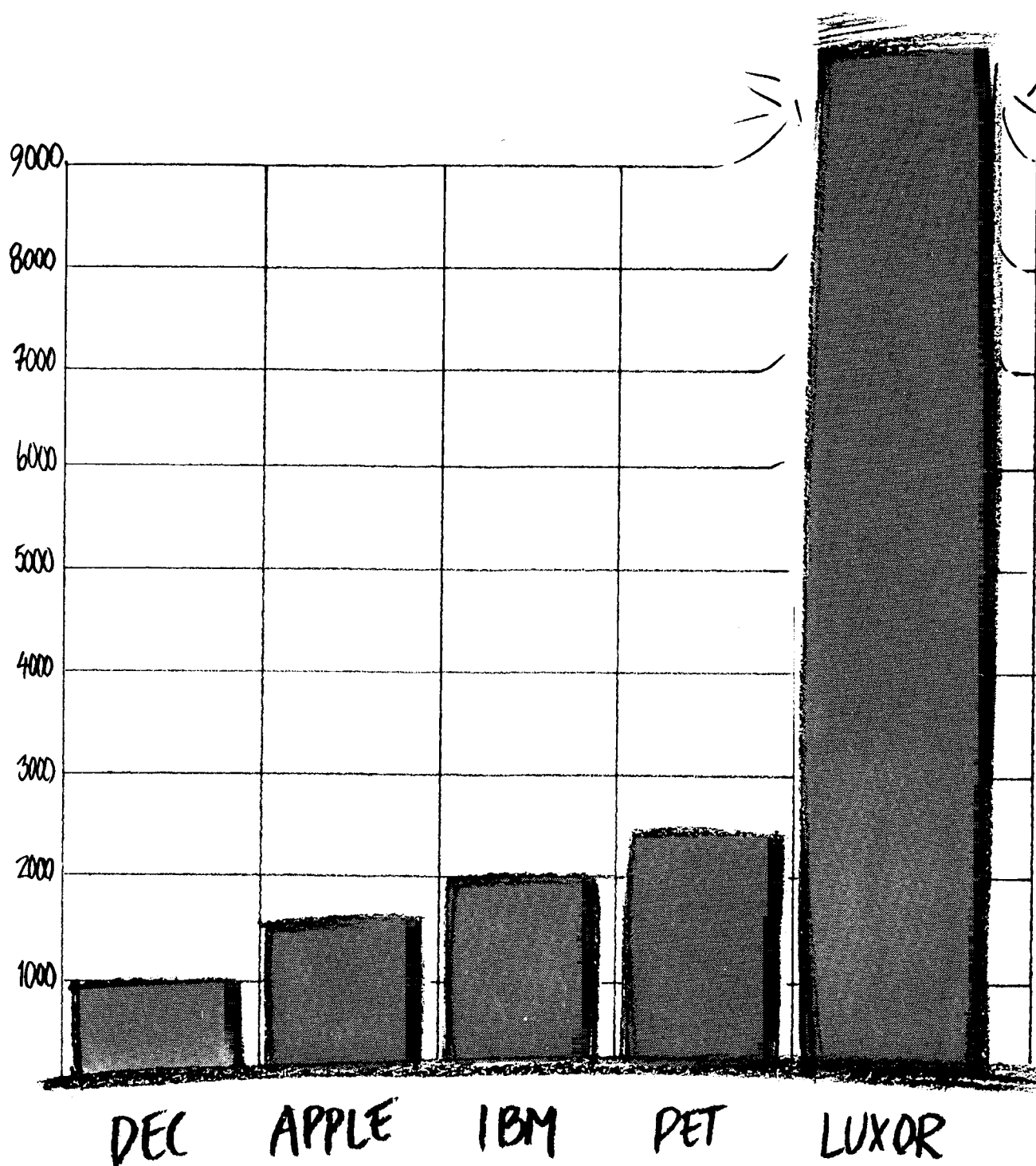


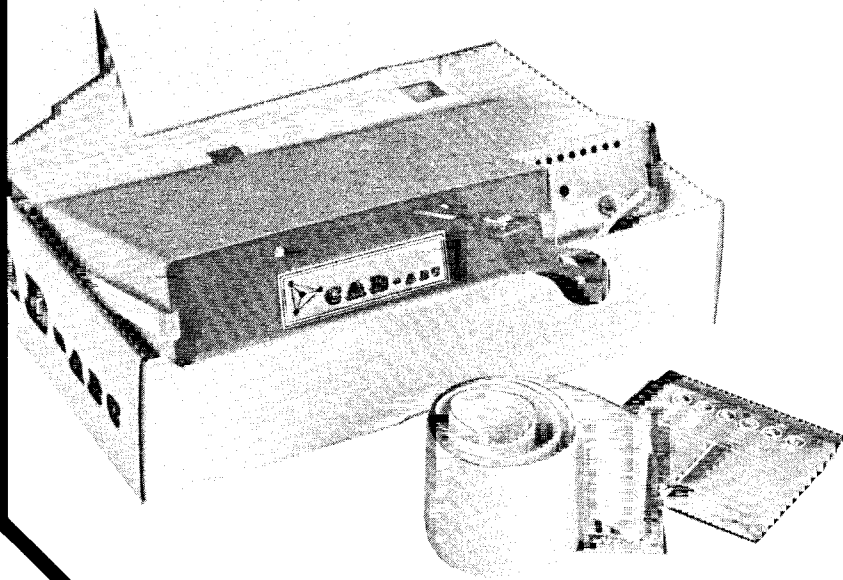
ABC BLADET

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

NUMMER 1 1984



CAD med ABC 80



CAD-paketet som betalar sig på 5–10 kretskort!

CAD-ABC är en komplett arbetsplats för framställning av ritningar, schemor och mönsterkortsunderlag. Systemet är baserat på ABC 80 i standardutförande kompletterat med ytterligare 16 kbyte RAM och en dubbel flexskiveenhet.

CAD-ABC – utvecklad av användare för användare!

CAD-ABC har snabbt blivit det mest sålda CAD-systemet i Sverige. Alla som har behov av CAD kan nu skaffa sig en egen anläggning utan att investera 6-siffriga belopp. Småföretag får fördelarna att kunna tjäna pengar och tid med en egen CAD i huset. Storföretags stora CAD-anläggningar får avlastning genom att alla småjobb klaras med en CAD-ABC på respektive arbetsplats. Med CAD-ABC-tillsats och programvara blir den ett komplett system för framställning av ritningar och layouter för kretskort.

Du kan systemet på två dagar genom självstudiekurs.

Kontakta avd. Mikrodatorer för ytterligare information.

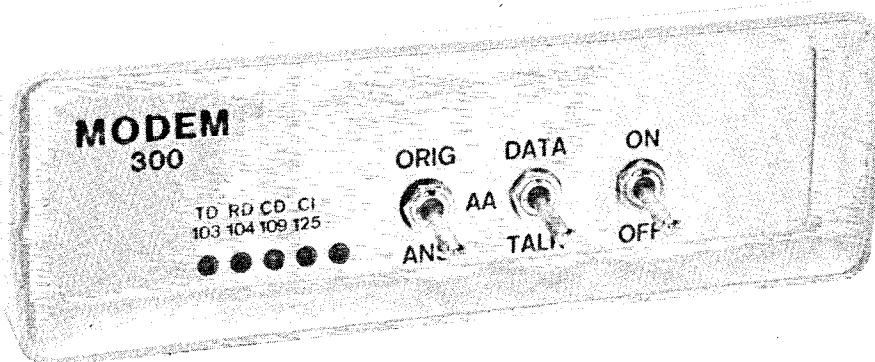
ELFA
RADIO & TELEVISION AB
171 17 SOLNA
INDUSTRIVÄGEN 23 • 08-730 07 00

KLIPPET.

SELIC's Pris 1550 kr + moms.

(Televerkets motsvarande modem kostar 2400 kr + moms. Dat 84.02.23)

A- och B-kanal samt autosvar.
Godkänd av televerket.



SELIC AB

Box 44002 • 40076 Göteborg. • Tel 031-22 3310.

Medlemsorgan för ABC-klubben

Vidängsvägen 1
161 33 Bromma
ISSN 0349-3652

Ansvarig utgivare: Gunnar Tidner
Redaktör: Rune Mattsson
I redaktionen: Odd Rolander
Claes Schibler
Ulf Sjöstrand

ABC-klubbens postgiron:
Medlemsavgifter: 15 33 36-3
Publikationer: 62 93 00-5
Q-Zentralen: 43 51 74-8

Telefoner:
08-80 15 22 (automatisk telefonsvarare med
aktuell klubbinformation)
08-80 15 23 (modem med ABC-Monitor)
08-53 57 50 (kansli)

Tryck: Märstatryck AB 1984

Lämnad till tryck 1 mars 1984.
Upplaga 8 000 ex.

NUMMER 1, 1984

INNEHÅLL

Omslag: Levererade smådatorsystem 1983 i prisklass
ca 20.000-70.000 kr.

Ledare av Gunnar Tidner	4
Snabbreferat från årsmötet	4
Datorpolitik av Herbert Söderström	5
FORTH av P-E Martin	8
FrågeBITen av Kalle Lindström	11
ABC-klubben har fått nytt kansli av Gunnar Tidner	12
Historik av Bo Kullmar	12

Inlägga Kort och Brett om DEC-10

ABC-datorer – nytt hjälpmedel för släktforskarna av PO Bergman	13
Rymdäventyrarna av Sven Wickberg	14
ABC80 som rammaskin av Kjell-Åke Johansson	16
Programmering BASIC II av Bo Kullmar	17
Hardcopy för ABC800 av Bo Kullmar	18
Kommandon i program av Bo Kullmar	18
800-program på kassett av Bo Kullmar	19
Radannonser	19
Disassemblern, vad är det? av Kalle Lindström	20
DIV:	20
Brev till ABC-klubben	21
ABC Väst	22
Utgivningsplan för ABC-bladet	23

Annonspriser fr o m nr 4, 1983

1/1-sida 185 × 260 mm	2.400:-
1/2-sida 185 × 128 mm, eller 90 × 260 mm	1.450:-
1/3-sida 185 × 85 mm, eller 60 × 260 mm	975:-
1/4-sida 90 × 128 mm	780:-
2 st 1/1-sidor i uppslag	5.400:-
2:a omslagssida	3.000:-
3:e omslagssida	2.875:-
4:e omslagssida 185 × 225 mm	3.300:-
Begärd placering 10% förhöjning.	

Tidningen ansvarar ej för att införda programlistningar är korrekta.

Utgivningsplan se sid. 23.

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

Upphovsrätt gäller för införda program om inget annat anges.

Medlemsavgifter 1984

Seniorer 140 Skr
Juniorer 80 Skr

Junior räknas man t o m det kalenderår man fyller 18 år. Ange därför personnummer när Du betalar medlemsavgifter.

Medlemskapet är personligt och avser fysisk person. Medlemsskapet räknas per kalenderår och Du får automatiskt det löpande årets förmåner retroaktivt om Du inte markerat annat årtal på talongen när medlemsavgiften betalas in.

Medlem blir Du enklast genom att sätta in medlemsavgiften på ABC-klubbens postgiro-konto 15 33 36-3 och ange en entydig avsändare.

ABC-klubbens styrelse för 1984

(enligt årsmötet 1984-02-18)

Ordförande: Gunnar Tidner

Vice ordförande: Stig Löfgren

Ledamöter: Bo Kullmar

Rune Mattsson

Jan Holmberg

Torsten Ljungström

Gösta Stenhorn

Suppleanter: Kalle Lindström

Björn Sjöborg

ABC-klubbens postgiron:

Medlemsavgifter: 15 33 36-3

Publikationer: 62 93 00-5

Q-Zentralen: 43 51 74-8

LEDAREN.

Det är nu mer än fem år sedan ABC80 kom ut på marknaden. Fem år är en ganska lång tid när det gäller datorer. Det är imponerande hur väl ABC80 har kunnat hänga med i utvecklingen under denna tid. Förklaringen torde vara dels den ursprungliga försäljningsframgången dels en från början ovanligt vällyckad konstruktion. Tack vare den stora spridning som ABC80 fick har det funnits en marknad för kompletteringssatser som gjort det möjligt att upgradera grundutrustningen t ex vad gäller minneskapacitet och 80 teckens bildskärm utan att kompabiliteten till tidigare program gått förlorad. Att det funnits en livskraftig användarklubb där medlemmarnas erfarenheter kunnat tas till vara och spridas till övriga användare har säkert också varit en bidragande förklaring.

Hur länge till ABC80 klarar att hänga med är en öppen fråga.

ABC80 i grundutförande med endast 16 K Ram och 40 teckens bildskärm kan med dagens mått inte anses fylla kraven på en persondator för allroundbruk. Men den kan fortfarande vara det ekonomiskt bästa alternativet vid vissa speciella tillämpningar. Rent tekniskt kommer apparaten att hålla för många års ytterligare användning. Men förr eller senare kommer den dag då man byter till en modernare utrustning. Det

måste vi som blivit känslomässigt fästa vid våra apparater inse.

De som nu skaffar sig datorutrustning väljer i stället någon ur ABC800-serien. För professionellt bruk torde de flesta välja ABC806.

För den som redan har en ABC80 och investerat mycket i egna program kan det dock fortfarande vara ett alternativ att gradera upp sin utrustning.

Man skulle kunna uttrycka det så att det pågår en form av skalömsning bland våra medlemmar och datoranvändare. Det är en process som inte alltid går helt smärtfritt.

Det är nu fyra år sedan ABC-klubben bildades.

År 1983 har även på ett annat sätt inneburit en skalömsning för klubben. Den kraftiga ökningen av antalet medlemmar - medlemsantalet passerade under året 4000 - har medfört vissa påfrestningar på klubbens administrativa rutiner. Tidigare har klubbadministrationen kunnat skötas på huvudsakligen ideell basis. Mot slutet av året visade sig detta allt svårare. Det ledde till en akut kris som inte löstes förrän klubben i december fick ett nytt kansli med halvtidstjänstgörande kanslist.

För ABC-klubben blir det nödvändigt att anpassa sig efter de ändrade förhållandena

om vi vill göra anspråk på att vara en klubb för alla användare av ABC-datorer. Och det är vad som beslöts på klubbens årsmöte.

Utvecklingen under de gångna åren visar att det finns ett stort intresse och behov av en användarklubb där vi kan utbyta och dela med oss av våra erfarenheter och få hjälp av varandra. Om vi skall kunna bibehålla det stora intresse som nu finns för klubben så måste vi kunna tillfredsställa vissa grundläggande krav från våra olika kategorier av medlemmar. Vi måste ge ut en tidning där alla kan finna saker som intresserar honom eller henne. Det måste även i fortsättningen komma fram nyttiga och bra användarprogram och vi måste på något sätt klara av att distribuera dem även till de som inte har någon kassettbandspelare.

ABC-klubben eftersträvar goda och förtroendefulla relationer till Luxor och andra leverantörer och tillverkare. Men klubben måste slå vakt om sin självständiga, av ekonomiska intressen oberoende ställning. När vi stöter på svagheter i hårdvaran eller mjukvaran är det vår uppgift att säga ifrån. ABC-klubben representerar tillräckligt många och kunniga medlemmar för att vi skall ha rätt kräva att våra synpunkter skall bli tagna på allvar.

Det är inte särskilt ovanligt att vi är först med att upptäcka ett systemfel eller annan svaghet i viss utrustning, fel som någon medlem i klubben hittat och rapporterat. I flera fall har det varit i samband med en udda tillämpning i klubbens egen verksamhet som ett fel först avslöjats. Det är ett rimligt krav att felet eller bristerna så småningom rättas till i nyproduktionen och att användare som drabbats av felet kan få den aktuella enheten utbytt om inte helt kostnadsfritt så i varje fall till en låg självkostnad.

Vi anser oss också ha rätt att ställa krav på att få tillgång till ordentlig dokumentation av utrustning och systemprogram. Här finns fortfarande allvarliga brister.

Gunnar Tidner

Snabbreferat.

Detta är ett snabbreferat av ABC-klubbens årsmöte. Avsikten är att som brukligt ha ett fylldt referat av årsmötesförhandlingarna. Detta är främst med tanke på att mötet besöks av en mycket liten andel av medlemmarna.

ABC-klubben hade årsmöte lördagen den 18 februari. Lokalen var Brommasalen, Gustavlundsvägen 168, Bromma. Samtidigt hade ordnats en ABC-dag där leverantörer av tillbehör m m till ABC-datorerna visade upp sina sortiment och diskuterade sina produkter med intresserade besökare.

Årsmötet öppnades av ABC-klubbens ordförande Gunnar Tidner som hälsade alla välkomna. Till mötesordförande valdes Torbjörn Alm och till mötessekreterare Bo Kullmar.

Årsmötet godkände verksamhetsberättelsen, resultat- och projektredovisning. Revisorerna redovisade revisionsberättelsen och tillstyrkte att styrelsen beviljas ansvarsfrihet för år 1983, vilket årsmötet beviljade.

Vidare beslöts att fastställa balansräkningen och att årets överskott, kronor 9.849:41, skall föras i ny räkning för verksamhetsåret 1984.

Beslöts att anta Stig Löfgren m fls motion om att stödet till lokalavdelingar

inte skall vara tidsbegränsat till de två första åren.

Beslöts att fastställa styrelsens förslag om medlemsavgiften för 1984 till 140 SEK för seniorer respektive 80 SEK för juniorer.

Beslöts att ge styrelsen mandat att besluta om en preliminär medlemsavgift för 1985 om högst 160 kronor för senior och 100 kronor för juniorer.

Beslöts att med en mindre ändring fastställa budgeten för 1984 enligt styrelsens förslag.

Beslöts efter votering att välja Gunnar Tidner som ordförande.

Beslöts att välja Stig Löfgren som vice ordförande.

Beslöts att välja följande till ledamöter i ABC-klubbens styrelse:

Bo Kullmar omval

Rune Mattsson omval

Gösta Stenhorn nyval

Torsten Ljungström nyval

Jan Holmberg nyval

Beslöts att till suppleanter välja

Kalle Lindström och Björn Sjöborg.

Beslöts att välja Nils Larsson och Lennart Jansson till revisorer, och att välja Lars-Göran Göransson till revisorssuppleant.

Till valberedning utsågs Göran Sundqvist (sammanställande), Magnus Hedner och Allan Larsson.

Beslöts att ändra §1 till följande lydelse:

ABC-klubben är en ideell sammanslutning av användare av ABC-datorer. Klubben skall verka för att tillvarata gemensamma intressen av datorer och datortillämpningar, verka för ökade kunskaper inom dessa områden till nytta, utbildning och nöje.

Beslöts att §5 skall ha följande lydelse:

Medlemsavgiftens storlek bestäms av årsmötet. Medlemskap träder i kraft när medlemsavgiften erlagts. Medlemsavgift skall erläggas årligen före den 15 mars.

Dessa beslut noterades vara enhälliga.

Beslöts att styrelsen innan styrelsen tar beslut med större ekonomiska konsekvenser skall styrelsen rådgöra med revisorerna.

Beslöts att styrelsens redovisningshandlingar till årsmötet skall innehålla en lista över inventarier. Listan skall upprättas i samförstånd med revisorerna.

Styrelsen meddelade att man avser att utarbeta ett förslag till reviderade stadgar för ABC-klubben. Årsmötet uttalade sig för att förslaget skall behandlas på ett extra årsmöte under hösten 1984.

Beslöts att på förslag av Gunnar Tidner vid avtackningen av avgående styrelsemedlemmar att utse Ulf Sjöstrand, Kjell-Åke Johansson, Marianne Forsman och Joe Johansson till hedersmedlemmar i ABC-klubben. Gunnar Tidner framhöll att dessa har under tre mandatperioder haft mycket arbetskrävande uppgifter inom ABC-klubbens styrelse.

Datorpolitik.

Hösten 1983 deltog jag i en datadebatt i Falun, arrangerad av Dala-Forum, en organisation för debatt och analys av Dalarnas ekonomi. ganska trista såg de ut så dags. Kanske har de ljusnat något sedan dess.

I materialet talades det mycket om industrihotellet och andra stimulerande samhällsätgärder, samt något ganska vagt om datautbildning. Jag hade försökt få veta lite om datoriseringen i Dalarna, men ingen visste något säkert: Några stordatorer kände man till, men mini- och mikrodatorbestyckningen hade man inget grepp om, än mindre vilket datakunnande Kopparbergs län kunde mobilisera.

Televerket, som ju har ordning på sina grejor, kunde emellertid exakt ange antalet telefonmodem i länet, strax under 1.000 var det då. Men hur många datorer har modem? Hur många har det inte? Låt oss anta, att det på varje modem går fyra som saknar. Det skulle då i Kopparbergs län finnas ca 5.000 datorer. Låt oss räkna lågt och anta, att varje anläggning i genomsnitt kostat 15.000 kr, en blygsam summa, eftersom nog de flesta mikrodatorerna fortfarande finns på företag och institutioner. Det skulle innebära, att enbart i Dalarna står det mikrodatorutrustning för 5.000×15.000 kr = 75 miljoner kr! Det är en summa många gånger större än Dalarnas alla samlade och mycket mer omskrivna industrihotellet.

Hur har man tänkt sig bedriva datapolitik utan att veta just något om hur datalandskapet ser ut? Kan man bedriva datapolitik och tänka bort datautvecklingen? Studerar man partiernas datapolitiska skrifter, skall man finna att ordet mikrodator ännu inte trängt in i de politiska texterna.

Sverige är i och för sig ett relativt avancerat elektronikland. Vi låg 1980 enligt Mats Bäck: Datorisering och datapolitik (Liber 1982) efter Japan, USA, Västtyskland, England och Frankrike men före Holland, Belgien, Schweiz och Österrike, för att inte tala om Danmark, Finland och Norge - om man räknar elektronikproduktionen som andel av bruttonationalprodukten.

Nu är datorer och elektronik inte detsamma, men några betecknande iakttagelser finns dock: Luxor har den högsta hemmamarknadsandelen av alla datortillverkare i världen, åtminstone några av de svenska mjukvaruhusen har halva sin vinstprognos utomlands, vilket tyder på att vår mjukvaruexport verkligen kommit i gång. Men att säga hur det "egentligen" står till, är omöjligt, ingen vet exakt hur marknadsbilden ser ut. Vi vet exakt hur många betäckta modersugor vi har i Sverige - viktigt för att veta tillgången på skinka till jul! men eventuellt vore det lika viktigt att veta hur datoriseringen fortskrider, inte minst om man har som deklarerad avsikt att sprida datakunnandet i Sverige, något som regeringen sagt sig vilja.

En enkel kalkyl:

Låt oss anta, att de som nu går ut gymnasiet får åtminstone ett minimum av datakunnande, tillräckligt för att snabbt lära sig köra en dator med smör- & brödrutiner, ett antagande som är väl optimistiskt, men låt gå! Om vi vidare antar, att bara personer under 55 år realistiskt kan antas vara beredda att lära sig, så har vi

grovt taget 35 årskullar som behöver kompletterande utbildning. Alla i en årskull behöver inte kunna köra dator (frisörer och taxichaufförer, dagisfröknar och operasångerskor kan t ex få vänta ett tag), låt säga en fjärdedel av varje årskurs i en första utbildningsvåg: $0.25 \times 35 \times 100.000 = 875.000$ personer som skall ges utbildning.

Av egen erfarenhet vet jag följande:

o Man kan inte undervisa mer än 12 åt gången.

o Det tar två kursdagar att lära sig köra ordbehandling på ABC 80.

o Det tar lika lång tid att lära sig t ex Visicalc - eller för all del att programmera matematik i BASIC, det är hugget som stucket.

o Ett lärarår omfattar 40 veckor a 5 dagar, alltså 200 lärardagar per år.

Sålunda behövs det för 875.000 elevdagar ca 72.900 lärardagar.

72.900 lärardagar = 365 lärarår.

Det är ingen oöverstiglig skolstrategisk operation - förutsatt att lärarkunnandet finns, men det vet man ju inte i dag.

Låt oss vidare anta, att vi vill genomföra operationen på fem år, så krävs det en lärarinsats av 365/5 lärare per år = 73 lärare. Redan nu börjar projektet anta mänskliga proportioner.

Låt oss vidare anta, att en tiondel av varje årskull behöver 10 dagars påbyggnadskurs utöver de två i grundkursen, så blir det $0,1 \times 10 \times 20 \times 100.000 =$ ytterligare 2 miljoner elevdagar. Med samma matematik som nyss krävs för detta ytterligare 833 lärarår - verken, fördelat på 5 år ger 167 lärare. Sammanlagt skulle det under fem år behövas $73 + 167$ lärare = 240 lärare på heltid. Inte heller detta ter sig särskilt överkomligt - om man nu visste var lärarna fanns.

Ungefär hälften av alla mikrodatorkunniga när det gäller professionellt arbete vid maskinen, arbetar med ABC 80 eller ABC 800. Jag bortser nu från alla kunniga entusiaster på ZX 80-81, Spectrum, Atari och Vic 20, som jag har all respekt för, men för att lära ut professionell ordbehandling eller kalkyl i kontorsmiljö räcker inte de maskinerna till.

ABC-klubben har vid det här laget ca en fjärdedel av de ABC-kunniga som medlemmar (en grov skattning!), alltså en åttondel av alla Sveriges mikrodatorkunniga. Vore det omöjligt att inom den kretsen hitta, låt oss säga, 100 personer, som vore beredda att ta tjänstledigt en månad per år för att försöka sig som datalärare? 100 personer med 4×40 lärartimmar ger 16.000 lärartimmar per år, på fem år 80.000 lärartimmar, 960.000 elevtimmar av beräknade 2.875.000 elevtimmar, d v s en tredjedel!

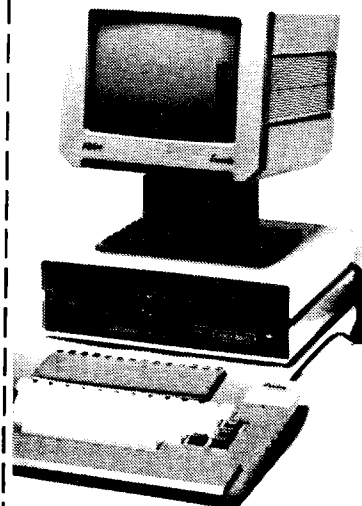
En del av oss kan undervisa i ordbehandling, andra i kalkyl, åter andra i programmering eller handhavande av grafikprogram. Hur vore det om ABC-klubben, bara för att få statsmakterna att ändå börja se sig om efter resurser, gjorde en medlemsinventering av den lärarpotential som rimligen borde finnas inom klubben?

I varje fall vore en sådan inventering värd minst lika mycket som kunskapen om antalet telefonmodem i Kopparbergs län.

Herbert Söderström



SLIM LINE FLOPPY TILL ABC



Liten men naggande god.
Se bara:

DataDisc 52—2 x 160 kB

DataDisc 54—2 x 320 kB

DataDisc 56—2 x 640 kB

Passar till ABC 80, 800, 802, 806.

Välkommen till T-D-X Norr
eller till T-D-X City!

T-D-X

SmåDatorerAB

T-D-X NORR: SOLLENTUNA-
VAGEN 225, 191 23 SOLLENTUNA
T-D-X CITY: KUNGSGATAN 79,
KUNGSHOLMEN STOCKHOLM
GEMENSAMT TEL NR 08-96 01 80

HELLO IBM. HEJ ERIC TJOSAN DEC. MORR'N

ABC 800 SOM TERMINAL- DATOR.

En handbok från Luxor Datorer.

ARNEK ANNONSBYRÅ AB

Hur vi än använder våra smådatorer. Alla behöver vi datautbildning. Anmäl dej till Dataskolan Te

GÖTEBORG TEL. 031-815300 · HELSINGBORG TEL. 042-18 00 65 · KRISTINEHAMN TEL. 0550-156 40 · LINKÖPING TEL. 013-1154 70 · MALMÖ TEL. 040-10 05 70 · NORRKÖPING TEL. 011-10 81 50 · STOCKHOLM T

NÄRMASTE TEAM 100-HANDLARE HITTAR DU I TELEFONKATALOGENS G

ERICSSON. GODDAG CDC. UNIVAC.

Det finns två sorters terminaldatorer. Intelligent och dumma. Den dumma terminalen kan bara hämta och sända data, inte behandla den. Därför är den alltid beroende av en centraldator.

Den intelligenta terminalen däremot jobbar helt självständigt.

En smådator i ABC 800-familjen är ett bra exempel på en intelligent terminal. Den klarar de flesta av dagens terminalfunktioner och är förberedd för morgondagens.

Dessutom klarar den alla traditionella smådatorfunktioner. Att t.ex. hantera företags löner och administration. Att ta hand om kalkylering och ordbehandling.

Detta gör ABC 800 till ett värdefullt arbetsverktyg. Du kan söka och sprida i stort sett vilken information du vill. Lokalt. Nationellt eller internationellt. Inom huset, i olika databaser, i ditt moder- eller dotterbolag. Och du kan behandla alla data i din ABC 800. T.ex. kan du plocka data från en stordator och köra den mot dina standardprogram eller kontrollera alla indata innan du kör dom vidare på en dyrbar stordator.

Hela tiden har du dessutom valfriheten att jobba mot olika datorfabrikat, IBM, DEC eller Univac för att nämna några. Och inte nog med det. Tack vare ABC 800 kan du också låta de olika datorfabrikaten tala med varandra. Du kan transportera data kors och tvärs mellan maskinerna, även om dom är av olika fabrikat och normalt inte kan utväxla ett enda tecken sinsemellan.

Det är därför vi säger Hello till IBM, Hej till Ericsson och Morr'n till Univac.

Inte bara som en vänlig hälsning. Utan mera som en inbjudan. Vi erbjuder oss att tjänstgöra som tolk närhelst som ni behöver prata med varandra.

*Hälsningar
ABC 800.*

Ja, jag tror på en öppen dialog och vill veta mer. Skicka mej den nya handboken "ABC 800 som terminaldator"

Namn _____

Företag _____

Adress _____

Postnr/Ort _____

Tel _____

Sänd kupongen till Luxor Katalogservice, Box 10000, 434 01 Kungsbacka.

LUXOR
Datorer

Team 100. Hämta kursschema hos Team 100-handlarna eller ring direkt till Dataskolan Team 100.

TEL. 08-16 00 60 · SUNDSVALL TEL. 060-15 38 03 · UPPSALA TEL. 018-15 52 20 · VÄXJÖ TEL. 0470-170 39, 130 20 · YSTAD TEL. 0411-187 17, 198 18 · ÖREBRO TEL. 019-10 06 10 · ÖSTERSUND TEL. 063-1100 60
EN; GULA SIDOR UNDER RUBRIKEN "DATAMASKINER OCH TILLBEHÖR".

FORTH.

Hej!

Översänder här ett stränghanteringspaket för Forth att publiceras/distribueras till ABC-klubbens medlemmar. Programmen skrevs när jag fortfarande var ganska nybliven Forth-programmerare så det föreligger inga anspråk på att det är det 'elegantaste' sättet att lösa problemen.

Huruvida man verkligen har användning för ett sån't här paket överlåter jag åt var och en att avgöra. Programmen skrevs i första hand som programmerings övning.

Förhoppningsvis är alla 7 skärmarna 'bug'-fria men skulle någon hitta något, eller ha andra synpunkter på paketet, så hör av er.

P-E Martin <2277>
Västerås

PS

Här är några Forth-ord som används av sträng-rutinerna följt av några enkla exempel för att 'komma igång'. I övrigt hänvisas till manualen.

```
: 1- 1 - ;
: 2- 2 - ;
: PICK SPÉ SWAP 2 * +
      DUP SO É 1 - >
      IF ." Out of bounds!"
        DROP CR
      ELSE É ENDIF ;
: 2ROT 6 PICK 6 PICK SPÉ 2- DUP
       12 + DO I É I 4 + !
       -2 +LOOP 2DROP ;
: 2OVER 4 PICK 4 PICK ;
: NDROP 0 DO DROP LOOP ;
: <ROT ROT ROT ;
```

Exempel:

```
15 $ FÖRNAME skapar variabeln
FÖRNAME med max 15 tecken-
positioner.
FÖRNAME $<-" Sverker" tilldelar
FÖRNAME innehållet Sverker.
```

FÖRNAME \$MAX lägger det maximala antalet positioner för variabeln på stacken som är 15.

FÖRNAME \$SIZE lägger den aktuella stränglängden på stacken, som är 7.

FÖRNAME \$. skriver ut innehållet i variabeln.

4 10 \$VECT FÄRG skapar en strängvektor med 5 element (0-4), med vardera 10 teckenpositioner.

2 FÄRG \$MAX ger 10 på stacken och 2 FÄRG \$SIZE ger 0 .

```
0 FÄRG $<-" GRÖN"
1 FÄRG 0 FÄRG $<-
2 FÄRG $<-" RÖD"
3 FÄRG $<-" GUL"
4 FÄRG $<-" SVART"
```

1 FÄRG \$CONC" ARE" ger element 1 innehållet GRÖNARE.

0 0 FÄRG \$V. ger utskriften

GRÖN GRÖNARE RÖD GUL SVART
medan

1 0 FÄRG \$V. ger utskriften

GRÖN
GRÖNARE
RÖD
GUL
SVART

0 FÄRG \$VCLR nollställer size-byten i samtliga element så vektorn kan betraktas som 'tom'.
Texten ligger dock kvar i det reserverade minnesutrymmet och kan raderas med
0 FÄRG \$BLANKS 1 FÄRG \$BLANKS osv

FORTH STRÄNG-PAKET V 1.0 av Martin

Symboler :

a adress till size byten i strängen.
m max antal tillåtna tecken i sträng.
s aktuellt antal tecken i strängen.
p position i strängen.
n ett enkeltal, oftast antal tecken eller index.
c ascii-kod.
b booleskt värde (Kan vara -1).

Observera att a normalt betecknar en adress men i texten används den för att symbolisera strängens innehåll. Till exempel vid \$COMP betyder al<a2 att strängen al's ascii-koder är mindre än a2's koder inte att adressen al är mindre än adressen a2.

\$
Anropas i formen m \$ namn som skapar strängvariabeln namn med reserverad plats för m tecken.
Variabeln får då formen :

```
Byte 1 2 3 4 ... m+2
- max size c1 c2 ... cm
```

där max är reserverad längd och size är aktuell längd. c är ascii-tecken. Vid anrop av namn lämnas adressen till size-byten (byte 2) på stacken.

Anropar : (\$)

\$ (a —)
Skriver ut texten i variabeln.

Anropas av : \$V.

\$< (al a2 — b)
b=1 om al<a2 annars är b=0.

Anropar : \$COMP

\$<- (al a2 —)
Strängen al tilldelas a2's innehåll utan att a2 förändras. Om a2 är längre än al's maxlängd kapas överskjutande tecken bort.

Anropar : \$MAX
Anropas av : (\$\$) \$MID \$CONC \$INS

\$<-\$ (a n —)
Tilldelar a en siffersträng skapad av enkeltalet n.

Anropar : (\$\$)

\$<-D\$ (a d —)
Som \$<-\$ fast med ett dubbeltal.

Anropar : (\$\$)

\$= (a1 a2 — b)
b=1 om a1=a2 annars är b=0.

Anropar : \$COMP
Anropas av : \$SEEK

\$> (a1 a2 — b)
b=1 om a1>a2 annars är b=0.

Anropar : \$COMP

\$BLANKS (a —)
Fyller hela reserverade utrymmet i a med mellanslag. Size = max.

Anropar : \$FILL

\$CLR (a —)
"Raderar" strängen genom att noll-ställa size-byten.
Texten finns alltså kvar.

\$COMP (a1 a2 — b)
Jämför två strängar och lämnar värdet b på stacken.
b= 0 om a1=a2
b= 1 om a1>a2
b=-1 om a1<a2

Anropar : 2ROT NDROP
Anropas av : \$= \$< \$>

\$CONC (a1 a2 —)
a1 tilldelas den sammanfogade strängen a1+a2. a2 förändras inte.

Mellanlagrar text i PAD, data i returnstacken.

Anropar : \$<-

\$DEL (a p n —)
Tar bort n tecken från och med pos. p ur a och "drar ihop" texten.
Mellanlagrar data i returnstacken.

Anropar : (\$IX2) \$ERROR 2OVER 1-

\$ERROR (n —)
Skriver ut felmeddelande n relativt rad 0 i screen 4, tömmer stacken och gör QUIT.

Anropas av : \$VECT \$STORE \$SEEK \$LAST \$GET \$PUT \$MID \$LEFT \$RIGHT \$INS \$DEL

\$FILL (a c —)
Fyller hela reserverade utrymmet i a med ascii-tecknet c. Size sätts lika med max.

Anropar : \$MAX
Anropas av : \$BLANKS

\$GET (a p — c)
Lägger ascii-tecknet c, från pos. p, på stacken.

Anropar : (\$IX1) \$ERROR

\$INS (a1 p a2 —)
"Klämmer in" a2 i a1 från och med pos. p. a1's längd + a2's längd måste rymmas i a1's maxlängd.
Mellanlagrar text i PAD, data i returnstacken.

Anropar : (\$IX3) \$MAX (\$IX4) \$ERROR 1- <ROT \$<-

\$LAST (a c —)
Lägger till ascii-tecknet c sist i strängen.
Mellanlagrar text i PAD, data i returnstacken.

Anropar : \$MAX \$ERROR

\$LEFT (a n —)
Strängen a tilldelas de n första tecknen från sig själv.

Anropar : \$MID

\$MAX (a — m)
Lägger maxvärdet för strängen på stacken.

Anropas av : \$VECT \$<- (\$V) \$LAST \$INS \$FILL

\$MID (a p n —)
Strängen a tilldelas n tecken från och med pos. p från sig själv.

Mellanlagrar text i PAD.

Anropar : (\$IX2) \$ERROR \$<-
Anropas av : \$LEFT \$RIGHT

\$PUT (a p c —)
Skriver över tecknet i pos. p i strängen med ascii-tecknet c.

Anropar : <ROT (\$IX1) \$ERROR

\$RIGHT (a p —)
Strängen a tilldelas de sista tecknen från och med pos. p från sig själv.

Anropar : \$MID

\$SEEK (a1 p a2 — p2)
Söker efter a2 i a1 från och med pos. p1. Om a2 hittas är p2 positionen annars är p2=0.
Mellanlagrar text i PAD, data i returnstacken.

Anropar : <ROT \$ERROR \$=

\$SIZE (a — s)
Lägger size-värdet för strängen på stacken.
Är ekvivalent med C%.

\$STORE (a1 p a2 —)
Skriver över text i a1, från och med positionen p, med a2.

Anropar : (\$IX3) (\$IX4) \$ERROR

\$V. (b a0 —)

Skriver ut texten i samtliga element i vektorn. Om b är 0 skrivs ett mellanslag mellan elementen, om b är 1 görs en radmatning.

OBS! Måste anropas med element-index 0 !

Anropar : (\$V) \$.

\$VCLR (a0 —)
"Raderar" elementen i vektorn genom att nollställa size-byten.
Texten finns alltså kvar.
OBS! Måste anropas med element-index 0 !

Anropar : (\$V)

\$VECT
Anropas i formen n m \$VECT namn som skapar strängvektorn namn med n+1 element vardera med m reserverade byts för text.
Vektorn får då formen :

Element	0	1	...	n
-	n m s cl-cm	m s cl-cm	...	m s cl-cm

där n är högsta index. I övrigt har varje element samma form som enkla strängar (Se \$).
Ett element anropas sedan i formen n namn där n är elementindex.
Vid anrop lämnas adressen till elementets size-byte på stacken.

Anropar : (\$) \$ERROR \$MAX

(\$) (n —)

Lägger n i översta lediga byten i ordlistan följt av en byte 0 och reserverar sedan n bytes i ordlistan.

Anropas av : \$ \$VECT

(\$\$) (a ud —)

Omvandlar ett dubbeltal utan tecken till en textsträng som variabeln a tilldelas.

Texten mellanlagras under PAD.

Anropas : 1- \$<-

Anropas av : \$<- \$ \$<-D\$

(\$IX1) (a p — a p b)

Testar om positions-index hamnar utanför strängen. Lämnar en boolesk flagga på stacken.

Anropas : <ROT

Anropas av : \$GET \$PUT

(\$IX2) (a p n — a p n b)

Se (\$IX1).

Anropas : 2OVER <ROT 1-

Anropas av : \$MID \$DEL

(\$IX3) (al p a2 — al p a2 p al)

(\$IX4) (al p a2 p n — al p a2 b)

Se (\$IX1).

Anropas i formen (\$IX3) xxx (\$IX4) där xxx är \$SIZE eller \$MAX. Vid anrop av (\$IX4) är n alltså size/max-värdet.

Anropas : 2OVER

Anropas av : \$STORE \$INS

(\$V) (a0 — steg gräns start)

Räknar om adressen till första elementet i en vektor till stegvärde, gränsvärde och startvärde för en DO+LOOP.

OBS! Måste anropas med element-index 0 !

Anropas : \$MAX 2-

Anropas av : \$VCLR \$V.

Extensions

\$"

Tar in text från inströmmen avgränsad med ett " och lägger den ovanför ordlistan där första byten innehåller antal tecken. HERE läggs på stacken.

Anropas av : Alla ord som har " som sista tecken.

\$<"

\$<"

\$="

\$>"

\$COMP"

\$CONC"

\$INS"

\$SEEK"

\$STORE"

Dessa anropas som sina huvud-ord med den skillnaden att den andra strängen skrivs direkt efter ordet, avslutat med ett " .

Exempel :

I uttrycket

PERSON NAMN \$<-

tilldelas strängen PERSON strängen NAMN's innehåll. Om NAMN innehöll 'Pelle Pettersson' så fås samma resultat med uttrycket

PERSON \$<-" Pelle Pettersson"

Följande uttryck lägger talet 14 på stacken

SUFF \$<-" son"

PERSON 1 SUFF \$SEEK

Utrycket

PERSON 1 \$SEEK" son"

ger samma resultat.

\$ 50
(STRÄNG 1 830226 PEM)

```
: $ERROR CR MESSAGE SP! QUIT ;
: ($IX1) OVER C$ OVER DUP
  1 < <ROT < OR ;
: ($IX2) DUP 2OVER SWAP C$ <ROT DUP
  1 < <ROT + 1- ROT > OR ;
: ($IX3) DUP C$ 2OVER SWAP ;
: ($IX4) OVER 1 < SWAP ROT
  - 1+ ROT < OR ;
: ($) DUP C, 0 C, ALLOT ;
: $ <BUILDS ($) DOES> 1+ ;
: $MAX 1- C$ ;
: $SIZE C$ ;
: $VECT <BUILDS SWAP DUP C, 1+ 0
DO DUP ($) LOOP DROP
DOES> OVER OVER C$ OVER
0 < <ROT > OR
IF 14 $ERROR ELSE 2+ DUP
$MAX 2+ ROT * + ENDIF ;
```

-->

\$ 51
(STRÄNG 2 830226 PEM)

```
: $<- SWAP OVER C$ OVER $MAX
2DUP > IF SWAP THEN DROP
2DUP SWAP C! ROT 1+ ROT
1+ ROT CMOVE ;
: $CLR 0 SWAP C! ;
: ($V) DUP DUP $MAX 2+ DUP ROT
DUP 2- C$ 1+ ROT * + ROT ;
: $VCLR ($V) DO 0 I C! DUP
+LOOP DROP ;
: $. COUNT TYPE ;
: $V. CR ($V) DO I $. OVER
IF CR ELSE SPACE THEN
DUP +LOOP DROP ;
: ($$) <$ $S SIGN $> OVER 1-
C! 1- $<- ;
: $<- $ DUP ABS 0 ($$) ;
: $<-D$ SWAP OVER DABS ($$) ;
```

-->

\$ 52
(STRÄNG 3 830226 PEM)

```
: $COMP COUNT ROT COUNT ROT 2DUP
2ROT 2SWAP MIN 0 DO OVER
I + C$ OVER I + C$ 2DUP
> IF 4 NDROP -1 LEAVE
ELSE < IF 2DROP 1 LEAVE
ELSE I 1+ I' = IF 2DROP 0
THEN THEN THEN LOOP DUP 0=
2SWAP ROT IF 2DUP > IF
3 NDROP 1 ELSE < IF DROP
-1 THEN THEN ELSE 2DROP
ENDIF ;
: $= $COMP 0= ;
: $< $COMP -1 = ;
: $> $COMP 1 = ;
: $STORE ($IX3) C$ ($IX4)
IF 16 $ERROR ELSE
COUNT 2SWAP + SWAP CMOVE
ENDIF ;
```

-->

FrågeBlTen.

\$ 53
(STRÅNG 4 830226 PEM)

```

: $SEEK <ROT OVER C# OVER < SWAP
>R >R OVER C# OVER C#
OVER - DUP 0< R> OR
IF R> 2DROP $ERROR
ELSE 2+ SWAP PAD C! R>
2DUP > 0= OVER 1< OR
IF 16 $ERROR ELSE
DO DUP 1+ PAD 1+ PAD
C# CMOVE OVER PAD $=
IF 1 LEAVE ELSE 1+ 1'
= IF 0 THEN THEN
LOOP <ROT 2DROP ENDIF
ENDIF ;
: $LAST OVER DUP C# 1+ DUP
>R OVER $MAX >
IF 16 $ERROR ELSE
COUNT + C! R> SWAP C!
ENDIF ;

```

-->
\$ 54
(STRÅNG 5 830226 PEM)

```

: $GET ($IX1) IF 16 $ERROR ELSE
OVER + C# SWAP DROP
ENDIF ;
: $PUT <ROT ($IX1) IF
16 $ERROR ELSE + C!
ENDIF ;
: $MID ($IX2) IF 16 $ERROR ELSE
ROT DUP 2SWAP SWAP ROT
+ SWAP DUP PAD C! PAD 1+
SWAP CMOVE PAD $<-
ENDIF ;
: $LEFT 1 SWAP $MID ;
: $RIGHT OVER C# OVER - 1+ $MID ;
: $CONC 2DUP C# OVER C# + PAD C!
COUNT PAD 1+ 2DUP +>R
SWAP CMOVE COUNT R> SWAP
CMOVE PAD $<- ;

```

-->
\$ 55
(STRÅNG 6 830226 PEM)

```

: $INS ($IX3) $MAX ($IX4)
IF 16 $ERROR ELSE
ROT OVER C# OVER C# +
PAD C! ROT OVER 1+ OVER
1- PAD 1+ 2DUP +>R SWAP
CMOVE ROT COUNT R> 2DUP
+>R SWAP CMOVE OVER
<ROT OVER C# OVER - 1+
<ROT + R> ROT CMOVE
PAD $<-
ENDIF ;
: $DEL ($IX2)
IF 16 $ERROR ELSE ROT
DUP C# 2OVER + 1- >R
2DUP C# SWAP - OVER C!
ROT + DUP >R + R> R>
CMOVE
ENDIF ;

```

\$ 56
(STRÅNG 7 830226 PEM)

```

: $FILL OVER DUP $MAX OVER C! 1+
ROT $MAX ROT FILL ;
: $BLANKS 32 $FILL ;

: $" 34 WORD HERE ;
: $<- " $" $<- ;
: $COMP " $" $COMP ;
: $= " $" $= ;
: $<" " $" $< ;
: $>" " $" $> ;
: $SEEK " $" $SEEK ;
: $CONC " $" $CONC ;
: $INS " $" $INS ;
: $STORE " $" $STORE ;
;S

```

Hejsan igen!

Ja, nu är jag här igen. Jag gjorde ett uppehåll i förra numret p g a att det inte hade kommit några frågor av (vad vi bedömer) allmänt intresse till oss. Istället så har jag med desto fler frågor den här gången.

Rikard Edström från Järfälla har skrivit detta brev till klubben:

Hej

Jag är en kille på 14 år som har ett antal frågor om ABC80.

1. Jag har medlemsnummer 3513. varför står jag inte med i medlemsmatrikeln för 1983?
2. Jag tycker att det är alldeles för lite program i ABC-bladet. Det vore jättebra med en stor annonssida där man kan köpa, sälja eller byta program med varandra.
3. Hur är det egentligen med en joystick till ABC80? Kan man köpa vilken som helst och plugga in eller måste det vara något särskilt märke? I så fall, hur mycket kostar en?
4. Jag har hört talas om två stora adventure-program, det ena heter STUGA och det andra heter THE CAVE. Vad för sorts dator finns de på och kan man få köra ett sådant?

5. Finns det egentligen några avancerade adventure-program till ABC80? I så fall var och vad är priset?
6. Kommer alla program på ABC-- kassetterna från ABC-klubbens programbank?

Rikard Edström
Hjälmvägen 5
175 61 JÄRFÄLLA

Svar:

1. Medlemsmatrikeln innehåller alla medlemmar mellan nr 100 och 3492. Dessa togs ut ur registret i februari 1983 men har på grund av förseningar hos tryckeriet inte distribuerats förrän på hösten.
2. En sådan "stor" annonssida har vi redan, radannonserna. Det står var och en fritt att köpa, sälja eller byta vad han/hon vill.
3. Vilken som helst går inte att ansluta men jag vet de som har anslutit en joystick från ett Atari-spel till en ABC-80. Jag skall försöka pressa fram en artikel ur en kille som har gjort en sådan anpassning. Vad jag vet så finns det bara en joystick att köpa och det är den som Comporian AB har tagit fram och den kostar 300-400 Sek. Adressen dit är:
Comporian AB
Box 4037
580 04 LINKÖPING
Tel: 013-14 80 92

4. Adventureprogram finns på de flesta stordatorsystem. Jag vet inte var man kan få tag i eller spela The Cave men jag vet att STUGA finns på QZ och att man kan köra det till reducerad taxa om man loggar in på ABC-klubbens konto, dock går det endast på det dyrare kontot. För att starta STUGA när man har loggat in i QZ skriver man RUN GAM:STUGA <return> så startar spelet. Det kostar i genomsnitt 1 krona per minut att spela.

5. Det mest avancerade adventure-program jag har sett till ABC80 är UPPSJO som kom ut på ABC-kassett 10. Kanske någon annan har ett bättre? Hör i så fall av er.
6. Ja, det gör de. Dock så finns inte hela klubbens programbank tillgänglig från monitorn med anledning av att vår 8-tummare inte rymmer allt. När vi till sommaren får igång vårt nya monitorsystem så kommer hela programbiblioteket att bli tillgängligt.

Det har kommit ett brev från Torbjörn Andersson <3448> som lyder så här:

JAG VILL LÄRA MIG MASKINSPRÅK !

Jag skulle vilja att ABC-klubben hade någon/några sidor där man skulle kunna lära sig maskinspråk från början (med många programexempel). Det finns tyvärr inte särskilt mycket böcker om maskinspråk för ABC80 så därför tycker jag att någon skulle kunna genomföra det här förslaget.

Kom inte och säg att det inte finns någon medlem som kan maskinspråk på ABC-80 för det räcker med att titta på era kassetter för att se att det måste finnas några medlemmar som kan maskinspråk.

Det är säkert fler än jag skulle vilja att detta genomfördes så jag hoppas att ni skulle vilja göra det.

Torbjörn Andersson

Svar:

Det finns inga medlemmar som kan maskinspråk. Nej, så är inte i verkligheten. ABC-Stockholm har funderat på att göra en assemblerkurs för sina medlemmar och sedan ge ut den som en ABC-rapport eller som en artikel i tidningen. När det hela kommer i gång kan jag inte säga.

TÄVLING

Den som före 1984-04-01 har skickat in det bästa filräddningsprogrammet vinner 100 stycken C60-kassetter av fabrikat ACME. Programmet behöver bara fungera för disk eller kassett.

Ja, det var allt för den här gången. Skicka gärna in frågor. Ett tips: Fråga gärna vad det finns för list- och kopieringsskydd till ABC80/800.

Det var allt från nästa gång, vi skrivs.

The Computer Phantome

ABC klubben har fått nytt kansli.

I slutet av förra året träffade ABC-klubben avtal med ett företag om att sköta vissa kanslifunktioner åt ABC-klubben. Där tjänstgör sedan början av december Berit Gustavii på vanlig kontorstid som kanslist åt klubben. I hennes uppgifter ingår bl a att ta emot och fördela inkommande post, svara i telefon, effektuera beställningar och sända ut material till nya medlemmar, handha medlemsregistret och hjälpa till med bokföringen. Tills vidare kommer Berit att arbeta drygt halvtid med uppgifter för klubben.

Med det stora antalet medlemmar som ABC-klubben numera har - f n över 4000 och antalet tycks fortsätta att öka i oförändrad takt - är det inte längre möjligt att sköta klubbadministrationen på enbart ideell basis, i varje fall inte om klubben skall kunna svara upp till de berättigade krav på service som medlemmarna ställer. Det här innebär naturligtvis en ökad belastning på ABC-klubbens ekonomi och har påverkat budgeten för 1984.

Innan vi kom fram till denna lösning har vi försökt klara de löpande uppgifterna med provisoriska arrangemang. Det har tyvärr inte alltid fungerat tillfredsställande och många, särskilt medlemmar som kommit



Berit Gustavii

med i klubben under hösten har fått vänta orimligt lång tid på att få sitt material. Jag beklagar djupt detta och ber de som drabbats om en allvarligt menad ursäkt. Det är min förhoppning att det i fortsättningen skall fungera bättre.

ABC-klubbens postadress är oförändrad

Vidängsvägen 1, 161 33 Bromma. Kansliet har fått eget telefonnummer, 08-53 57 50. När Berit inte kan svara kopplar hon in en telefonsvarare som även kan ta emot meddelanden. På telefon 08-80 15 22 kommer liksom hittills att finnas en telefonsvarare med aktuell klubbinformation. Denna telefonsvarare kan numera inte ta emot meddelanden. Telefon nr 08-80 15 23 går liksom hittills till monitorn i Alvik. Den planerade extra ingående linjen till den nya monitorn kommer att ha telefonnummer 08-80 15 26. Om man vill nå en person som tillfälligt vistas i klubblokalen kan man ringa 08-80 17 25, i normala fall är det dock ingen som svarar på detta nummer.

Beställning av samlingsnummer och övriga publikationer skall i första hand göras genom att använda postgirokonto nr 62 93 00-5, ABC-klubben Publikationer. Ange på talongen vad du vill ha och betala in det framräknade beloppet - observera att vissa priser justerades i början av februari. Medlemsavgifter skall betalas till postgirokonto nr 15 33 36-3, ABC-klubben. Betalning av körningar på Q-Zentralen görs till postgirokonto nr 43 51 74-8, ABC-klubben Q-Zentralen.

Gunnar Tidner

HISTORIK.

ABC80

Historien om ABC80 började med att Scandia Metric i februari 1978 tog kontakt med Luxor för tillverkning av bildskärmar. Scandia Metric visade då upp en mikrodator, TRS-80. Man försökte då få Luxor intresserad av att sälja en sådan dator. Vid denna tidpunkt var mikrodatorn inte alls känd i Sverige.

Luxors dåvarande ledning var inte intresserad av mikrodatoriden. Man tog dock kontakt med Gunnar Markesjö på KTH. Gunnar Markesjö var full av entusiasm inför mikrodatoriden. Efter nya diskussioner mellan Luxor och Scandia Metric togs frågan upp om de nödvändiga kunskaperna för att konstruera en dator.

Dessa kunskaper fanns då inte inom Luxor. Scandia Metric föreslog att kontakt skulle tas med Lasse Karlsson och hans företag DIAB. DIAB och Lasse Karlsson tog hand om utvecklingen. Under projektet gick utvecklingen av minnen så fort att DIAB på eget bevåg utrustade datorn med större minne.

För att få projektet i hamn beställde sedan Scandia Metric 3000 maskiner, utan att datorn var konstruerad! Denna beställningen kom att utgöra den ekonomiska förutsättningen för projektet. Vid midsommar var en prototyp klar. Då trodde dock alla på Luxor, utom utvecklingschefen Bengt Lönnqvist, att projektet skulle misslyckas.

Mikrodatorn, som nu hade döpts till

ABC80, pressvisades i augusti 1978. Då var det en fullständig nyhet att ett svenskt företag hade utvecklat en mikrodator, nästan inga kände till projektet i förväg. Det blev succe direkt. Datorn var klar för tillverkning i november samma år. Dock kunde det inte ske eftersom TEXAS Instruments inte kunde leverera teckengeneratoren i tid. Datorer tillverkades ändå, för senare montering av teckengeneratoren.

En stor roll för tillkomsten av Sveriges första mikrodator har Gunnar Markesjö/KTH, Lars Karlsson/DIAB, Bengt Lönnqvist/Luxor och Karl-Johan Börjesson/Scandia Metric spelat. Scandia Metric har dock senare lämnat samarbetet kring ABC-datorerna.

I januari 1979 kom Luxorkraschen och Luxor inställde betalningarna. Detta förorsakade givetvis problem för mikrodatorutvecklingen. Trots allt kom sedan tillverkningen igång.

Luxor började senare att själv tillverka flexskiveminnen. En ny mikrodatorserie presenterades våren 1981 under namnet ABC800. Våren 1983 kom ABC802 och ABC806.

I januari 1984 meddelade industridepartementet att man hade sålt Luxor till Nokia i Finland. ABC-klubben ser med spänning fram emot vad detta kan innebära för utvecklingen av Luxor mikrodatorer. Nu under 1984 väntar vi på nya kraftfulla 16 bitars mikrodatorer från det svenska datorföretaget i Motala.

ABC-klubben

Diskussioner om att bilda en användarklubb kom fram i början på 1980. Detta understöddes av Luxor och Scandia Metric, vilket ledde till några möten i början på 1980. ABC-klubben bildades den 24 januari 1980 vid ett möte på KTH, där cirka 150 personer av de dåvarande 500 medlemmarna var närvarade. Sedan dess har verksamheten vuxit enormt.

Nu finns det över 4000 medlemmar! Medlemsutvecklingen har varit explosionsartad. Detta har inneburit att ABC-klubben numera måste anlita betald kontorshjälp för att få administrationen att fungera. Klubben är helt enkelt för stor för att den helt skall kunna drivas med ideellt arbete.

Bo Kullmar <1789>

Kort och Brett om

DEC-10.

Här kommer nu ytterligare 16 sidor av skriften Kort och Brett om DEC-10.

Skriften kommer att publiceras efter hand i mån om plats.

Den som vill ha tillgång till skriften i original kan beställa den direkt från QZ, 08-67 92 80. Tyvärr kan man inte beställa den genom ABC-klubben eller Q-Zentralen.

Att sända meddelanden

Meddelanden på högst en rad kan man sända från en ansluten terminal till en annan genom kommandot:

```
.SEND nnn meddelandetext
```

där nnn är numret (TTYnn) på den terminal meddelandet skall sändas till. Om mottagaren kan ta emot meddelandet återfår sändaren monitorpunkten omedelbart. Om mottagaren är upptagen, t ex av att köra ett program, får sändaren meddelandet "BUSY".

Meddelande till operatören sänder man (då Galaxy V4.1 eller senare är i drift) med:

```
.PLEASE
Enter text, terminate with Altmode or ^Z
Nu kan man skriva ett meddelande till operatören.
Obs att meddelandet kan vara flera rader långt.<ESC>
```

Om man istället avslutar meddelandet med vanligt RETURN eller ^Z så skickas meddelande till operatören, varefter PLEASE väntar på svar. Denna facilitet används ibland från batch-jobb, där <ESC> för övrigt skrivs ^A (se kapitlet om batchhantering). Ex:

```
.PLEASE Var vänlig skicka listan till A-enheten^A
```

Genom monitorkommandot TERMINAL GAG kan man hindra meddelanden från andra användare att nå fram om man inte är på monitornivå. Observera dock att inte heller förmodat viktiga meddelanden från operatören kommer fram då.

TERMINAL GAG upphävs av kommandot TERMINAL NO GAG.

Här man information eller klagomål man vill framföra till OZ kan man använda programmet GRIPE. Exempel:

```
.R GRIPE
Yes ? (Press ESCape key when you are through)
Meddelandetext, som kan omfatta flera rader.
kan omfatta flera rader...
Avslutas med <ESC>
Thank you.
```

Dessa meddelanden sparas i en särskild fil och läses av OZ-personal följande arbetsdag.

Itall man använder KOM kan man alternativt skicka brev till KOM-möbet "Gripe i KOM". Vidare information om KOM fås via HELP KOM. Dessutom finns ett MAIL-system med vilket man kan sända "brev" till DEC-10-användare via skivminnesfiler. HELP MAIL ger vidare information om MAIL.

Radskrivarutskrift

Till DEC-10 är kopplad en central radskrivare, utrustad med både versala och kemena typer, på vilken man kan få sina listor utskrivna.

Till varje PPN eller varje projektnummer kan man sätta en sändningsadress till vilken alla listor skickas. Om man inte satt någon sändningsadress måste man själv hämta listorna vid OZ:s kundmottagning. Gör "HELP SEND" för mer hjälp.

Filer lägs i kön till radskrivaren genom monitorkommandot:

```
.PRINT filspecifikation
```

Härvid kommer de uppräknade filerna så småningom att skrivas ut på radskrivaren försedda med sidnummer. Om man inte önskar dessa sidnummer, t ex om sådana redan finns i filen, ger man i stället kommandot:

```
.PRINT/NOHEAD filspec
```

Man bör observera att sidnumreringen underlättar kontrollen av att man verkligen fått ut alla sidor och att OZ inte kan svara för bortkomna sidor i onummerade listor.

I FORTRAN utnyttjas ofta första tecknet i varje rad som styrtecken för att ange eventuell rad- och sidframmatning. Om man vill att första tecknet i varje rad skall tolkas som styrtecken på FORTRAN-vis måste man använda kommandot:

```
.PRINT/FILE:FORTRAN filspecifikation
```

Filer med extension DAT tolkas vid radskrivarutskrift som FORTRANs datafiler med styrtecken i kolumn ett. Om man inte skrivit sådana styrtecken måste /FILE:ASCII anges

```
.PRINT/FILE:ASCII KALLE.DAT
```

Om man vill ha flera kopior av en utskrift, säg 5,0, ger man kommandot:

```
.PRINT/COPIES:7 filspec
```

Filer som endast är avsedda att skrivas ut på radskrivaren har ofta extension LST.

Antag att man skapat filen KALLE.LST. För att skriva ut denna på radskrivaren räcker kommandot:

```
.PRINT KALLE
```

När filer med extension LST har skrivits ut tas de automatiskt bort av systemet.

Man kan när som helst få reda på hur många radskrivarijobb som ligger i kön genom att ge ett PRINT-kommando utan argument. Härigenom kan man se om man fått sina filer utskrivna eller inte.

Satsvisa jobb

DEC-10 kommer bäst till sin rätt vid dialoganvändning, men även satsvisa jobb ("batch") kan behandlas. Satsvisa jobb initieras vanligen med SUBMIT-kommandon.

Principen för SUBMIT-kommandot är enkel. Man tillverkar en fil med alla de komandon och indata som skulle krävas för att utföra arbetet i dialog. Därefter ger man kommandot:

```
.SUBMIT filspecifikation
```

Kösystemet loggar då in ett nytt jobb, utför alla komandon i filen, bokför alla åtgärder och resultat i en loggfil, samt loggar ut jobbet när allt är klart.

Filer som innehåller komandon ansedda att exekveras som satsvisa jobb kallas "kontrollfiler" och har vanligen extension CTL ("control"). Om en kontrollfil har extension CTL behöver man inte ange denna vid SUBMIT-kommandot.

Innehållet i en kontrollfil kan vara av två slag:

1) Monitorkomandon. I detta fall skall raden inledas med en punkt, symboliserande monitorpunkten.

2) Indata till program. Raden skall inledas med en asterisk.

Exempel: Programmet BTEST skall köras som satsvist jobb. BTEST väntar på två heltal som första indata och därefter ett reellt tal. Man skapar kontrollfilen BTEST.CTL som innehåller de tre raderna

```
.RUN TEST
#23 57
#1.4142
```

Jobbet körs genom kommandot:

```
.SUBMIT BTEST
```

Genom att ge ett SUBMIT-kommando utan argument får man ut en utskrift på terminalen av kösituationen. Här kan man se vilka jobb som finns i kön, om de har startat eller när de kommer att starta, samt när de är klara (försvinner då ur kölistan).

När jobbet är klart kan man läsa resultaten i loggfilen, som har samma namn som kontrollfilen, men extensionen LOG. (BTEST.LOG i ovanstående exempel). Loggfilen skrivs även ut automatiskt på radskrivaren. Detta kan undertryckas genom att man fogar "/OUTPUT:NOLOG" till SUBMIT-kommandot:

```
.SUBMIT BTEST/OUTPUT:NOLOG
```

Om man vill att ett jobb inte skall starta förrän efter ett visst klockslag (för att utnyttja lägre taxor), såg halv elva på kvällen, kan man ge kommandot:

```
.SUBMIT BTEST/AFTER:22:30
```

Vill man dessutom försäkra sig om att jobbet inte startar efter ett visst klockslag, såg sex på morgonen efter, kan man foga "/DEADLINE:6:00" till kommandot ovan.

Normalt tillåts köade jobb att exekvera i högst fem minuter maskintid. Denna gräns kan ändras för ett jobb till T timmar, W minuter och S sekunder genom kommandot:

```
.SUBMIT BTEST/TIME:T:W:S
```

/TIME och /AFTER kan anges i samma SUBMIT-kommando. Om man t ex har ett tungt jobb som kräver en halv timmes maskintid kan man med fördel köra det på natten (kl 1) genom kommandot:

```
.SUBMIT BTEST/TIME:0:30:0/AFTER:1:0/DEADLINE:6:00
```

Gör även HELP BATCH för information!

Rader skrivs om (ersätts) genom R-kommandot ("Replace"), vilket har samma effekt som ett D-kommando följt av ett I-kommando.

Rspec

T ex ersätter man rad 400 genom

```
SOS>P400
00400 Här skriver man in den nya rad 400
```

Delar av en fil skrivs ut med P-kommandot ("Print"), som har formen

Prspec

Exempel:

```
Här aktuell sida skrivs ut genom
```

P/.

Raderna 600 till sidans slut skrivs ut genom

```
P600:*
```

Aktuell rad skrivs ut genom

P.

För att skriva ut föregående rad eller efterföljande rad kan man använda kommandona <ESC> respektive <LF>.

4. Avslutning av redigering

SOS har många fler funktioner, varav en del är beskrivna i efterföljande avsnitt, men redan de som nu beskrivits torde vara tillräckligt för att skapa och modifiera filer. Vi vill därför redan här beskriva hur redigeringen avslutas.

När man modifierar en fil med SOS, så arbetar man på en kopia av filen. Det är först när redigeringen avslutas som det gamla originalet försvinner och kopian blir det nya originalet.

Först ett säkerhetskommando. Det kan inträffa att förbindelsen med DEC-10 bryts eller att DEC-10 stannar. Har man då suttit en längre tid och redigerat, riskerar man att arbetet har varit förgävs och måste göras om, eftersom man i SOS endast arbetar på en kopia av originalet. För att undvika detta kan man när som helst ge W-kommandot ("save World")

```
SOS>W
```

```
DSKD:NAME.EXT
```

som innebär att alla gjorda modifieringar överförs från kopian till originalet. Ett W-kommando var tionde minut rekommenderas.

Introduktion till SOS

1. Allmänt

SOS kan användas för att skapa nya textfiler eller redigera redan existerande textfiler.

Här nedan beskrivs endast de elementäraste funktionerna hos SOS. För utförligare beskrivning hänvisas till SOS-handboken som kan beställas från OZ. En förteckning över SOS-kommandon och deras format fås även genom monitorkommandot HELP SOS. Se även Appendix 4.

Allt redigera med SOS innebär att man arbetar med numrerade rader. Man kan sätta in nya rader, ta bort rader, kopiera rader (även från annan fil), flytta rader eller ändra innehåll i en rad. Man kan även söka efter eller byta ut textföljder varhelst de förekommer i en fil.

SOS startas genom kommandot:

```
.sos name.ext
```

Om NAME.EXT utelämnas uppfattar SOS det som att man vill redigera den fil man senast skapat eller redigerat med SOS under pågående terminalsession. Exempel:

```
.sos
Edit: DSKB:NAME.EXT
SOS>
```

Om filen NAME.EXT redan existerar svarar SOS med "SOS>", som indikerar att man kan ge kommandon till SOS för att skriva ut eller modifiera delar av filen. Närhelst SOS skriver ut "SOS>" och väntar sägs man vara i SOS kommandomod.

Om filen NAME.EXT inte existerar svarar SOS med ett radnummer som indikerar att SOS är beredd att att ta emot löpande text för en ny fil, som kommer att få namnet NAME.EXT. Exempel:

```
.sos finns.ej
Input: FINNS.EJ
00100 Nu skriver jag in text i en ny fil.
00200
```

När SOS skriver ut radnummer och väntar sägs man vara i insättningsmod. För att övergå från insättningsmod till kommandomod ger man SOS-kommandot <ESC> (tangenten märkt "ESCAPE", "ESC", "ALT" eller "ALTMOD") nedtryckes.

Radnumren i SOS startar normalt med 100 och har inkrementet (=steget) 100, dvs efterföljande rader får numren 200, 300, 400, etc. När vi här talar om exempelvis rad 700, menar vi alltid raden med SOS-numret 700, inte den 700:de raden. Anledningen till denna numrering är att det skall finnas gott om "lediga" nummer mellan raderna för att sätta in nya rader på. Användaren kan själv välja nytt inkrement eller numrera om filen.

Observera att SOS radnummer ingenting har att göra med FORTRAN:s satsnummer. Man kan ta bort radnumren då man lämnar SOS, men standard är att de behålls. De flesta kompilatorer ignorerar SOS radnummer då de förekommer. Däremot kan de användas vid avslutning och inkörning av program i SIMULA & FORTRAN bl.a.

Man kan även indela en fil i s k "sidor", där varje sida har sin egen numrering. Man kan när som helst sätta in en ny sidamarkering mellan två godtyckliga rader. Man kan ta bort sidamarkeringar, men numreringen räkar då ofta i ordning, vilket man avhjälpas genom omnumrering av sidan.

2. Radspecifikation

Det centrala begreppet i SOS kallas radspecifikation (ibland radgränsuttryck) och förkortas här **rspec**.

Den generella formen är

R1/S1:R2/S2

Vilket utläses "från och med rad R1 på sidan S1 till och med rad R2 på sidan S2". Ett par enklare former är

R/S "rad R på sidan S"

R/SIN "N rader från och med rad R på sidan S"

Ibland kan man utelämma delar av en radspecifikation. Med uttrycken "aktuell sida" och "aktuell rad" menar vi den sida respektive rad, som man senast refererade till, dvs skrev ut, modifierade eller som på annat sätt förekom i den senaste gjorda radspecifikationen. Utelämnas sidangivelse tolkas detta som aktuell sida. Exempel:

R "rad R på aktuell sida"

R1:R2 "rad R1 t o m rad R2 på aktuell sida"

Följande symboler har speciell innebörd i radgränsuttryck:

• aktuell rad eller sida. Exempelvis "+1" anger sidan efter aktuell sida, "-3" anger raden tre rader före aktuell rad. Avser ett kommando hela den aktuella sidan, ger man "/" som radspecifikation.

• första raden på en sida eller första sidan i filen.

• sista raden på sidan eller sista sidan i filen. T ex kan hela sidan sju refereras till genom "+7.*".

EF betyder hela filen (Entire File)

3. Insättning, borttagning och utskrift

Låt oss nu antaga att man har startat SOS för att skapa en ny fil. Om filen inte fanns tidigare bjuder SOS direkt på radnumret 00100 och det är bara att skriva in text. Varje gång man avslutar en rad genom att trycka på <CR> får man en ny rad med nytt radnummer med inkrementet 100.

Då man tycker sig ha skrivit nog med rader eller vill gå tillbaka för att ändra något trycker man på <ESC> och övergår då till SOS kommandomod, som indikeras av texten "SOS" (kallas ledtext).

Om man vill sätta in en ny rad mellan raderna 500 och 600 ger man I-kommandot ("insert")

SOS>1500

00550 ny text

SOS beräknar alltså själv ett radnummer lika med medelvärdet av angiven rad och den efterföljande i detta fall alltså 550. Då raden är skriven avslutar man med <CR>, varvid man återkommer till SOS kommandomod. Man kan även explicit ange ett radnummer som inte tidigare fanns, t ex

SOS>1567

00567 nyare text

varmed rad 567 sätts in.

Man kanske har behov av att sätta in flera rader mellan två befintliga. Vill man sätta in raderna 510, 520 t o m 590 kan man ge kommandot:

SOS>1500;10 (obs semikolon)

varvid SOS genererar radnummer med inkrementet 10, fr o m 510 och så länge man inte hittar en redan existerande rad. Kommandomod återfås genom <ESC> eller när rad 590 är klar.

Rader tas bort med D-kommandot ("Delete")

Drspec

T ex kan raderna 400, 900 t o m 1400, 2100 och 5 rader framåt tas bort genom

SOS>d400

1 line (00400/1) deleted

SOS>d900:1400

6 lines (00900/1:01400) deleted

SOS>d2100:5

5 lines (02100/1:02500) deleted

SOS>

SOS bekräftar härvid vilka rader som tagits bort. Efter borttagning är motsvarande radnummer lediga att sätta in ny text på.

7. Kort om övriga SOS-kommandon

C-kommandot ("Copy") kopierar rader från ett ställe i en fil till ett annat. Exempel: kopiera till rad 700 raderna 300:500

C700,300:500

C-kommandot kan även användas för att kopiera rader från en annan fil: C700=filspec/S där /S står för "select".

T-kommandot ("Transfer") förflyttar rader från ett ställe till ett annat. T fungerar som C, fast tar dessutom bort de rader man kopierat från.

X-kommandot ("extend") ger möjlighet att utvidga en rad, dvs fortsätta skriva på en tidigare skapad rad.

A-kommandot ("Alter") ger möjlighet att ändra inne på en rad. Detta kommando har flera "subkommandon" för att ta bort, sätta in, skriva ut tecken etc.

J-kommandot ("Join") sår ihop två konsekutiva rader till en.

G-kommandot ("Go") innebär att redigeringen avslutas på samma sätt som vid E-kommandot och att därefter det senast givna EXECUTE-, COMPILE- eller LOAD-kommandot utförs.

Flera av kontrolltecknen (t ex CTRL-C, CTRL-Z, <ESC>) kan inte skrivas in med SOS som vanliga tecken. Det finns dock metoder för att få in godtyckliga tecken med hjälp av SOS. Se SOS-handboken för närmare information härom.

SOS erbjuder även möjligheter att få en text radjusterad, t ex med rak högermarginal och given radbredd. I SOS-handboken är dessa kommandon beskrivna under nyckelordet "justify".

Gör "HELP SOS" för kort presentation av alla kommandon eller beställ SOS-manualen från QZ. Se även Appendix 4 för en sammanställning av SOS-kommandon och deras format.

Man kan ge ett stort antal s k "switchar" i startkommandot, t.ex.

.sos kort.brt/UNSEQUENCE

(som medför att radnummer tas bort, jfr ES). Denna switch kan man lägga i en speciell fil med namnet SWITCH.INI. Man tillverkar en fil med SOS s:ativ:

```
.sos switch.ini
Input: SWITCH.INI
00100 SOS /UNSEQUENCE<ESC>
SOS>es
```

avsluta med <ESC>
detta vet SOS inte än

I fortsättningen kommer SOS att läsa SWITCH.INI varje gång du startar en editering.

Ytterligare ett säkerhetskommando. Man kan på olika sätt råka förstöra en fil vid redigering, exempelvis ta bort alla rader eller införa alldeles felaktiga modifieringar. Här kommer finessen med att arbeta på en kopia av originalet väl till pass. Man ger kommandot:

BQ

som innebär att redigeringen avslutas utan att originalet ändras ("End Quit"). Inga redigeringar överförs till originalet.

Det enklaste sättet att avsluta en redigering är att ge E-kommandot ("End")

E

varvid arbetskopian blir det nya originalet och alla radnummer bibehålls i filen. Dessutom sparas det gamla originalet, vars extension ändras så att första bokstaven blir "Q". Om man t ex avslutar redigeringen av FOURIE.SIV med E-kommandot så kommer det gamla originalet att finnas kvar under namnet FOURIE.QIV. På så vis har man alltid den närmast föregående versionen intakt.

Några andra varianter av E-kommandot:

ES

tag bort radnummer

EQ

behåll ingen skyddskopia (filnamn.oxx)

5. Sidor och sidmarkeringsnumrering

Antag att man skrivit raderna 100-3000 och vill börja på ny sida här efter. I stället för att skriva in text på rad 3100 trycker man på <ESC> för att komma i kommandomod och ger därefter M-kommandot ("Mark")

M3100

vilket innebär att en sidmarkering sätts in efter rad 3000. Om man vill att radnumreringen på den nya sidan skall börja med 3100 behöver man bara ge ett I-kommando utan argument för att komma i insättningsmod. Vill man däremot börja om med radnummer 100 måste man ge kommandot I100.

Skulle en sida ha blivit för stor kan man sätta in en sidamarkering genom kommandot:

Mrspec

där rspec definierar den rad som skall bli den första på den nya sidan.

Observera att då man skjuter in en ny sidamarkering mellan två gamla kommer de efterföljande sidorna få ökat sidnummer.

En sidamarkering tas bort genom K-kommandot ("Kill page mark")

K/sida

och innebär att sidorna "sida-1" och "sida" sås ihop. Alla sidamarkeringar i en fil kan tas bort genom upprepade kommandon K/2 eller enklare ett enda K/#!/.

Då man tar bort sidamarkeringar råkar radnumreringen ofta i oordning, vilket SOS meddelar genom utskriften

```
%out of order
```

man bör då numrera om filen eller sidan, vilket görs med hjälp av N-kommandot ("re-Number")

Ninkr,rspec,startrad

där inkrr står för det önskade inkrementet (vanligen 100) och rspec talar om vilka delar av filen man vill ha omnumrerad. Startrad specificerar vilket nummer man vill att sekvensen ska börja med.

Om man vill ha hela filen omnumrerad med inkrementet 50 räcker det med kommandot:

N50

eller

N50,50

varvid raderna får nummer 50, 100, 150 osv.

Vill man endast ha sidan 7 omnumrerad med inkrementet 10 ger man kommandot:

N10,/7

6. Textsökning och utbyte av text

SOS kan leta reda på teckenföljder och skriva ut den rad de förekommer på. Den enklaste formen är F-kommandot ("Find")

Fabc...<ESC>rspec

och innebär att SOS letar efter texten "abc..." inom det område som definieras av rspec. Om rspec utelämnas sker sökningen från aktuell rad och framåt. Om teckenföljden inte återfinnes inom rivet område får man meddelandet

```
%Search fails
```

Om textföljden hittas skrivs hela motsvarande rad ut. Denna rad är i detta läge aktuell rad. Ett nytt F-kommando utan argument ger ny sökning från raden efter aktuell rad och framåt.

Exempel:

Man önskar ta bort alla rader som innehåller texten "blurr".

```
Fblurr<ESC>"/1:#!/
.....blurr.....
D.
F
.blurr.....
D.
F
%Search fails
(SOS har hittat en dylik rad)
(tar bort aktuell rad)
(fortsätt sökningen)
(nästa napp)
(tar bort raden)
(fortsätt sökningen)
(fanns inget mer "blurr")
```

Man kan även automatiskt leta upp och byta ut teckenföljder. Enklaste varianten av S-kommandot ("Substitute") är

Sgamla<ESC>nya<ESC>rspec

vilket innebär att texten "gamla" byts ut mot "nya" överallt inom rspec.

Exempel: Byt ut Karlsson mot Olsson på aktuell sida.

SOS>Skarlsson<ESC>Olsson<ESC>/.

Observera. Om man skriver fel i F- eller S-kommandona, så kan man inte backa (raderna) över ett <ESC>. För att inibera F- och S-kommandon använder man två CTRL-G. Oförsiktigt använda S-kommandon är utmärkta till att fördräva filer och skapa problem. För att inte göra felaktiga substitutioner kan man lägga till en substitute-option "D" som i decide. Exempel:

```
SOS>Smat<ESC>föda<ESC>,D      :byt "mat" mot "föda"
00H60 och vi fick mycket god föda
SOS,D>h Substitute Decide
```

Type one of the following letters:

<space> To accept this line with the substitutions

<delete> To restore original line and continue

Q To restore original line and end search

E Same as Q

A To go into Alter mode on this line

G To accept this line and all the rest without asking

SOS,D>qNo changed

Appendix 3. Monitorkommandon

För monitorkommandon gäller att de får förkortas så mycket att de fortfarande är entydiga. T ex kan LOGIN förkortas till LOG, men inte till LO, ty detta skulle även kunna tolkas som LOAD. Undantag finns dock från denna regel så att vanliga kommandon kan förkortas mer. T ex kan man förkorta DIRECT till DI trots att det blir konflikt med t ex DISMOUNT.

Kommandon och filspecifikationer kan ges med såväl versaler som gemena tecken. Tecknen A, Å och Ö betraktas som specialtecken, vilka inte är tillåtna i filspecifikationer.

Argumentlistan omfattar endast den vanligaste varianten. För flertalet kommandon kan den vara betydligt fylldare. För närmare beskrivning hänvisas till handboken "Operating System Commands".

Till alla kommandon finns hjälpfil. Gör "HELP kommando".

fys: fysiskt enhetsnamn t ex MTA; DTA; DSK; eller LPT:
log: logiskt enhetsnamn t ex KALLE; 10:
p:pn projekt,programrarnummer
filspec enhet:filnamn.extension&PP&skyddskod
lista en följd av filnamn med kommatecken mellan
namn filnamn oftast utan extension

Kommando

Förklaring

ASSIGN fys: log: skapar logiskt (synonymt) enhetsnamn
ATTACH jobn<p,pn> återknyter kontakt med jobb
BACKSPACE log:n FILES backar ett magnetband n st filer
BACKSPACE log:n RECORDS backar ett magnetband n st poster
CLOSE stäng alla öppna filer
COJOB log:fil=filspec arg1,... startar batch-jobb under MIC
COMPLETE lista kompillerar angivna källkodfiler
CONTINUE fortsätter avbruten exekvering
COPY till=från kopierar filer
CORE kontrollerar använd kårminnesstorlek
CPUNCH lista filer köas för hålkortsutmatning
CREF tillverkar korsreferenslistningar
D ändra angiven cell (Deposit) jmf E
DAYTIME ger klockslag och dagens datum
DDT startar avlusningsprogrammet DDT
DEASSIGN log: upphäver ASSIGN
DEBUG lista laddar och startar avlusningsprogram
DELETE lista tar bort uppräknade filer
DETACH avbryter kontakten med jobbet
DIRECT lista ger lista med filnamn
DISMOUNT log: avmonterar enheter, som t ex magnetband
DO filspec arg1,... utför kommandona i angiven fil (MIC)
DSK info om antalet lästa/skrivna block
E examina angiven cell (Examine) jmf D
EOF log: starta editering av fil
skriver End-Of-File-märkning på band

Appendix 1. Kort ordlista

Då de flesta handböcker rörande DEC-10 är skrivna på engelska ger vi här en kort ordlista med en förklaring av de vanligaste tekniska termerna.

allocate reservera (utrymme på yttre minne)
ASCII en vanlig teckenkod
attach återknyta kontakt med jobb
backup arkivkopia av filer
batch satsvis bearbetning
block 128 stycken 36-bitars ord
buffer minnesarea avsett för mellanlagring av data
bug lus, fel i program
carriage return motsvarar varnretur på en skrivmaskin
close stänga en fil
cluster den minsta enhet skivminne som kan reserveras
cluster size antal skivminnesblock i ett "cluster"
debug "avlus", rätta fel i datorprogram
density töthet för magnetband (ex 800, 1600, 6250)
DEC-tape DEC-band (speciella små magnetband)
delete ta bort (en fil)
detach avbryta kontakt med jobb
device enhet
directory filregister
disk skivminne
dump utskrift av en fil eller del av minne
editor i format som inte skiljer mellan program, data, bindr- eller källkod
enter textredigeringsprogram (t ex SOS, TECO)
ersatz device skapa en fil att skriva på
extension skivminnesarea med specifikt namn, ex PUB: namntillägg till filnamn (ex .SYM)
file fil
file form feed tecken som ger ny sida på skrivare
feed kort meddelande till systempersonalen
grape tecken som ger ny rad
link länka ihop huvudprogram och subrutiner
log file logg över satsvisa jobs exekvering
lookup leta rätt på en fil för läsning
map en lista över moduler, rutiner och deras minnesadresser i ett program
module den minsta programenhet som kan länkas ihop med andra moduler till ett program
monitor operativsystem
open öppna en datakanal till en fysisk enhet
overlay överlagring av programsegment
password lösenord
PPN projekt-programrarnummer
printer radskrivare
protection skyddskod (för skivminnesfiler)

quota	det största antal skrivminnesblock som en användare får ha
release	stänga en datakanal
rename	döpa om (en fil)
RIB	Retrieval Information Block, speciellt skrivminnesblock för varje fil. Innehåller information till monitorn om filens status och fysiska adress.
runtime system (RTS)	programspråk behöver ofta en samling rutiner för att klara kontakten med operativsystemet, in- och utmatning mm
slash	tecknet "/"
spooled device	en enhet som endast privilegierade jobb har tillgång till, medan användare kan lägga jobb i kö för att monitorn skall utföra jobb på enheten. Exempel: radskrivare, kortläsare. skrivminnesstruktur, en eller flera skrivminnesenheter med ett gemensamt namn, t ex DSKB, DSKD, DSKF
swap	flytta program mellan primär- och sekundärminne
switch	parameter till kommando. Anges ofta i formatet "kommando/switch:värde", t ex "PRINT/COPIES:7"
submt	köra jobb för satsvis bearbetning
TTY, teletype	terminal med tangentbord
UFD	till varje PPN hör en fil kallad "User File Directory" som innehåller namn på alla skrivminnesfiler som hör till detta PPN
UUO	subrutin, tillhandahållen av monitorn ett för användaren automatiskt system som medger att ett exekverande program endast delvis behöver finnas i minnet i varje ögonblick
virtual memory	tecknet "m" i filspezifikation står för godtyckligt värde (namn eller PPN)
Wild card	tecknet "?" står för godtyckligt tecken
Wild character	

Appendix 2. Vanliga extensioner

Extension är det namntillägg som filer kan ha och som oftast används för att ange vilken sorts innehåll filerna har.

Nedan följer en beskrivning av de vanligaste extensionernas användning. Det brukar underlätta att i största möjliga mån följa dessa konventioner. Filer med asteriskmärkt extension bör inte skrivas ut på terminal eller radskrivare (helt eller delvis binära). Gör HELP EXTENT för mer information.

Extension	Binär?	Betydelse
ALG		ALGOL källkod
ATR	*	SIMULA attributfil (skapas vid kompilering)
BAK		TECO:s backup-fil
BAS		BASIC källkod
BLI		BLISS-10 källkod
CBL		COBOL källkod
CMD		kommandofil
CRF	*	korsreferensfiler, indata till programmet CREF
CTL		kontrollfil för satsvist jobb
DAE	*	dumpriler skapade av programmet DAEWON
DAT		FORTRAN datafil (styrtecken i kolumn ett)
DIR		directory- (filregister) fil
DMx		filer för databssystemet 1022
DOC		dokumentationsfil
ERR		fil med felmeddelanden
EXE	*	primärminnesbild, direkt exekverbar
F4		FORTRAN-40 källkod (sammat FORTRAN-version)
FOR		FORTRAN-10 källkod
HLP		hjälpfil
INI		initieringsfil
LIB	*	med LIBMAN ihop-packade filer
LOG		loggfli skapad av satsvist jobb
LST		fil avsett för radskrivaren
MAC		MACRO-10 källkod (assembler)
MAN		manual-fil
MAP		LINK-10 map
MIC		fil för MIC-kommandon (se kapitel om MIC)
PAS		Pascal källkod
Oxx		SOS backup-filer
QTR	*	backup av SIMULA's .ATR-fil
REL	*	relokerbar kod (kompilerad källkod)
RNX	*	RUNOFF-fil (för speciell textredigering)
SAV	*	primärminnesbild (kompakterad .EXE-fil)
SFD	*	Sub File Directory (se appendix 9)
SIM	*	SIMULA källkod
SYS	*	speciella systemfiler
TMP		temporära filer
TXT		text-fil
UFD	*	User File Directory (se kapitel om filhantering)

ABC-DATORER - NYTT HJÄLPMEDEL FÖR

SLÄKTFORSKARNA.

Släktforskaren har en stor mängd uppgifter att söka i och ta hand om i sin släktforskning. Han letar efter försvunnen släkt i kyrkböcker, domböcker, bouppteckningar mm och om släkten emigrerat till t e USA bl a i deras folkräkningshandlingar. Endast till en mindre del av dessa handlingar finns personregister. Ett stort behov föreligger därför att ta fram personregister till ytterligare handlingar alternativt ordna databaser med sökmöjlighet. De uppgifter släktforskaren funnit om den egna släkten behöver vidare tas hand om, ordnas på lämpligt sätt och presenteras - gärna grafiskt - t e som an- eller släkttavlor. Dessa uppgifter är sådana som lämpar sig väl för databehandling. En särskild förening - DIS - bildades för några år sedan för att undersöka möjligheterna att använda datorer i släktforskningen. I föreningen valde man att arbeta med ABC 80 och har idag utvecklat program för registerframtagning och för att hålla reda på uppgifterna om den egna släkten och presentera dem på olika sätt. I dag finns en serie program som täcker släktforskningens grundläggande behov och problemet är närmast hur alla släktforskare skall få tillgång till ABC 80. Kanske kan ett samarbete med ABC-klubben ge något.

Valet av dator gjordes redan 1980. ABC 80 bedömdes då vara den bästa med hänsyn till pris, prestanda och förekomst. Av betydelse blev också ABC-klubbens stöttning av denna dator. I dag håller vi fortfarande fast vid ABC 80 med 16 k arbetsminne som huvudalternativ även om programmen till del även finns på ABC 802.

De första programmen som togs fram gällde registrering av kyrkboksdata för registerframtagning. Då antalet uppgifter som skall matas in är stort var det viktigt att utforma programmen för att underlätta inmatningen. Detta skedde bl a genom att använda förkortningar vid inmatningen, så att antalet tangenttryckningar väsentligt kunde reduceras. Anpassning gjordes också till de olika typerna av kyrkböcker. I början gjordes enbart registreringen på ABC 80 och sorteringarna till register på stordator med telefonöverföring i 300 baud av uppgifterna till den senare (Det tog timmar att överföra). Registren togs ut på band och lämnades över till en Stockholmsfirma för mikrofilmframtagning. I dag finns program för sortering till register på ABC 80, vilket innebär att hela processen med slutresultat register på papper klaras av med ABC 80. Program finns också anpassade till registrering av domboksuppgifter, mantalslängder mm. Gjorda register har hittills gällt 1700-talsmaterial.

För att hålla reda på och presentera uppgifterna om egna släkten togs för ungefär ett år sedan fram ett programsystem kallat DISGEN. Ett program tar hand om inmatningen av uppgifter samt rättning. Av hanterings-skäl delas personerna upp i grupper - kallade flockar - om max 1000 personer. En sådan flock rymmer normalt på en diskett med "double density". Genom att ha fortsättning i en annan flock kan flera flockar samköras. Uppgifterna om varje person delas upp i notiser. Notiserna är dels av typ med fast indelning för datum, ort etc och dels av typ fri text. Primärt kan man ha ca 20 sådana notiser för en person men möjlighet finns att fortsätta på nytt nr och därigenom

få tillgång till utrymme för ytterligare 20 notiser etc. Ett program kontrollerar inmatade uppgifter så att inga bokstäver kommit in där det skall vara siffror, att antalet siffror i datum är rätt etc. Program finns sedan för att ta fram antavlor av olika typ och släkttavlor typ "ättlingarna till" av olika slag. Person- och ortregister kan också tas fram till an- och släktavlorna. Hela släktböcker kan tas fram med person- och ortregister. Med ett program kan man få tidsdiagram och grafiskt se när de olika personerna levde.

DISGEN-programmen kan ses som en för släktforskning specialiserad typ av ordbehandlings/utskrifts-program, som används för utskrift av an- och släkttavlor mm. De ger härutöver en ny möjlighet att hantera materialet. I stället för kanske en massa svår-lästa anteckningar får man ett ordentligt ordnat material från vilket utskrifter snabbt kan göras när så önskas. Det ger också nya möjligheter att hantera släktuppgifterna och lära känna sin släkt genom lättheten att ta fram och skriva ut antavlor till olika personer och släkttavlor till likaledes olika personer och med olika antal generationer medtagna.

Även om de för släktforskningen grundläggande behoven är täckta genom framtagna program kan förutses behov av en fortsatt programutveckling som både syftar till nya program och till den anpassning som den tekniska utvecklingen nödvändiggör. Vi kan t e ännu inte ställa frågan till datorn hur två utpekade personer är släkt med varandra, vilket var med som exempel på LISP-program i en av ABC-klubbens tidigare programkassetter. Det var då utan större praktiskt värde då minnet mycket fort blev otillräckligt. Kanske skall man fundera på en FORTH-version av programmen för att minska minnesbehovet och förbättra möjligheten att flytta programmen till andra datorer? Redan överflyttningen till ABC 802 vållar svårigheter vid förfrågningar och uppdatering då föreningen endast har ABC 80 och ingen ABC 802. Överföring till CPM har diskuterats men mer har det inte blivit.

Programsystemet DISGEN för att hålla reda på uppgifterna om den egna släkten börjar nu så smått bli populärt bland släktforskarna. Föret har varit trögt, då många släktforskare har en starkt humanistisk laggnings och är mycket främmande för de tekniska. Men exempel finns på att släktforskare som aldrig tidigare sett en dator ganska snabbt lärt sig hantera den i sitt arbete. Själv har jag funnit den vara ett bra hjälpmedel både för registerframtagning och för hantering av de egna insamlade släktuppgifterna.

Föreningen DIS har nu närmare 400 medlemmar, varav bortåt 100 i Stockholms-trakten. Klubblokal finns i Linköping med en komplett utrustning ABC 80 med diskett-enhet och skrivare. Medlemsbladet Släkt-Forskar-Nytt" ges ut 4 gånger om året. Föreningens ordf är Lars Blomberg som haft huvudansvaret för programutvecklingen. Själv är jag sekreterare och för pennan. Föreningens adress är Hjortronvägen 89 59054 Sturefors. Jag brukar gå in i KOM då och då och kan nås där.

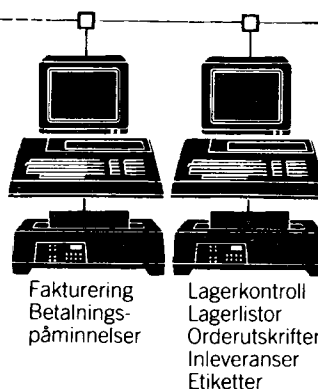
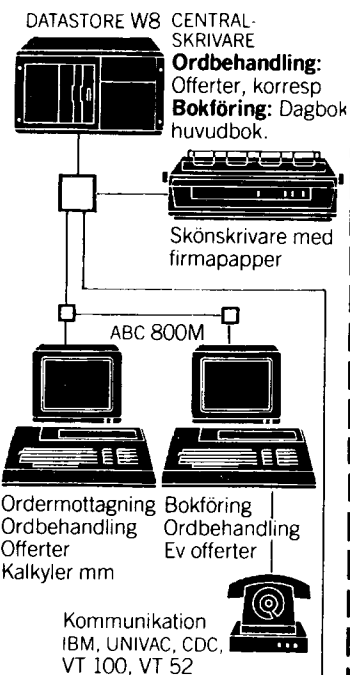
PO Bergman



ABC NET

MULTIUSER

EXEMPEL

Fleranvändarsystem.
Administration.

Fler exempel på fleranvändarsystem i vår specialbroschyr. Ring, skriv eller slink in!

T-D-X

SmåDatorer AB

T-D-X NORR: SOLLENTUNA-
VÄGEN 225, 191 23 SOLLENTUNA
T-D-X CITY: KUNGSGATAN 79,
KUNGSHOLMEN STOCKHOLM
GEMENSAMT TELNR 08-96 01 80

RYMDÄVENTYRARNÄ.

Här kommer ett förhoppningsvis litet annorlunda spelprogram. Det gäller inte att "skjuta fiender" utan bara att överleva en våldig rymdfärd.

Programmet simulerar en simulator... Skärmen blir en radarskärm, som visar hur det ser ut i rymden mellan basen (jorden) = vänsterkant och en rymdbas = högerkant. Farkosten skall ta sig från vänster till höger utan att fastna i något av de farliga svarta hålen (" ") eller förrädiska rymdgroparna.

Komplikationen är de märkliga rymdklumpar som dyker upp, ett slags meteoriter som dras mot rymdfarkosten. En krock blir ödesdiger. Antingen måste man rädda sig in på basen, dit klumparna inte kan ta sig in, eller också ordna så att de på sin farliga färd hamnar i ett svart hål.

Slutligen gäller det att ta sig hem innan syret är slut.

Farkosten styrs med tangenterna R T Y upptåt, V B N nedåt, R F V åt vänster, Y H N åt höger. G håller farkosten stilla.

Programmet ger en god överblick över olika programmeringstekniker. Hackern kan med lätthet fixa till ytterligare förvecklingar och bearbetningar.

Kommentarer till programlistan:
(För enkelhets skull hoppar jag över alla %-tecken)

De olika "pjäserna" på skärmen betecknas S\$(), där I är rymdfarkosten, P1-P9 är rymdklumparna, H1-H9 de svarta hålen och G1-G9 rymdgroparna.

Klumparna är t v tomma - de dyker upp efter hand. 200-240 gör i ordning spelplanen för start med hjälp av subrutinerna 1500, 1810 och 1820, där man slumpvis placerar in det bestämda antalet gropar och hål och till sist placerar spelaren i vänsterkant någonstans.

1000-1190 är huvudrutinen som genomlöps

för varje dragomgång. Det är spelarens drag och ev uppträdande klumpars drag. 1012 släpper in en ny klump då och då, ibland upptill och ibland nedtill (subrutinen 1600-1660).

1013 kollar syreåtgången och stoppar spelet efter tusen omgångar, ev efter varningstext.

1015 hoppar genast till slingans slut om det inte finns någon pjäs.

1020 avläser tangentbordet och förvandlar små bokstäver till stora. Det är spelarens drag som kommer här. För att kunna "flytta" pjäserna måste man sudda den gamla positionen och skriva dem i den nya. 1025 suddar, 1100 skriver.

Dessemellan undersöks vart den aktuella pjäsen skall flyttas. Spelaren flyttar på raderna 4000-4150, medan klumparna flyttas på 2000-2140. Det kanske tarvar några förklaringar.

För spelaren avläses de aktuella tangenterna, och sedan justeras värdena för rad och kolumn. Spelaren kan "smita över kanten" upptill och komma fram nedtill och tvärtom. Det kan däremot inte klumparna eftersom de alltid rör sig mot farkosten. De kan inte heller ta sig in i "boet" alltså i rymdbaserna (2135-7).

Men klumparna skall röra sig i riktning mot farkosten vilket ordnas i 2010-2130. Man tar reda på vem som har lägsta nummer på rad resp kolumn och justerar klumpens läge efter det.

När alla rörliga pjäser flyttats kollar man om det föreligger någon krock (1135 med hopp till 3000). 1137-40 ger beställd fördröjning (T=0 ger realtidseffekt, T=9 ger förflyttning endast vid knapptryck).

Nu till krockarna! På 3000-3060 undersöks systematiskt ifall två pjäser har samma koordinater. Tomma pjäser hoppas över. Vid

krock hoppar man till 3500 och där sker en mängd saker.

Om krocken skett med en grop är J mellan G1 och G9. Då sker för det mesta en sidförflyttning (3700) med ett visselljud. I 10% av fallen sker katastrof som vid en grop och man stannar i denna rutin. 3550 sänder ut ett väsljud som talar om att någon pjäs förstörts. 3600 nollställer pjäsen och 3605 flyttar bort den till ett hörn - annars skulle man få krocksignaler hela tiden. Medan ljudet pågår lyser en ruta upp. När det hela är över återställer man gropen/hålet.

Men om det är farkosten som krockat (3630) sker hopp till slutrutinen på 5000.

I samband med spelarens drag sker kontroll om farkosten nått fram till höger kant (basen). I så fall sker (5100-) signal, text, och vändning av tecknet. Samtidigt sätts en flagga Q=-1 för att markera att man passerat basen på sin hemfärd.

Har man kommit tillbaka till vänsterkant efter att ha nått högerkant sker en mindre "flaggning" och man får segertext.

Tar syret slut, stannar spelet, tillsammans med sorgetext.

Detta är ett långt program och kanske litet mödosamt att knappa in. Ta det i etapper och prova varje etapp för sig! Först t o m 250, inkl 1500-1540 och 1800-20. När det fungerar prova spelarens drag (4000-4150 tillsammans med bara de rader i huvudprogrammet som behövs: 1000-1130 minus 1012-3, 1090. Prova också in sluttexterna. Därpå stoppar man in klumparna (2000-2140), sedan krockrutinerna.

Lycka till med spelet! Det är inte så lätt att ta sig hem igen. Tur att man kan göra det i simulatoren!

Sven Wickberg

```

10 REM rymdresa 83 11 21 SW
11 REM (c) Sven Wickberg
12 REM 0750-50456
20 T=0 : REM tidsfaktor
30 ; CHR$(12%)'H RYMDRESA H' : ; ;
; 'Vill du ha information (N)'; : INPUT W$
40 IF W$='J' OR W$='j' GOSUB 7000
45 GOSUB 7310
90 RANDOMIZE
99 REM -----
100 REM dim och def
105 P1%=2% : P2%=10% : H1%=P2%+1% : H2%=H1%+30%
106 G1%=H2%+1% : G2%=G1%+10%
108 T%=11% : G%=5% : REM ljudet vid krock
110 DIM R$(G2%),K$(G2%),S$(G2%)-1%
120 S$(1%)='>' : REM spelaren själv
130 FOR I%=P1% TO P2%
135 S$(I%)='' : REM rymdKLUMP tom t v
140 NEXT I% : REM svarta hål
145 FOR I%=H1% TO H2%
150 S$(I%)='.' : REM svarta hål
155 NEXT I%
160 FOR I%=G1% TO G2%
165 S$(I%)='?' : REM rymdGROP
170 NEXT I%

```

```

190 REM -----
200 REM gör spelplan
205 ; CHR$(12%);
210 FOR P%=H1% TO G2%
220 GOSUB 1500
230 NEXT P%
240 P%=1% : GOSUB 1500
250 GET W$ : REM vänta på start
990 REM -----
1000 REM huvudprogram
1010 FOR I%=1% TO P2%
1012 IF RND<.01 GOSUB 1600 : REM rymdklump in
1013 S%=S%+1% : IF S%>500 GOSUB 9000 :
REM SYRET SLUT?
1015 IF S$(I%)='' THEN 1130 : REM d&d
1020 W%=INP(56%) AND 95%
1025 ; CUR(R$(I%),K$(I%))' ' : REM sudda
1030 IF I%=1% IF W%>13% GOSUB 4000 :
REM spelaren flyttar
1090 IF I%>1% W%=0% : GOSUB 2000 :
REM klump flyttar
1100 ; CUR(R$(I%),K$(I%))S$(I%);
1130 NEXT I%

```

```

1135 GOSUB 3000 : REM krock?
1137 IF T=9 THEN GET W$: GOTO 1190
1140 FOR I=1 TO T*200% : NEXT I
1190 GOTO 1000
1195 REM -----
1500 REM placera pjäser
1510 GOSUB 1810 : GOSUB 1820 : REM dra R,K
1520 IF P%=1% K%(P%)=0%
1530 : CUR(R%(P%),K%(P%))S$(P%);
1540 RETURN
1590 REM -----
1600 REM rymdklump in
1605 IF P9%>P2% THEN 1660
1610 P%=P1%+P9%
1620 P9%=P9%+1%
1630 S$(P%)='S'
1640 GOSUB 1820
1650 IF RND<.5 THEN R%(P%)=22%
1660 RETURN
1800 REM dra R,K-----
1810 R%(P%)=RND*20% : RETURN
1820 K%(P%)=RND*38%+1% : RETURN
1999 REM -----
2000 REM klump-drag
2010 W%=SGN(R%(1%)-R%(I%))
2050 R%(I%)=R%(I%)+W%
2080 W%=SGN(K%(1%)-K%(I%))
2130 K%(I%)=K%(I%)+W%
2135 IF K%(I%)<1% THEN K%(I%)=1%
2137 IF K%(I%)>38% THEN K%(I%)=38%
2140 RETURN
3000 REM krock? -----
3010 FOR I1%=1% TO P2%
3015 IF S$(I1%)='' THEN 3050 : REM redan borta
3020 FOR J%=I1%+1% TO G2%
3030 IF R%(I1%)=R%(J%) AND K%(I1%)=K%(J%)
GOSUB 3500 : REM krock!
3040 NEXT J%
3050 NEXT I1%
3060 RETURN
3100 REM -----
3500 REM krock!
3510 IF RND<.9 IF J%>=G1% AND J%<=G2% THEN 3700 :
REM rymdgrop
3520 REM krock med svart hål-----
3550 FOR L=1 TO 200 : OUT 6%,T% :
: CUR(R%(J%),K%(J%))CHR$(127%); : NEXT L
3600 S$(I1%)='' : REM död
3605 R%(I1%)=23% : K%(I1%)=0% :
REM flytta bort död
3610 OUT 6%,0%
3620 : CUR(R%(J%),K%(J%))S$(J%); : REM återställ
hål/grop
3630 IF I1%=1% THEN 5000 : REM slut
3640 GOTO 3800 : REM slut krock
3700 REM krock med rymdgrop-----
3710 FOR L=1 TO 200 : OUT 6%,G% : NEXT L
3720 REM ev död?
3730 IF RND<.8 P%=I1% : GOSUB 1810 : REM ny rad
3740 IF RND<.1 P%=I1% : GOSUB 1820 : REM ny kolumn
3750 : CUR(R%(I1%),K%(I1%))S$(I1%);
3760 : CUR(R%(J%),K%(J%))S$(J%); : REM åter-
ställ grop
3770 OUT 6%,0%
3800 RETURN : REM slut krock
3990 REM -----
4000 REM spelarens drag? (RTY uppåt etc)
4010 POKE 65013%,0%
4020 R%=0% : K%=0%
4030 IF W%=82% OR W%=84% OR W%=89% R%=-1%
4040 IF W%=86% OR W%=66% OR W%=78% R%=1%
4050 R%(1%)=R%(1%)+R%
4060 IF R%(1%)>20% R%(1%)=0%
4070 IF R%(1%)<0% R%(1%)=20%
4080 IF W%=82% OR W%=70% OR W%=86% K%=-1%
4090 IF W%=89% OR W%=72% OR W%=78% K%=1%
4100 K%(1%)=K%(1%)+K%
4110 IF K%(1%)>39% K%(1%)=39%
4120 IF K%(1%)<0% K%(1%)=0%
4130 W%=0%
4140 IF NOT Q% IF K%(1%)=39% GOSUB 5100 :
REM halvvägs framme!
4145 IF Q% IF K%(1%)=0% GOSUB 6000 : REM
klart! återkomst!
4150 RETURN

```

```

4900 REM -----
5000 REM slut
5005 GOSUB 5500 : REM sudda
5010 : CUR(20%,0%)'Sorry, du strök med! - otur'
5020 GOTO 9800 : REM slutrutin
5090 REM -----
5100 REM spelaren klarat sig till basen
5110 : CHR$(7%)CUR(20%,0%)'Du har kommit halvvägs!'
5120 : 'KAN DU TA DIG TILLBAKA?'
5125 Q%=-1% : REM flagga för returfärd
5130 S$(1%)='<' : : CUR(R%(1%),39%)S$(1%);
5150 RETURN
5490 REM -----
5500 REM sudda läsraderna
5510 : CUR(20%,0%)SPACE$(79%); : RETURN
5520 REM -----
6000 REM hjälten återkomst!
6100 GOSUB 5500
6105 FOR K%=0% TO 20% :
: CUR(K%,0%)CHR$(7%,127%); : NEXT K%
6110 GOSUB 5500 : : CUR(20%,0%)' HURRA!
Du klarade dig!'
6120 END
6900 REM -----
7000 REM information
7010 : CHR$(12%)'H RYMDRESA H' : ;
7020 : 'Testprogram för blivande rymdflygare.'
7025 : ; 'Du skall flyga en rymdfarkost'
7030 : '(märkt > på radarskärmen) från skärmens'
7040 : 'vänstra sida till den högra och tillbaka'
7050 : ; 'På skärmen ser du SVARTA HAL (.)'
7055 : 'Det som fastnar där förintas.' : ;
7060 : 'Den som hamnar i en RYMDGROP (?)';
7070 : 'kan bli kastad ut till en annan del av'
7080 : 'rymden - men kan också bli förintad.'
7100 : ; 'Magnetiska RYMDKLUMPAR ($) dras mot'
7110 : 'ditt skepp. En kollision är fatal.'
7120 : 'Se till att du lockar dem ned i'
7140 : 'ett svart hål så fort som möjligt.'
7150 : ; '(Tryck RETURN när du läst klart)'
7160 GET W$
7200 : CHR$(12%)'Du styr med tangenterna'
7210 : ; TAB(10%)'R T Y'
7220 : TAB(10%)'F G H'
7230 : TAB(10%)'V B N'
7240 : ; 'R = snett uppåt vänster'
7250 : 'T = rakt uppåt'
7260 : 'Y = snett uppåt höger'
7270 : 'V B N = på samma sätt nedåt'
7280 : 'F = rakt åt vänster'
7290 : 'H = rakt åt höger'
7295 : 'G = stå still'
7300 : ; '(Tryck RETURN)'
7305 GET W$
7310 : CHR$(12%)'Vilken hastighet vill du ha:'
7320 : ; '0 = (nolla) = full fart'
7325 : '1-8 = allt långsammare'
7330 : '9 = inspektion (flyttar endast vid tryck'
7340 : ; 'Välj fart (T):'; : GET W$ : ; W$
7350 ONERRORGOTO 7360 : T=VAL(W$)
7360 : ; ; ; ; ; 'Vill du läsa
information igen'
7370 : 'Tryck I, annars börja med RETURN'
7380 GET W$
7390 IF W$='I' OR W$='1' 7000 ELSE RETURN
9000 REM SYRET SLUT?
9010 IF S%>1000% GOTO 9050
9020 : CUR(20%,0%)'BARA'(100%-S%/10%)'% AV
SYRET KVAR!';
9025 IF S%>700% : ;CHR$(7%);
9030 RETURN
9050 GOSUB 5500 : : CUR(20%,0%)'SYRET SLUT - DIN
FÄRD OCKSA!' : GOTO 9800
9800 REM slutrutin
9810 IF INP(56%)>127% THEN 9810
9820 POKE 65013%,0%
9830 END
=====
OBS!
Av layouttekniska skäl har raderna
30,1013,1030,1090,3030,3510,3550,3605,3620,3760,
4140,4145,6105,6110,7360,9020,9050
delats på två rader.
Andras till en rad i programmet.
=====

```


ABC80 SOM RAMMASKIN.

När man har ABC80 ombyggd som rammaskin alltså med 64K adresserbart minne, så får man en rad nya möjligheter. Dessa möjligheter har alla samband med att man på olika sätt kan gå in och ändra i BASIC-tolken. En av de mer intressanta möjligheterna är att man kan ändra tidskonstanterna i kassettrutinen. För den som inte har floppy är denna möjlighet extra intressant eftersom man kan få snabbare inläsning av program. Tidskonstanterna för kassettrutinen ligger på de decimala adresserna 1144, 1153 och 1439 om man har check-summa 11273, annars ligger de på adresserna 1153, 1162 och 1439.

Med en av följande små snuttar kan du bekvämt ändra tidskonstanterna:

```
1 REM för check-summa 11273
10 INPUT A
20 POKE 1144,A
30 POKE 1153,A
40 POKE 1439,A*.8
```

```
1 REM för checksummorna 10042 & 9913
10 INPUT A
20 POKE 1153,A
30 POKE 1162,A/2
40 POKE 1439,A*.8
```

Tidskonstanterna är i hårdvaruversionen av ABC80 för checksumma 11273 150,150 och 120 det vill säga de två första har samma värde medan den tredje är 4/5 av detta värde. Tidskonstanterna är för de övriga checksummorna 150, 75 och 120. Man har helt enkelt tagit bort en skiftinstruktion, det vill säga en division med 2, när man ändrat i Basic-tolken.

Att ändra inspelningshastighet

Den normala överföringshastigheten till kassett är för ABC80 700 Baud och den hastigheten uppnås när parametrarna har standardvärdena 150, 150 resp 120. Det finns alltså utrymme för både långsammare och snabbare kassettrutin. Jag har gjort försök med båda delarna. Min utrustning har varit en ABC80 med checksumma 11273 och den gamla ABC80-bandspelaren och en ordinär kassett av samma typ som ABC-kassetterna. Vilken ramombyggnad man har spelat sannolikt ingen roll. Jag har en utan ram-floppy.

Läsning och skrivning i hastigheter lägre än 700 Baud

Jag ska börja att berätta om mina försök med att sänka hastigheten på kassettrutinen. Många ställer sig antagligen frågan: Vad ska det vara bra för? Frågan är högst begriplig, kassettrutinen går tillräckligt långsamt redan nu. Det finns knappast någon stark opinion som kräver långsammare kassetter. Det var ren och skär nyfikenhet som fick mig att försöka sänka kassettrutinen genom att mata in värdet 255 i programsnutten ovan. När jag gjort det läste jag in ett program från disk, utan flexskiveenhet hade det blivit besvärligt, sedan jag ändrat parametrarna kan jag ju inte läsa in standardkassetter.

Programmet sparades på kassett och när det var klart backade jag och laddade in programmet med load. Det lät dovt och

ovant, men hast du mir gesehen FOUND .BAC det hade lyckats! Överföringshastigheten borde vara så där en 412 Baud. Det var emellertid nu som de egentliga experimenten började. Jag ville alltså ta reda på hur känslig kassettrutinen är för variationer i hastighet hos bandspelaren. Eftersom det är svårt att variera hastigheten hos bandspelaren på ett kontrollerat sätt är det bättre att variera hastigheten på kassettrutinen.

Kassettfilen som jag skulle försöka läsa in var alltså inspelad med 412 Baud och tidskonstanterna var vid inspelningen i ordning 255, 255 och 204. Hur mycket skulle jag våga ändra? Jag började försiktigt med ett par procent och matade in 250 i programmet för ändrad skriv- och läshastighet på kassett. Det gick bra och jag gjorde ett nytt försök med 240, vilket också gick bra. Nu blev jag djärvare och provade med 230, 210, 200, 190, 170 och 165. Det blev inte stopp förrän vid 165, allt annat gick perfekt. Det innebär att kassettrutinen gick drygt 30% snabbare vid det sista lyckade försöket än den gjorde då jag skrev in test-filen på kassetten.

Vi som arbetat med ABC-kassetterna inom klubben vet att det så kallade fönstret är relativt stort, det vill säga kassettrutinen är relativt okänslig för variationer i hastighet hos bandspelarna. Det är tur annars skulle det gå mycket sämre att distribuera program på kassett. För att få en uppfattning om fönstrets storlek bör man variera hastigheten vid inläsning så att man provar med såväl snabbare som långsammare kassettrutin. Den första test-filen var inspelad med långsammast tänkbara kassettrutin och någon möjlighet att prova inläsning med en ändå långsammare förelåg alltså inte. Jag tog då en ABC-kassett nr 10 och satte kassettrutinen på en något högre hastighet med hjälp av ändringsprogrammet i vilket värdet 140 matades in.

ABC-kassetterna är inspelade med 700 Baud och alltså standardvärdena på tidskonstanterna dvs 150, 150 och 120. Jag började alltså med att snabba upp rutinen i förhållande till testfilen.

Det första försöket gick bra, jag fortsatte med värdena 130, 120, 110 och 105. Det sista, 105, gick inte alls, de övriga gick perfekt. Nu var det alltså dags för den andra delen av försöket, att försöka läsa med en rutin som går långsammare än den rutin som användes vid inspelning. De värden jag provade var 160, 165 och 170. Det gick bra till och med 165 sedan var det stopp. Det betyder att jag kunde läsa med värden mellan 110 och 165, vilket faktiskt innebär en hastighetsskillnad om ca 50%.

Att skriva och läsa i mer än 700 Baud

Det är framför allt möjligheten av en snabbare kassettrutin som hägrar för de som hängt med ända hit, eller hur? Jag tror inte ni ska bli besvikna. Enligt en artikel i Mikrodatorn skulle det vara möjligt att snabba upp kassettrutinen till det dubbla med denna metod, jag har lyckats att få den att gå med nästan fyrdubbla hastigheten. Dubbla hastigheten får man genom att mata in 75 i ändringsprogrammet och det är inga större problem med att läsa tillbaka från kassetten. Svårigheterna kommer med högre

skriv- och läshastigheter. Jag var alltså benägen att ge upp efter flera timmars experimenterande med skrivhastigheter över 1500 Baud. Det var först sedan jag funderat över resultaten från de tidigare försöken som jag kom på en möjlighet. Det hade ju visat sig att det inte var någon nackdel om kassettrutinen vid läsning var snabbare än den varit vid skrivning på kassetten. Varför inte göra ett försök? Tanke och handling äro som bekant ett hos data-amatören, varför jag genast grep mig verket an. Jag ändrade kassettrutinen genom att mata in 60 och skrev sedan in en fil på kassetten. Därefter laddade jag på nytt in ändringsprogrammet och gav 50 som invärde. Det var nu mycket spännande att se om programmet skulle gå in. Jag gav kommandot load cas: och fick FOUND efter en stund. Det var hastighetsrekord. Nu fortsatte jag att systematiskt pröva skrivning och läsning i höga hastigheter. Hela tiden gjorde jag så att jag lät kassettrutinen vid läsning vara ca 20% snabbare än vid skrivning på kassett. Sista lyckade försöket gjordes vid skrivning med kassettrutin-parametrarna 40, 40 och 32, följt av läsning med motsvarande parametrar satta till 33, 33 och 26. Skrivhastigheten bör då ha varit ca 2600 Baud. Det är naturligtvis inget tvång att gå på marginalerna och en lagom hastighet är 2400 Baud. Jag rekommenderar att man håller sig till denna hastighet, vilket man gör om man matar in 44 i ändringsprogrammet före skrivning och 37 före läsning. Dessa värden är naturligtvis bara förslag, men på sikt kan det finnas skäl att komma fram till en standard så att vi som har ram-maskiner kan kommunicera med varandra på kassett.

Säkrare läsning och skrivning på kassett utan internklocka

Genom att stänga av klockan i ABC80 får man något färre fel vid läsning från kassett och kanske också snyggare fyrkantvåg på sina kassetter. Det är mycket enkelt att stänga av klockan, det räcker med att poka in en enda instruktion i Basic-tolken. På adress 102 (decimalt för alla checksummorna) lägger man in instruktionen RETN, return from nonmaskable interrupt. Detta sker med POKE 102, 237,69 varvid klockan stannar. Den kan startas på nytt med POKE 102,24,223. När klockan stängs av hoppar ABC80 över klockrutinen och processorn får mindre jobb.

Ingen kompatibilitet med ABC800

De kassetter som spelas in med förhöjd hastighet på det ovan beskrivna viset kan ej läsas in utan vidare av en ABC800. Det beror på att kassettrutinen i ABC800 arbetar med ett inledande block vars hastighet är 700 Baud. I detta block finns uppgift om hur fort kassettrutinen överför de följande blocken. När man snabbar upp kassetterna på det här beskrivna viset går överföringen med samma hastighet hela tiden. Det spelar naturligtvis ingen större roll för den som bara vill kunna spara och ladda in sina egna program snabbare. Problem uppstår först om man vill använda denna metod exempelvis för att få plats med fler program på en ABC-kassett.

Kjell-Åke Johansson

Programmering BASIC II.

BASIC II har några ingredienser som gör det mycket enkelt att skriva strukturerad kod. Långa namn på variabler och funktioner, WHILE loopar och flerradiga funktioner. Om man använder en programutvecklingsmetod som JSP så förenklas själva kodningen av BASIC II. Dock vill jag för att undvika missförstånd påpeka att JSP kan tillämpas oberoende av programspråk.

Namn på variabler och funktioner.

I BASIC II kan man använda långa namn på variabler och funktioner. Just långa namn på variabler och funktioner gör att BASIC II-programmen blir mycket enklare att förstå och skriva än program i BASIC-dialekter med enbart korta variabelnamn.

Det finns egentligen ingen begränsning på längden av ett variabelnamn. Dock begränsas det ändå eftersom man inte kan mata in mer än 160 tecken på en rad. Alla långa namn på variabler och funktioner finns i en lista längst bak i programmet. Detta gör att man enbart får obetydligt längre kod med långa variabelnamn än med korta dito.

WHILE-satsen

Normalt använder man sig flitigt av "read ahead"-tekniken om man utnyttjar JSP. "Read ahead" står för "läs i förväg" och kan användas även om man inte tillämpar JSP. Läs i förväg innebär att man alltid läser en post i förväg. Detta gör att man t ex alltid klarar av att behandla tomma filer utan att speciellt ta hänsyn till det. En enkel tillämpning av "read ahead"-tekniken visar jag här i ett exempel där jag läser tal från tangentbordet och adderar dem tills man matar in 0 (=noll). Jag bortser från radnummer.

```
Summa = 0
INPUT Tal ! Read ahead
WHILE Tal<0
  Summa = Summa + Tal
; Summa
INPUT Tal ! Read ahead
WEND
```

Detta innebär att man måste skriva läs-satsen på två ställen, men det gör ju inget. En evighetsloop kan uttryckas med en WHILE-loop på ett snyggt sätt. Villkoret -1 är ju alltid sant!

```
WHILE -1
  Z=FNDemo
WEND
```

Flerradiga funktioner

GOSUB-satsen i BASIC är inte särskilt bra eftersom den är bunden till ett radnummer som inte avsöjar särskilt mycket av vad som skall utföras. Det är också lite knöligt att skicka med parametrar tillsammans med ett anrop av en subrutin med GOSUB. Dessutom är anrop till flerradiga funktioner snabbare än motsvarande anrop med GOSUB.

Flerradiga funktioner i BASIC II är så

bra att man normalt använder dem i stället för anrop med GOSUB. Även om det inte är frågan om riktiga funktioner, dvs man har inget värde och skicka med och inget värde returneras så använder man flerradiga funktioner.

Observera att man inte behöver låta programmet genomlöpa koden i en funktion innan man kan använda den. Man behöver alltså inte som i Pascal först skriva sina funktioner och sedan huvudprogrammet.

En annan fördel med flerradiga funktioner är att man kan använda lokala variabler i en flerradig funktion. Det innebär att de lokala variablerna enbart finns vid själva anropet till funktionen. Motsatsen kallas för globala variabler som gäller i hela programmet. En global variabel t ex variabeln Summa är alltså inte samma sak som den lokala variabeln Summa.

Typen på namnet för en flerradig funktion måste vara samma som returvariabeln, dvs skall funktionen returnera en sträng så måste funktionen t ex ha namnet FNTest\$. En flyttalsfunktion kan dock returnera ett heltal och tvärt om.

Man får inte lov att tilldela en strängvariabel, som används som inparameter, något värde i funktionen, dvs "insträngen" får ej stå till vänster om ett "="-tecken. Detta resulterar i felmeddelande nr 191, ej tilldelningsbar i funktionen:

```
; FNAdd$(ABC)
DEF FNAdd$(Sträng$)
  Sträng$ = Sträng$ + '-Klubben'
  RETURN Sträng$
FNEND
```

I stället får man skriva funktionen så här:

```
DEF FNAdd$(In$) LOCAL Sträng$=80
  Sträng$ = In$
  Sträng$ = Sträng$ + '-klubben'
  RETURN Sträng$
FNEND
```

I det senare och bugfria (felfria) exemplet används en lokal variabel. Det behövs inte för variabeln In\$ och i det felaktiga exemplet eftersom inparametern alltid är lokal. Notera också att man måste deklarera längden på lokala strängar (här har jag tagit till kraftigt, mer än vad som behövs för exemplet).

Man kan med fördel använda flerradiga funktioner när man egentligen behöver en procedur som inte skall ha några parametrar varken in eller ut. Då anropar man funktionen utan parameter och returnerar en tom sträng eller, om ett tal skall returneras, så skickar man tillbaka 0 (=noll).

Två exempel på detta:

```
DEF FNSkriv
; 'Detta är ett exempel på en funktion'
; 'som inte returnerar något värde'
RETURN 0
FNEND
```

```
DEF FNSkriv$
; 'Detta är ett exempel på en funktion'
; 'som inte returnerar något värde'
RETURN "
FNEND
```

Egentligen spelar det ingen roll vilken typ ovanstående funktion är, men skall den anropas med ;FNSkriv så spelar det roll. I det första exemplet skrivs då noll eftersom funktionen returnerar detta. Om man anropar funktionen FNSkriv\$ så skrivs givetvis inget alls. Egentligen skrivs det en tom sträng, men den syns ju inte.

Man kan låta en funktion returnera ett booleskt värde, dvs sant eller falskt (-1 eller 0). Sedan kan man anropa funktionen i en IF-sats.

```
INTEGER
IF FNUdda(Tal) THEN ; 'Talet är udda'
ELSE ; 'Talet är jämt'
DEF FNUdda(Tal)
  IF (Tal AND 1) = 1 THEN RETURN
  -1 ELSE RETURN 0
FNEND
```

Notera också att om man använder JSP-teknik så gör det absolut inget om man har flera uthopp ur en funktion. Att man tidigare inte ville ha det berodde det nog mest på att vanliga flödesscheman då blir lite grötiga.

I de första versionerna av BASIC II-tolkarna kunde man hoppa ut ur flerradiga funktioner med GOTO, GOSUB och ON ERROR GOTO satser. Detta kunde dock medföra att datorn dök i vissa fall. Därför kan man inte på senare versioner av BASIC II hoppa ut ur en flerradig funktion med någon form av GOTO-hopp. Felhanteraren är alltid lokal i en funktion.

En loopvariabel för en FOR-NEXT-loop kan dock inte vara en lokal variabel eftersom FOR-NEXT-loopen är optimerad så att den skall vara snabb. För att en loopvariabel skall var lokal så kan man i stället använda sig av en WHILE loop.

```
DEF FNLoop LOCAL I
FOR I=1 TO 10
; I
NEXT I
RETURN I
FNEND
```

Detta är alltså en felaktig funktion och felmeddelande nr 186. Man kan i stället skriva så här:

```
DEF FNLoop LOCAL I
WHILE I < 10
  I = I + 1
; I
WEND
RETURN I
FNEND
```

Denna loop blir dock obetydligt långsammare än det första exemplet. Det första exemplet är körbart om I inte är en lokal variabel. Man kan alltså behålla FOR-NEXT-loopen om man gör I till en global variabel. Notera att I ej behöver nollställas i exemplet med WHILE-loop eftersom det är en lokal variabel som alltid nollställs vid anropet. Variabeln I för lokala variabler finns enligt uppgift på stacken och då enbart under själva anropet.

Glöm slutligen inte att använda radnummer om du själv vill testa några exempel på en dator. Eftersom det inte har någon betydelse så har jag utelämnat dem, men jag vill också visa att man för logikens skull egentligen klara sig utan dem.

Bo Kullmar <1789>

HARDCOPY FÖR ABC800

Hardcopy är engelska för utskrift av bildskärmens innehåll på skrivare.

I ABC800 C är bildminnet organiserat på samma sätt som i ABC80. Däremot räknar optionsprommet 7.1 ut bildskärmsadressen, så någon motsvarighet till den tabell som finns i ABC80 saknas. Sannolikt räknas bildskärmsadressen också ut i optionsprommen 7.2 och 7.21. Därför måste man vid hardcopy för ABC800 C räkna ut bildskärmsadressen rad för rad.

I de andra 800-orna är bildminnet organiserat linjärt från adress 30720. Man kan då göra en utskrift av bildminnet genom att läsa av 80 bytes för varje rad. Det finns ett enklare sätt att utföra detta på!

Det går att med VAROOT flytta värdepekaren för en variabel. Man kan därför flytta värdepekaren för en variabel till bildminnet och pokea in dimensionerad och aktuell längd till 2048 bytes. Sedan är det bara att skriva ut denna variabel på skrivaren! Observera att 6:e tecknet i parametern för printern måste vara A eller B!

Här nedan finns en hardcopyrutin för ABC806, som tar alla 25 raderna. Observera att den 25:e raden finns på olika adress i ABC806 och ABC800 M. I ABC800 M börjar den på adress 32688 enligt Bit för Bit, medan den i ABC806 börjar på adress 32640. Detta innebär att på ABC800 M måste man behandla den 25:e raden speciellt om man vill ha med den ut på skrivaren.

```
10 DEF FNHardcopy806
20 DIM Screen$:POKE VAROOT
  (Screen$),0,8,0,120,0,8
30 PREPARE 'PR:VSA36A72.55' AS
  FILE 9
40 ; $9,LEFT$(Screen$,2000)
  ! 2000 ändras till 1920 för 24 rader
50 CLOSE 9 : RETURN 0
60 FNEND
```

När det gäller ABC806 får man inte "tag på" attributminnet på detta sätt. I bildminnet finns tecken med dubbel textbredd (EL) i två byte och tecken med dubbel textbredd och höjd (DBLE) i fyra byte. EL-moden är samma som 40 teckens-mode. Om man gör en hardcopy enligt ovan på ABC806 och har text med DBLE så får man följande resultat:

```
TT EE SS TT      Motsvaras av PRINT DBLE
"TEST"
```

Eftersom jag inte känner till någon metod för att läsa av attributminnet, så vet jag inte hur problemet med DBLE och EL kan lösas.

Nedan finns en rutin som bör fungera bra på alla 800 datorer, utom på ABC806 och ABC802 i 40 teckens mode. Adress 65364 innehåller bredden för bildskärmen, men kan man såvitt jag vet inte ta reda på om det är en ABC800 C eller en ABC806/2 i 40 teckens mode. Därför antar jag att om det är 40 teckens skärmbredd, ja då är det en ABC800 C! Rutinen skriver bara ut de 24 normala raderna för ABC800 M, 802 och 806. Den 25:e raden finns inte på ABC800 C.

```
65000 DEF FNHardcopy LOCAL Max,Adress,
  Teck
65010 ! Om 40 tkn, så antas det att det
  är en ABC800 C, 40 tkn mode
65020 ! ABC802 och ABC806
  kräver en annan rutin som avläser
  vart annat tkn!
```

```
65030 DIM Screen$:0 : POKE
  VAROOT(Screen$),0,8,0,120,0,8
65040 PREPARE 'PR:VSA36A72.55'
  AS FILE 9
65050 IF PEEK(65364)=80 THEN
  ; $9,LEFT$(Screen$,1920) :
  GOTO 65160
65060 FOR I=0 TO 23
65070 ; $9,TAB(20);
65080 Adress=31744+I*128-I/8*984
65090 FOR Y=Adress TO
  Adress+Max-1
65100 Teck=PEEK(Y)
65110 IF Teck<32 THEN Teck=32
65120 ; $9,CHR$(Teck);
65130 NEXT Y
65140 ; $9
65150 NEXT I
65160 CLOSE 9 : RETURN 0
65170 FNEND
```

Vill man tillfälligt spara undan bildminnet och återskriva det senare kan man göra det med samma teknik. Observera att detta

bör gå även för ABC800 C. Ev kan man behöva ändra 8:orna i VAROOT:en till 4:or eftersom bildminnet bara är 1024 bytes i ABC800.

```
750 DIM Screen$:0 : POKE
  VAROOT(Screen$),0,8,0,120,0,8
9100 DEF FNSavescreen
9110 Curpos=PEEK2(65362)
9120 Tscreen$=Screen$
9130 RETURN 0
9140 FNEND
9200 DEF FNPrintscreen
9210 Screen$=Tscreen$
9230 POKE 65362,Curpos,SWAP%
  (Curpos)
9240 RETURN 0
9250 FNEND
```

Man kan också utnyttja denna teknik för att på ABC800 M och 802/6 blanka ut delar av bildskärmen genom att t ex skriva Screen\$=SPACE\$(960). Detta påverkar inte cursorn. Det går även att skriva i bildminnet genom att tilldela variabeln Screen\$ ett värde. Man kan t ex skriva Screen\$=STRING\$(1920,35). Detta bör fungera även på ABC800 C. Ett annat exempel är Screen\$=Space\$(960)+STRING\$(960,42).

Av Bo Kullmar

KOMMANDON I PROGRAM

för ABC800 datorer

I likhet med vad som beskrivs i ABC-Bladet nr 2 1982 kan man i ABC800, ABC802 och ABC806 använda kommandon i program.

Kommandon genomföres i ett program genom ett CALL till en rutin som laddar adressen till parametern i HL samt sedan hoppar till adressen för rutinen, vilken lägges i DE registret.

På detta sätt anropas RUN rutinen i ABC806:

```
10 Z=FNFunk("")
20 DEF FNFunk(P$) LOCAL Par$=20,Par
30 Par$=P$+CHR$(13)
40 Par=VARPTR(Par$)
50 POKE 64512,213,33,Par,SWAP%(Par),
  201
60 Z=CALL(64512,4907)
70 RETURN 0
80 FNEND
```

Här används en tom parametersträng som bara innehåller ASCII-tecknet 13 (RETURN). 64512 är en adress i dosbuffert nr 7, HL laddas med adressen till parametern som man får med instruktionen VARPTR(Par\$). Därefter sker hopp till run-rutinen på adress 4907. Dosbuffert 7 användes eftersom det i ABC806 och andra 800-or som har UFD-doset, saknas en POKE-area.

Vilka är adresserna för kommandorutinerna då? Det är ganska enkelt att ta reda på. Däremot är det meningslöst att här ange några adresser eftersom det finns fyra olika ABC800 datorer och en del av dem finns ute i flera olika versioner vad gäller operativsystemet.

Någonstans på adress sextontusen finns det en lista i klartext över de olika kommandona. Denna omfattar ej BYE, men det gör inget eftersom BYE kan användas direkt i program. För att hitta denna lista kan man skriva ett enkelt program som läser/peekar minnet byte för byte.

I den första ABC800 C började denna lista på adress 16382 och i den första ABC806 börjar denna lista på adress 16514. Sedan läser man listan tills den är slut och hittar då talet 255, vilket markerar slut på listan. Därefter följer en tabell med de sökta adresserna!

Adresserna finns i två byte och kan erhållas med PEEK2. Man skall tolka adress-tabellen i samma ordning som listan över kommandon i klartext, men observera att SCR och NEW samt RENUMBER och REN refererar till en och samma rutin eftersom detta bara är synonymer!

En underlighet är att RESUME finns i kommandotabellen trots att detta inte kan användas som ett kommando. Jag känner inte till någon förklaring till varför det är så.

I anslutning till RUN rutinen finns det några inhoppsadresser för RUN utan filnamn och RUN utan CLEAR på samma sätt som för ABC80. Dessa adresser kan lätt experimenteras fram med "trial and error" metoden.

Vissa kommandorutiner kan inte användas i program, andra kan. Prova!

De jag kan rekommendera i program är:
RUN

Om man lägger BOFA och EOFa i common-variabler och sedan höjer BOTTOM kan man CHAIN:a in en subrutin. Sedan kan man ställa tillbaka gamla BOFA och EOFa, samt anropa RUN utan filnamn. Då har man fått en virtuell subrutin på ett enkelt sätt.

LIST

Program kan lista sig själva antingen på bildskärmen eller på valfri enhet.

RENUMBER/REN

Program kan omnumrera sig själv! Observera att radnummer inte har någon funktion för programmet alls! De är endast till för editering! Man kan ta bort dessa radnummer, under förutsättning att programmet anpassas till detta (pekare måste ändras). Programs radnummer kan tas bort med Luxors program Squeeze 800.

NEW/SCR

Anrop till denna rutin kan ske. I BASIC är det enklare att göra CHAIN 'NUL:' vilket ger samma resultat.

Av Bo Kullmar

800program på kassett.

ABC-kassett nr 9

CASDISK.800

Denna versionen fungerar inte på grund av att mellanrummet mellan blocken på banden har minskats. Versionen på kassett nr 10 är anpassad för detta.

TV.800

TV-editorns funktionen för att ta reda på hur mycket diskutrymme som finns kvar på disketten är avsedd för disk typ ABC830 med gammalt DOS. Det kan dock enkelt ändras till annat DOS och disk.

MASTMIND.800

är ett program för ABC800C, dvs det är ett mastermindspel som genom ABC800:as funktionstangenter använder sig av fåger. Därför kan programmet inte köras på ABC800M eller ABC802. Det går att köra på ABC806 med en färgskärm, men eftersom färghanteringen fungerar på ett annorlunda sätt än vad som är fallet med ABC800C så blir utskrifterna på bildskärmen fula.

FILTRANS.800

kan inte bara användas vid filöverföring mellan två ABC800 utan även mellan ABC800/802/806 och ABC80. Dessutom kan man använda programmet vid kommunikation med ABC-klubbens nuvarande monitor. Över-

för man filer från en RAM-floppy om man kör ABC800:an som passiv monitor så måste någon fördröjning vid filslut ökas, speciellt om den andra (aktiva) datorn är en ABC80.

ABC-kassett nr 10

YATZY.800

Det verkar som det finns vissa överföringsfel i YATZY.800 som härrör från den tidpunkten när programmet skickades in till klubben via monitorn. Jag får i varje fall upp konstiga tecken på skärmen när jag kör programmet.

FÄRGKART.800

Programmet FÄRGKART.800 ger ingen färggrafik, utan visar en färgkarta över de kombinationer av text med färg och bakgrund i färg som kan användas på ABC800C och ABC806.

HARDCOPY.800

Se en artikel i detta numret av ABC-bladet.

LÄSDISK.800

Programmet LÄSDISK.800 är ett enkelt program för att läsa sektorer blockvis på en ABC830-diskett. Programmet är avsett för ett gammalt dos. Därför fungerar det inte med UFD-DOS:et, som finns som standard till ABC802 och ABC806. Det är dock

ganska enkelt att ändra det så att det fungerar för UFD-DOS:et.

I alla CALL satser som läser på diskett står det nu inom parantesen och efter kommat A%*256%+B%*32%. Detta skall ändras till: A%*8%+B%. Sedan kan man ev ändra driveselect från 0 till 12 respektive 1 till 13 om man inte gör så att MOx tolkas som DRx.

COPYFAST.800

COPYFAST.800 är avsett för disk typ ABC830 och gammalt dos. Det fungerar inte för andra diskar och DOS. Eftersom den rutinen som måste ändras enbart finns i maskinkod i programmet, så är det lite svårt att enkelt ändra programmet. ABC-klubben har för avsikt att göra de versioner av programmet som behövs för andra diskar och dos.

ABC-kassett nr 11

SNÖ.800

SNÖ.800 är ett mycket enkelt grafikprogram som ger snö på bildskärmen. Det använder sig av lågupplösningsgrafik och kan därför bara köras på ABC800C och ABC802.

MÖNSTER.800

MÖNSTER.800 är ett program för högupplösningsgrafik och kan därför inte köras på ABC802. Man kan köra programmet på ABC800C- och M-maskinerna under förutsättning av datorn är utrustad med ett kort för högupplösningsgrafik.

TJUT.800

TJUT.800 ger ljud i ABC800C- och M-maskinerna. Sannolikt fungerar programmet för ABC802 också, men det är helt klart att det inte fungerar på ABC806.

Bo Kullmar <1789>

Radannonser.

PERSONLIGT HOROSKOP.

Dataprogram, önskas köpa. Svar till: Bernt Bergström Odaltorget 3 602 29 NORRKÖPING

SÖKES - KÖPES.

Jag har en Centronics 101AL skrivare med TTL-kretsar. Söker en manual/servicehandbok för att kunna göra smärre justeringar.

Björn Wänman (3298)
Tallstigen 50, 915 00 Robertsfors
0934/105 60

KÖPES:

Beg eller defekt ABC-dator (80,800,802, 806) Till hyggligt pris.

Ska använda den till att göra ett examens arbete på löda i den m m (vill då ej köpa en ny, för dyrt).

Jan-Åke Larsson <2461>
Tranbärsvägen 5
633 53 Eskilstuna

Tel 016-14 12 06 efter 16.00

SÄLJES

Fabrikstillverkad joy-stick, modifierad till ABC80. 8 riktningar och avtryckare. Drivrutin, en rad BASIC medföljer. Pris 190:- inkl porto.

Erik Hänel
Älvdansvägen 16
436 00 ASKIM

telefon 031-28 56 76

SÄLJES

1 st ABC80 med bildskärm & bandspelare 3.700.-

1 st ABC Rapport nr 1 (obeg) 50.-

Göran Engström <3457>
Box 200
685 00 TORSBY

Tel 0560-107 02

KÖPES

Beg dator ABC80 med kassetbandspelare.

Bertil Piculell <4492>
Sigfridsgården 4891
262 07 HJÄRNARP

Tel 0431-281 24
arb 0431-310 60

SÄLJES

Smartaid III till ABC80 pris 750:-

KÖPES

Fortran till ABC80
Staffan Marcusson <2030>
Brattåsvägen 16
483 00 Landvetter

Tel 031-71 02 33

BREVVÄN

Jag önskar kontakt med någon jämnårig medlem att brevväxla (byta program) med.

Simon Lusua <3743>
Sätravägen 34
733 00 SALA

Födelseår 1968.

DATADISK 82 TILL SALU!

Den är köpt ny våren -82, nya beige/bruna designen. Inkl. DOS och printerdrivrutin för ABC80. Passar fullt ut till ABC800.

OBS! Ena drivern dubbelsidig, 320 k.

Priset 7500.- + moms.

Intresserad? Ring:

Christer Jonsson <681>
tel 0910-756 02 e. 17.00.

DISASSEMBLERN,

VAD ÄR DET?

Disassemblern är den första utgåvan i serien ABC-rapporter och den behandlar hur maskinvaran (Basicolken) i ABC80 är uppbyggd. Den förklarar hur de vanligaste rutinerna fungerar, vilka internkoder som används till olika instruktioner och argument, vilka systemvariabler som finns samt den snuddar vid hur filhantering går till via DOS:et. Listan är till 75-80 % kommenterad. Den är skriven med skönskrivare i "fotostat-kopieskick" men kommentarerna är handskrivna, dock fullt läsbara.

Vad kan man nu använda dissassemblern till? Jo, låt oss se:

1. Du har hittat en mysko systemvariabel i ett program som inte finns upptagen i bruksanvisningen. Om den används av basicolken så står det vad den används till i dissassemblern.
2. Du håller på att göra en drivrutin till ett mätinstrument men den måste vara skriven i assembler. Var finns rutinen som skriver ut indata från instrumentet på flexskiva eller printer? Svaret står i dissassemblern.

3. - "Nu har jag äntligen lärt mig maskinspråk! Man kanske skulle ta och göra sig en egen Hjälpare? Jag undrar hur kommandotolkningen fungerar..." Titta i dissassemblern så löser det sig.

4. - "Det skulle vara kul om man kunde lära sig assembler. Jag har visserligen köpt den här boken "Programming the Z80" men inga programexempel passar ju ABC80 och så är den på engelska också." Använd dissassemblern. Den och en annan lärobok utgör ett perfekt hjälpmedel för att lära sig assembler. Och att "programexemplet" passar till ABC-80, det kan du vara säker på!

Du beställer dissassembler enklast genom att sätta in 100 Sek på postgiro nr 62 93 00-5. Du kan också sända in din beställning till:

ABC-klubben
Vidängsvägen 1
161 33 BROMMA

The Computer Phantome 1984-01-24

DIV:

Kassett nr 9

Enligt telefonmeddelande från en medlem har programmet RESULTAT.BER har följande brister:

Sortering sker efter ASCII och ej efter numeriska tal.

I resultatlistan suddas ej gamla decimaler när den visas på bildskärmen.

TEXT FRÅN QZ-KOM

(Text 174881) 84-02-01 23.21 Mats Knuts ABC Klubben

Mottagare: ABC-800 erfarenhetsutbyte <403>

Ärende: Odokumenterad finess

Om du har en 806 med färgskärm kan du prova följande

```
ATTRIBUTE 4
; GRN
```

Det går också bra med ATTRIBUTE 5, 6 eller 7 dessa attribut motsvarar 0 - 3 med ett viktigt och (tycker jag) trevligt undantag. Nyfiken? Nej jag tänker inte avslöja vad som händer du får prova själv.

Om du inte har färgskärm kan du prova med följande:
ATTRIBUTE 4
; FLSH

(Text 174881)

Insänt av Bo Kullmar

PRESSMEDELANDE

Datakraft och Luxor Datorer går skilda vägar.

Avtal har träffats mellan Datakraft och Luxor Datorer innebärande att samarbetet avslutas. Luxor behåller alla rättigheter till ADMINISTRATION 800, men tar själva över underhåll och vidareutveckling. Datakraft behåller alla rättigheter till det s k SOL-konceptet, som säljs bl a under varumärket DTC ADMINISTRATION (Facit/Ericsson) och Ekonomi ABC (Software of Scandinavia/Data-Sweden).

Avtalet, som har utarbetats av parternas respektive advokater, har godkänts av resp VD för Datakraft och Luxor Datorer, och gäller med omedelbar verkan.

Ulf Lingärde
Datakraft

Vet Du att ABC-klubben har distrubrat programspråket

FORTH

på ABC-kassett nr 4 Det finns olika versioner beroende på vilken maskinuppsättning man har.

Artiklar om FORTH har funnits i ABC-bladet nr 2, 1982 och nr 1 och 2, 1983.

RÄTTELSE

Spara argument för ABC806:

I artikeln står det:

Använd ej Attribut som variabel, eftersom detta är ett BASIC-verb. Det skall naturligtvis vara "Använd ej Attribute ..." Attribut utan "e" går bra att använda som en variabel.

Funktionstangenter i ABC800-datorer

Här har jag "vänt" på bitarna som några säkert har upptäckt. Det jag i artikeln kallar för bit 0 är bit 7 osv.

Bo Kullmar <1789>

Fel på kassettbandspelaren?

Bengt Sandgren <2776> har berättat om en av de besvärligheter som man kan råka ut för med kassettbandspelaren Lösningen har stått i LUXORS serviceinformation av den 23 juni 1983.

Vi citerar:

Tänkbar felorsak vid anslutning av kassettminne ABC 821 till tidiga modeller av ABC80

Om ABC80:s datorkort har variantnummer -03 (55 10470-03) saknas en folie vid kontakt märkt "TAPE". Detta medför att inspelning inte kan ske på kassettminnet.

Följande modifiering rättar till problemet:

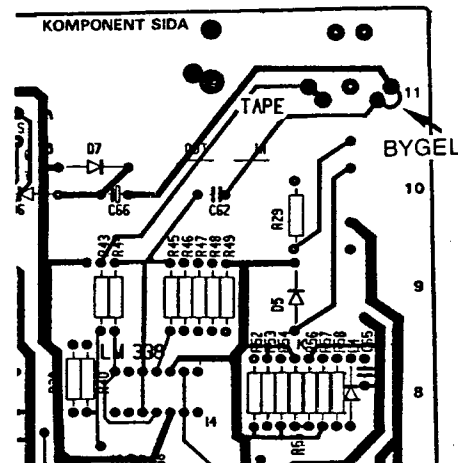
1. Dra ut nätkontakten.
2. Lossa kåpan och öppna datorenheten.

3. Löd in en överkoppling mellan stift 1 och 4 på kontakten märkt "TAPE" enligt figuren.

Bengt bifogar också två kommentarer.

* Märkningen med nummret hittar Du mitt under kylflänsen på moderkortet.

* Lödningen går lättast om man monterar ur hela moderkortet och löder på baksidan.



Del av ett kretskort.

Brev till ABC klubben.

Hej alla datanissar!

Ojoj! När jag skickade in UPPSJÖ till ABC-klubben väntade jag mig inte att programmet skulle bli så omtalat som det blev. Ärligt talat så ångrade jag att jag över huvud taget lade det på brevlådan.

Ett heldumt spelprogram utan början, utan slut och med tveksamt innehåll. Kan det verkligen vara något?

Hursomhelst, jag har skissat upp ett nytt adventure med spökslott, vargar och underjordisk labyrint. Frågan är: Ska jag fortsätta?

IF Svar = JA THEN Fråga =

Hur ska det hela se ut? För att få plats med det hela i ABC:ns minimala minne borde man kanske dela upp det i flera avsnitt. Ordlistan i Uppsjö borde nog utvidgas för att programmet ska ge ett bättre intryck. Dessutom borde hjälpfunktioner byggas in, så att man får ledtrådar när man virrat omkring i labyrinterna tillräckligt länge utan att hitta ut.

Beskrivningarna av de olika platserna kunde gärna vara mer utförliga, men det visar sig då att ett stort problem gör sig gällande: minnesbrist!

Ett annat problem är att författare till dylika program fastnar i samma hjulspår, varför programmen kommer att likna varandra mycket. Friska ideer skulle inte skada.

Alltså: Om det finns intresse för en ny datoriserad skräckhistoria, skriv då gärna en rad med synpunkter m m. I annat fall kan detta brev lämnas utan åtgärd.

Med vänliga hälsningar
Erik Åström
Köpmangatan 7
960 40 JOKKMOKK
0971 - 103 74

Ibland Brogatan 8 951 38 LULEÅ 0920 - 945 44

Svar till ABC-kassett nr 10 - program KALENDER.BER

I den Gregorianska kalendern från år 1583, finns undantag för sekularår (slutar på 00) varav de som är delbara med 400 blir skottår. Genom denna regel uppkommer en dags fel först efter 3000 år, så på slutet av 47:e seklet får de då levande bestämma när den extra dagen skall infalla.

Sverige hade Julius Caesars kalender före år 1753.

KALENDER.BER programmet behöver ändras med följande rader:

```
135 IF 1/100-FIX(1/100)=0 AND 1/400<>INT(1/400) THEN 150
212 IF X/100-FIX(X/100)=0 AND X/400<>INT(X/400) THEN 220
```

så blir resultatet rätt.

Bo Wernfeldt <4584>

Synpunkter på programmet KALENDER (ABC-kassett nr 10).

Programmet gäller endast Gregoriansk kalender, och för svensk tideräkning är det användbart endast från 1 mars 1753.

När huvuddelen av den katolska världen år 1582 gick över från Juliansk (gamla tiden) till Gregoriansk (nya tiden) innebar det att man hoppade över 10 dagar genom att låta torsdagen den 4 oktober följas av fredagen den 15 oktober.

Att i Sverige följa ett sådant påvligt beslut var ju otänkbart. Reformationen låg alldeles för nära i tiden. Men inför skottåret 1700 tyckte man att det fanns tillfälle att starta en inhämtning av de 11 dagar som nu skilde den svenska tiden från nya tiden. Därför tog man bort skottdagen, vilken ju år 1700 inte fanns enligt Gregoriansk kalender. Avsikten var att i efterhand komma ifatt den nya tideräkningen. Men någon uppföljning skedde inte, och Sverige hade följaktligen en alldeles egen tideräkning, som låg 1 dag före den gamla Julianska kalendern och 10 dagar efter den nya Gregorianska kalendern. Detta stod man ut med fram till år 1712. Detta år infördes en extra skottdag, vilket innebar att februari kom att innehålla 30 dagar, och tideräkningen stämde återigen med den gamla.

År 1753 gick Sverige över till den Gregorianska kalendern genom att hoppa över 11 dagar i februari. Onsdagen den 17 februari följdes omedelbart av torsdagen den 28, och ordningen var återställd.

Med vänlig hälsning

Kurt Steinrud
Tullingebergsvägen 19
146 00 TULLINGE

Hej!

På lappen som följde med ABC-kassett nr 10 frågas om programmet KALENDER.BER är korrekt. Programmet är inte helt korrekt. Åren 1700, 1800, 1900, 2100, 2200 och 2300 blir felaktigt skottår. Felet rättas med följande tillägg:

```
207 IF X/100=INT(X/100) AND NOT(X/400=INT(X/400)) THEN 220
```

Man bör uppmärksamma att KALENDER räknar med den Gregorianska kalendern. I de romerskt katolska länderna infördes den före år 1600, men i Danmark och Norge år 1700 och i Sverige och Finland först år 1753!

Carl von Linn lär oss i beskrivningen av sin resa till Öland och Gotland att den 7 juni 1741 var en söndag. KALENDER påstår emellertid att denna dag var onsdag. Linn använde dock den Julianska kalendern. Skillnaden mellan de båda kalendrarna var vid den här tiden 11 dagar. Den 7 juni motsvaras av den 18 juni enligt den gregorianska kalendern och denna dag anger KALENDER helt riktigt som söndag.

Mer om kalendrar kan du läsa i boken "Tid" av Lars-Olof Loden (Bonniers Uggleböcker 1968).

Hälsningar
Per Ahlin <1911>

Än en gång ABC-kassetterna

Visst går det att få en ABC80 att läsa och begripa det mesta i kassettväg. Och ABC-klubbens kassetter behöver inte ställa till alltför våldsamma problem.

Med anledning av de skrivelser som förekommit har jag inte utan viss förvåning kunnat konstatera att jag hittills klarat mig alldeles utmärkt utan Luxors specialbandspelare, utan istället använt lite olika spelare som kommit i min väg. Så t ex har jag använt en Sony minibandspelare, en Marantz bärbar i stereoutförande osv.

Eftersom jag och det bestämdaste ander att ljudkassetter är utom i ett fåtal undantagsfall helt värdelösa som datalagringsmedium, har jag valt att istället satsa på disketter. Men så kommer ju ABC-klubbens kassetter.

Problemet löstes efter det enkla konstaterandet att ABC:ns ingångskretsar helst vill ha något som i alla fall liknar något av fyrkantpulser. Sådana går att åstadkomma i de flesta LF-förstärkare genom överstyrning. Alltså tog jag en kassettspelare, hängde en LF-förstärkare efter, kopplade ABC:n till högtalarutgången och drog på så att förstärkaren överstyrdes. Lite kontroll på ett oscilloskop gav besked om lämplig justering av nivån.

Och hör och häpna. ABC80:n svalde glatt jämmertjuten från förstärkeriet. Läste och lade över på diskett. Det enda bekymret var att filerna ligger lite nära varandra på kassetten. Normalt förutsättes ju att kassetten stoppas när det går för snabbt. Det fick jag sköta manuellt men bandspelarens paustangent för att ABC:n skulle hinna avsluta filerna innan nästa fil kom på kassetten. Men det är ju en enkel operation att göra manuellt.

Samtidigt måste jag trots en del förklaringar som förekommit i ABC-bladet fråga mig om det inte vore enklare för klubben att kopiera till disketter än till kassetter. Det är ju inget som hindrar att man kör i princip hur många diskdrivers som helst mot en controler och en dator. Att kopiera en omgång disketter skulle då ta en till två minuter med tillägg för tid för diskettbyte. Jag tvivlar på att klubben kan kopiera kassetter med den hastigheten.

Victor Falkteg <4068>

ABC-klubben kopierar ej kassetterna själv, utan lämnar ut det jobbet till ett företag. Klubben färdigställer en open-reel-master på ett specialband som kopieras i batchar om 21 kassetter och med 8 gånger snabbare hastighet.

Packningen i de små trevliga omslagen sker sedan i klubblokalen av intresserade medlemmar på kvällstid. Belöningen för dem är att få sin kassett någon dag tidigare än medlemmarna ute i landet, och kanske något tips om vad som det finns på kassetten. ABC-klubben är lyckligt lottad att det kan fungera på detta sätt. Man lovar ju bara att komma en kväll och gör en insats som man snabbt inser att många har glädje av.

Klubben skulle troligen inte kunna rå med att även lägga ut denna del och samtidigt ha en kostnad per kassett i handen hos den enskilde medlemmen som f n understiger 10 kr.

Den dag det finns ett företag som gör det Du talar om i den kvantitet (och kvalitet) som klubben behöver skall vi diskutera saken och överväga kostnaden i sammanhanget.

ABC-klubben
Kassetredaktionen/KAJ

ABC VÄST

Klubbträffar under mars och april 1984

5 mars
Demonstration av FACIT DTC 2 med standardprogram
19 mars
Demonstration av 2 hobbydatorer: VIC och SPECTRUM
2 april
Demonstration av SUPERSMARTAI

Förutom programbyte och genomgång av ABC-kassetter utlyses under februari en programtävling med flera priser. Tävlingsbidragen måste ha skrivits/skapats av de tävlande själva och programmen får utföra vilket arbete eller beräkning som helst men skall vara någorlunda väl dokumenterade. De skall efter tävlingstidens slut kunna läggas på monitorn för distribution till ABC-medlemmar. Styrelsen för ABC-väst är enväldig jury och beslutet om vinnande program kommer inte att kunna överklagas. Sista datum för inlämning av tävlingsbidrag är sista april.

Klubblokalen, Järntorget 3, kommer som vanligt att hållas öppen varannan måndag i jämna veckor kl 1800 till ca 2200. Aktiviteterna börjar kl 1900.

Medlem i ABC-väst blir man enklast genom att betala Kr 30:- på postgiro 15 33 36 - 3 Ange Ditt medlemsnummer och skriv "ABC VÄST" på talongen. ABC-väst informerar på klubbträffar, i ABC-bladet och på ABC-monitorn. Information även från

Hans Sturk, telefon dagtid 031-616955
Anders Lundberg telefon kvällstid 031-117572

ABC-VÄST VERKSAMHETSBERÄTTELSE Under året 1983 har hållits:

- 4 st styrelsemöten
- 4 st klubbträffar under våren
- 8 st klubbträffar under hösten
- 5 st föreläsningar i elektronik om vardera 3 tim
- 7 st lektioner i assembler om vardera 3 tim
- 1 st besök på Torslanda datamässa
- 1 st besök på Data Interface fristående dataskola
- 1 st besök på Kungsladugårds vuxen-gymnasium
- 1 st besök på Medborgarskolan
- 2 st extra träffar för "hårdvaruarbeten"

Under klubbafnarna har olika - icke till datum angivna - demonstrationer ägt rum: ABC80 med inverterat skärmljus, utbildningsdator, joystick, styrning av periferenheter (bandspelare) från V24-kontakten, "hembygge", modemkommunikation mot ABC-monitorn och QZ-centralen, lödning av modemkabel, printer, smartaid, ABC800 med färggrafik mm.

Två externa kurser har givits varav den ena avslutas först under februari månad 1984. Teknisk hjälp har givits i de fall detta varit möjligt. En kartläggning av medlemmarnas tillgång till ABC-maskiner och olika intressen har gjorts. Program har delats ut och programbyten har skett under klubbafnarna. Närvaron vid klubbafnarna har varit god och deltagarantalet har ingen gång varit mindre än 10.

ABC VÄSTs styrelse 1984

Ordförande
Hans Sturk
Pennygången 88
414 82 GÖTEBORG
Arb 031-616955, hem 031-415395

Sekreterare
Anders Lundberg
N:e Kvarnbergsgatan 18
411 04 GÖTEBORG
Arb 031-612293, hem 031-117572

Kassör
Nils Björnberg
Kuttergatan 1 A
421 76 VÄSTRA FRÖLUNDA
Hem 031-290049

Ledamot
Bo Olsson
Ulla Lyckas väg 114
430 50 KÄLLERED
Hem 031-722538

Ledamot
Anders Ripa
Tellanders väg 1
443 00 LERUM
Hem 0302-12054

Ledamot
Carl G Blomqvist
Häradsgatan 17
431 42 MÖLNADAL
Hem 031-279242

Suppleant
Carl G Hillefors
Poetgatan 22
422 55 HISINGS BACKA
Hem 031-581447

ORDBEHANDLINGSPROGRAM

En kraftfull och prisvärd ordbehandling för ABC80 eller ABC800.
Innehåller bland annat följande funktioner:

- Automatisk avstavning
- Tabulatorinställning
- Cursoradresserad editor
- Flyttning av stycke
- Rak högermarginal vid utskrift
- Understrykning
- "Huvud" eller rubrik på varje sida

Passar till 40/80 Tkn bildskärm ABC80 eller 80 Tkn ABC800

Pris: Diskett 385:- Kasset 345:- Priser inkl. Moms

Begär ytterligare information från

Råda Elektronik

Brattåsvägen 16, 43800 Landvetter PG 4654957-2

TÄNK

om det här var Din annons.

Då skulle alla tidningsläsare kunna läsa om ditt erbjudande just nu.

Precis som Du läser
vårt erbjudande:

Annonsera i

ABC BLADET

CaSturbo... för ABC80

CaSturbo... snabbar upp ditt kassetminne 300%.
Med denna nya suveräna kassettrutin kan du nu skriva och läsa med 2400 baud i stället för 700 som standard.

- **CaSturbo...** är snabbare och kraftfullare än rutinen i ABC80 men tar ändå bara ca 1 25 Kbyte. Den är självrelokerande och läggs in under BOFA.
- **CaSturbo...** kan utföra bland annat:
Skrivning och läsning rad eller blockvis, med eller utan bandstopp mellan blocken.
Skrivning med blockmellanrum 0.02—2.15 sek.
Skippa ERR 35. Möjliggör reparation av skadad fil.
- **CaSturbo...** känner automatiskt vilket baudtal en fil är inspelad med och kan alltså även läsa filer från ABC800.
- **CaSturbo...** är efterlängtat och oombarligt för alla kassetanvändare. Även idealisk när man ska göra backup på band.
- **CaSturbo...** kassettskiva innehåller även flera mycket bra hjälpprogram:
- **CASTURBO.BAC.** för läsning och skrivning i 2400/700 baud på kasset.
- **RUN2400.BAC.** för läsning av fil som är skapad av ABC800 eller CaSturbo.
- **CASLIB.BAS.** skriver ut en lista över kassetens filer.
- **CASCOPY.BAS.** kopierar alla filtyper, från kasset till skiva.
- **CASCAS.BAS.** kopierar från en kasset till en annan.
- **FILECOPY.BAS.** kopierar alla filtyper, från skiva till kasset eller från skiva till skiva.
- **LASPRINT.BAS.** printar ut filer från kassettskiva.

MODIFIERINGSSATS för bandspelaren ingår i paketet.

- ☐ Hela paketet, inkl ombyggnadssats: 475 kr inkl moms och porto.
Rabatt för medlem i ABC-klubben (medlemsnr): 75 kr.
Tillägg för flexskiva: 25 kr.

ÖVRIGT:

KVALITETSKASSETTER SÄLJES:

- ☐ C60, 10-pack 100 kr inkl moms.
- ☐ FLEXSKIVOR AV HÖG KVALITET: 10-pack 250 kr inkl moms.

Portotillägg 10 kr. Två 10-pack portofritt.

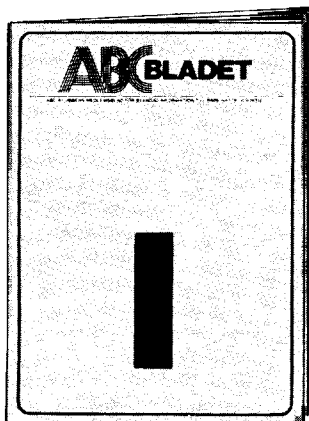
Vid betalning på postgiro 88 25 27-5 inget postförskottstillägg.
Beställes från:

L-system, tel 08-756 18 29. Box 5528, 183 05 Täby.

UTGIVNINGSPLAN

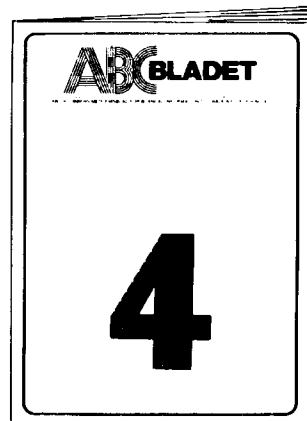
1984

Nr 1,84
Manusstopp 1 februari
Annonsbokning 6 februari
Materialdag 20 februari
Till tryck 1 mars
Medlemmarna 27 mars



Nr 2,84
Manusstopp 1 maj
Annonsbokning 7 maj
Materialdag 21 maj
Till tryck 1 juni
Medlemmarna 27 juni

Nr 3,84
Manusstopp 15 augusti
Annonsbokning 20 augusti
Materialdag 10 september
Till tryck 17 september
Medlemmarna 11 oktober



Nr 4,84
Manusstopp 26 oktober
Annonsbokning 6 november
Materialdag 20 november
Till tryck 26 november
Medlemmarna före jul

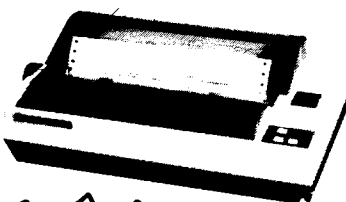


Kom ihåg att anmäla adressändring i tid

DATORER PROGRAM KONSULTING UTBILDNING

Jet Computer

NYHET! SKRIVARE



Pris 3.595:- exkl moms

- 80 tecken/sek
- Alfnumerisk/Grafisk
- Traktor och friktionsmatn
- Professionell kvalitet
- Epsonkompatibel
- Serieinterface finns

☐ SÄND INFORMATION ☐ KONTAKTA MIG

Namn: _____ Tel: _____

Företag: _____

Adress: _____

Jet Computer Corp.
Box 138, S-182 12, Danderyd, Sweden.
Tel. 08-75331 45, Telex: 11019.

DATORER PROGRAM KONSULTING UTBILDNING

Jet Soft

Jet Soft representerar välkända utländska och svenska programföretag såsom DIGITAL RESEARCH™, MICRO-CAL, ATI, MICROPRO, PDATA m fl.

Jet Soft säljer endast utprovade och rekommenderade programvaror:

sätt x:

- | | |
|---|---|
| ☐ Adm. system | ☐ Kalkylprogram |
| ☐ Operativsystem | ☐ Databas/register |
| ☐ Högnivåspråk | ☐ Ordbehandling |
| ☐ Datakommunikation | ☐ Skräddarsydda program |
| ☐ Databöcker | ☐ Programvaror från DIGITAL RESEARCH |
| ☐ Självstudiekurser ATI, MICROCAL
(Basic, CP/M och Cobol) | ☐ Program för Jet 80, ABC 800, IBM PC m fl |

Ja, Jag är intresserad av förkryssad produkt, sänd mer information. ☐ Ring mig!

Namn: _____ Tel: _____

Företag: _____ Avd: _____

Adress: _____

Jet Soft ab
Box 138, S-182 12, Danderyd, Sweden.
Tel. 08-753 31 45, Telex: 11019.

DATORER PROGRAM KONSULTING UTBILDNING

Jet Computer

PRIS/PRESTANDA- GENOMBROT

SVENSK
SENSATION!

- ☐ Enkortsdatorer
- ☐ Smådatorsystem
128 K RAM 0.4 Mbyte FD Kr 9.980:-
10 Mbyte +0.8 Mbyte Kr 29.000:-
- ☐ Lokalt datanät
- ☐ Matris skrivare CP-80
Epsonkompatibel Kr 3.595:-
- ☐ Lågprispildskärm Kr 4.980:-
- ☐ Printer/terminalkablar

Priser exkl. moms

Sänd mer information om förkryssade produkter.

Namn: _____ Tel: _____

Företag: _____

Adress: _____

Jet Computer Corp.
Box 138, S-182 12, Danderyd, Sweden.
Tel. 08-75331 45, Telex: 11019.

Varsågod 700 smådatorkurser!



Praktiska intensivkurser:

- ABC 800 / DTC
- IBM PC, Columbia
- PET, VIC, Xerox
- ABC 80, Texas, Sinclair
- CP/M-, MS DOS- och UNIXkurser
- Användar- och programmeringskurser
- Special- och skräddarsydda kurser
- Hemdatorkurser
- Programmeringskurser

Kursexempel: Nybörjarkurser, Ordbehandling, BASIC, Kopa dator, Teledata, Kalkyl, Basregister, Bokföring, Mät & Styr, Datakomm, Nätverk, Databas, Systemering, JSP, CP, M, Assembler, C, Unix, Pascal, Cobol, FORTRAN.

JA, sänd er 132-sidiga kurskatalog GRATIS.

Namn: _____

Företag: _____

Postadress: _____

LEXICON
utbildning på smådatorer

Box 136, 182 12 Danderyd. Tel. 08-75331 40