

PROGRAM

nittinian

BITEN

99-1

Inför årsmötet



Innehåll

Redaktören	2
Kallelse till årsmötet	3
McGoverns XB-skola	4
Superbug II	5
Menu version 7.1	6
Star-recension	7—8
Myarcs diskkontrollkort	8
Geneve	9—14
TI-74 BASIC	15
EDP — en recension	16
Variabelöverföring	16—17
Myarc Extended BASIC	18—20

ISSN 0281-1146

Redaktören ...

Detta nummer består till stor del av beskrivningar av nya program och tillbehör till 99:an. Myrars nya dator *Geneve* beskrivs ingående av Sören Bernle. Han beskriver också deras diskkontrollkort.

Recensioner av freeware börjar vi också få in. Denna gång är det programmen *Star* och *Enhanced Display Package* som beskrivs. De finns förstås i Programbanken.

Årsmötet

Detta nummer har satts ihop i viss hast för att vara klart till årsmötet. Därför är det bara på 20 sidor denna gången.

Ny Freeware

Funnelweb version 4.0 med datum DEC 22/87 på 2 diskar
Disk Manager 1000 version 3.8 på 1 disk
Disk Manager 1000 källkod på 2 diskar
MENU version 7.1 för Horizon RAM-disk från Miami TI Users Group på 1 disk
Disk Utilities version 4.0 finns i programbanken utan manual. Manualen måste beställas direkt från John Birdwell.
TEXTLOAD/EA5LOAD från Pargon Computing på 1 disk
MACRO ASSEMBLER version 6 på 2 diskar
LINKER version 2 på 1 disk
LINKER LIBRARIAN av Tom Bentley på 1 disk. Detta är ett tillägg till R.A.G. LINKER ovan.
TI-FORTH med laddare för Extended Basic

Computer Kontakt

Den tyska tidningen Computer Kontakt som även har behandlat TI-99/4A har slutat att komma ut. Resterande prenumerationsavgift har betalats tillbaka. Sista numret blev nr 2/3 1988.

Ny katalog från Asgard

Det har kommit en ny katalog på 14 sidor från:
ASGARD SOFTWARE
P.O. Box 10306
Rockville, MD 20850
USA

Den innehåller bl a följande:

High Gravity (rymdspel)	USD 15
Legends (grafik-adventure)	USD 28
(se Micropendium nov 1987)	
Font Writer II (blanda grafik och text)	USD 25
(se Micropendium okt 1987)	
Total Filer (databas)	USD 20
EZ-Keys	USD 15
Music Synthesizer	USD 10
Pre-Scan It!	USD 10
(se Micropendium jun 1987)	

Det finns även 7 spel och tillägsskivor till TI-Artist och GRAPHX. Nästan alla program kräver 32 kb och disk.

Du som vill ha en kopia på katalogen kan sända 5 frimärken (rabattmärke eller brevporto) till:

Jan Alexandersson
Springarvägen 5, 3 tr
142 00 TRÅNGSUND

I redaktionen:

Redaktör	Peter Odelryd
Utmaningsredaktör	Anders Persson
Forth-redaktör	Lars-Erik Svahn
Programförmedlare	Börje Häll
Allt-i-allo	Claes Schibler

Föreningens och redaktionens adress:

Föreningen Programbiten
c/o Schibler
Wahlbergsgatan 9, nb
12146 Johanneshov

Datainspektionens licensnummer: 82100488

Postgiro: 198300-6

Medlemsavgiften för 1987 är 120:-

Annonser, insatta av enskild medlem (ej företag), som gäller försäljning av moduler eller andra tillbehör i enstaka exemplar är gratis.

Övriga annonser kostar 1000 SEK per helsida, 500 SEK per halvsida. För lösblad som skickas med tidningen gäller 1000 SEK per blad.

För kommersiellt bruk gäller följande:

Mångfaldigande av innehållet i denna skrift, helt eller delvis, är enligt lag om upphovsrätt av den 30 december 1960 förbjudet utan medgivande av Föreningen Programbiten. Förbudet gäller varje form av mångfaldigande genom tryckning, duplicering, stencilering, bandinspelning, diskettinspelning etc.

Föreningens tillbehörsförsäljning

Följande finns att köpa för medlemmar genom att motsvarande belopp insätts på postgiro 198300-6.

För TI-99/4A:

Användartips med Mini Memory	60:-
Nittinian T-tröja	40:-
99'er magazine nr 12/82, 1-5, 7-9/83 (per styck)	20:-
Programbiten/Nittinian 1986 (TI 59+99)	80:-
Programbiten/Nittinian 1985 (TI 59+99)	80:-
Programbiten/Nittinian 1984 (TI 59+99)	80:-
Nittinian, årgång 1983 (TI-99/4A)	60:-

För TI-59 mm

Programbiten, årgång 1983 (kalkylatorer)	40:-
Programbiten, årgång 1982 (kalkylatorer)	40:-
Programbiten, årgång 1981 (kalkylatorer)	40:-
Programbiten, årgång 1980 (kalkylatorer)	40:-
Programbiten, årgång 1978/79 (kalkylatorer)	40:-
Programbiten, fem årgångar 1978-83	160:-
Programbiten, sex årgångar 1978-84	220:-
Programbiten, sju årgångar 1978-85	300:-
Katalog med belgiska och engelska program för räknare TI-57, TI-58, TI-59	20:-
Föreningens programmeringsblanketter (TI-59), olika typer, block om 50 blanketter (se PB 83-1 sid 30) per block	11:-
Patenthandlingar TI-59	25:-
Tom magnetkortsplånbok	10:-

 K-TRYCK AB

FÖRENINGEN PROGRAMBITEN

c/o Schibler
Wahlbergsgatan 9 N.B.
121 46 JOHANNESHOV

KALLELSE TILL ÅRSMÖTE

Medlemmar i föreningen Programbiten kallas härmed till Årsmöte
Lördagen den 12 mars 1988 kl 14.00 på Militärhögskolan, Valhallavägen 117, Stockholm

Förslag till dagordning:

1. Mötet öppnas
2. Val för mötet av ordförande, sekreterare och två justeringsmän.
3. Mötets behörighet
 - utlysning
 - röstberättigade närvarande.
4. Beslut om dagordning.
5. Styrelsens årsredovisning för 1987
Verksamhetsberättelse och kassarapport delas ut på mötet.
6. Revisorernas berättelse för 1987.
7. Om ansvarsfrihet för styrelsen för 1987.
8. Val av styrelse för 1988
Årsmötet väljer enligt stadgarna en styrelse bestående av ordförande, sekreterare, kassör och redaktör, vilka samtliga utses till befattning på årsmötet, samt därtill en och högst fem övriga ledamöter
9. Val av revisorer för 1988
Mötet väljer två revisorer och en suppleant
10. Val av valberedning
Mötet väljer två ledamöter varav en sammankallande
11. Om årsavgift 1988
Styrelsens förslag är oförändrat 120:—
12. Årsmötets formella del avslutas

Allmänt medlemsmöte försätter med visning av nya program m m. Tag gärna med något eget som du vill visa. Det är viktigt att du som tycker att möten med visningar av detta slag skall fortsätta också kommer på mötet. På de senaste mötena har deltagandet varit litet. Skriv och berätta vad du tycker skall vara med på mötet.

För Styrelsen

Jan Alexandersson
Springarvägen 5, 3 tr
142 00 TRÅNGSUND
Tel. 08/771 05 69

Extended BASIC-lektioner från Funnelweb Farm

(fortsättning från PB 87-4)

TI Basic kommer att göra slut på sin ynkligt lilla 16K-hjärna och avbryta exekveringen med felmeddelandet BAD NAME IN 100, medan Extended Basic, som är lite klyftigare, försöker utföra rad 100, men avbryter med meddelandet SUB-PROGRAM NOT FOUND IN 100.

Extended Basic-handboken kräver att alla SUB-program kommer sist i programmet, d v s att inget utom SUB-program får förekomma efter den första SUB-satsen (bortsett från REMARK-satser som programmet ju alltid struntar i). Extended Basic löper sedan igenom och reserverar nytt minnesutrymme för variabler, med början i variabelnamnen i SUB XXX(parameter-lista), för varje sub-program från SUB till SUBEND, som om det var ett separat program. Det verkar som om Extended Basic bara håller sig med en enda lista på underprogram-namn, oavsett var de finns, och som konsulteras varje gång interpretatorn/BASIC-tolken kommer till en CALL-sats under programkörningen. Alla DATA-satser hamnar också i en och samma lista. Det kan du övertyga dig om genom att köra detta lilla program:

```
100 DATA 1
110 READ X :: PRINT X :: READ X :: PRINT X
120 SUB NOTHING
130 DATA 2
140 SUBEND
```

När du kör detta program spelar det ingen roll att den andra DATA-satsen till synes står inne i ett sub-program. IMAGE-satser fungerar på samma sätt. DEFINierade funktioner — om du bryr dig om att använda sådana — är strikt begränsade till den del av programmet där de står, vare sig det är huvudprogrammet eller ett SUB-program. Under pre-scan hålls DEF-namn isär alltefter vilket SUB-program de står i eller om de återfinns i huvudprogrammet. Prova gärna denna programsnitt för att kolla att det stämmer:

```
100 DEF X=1 :: PRINT X; Y :: CALL SP (Y) :: PRINT X; Y
110 SUB SP (Z) :: DEF X=2 :: Z=X :: DEF Y=3
120 SUBEND
```

Detta framgår inte av Extended Basic-handboken och har varit föremål för vilseledande eller felaktiga kommentarer i tidskrifter och medlemstidningar. Lite eftertanke om hur Extended Basic beter sig brukar reda upp missförstånden.

Under pre-scan tilldelar TI:s BASIC-dialekter nominella värden till alla variabler som förtecknas i programmet, noll för numeriska variabler och en tom sträng ("") för strängvariablerna. Detta är en olikhet mot andra programmeringsspråk (t o m vissa BASIC-dialekter) där man får ett felmeddelande om man tar en odefinierad variabel i bruk. Det betyder att Extended Basic inte kan fungera som TI LOGO som har en regel som säger att om det stöter på en odefinierad variabel inne i en procedur, så kollar det kedjan av CALL-procedurer tills det finner ett värde. Men till skillnad från Pascal, som utplånar all information inom en procedur när den är genomlöst, bibehåller Extended Basic variabelvärdena från CALL till CALL helt inneslutna i resp SUB-program. Variabelvärden som förs över till ett SUB-program genom en parameterlista kommer naturligtvis att anta detta värde varje gång underprogrammet anropas med denna SUB-sats. Ett litet program visar skillnaden.

```
100 FOR I=1 TO 9 :: CALL SBPR (0) :: NEXT I
110 SUB SBPR (A) :: A=A+1 :: B=B+1 :: PRINT A; B
120 SUBEND
```

Den första variabeln (A), som skrivs ut återställs till 0 varje gång underprogrammet SBPR anropas, medan den andra, B, ökas med ett varje gång. Fält med variabler lagras i sin helhet någonstans i programmet, i huvudprogrammet eller i det SUB-program där DIM-satsen står. Extended Basic godtar inte försök att ändra på DIMensioneringen av ett fält, så information om fältstorleken kan bara överföras längs kedjan av underprogram i en riktning. Varje försök av ett SUB-program att anropa sig självt, vare sig direkt eller indirekt genom ett underprogram som anropats av det första under-

programmet — oavsett hur många mellanled det finns — resulterar i ERROR. "Rundgång" i procedurerna, en viktig beståndsdel i TI LOGO, kan inte utföras av SUB-program i Extended Basic, då att anropa ett underprogram med CALL inte etablerar en ny förteckning av variabelvärden.

Hela denna diskussion om hur Extended Basic beter sig kommer från programmeringserfarenhet med version 110 av Extended Basic på en TI-99/4A med titelskärm med årsangivelse 1981. Tidigare versioner och tangentbord är inte vanliga i Australien, men TI tycker i regel noga se till att nya programversioner är kompatibla med tidigare. Å andra sidan har TI också varit mycket förteget om hur Extended Basic fungerar i detalj. Editor/Assembler-handboken har mycket litet att berätta om detta, ännu mycket mindre än vad den innehåller om grund-BASICen. Jag känner f n inte till någon diskussion om syntaxen i dess GPL (Graphics Programming Language), än mindre källkoden för GPL-tolken som är inbyggd i varje 99/4A:s grundmoduls ROM.

Ännu ett enkelt programmeringsexperiment kommer att visa vid vi menar med att säga att XB sätter upp ett separat BASIC-program för varje underprogram. Kör följande:

```
100 X=1 :: CALL SBPR :: BREAK
110 SUB SBPR :: X=2 :: BREAK :: SUBEND
```

Kolla vilket värde variabeln X har efter BREAK genom att skriva in kommandot PRINT X, fortsatt sedan med CONTinue till nästa BREAK, som denna gång ligger i huvudprogrammet, där du än en gång kollar vilket värde X har.

Vi ska nu sammanfatta de egenskaper som utmärker underprogrammen i Extended Basic som procedurer i fullständiga Extended Basic-program, men spara detaljerna om hur hopkopplingen mellan de olika underprogrammen sker till nästa avsnitt.

- XB behandlar varje underprogram som ett separat program, och bygger upp en bestämd tabell av (med REF) benämnda och DEFINierade variabler för varje underprogram.
- Alla DATA-satser behandlas som om de återfinns i en gemensam pott, lika tillgängliga från alla underprogram och huvudprogrammet på samma sätt som IMAGE-satser, tecken definierade med CHAR, SPRITEs, COLORs, och FIL-specifikationer.
- All annan information flyttas från det anropande huvud- eller underprogrammet genom parameterlistorna i CALL- eller SUB-satserna. Extended Basic tillåter inte att man deklarerar samtliga för alla underprogram gemensamma variabler, vilket är möjligt i vissa program-språk.
- Variabelvärden som är begränsade till ett visst subprogram förblir oförändrade tills underprogrammet anropas nästa gång. Vissa språk, t ex PASCAL, utplånar alla spår av en procedur sedan den genomlöpts.
- Underprogram i Extended Basic får inte anropa sig själva, vare sig direkt eller indirekt i en sluten kedja. Annars får alla underprogram anropa varandra.
- MERGE-kommandot i Extended Basic med diskettstation (expansionsminne erfordras ej) tillåter att ett bibliotek av underprogram i Extended Basic lagras på diskett och in-förlivs med olika program när behovet uppstår.

Annonser:

Säljes:

Expansionsbox med 32K minnesexpansion och ursprunglig diskdrive med diskkontroller. Pris 1 000:—

Moduler:

TI Extended BASIC
TI Editor/Assembler
Spelmoduler

200:—
350:—
50:— st

Martin Flodén
Björkliden 22
183 41 Täby
tel 08/758 46 10

Köpes:

Corcomps Triple Tech-kort för expansionsboxen

Peter Odelryd tel 08/777 12 16

Superbug II

av Jan Alexandersson

Tillsammans med Editor/Assembler kommer en TI-Debugger med källkod på en av flexskivorna. TI har släppt en förbättrad version av denna till användarföreningarna. Denna Super Bugger 3.1 utvecklad av Navarone har funnits länge i programbanken.

Det finns en avsevärt förbättrad version av dessa utgiven av Edgar Dohmann i USA kallad *Superbug II* version 2.1. Nödvändig utrustning för att köra programmet är 32 kbytes expansionsminne samt Extended Basic, Mini Memory, Editor/Assembler eller Funnelweb. I programbanken har vi nu detta program i version 2.1 med manualen till version 1.0 på 40 sidor. Den fullständiga manualen på 52 sidor till version 2 har copyright men kan beställas från USA för USD 5 plus porto USD 3. Superbug II tar 8 kbytes av RAM-minnet och finns lagrat i tre olika format på flexskivan:

- SBUG är i PROGRAM-format med startadress >A000.
- SBUG6 är i PROGRAM-format med startadress >6000 avsett för RAM/GRAM-modul som t ex *Maximem* eller *Superspace*. Det räcker dock med en liten modul som endast har CPU-RAM >6000 till >7FFF. När programmet startas första gången skapas ett huvud som gör att Superbug II kan väljas direkt från TI-startmenyn.
- SBUGO är i compressed DIS/FIX 80-format med relokerbar kod.

Följande kommandon finns:

A	disassemblerar maskinkod
B	sätta brytpunkter i maskinkod
C	läsa och ändra CRU
D	dumpa minne som ASCII och HEX
E	starta maskinkod på adress
F	söka HEX-kod
CTRL/F	söka textsträng
G	ändra GROM-basadress
H	HEX-beräkningar
I	välja textsträng för sökning
CTRL/I	skriva textsträng
J	ändra skärmfärger
K	söka data som ej är lika
L	aktivera/stänga printer m m
CTRL/L	ladda in programfil
M	läsa och ändra minne
N	flytta minnesblock
O	ändra utenhet
P	jämföra minnesblock
Q	QUIT
R	läsa och ändra WP, PC, SR
S	enkelsteg genom maskinkod
CTRL/S	spara minne som programfil
T	skifta mellan SUPERBUG och egen skärm
U	ändra basic-offset
V	köra maskinkod tills värdet uppnås
W	läsa och ändra register
X,Y,Z	tillägg till adress
" "	decimalt till hexadecimalt
>	hexadecimalt till decimalt

Alla dessa kommandon kan adressera valfritt till CPU-minne, VDP-minne eller GROM-minne. Utskriften kan fås mot skärm, printer eller disk.

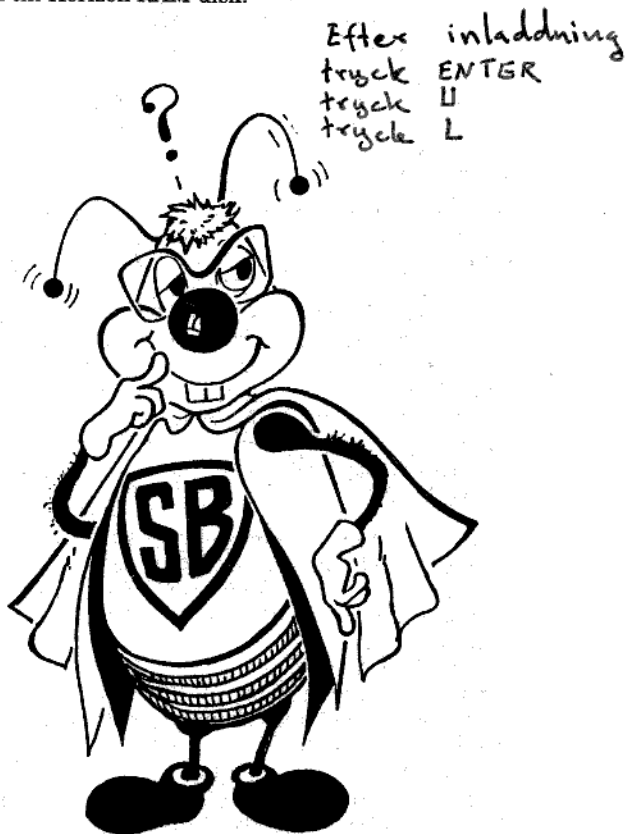
En mycket användbar funktion är att stega igenom ett assemblerprogram och se vad som händer med minne, adresser och register. Du gör det genom att sätta brytpunkter (max 11 st) eller genom att stega en instruktion i taget.

Kommandona M och D dumpar ut minnet med hexadecimal kod och ASCII-kod. Detta kommer ut i en mycket kompakt form så att mycket ryms samtidigt på skärmen eller en utskriven sida. Det är alltid lämpligt att undersöka minnet på detta sätt innan man börjar disassemblera med A-kommandot.

Det finns möjlighet att söka upp hex- eller ASCII-sträng i minnet samt även jämföra olika minnen med varandra. Jag har använt K-kommandot för att se vilken del av min GRAM-Kracker som ej håller minnet. Jag initierade först hela minnet med FFFF och sökte sedan upp vilka adresser som ändrats. K-kommandot skriver endast ut vad som ändrats. Resultat var att GRAM-adress >8000, >A000 och >C000 ändras om datorn stängs av.

Jag har även använt Superbug II till att omvandla DIS/FIX 80-fil till PROGRAM-fil för RAGCTLG (i låga 8 kb EM) som finns på skivorna till MACRO ASSEMBLER. SAVE-programmet från TI går ej att använda eftersom det ligger på samma adress som RAGCTLG.

PROGRAM-filer kan laddas in till valfri adress i minnet d v s man kan bryta mot den normala instruktionen till laddaren som finns på de sex första byten. Du kan t ex ladda in två versioner av samma program och få utskrivet exakt vilka bytes som skiljer sig åt. *Disk Utilities* av John Birdwell skriver alltid ut hela sektorn när du jämför filer vilket inte är lika bra som Superbug II. Hela minnesblock kan flyttas i minnet så det är möjligt att t ex jämföra två versioner operativsystem till Horizon RAM-disk.



- Powerful Assembly Language Debugger
- Leaps Tall Buildings with a Single Bound
- Faster Than a Locomotive
- Even Works in a Phone Booth!

MINIMUM REQUIREMENTS

- TI-99/4A Home Computer
- Extended Memory
- Disk Systems

© 1986 Edgar Dohmann

Menu version 7.1 till Horizon

av Jan Alexandersson

Menu för RAM-disk

Det har kommit en ny version av det menyprogram till Horizon RAM-disk som beskrevs i PB 87-4 nämligen MENU version 7.1 direkt från *Miami TI Users Group*. Det är radikalt förbättrat och alla allvarliga buggar är borta.

MENU består av ett nytt operativsystem ROS som helt ersätter originalfilen till Horizon och ett MENU-program samt CFG som konfigurerar RAM-disken. MENU och ROS tillsammans kommer att ta kontroll över datorn vid alla typer av RESET (QUIT, BYE, RESET-knapp eller strömbrytare) så att MENU-programmet laddas in och körs igång automatiskt utan att färgbalkarna eller startmenyn med 1 TI BASIC osv kommer upp.

CFG-programmet används för att ställa in alla grundparametrar för RAM-disken inklusive initiering. DM 1000 kan ej användas för initiering eller SWEEP DISK medan övriga funktioner som kopiering fungerar. Eftersom operativsystemet har krympts från 12 till 8 kbytes tilldelas RAM-disken 736 sektorer istället för det normala 720 sektorer. För dig som har en ensidig Horizon finns endast 356 sektorer vilket faktiskt är en försämring jämfört med version 6.4.

Den tidigare buggen med första filen på disken är borta och det finns inga problem med filhanteringen. Även interruptrutinen för XB-laddaren är borta så alla Extended Basic-program laddas utan problem.

Du kan även dela upp RAM-disken på två olika logiska RAM-diskar som sedan anropas med olika disknummer. Fördelningen av tillgängliga sektorer mellan de två kan göras helt valfritt (antalet sektorer delbart med 8).

Den andra logiska RAM-disken kan i sin tur delas upp på max nio olika RAM-diskar med olika skivnamn och egna sektor 0 och 1. Dessa kopplas in om anropet görs med DSK.SKIVNAMN.FILNAMN. Man kan även bläddra mellan dessa olika diskar med CALL TD. Om den andra logiska RAM-disken anropas med nummer kommer den disk som senast användes att vara aktiverad. Detta är mycket trevliga saker att ha vid MULTIPLAN som väntar sig att disken skall heta TIMP och PRBASE som kräver en disk som heter PRBASE.

Nya CALL-rutiner

MENU version 7.1 har följande nya CALL-rutiner som ej fanns i V6.4:

- CALL TD skiftar andra logiska disken
- CALL U1-U9 här finns nio användardefinierade CALL som laddar in och kör önskat program (XB eller assembler) varav ett måste heta MENU om du vill att power up skall ladda in MENU. Även CALL DM måste läggas in bland U1-U9 om man vill ha den kvar.

En nyhet i version 7.1 är att alla dessa CALL även kan anropas som DSR-rutin. Du kan anropa dessa i alla program som har någon typ av SAVE, LOAD eller DELETE som tex TI Writer. I Basic eller XB kan du göra det så här:

```
OLD U3
DELETE "U3"
RUN "U3"
SAVE U3      (om minnet ej är tomt)
OPEN #1:"U3" (i program)
CALL LOAD("U3") (om CALL INIT har gjorts)
```

Assembler-laddare

Denna OPTION-laddare har radikalt förbättrats och laddar utan problem alla nyttoprogram som tex PB-FORTH och RAGCTLG. Problemen med laddning av vissa spelprogram kvarstår dock beroende på att VDP inte initieras.

CALL- och DSR-laddaren eller val på tredje menyn ställer ej in VDP-registren på samma sätt utan följande program kommer att få blank skärm: TOMBSTONE CITY, BEYOND PARSEC, CUBIT och MUNCHMAN (efter det att tangent har tryckts). Alla program som själva initierar VDP fungerar dock helt utan problem.

Buggar i MENU

Förutom problemet med att endast sista valet i GROM3 visas och att assembler-laddaren ej ställer in VDP-minnet finns några små buggar. CALL- och DSR-anrop av egna namngivna rutiner sökes igenom med början i CRU >1000 medan val av samma rutin på tredje menyn först söker igenom efterföljande CRU. Om du använder två Horizon RAM-diskar på CRU >1000 respektive CRU >1200 och har MENU-ROS på den första och original-ROS på den andra kommer inte DM på tredje menyn att kunna laddas från första RAM-disken eftersom den försöker starta det från CRU >1200 där det inte finns något program.

Om du delar upp RAM-disken i två logiska diskar så är det den andra disken som sökes först med anrop av typen CALL DN(3) medan den första endast nås med det mer detaljerade CALL DN.1(3) där man efter punkten anger exakt vilken drive anropet gäller. Detta typ av anrop används även då du har mer än ett HORIZON-kort. CALL DN.1(3) fungerar dock endast från Basic men ej i Extended Basic (kommandomod).

De tidigare beskrivna problemen med power up av Editor/Assembler är lösta men problemet med Terminal Emulator II kvarstår så att SPEECH kommer ej att fungera. BACK från MENU ger dock power up av det som sedan väljs.

Ombygd Horizon med 392 kb

Jag har även lyckats använda MENU och ROS version 7.1 genom att dela RAM-disken i två logiska diskar. Den första på 192 sektorer och den andra på 1440 sektorer. DS/DD-disken med sektorer fungerar perfekt. Den första disken med 192 sektorer innehåller ett antal adresser som är ofullständigt avkodade vilket gör att det du skriver på en sektor även kan läsas på en annan sektor. Test med Disk Manager 2 COMPREHENSIVE TEST ger 112 fel med BAD ADDRESS/CODE. QUICK TEST och initiering avslöjar inga felaktiga sektorer.

Dessa problem beror på att det endast sitter en 8 kbytes-chip i position U11 så ett byte till 32 kbytes kanske är en lösning. Genom att markera de olämpliga sektorerna som upptagna i sektor 0 kan man eventuellt använda 72 av diskens 192 sektorer om nu inte sektor 0 och 1 tillhör de olämpliga. Delar av operativsystemet finns på sektor 0 och delar av ett teckenset finns på sektor 1.

Jag har provat 392 kb-disken med Horizons originaltest-program. Det visar fel på position U8, U16 och U7 med 8 kbytes både med enkel och dubbelsidig test.

MENU version 7.3

Ytterligare en version av MENU-programmet fick vi i början av februari direkt från Miami. Mer om detta i ett kommande nummer.

Tyska återförsäljare

Under oktober 1987 anordnades en mäsas för TI-99/4A i Bremen i Västtyskland. Det var endast två firmor som ställde upp och visade vad de hade att sälja:

HAGERA
Rausch Vertrieb
Postfach 320313
D-5300 BONN 3
Västtyskland

LEGIO COMPUTER CENTER
Albert Visser
Haagweg 169
NL-2281 AJ RYSWYK
Nederländerna

Hagera är det nya namnet på Rausch & Haub som nämndes i PB 87-3.

STAR

Recension av Anders Josefsson

Super TI Assembler Routines version 1.0

Ett program från Com-Link Enterprises.

Skrivet av Michael Riccio

STAR är en diskett med en hel hög assemblerrutiner till Extended Basic. Hela 53 st! Det finns rutiner för laddning av olika teckenset, ljud, kassettmotorkontroll, färger, direkt skrivning och läsning av VDP-minnet, skärmtrick, programstyrning, tangentbordskontroll, skrivning & läsning direkt till och från diskett, text-mode plus lite till.

Då trycker vi in disketten i driven får vi se vad som händer. Disken har ett LOAD, så man startar från scratch. Rutinerna laddas in, och när det är klart har man en snygg

grafik-bild på skärmen. Man trycker en tangent, och får en text med riktiga små bokstäver! Det har man ju inte sett förut utan att definiera om de små tecknen. Det räcker med att skriva CALL LINK("LOW"), så har man dom!

Men, en del av rutinerna finns också till Mechatronics XB II+, t ex direkt skrivning & läsning av VDP-minnet, skärmhantering och sprites. Även med den har man ju nytta av alla andra rutiner.

Manualen är mycket bra med ordentliga förklaringar till alla rutiner. Så man kan knappast göra fel vid användning av rutinerna. Man får veta det minimala systemet för användning, vilket är dator, expansionsminne, diskkontroll, kassettbandspelare. Rutinerna fungerar med MYARC 128/512K och RAM-disk, säger manualen. CORCOMPs och MYARCs diskkontrollkort är inte testade däremot.

Näväl, vi ska titta på en del smått fantastiska rutiner, såna som är unika annars. Alla rutiner är av CALL LINK-typ, så det skriver jag inte ut. Här kommer ett urval:

Rutinerna till olika teckenset:

"LGCAPS"

"LOW"

Laddar in titelskärmens stora tecken.

Riktiga små tecken med underslängar.

Färgrutiner:

("COLORS",fgr,bgr)

("BURST",tid)

("FLASH",fgr,bgr)

Ändrar färgen på alla 15 teckenseten.

Blåddrar igenom alla färger i rasande takt, vilket ger en enorm effekt.

Alla små tecken blinkar i dom färger Du matade in. Fortsätter att blinka tills CALL LINK ("NORMAL") beordras.

VDP-skrivning, läsning:

("VDPREG",register,byte)

Skriver direkt till något av de 7 VDP-registren.

Skärmtrick:

("RDSCR",strängfält)

("WTSCR",strängfält)

("SAVSCR",filnamn)

("GETSCR",filnamn)

Dumpar skärmen till en strängmatris med 24 fält DIM A(24), rad per element.

Skriver tillbaka strängfältet till skärmen.

Dumpar skärmen till ex disk, RAM-disk tillåten

Skärmen tillbaka.

Kontroll-rutiner:

"BYE"

"NEW": END

"NOQUIT","QUIT"

"NOMOVE","MOVE"

De är 6 st som styr programmet till viss del.

Går direkt till titelbilden, kan användas i programmet.

Sekvensen går till kommandomode med raderat program.

Kopplar ur resp in FCTN =.

Stoppar resp får igång sprites. Alla vet väl hur svårt det kan vara att få igång sprites samtidigt. Nu går det.

Tangentbordskontroll:

"LOCK","SHIFT","CTRL","FCTN"

Gör en keyscan och kollar upp Alpha Lock, Shift, Control och Function-tangenterna. Du får en etta om någon av tangenterna är nedtryckt, och en nolla om inte, tillbaka i en variabel.

Teckendefinitioner:

"COPY",cha,cha

Kopierar över definitionen för tecken till 1 till tecken 2. Samma som följande sekvens: CALL CHARPAT(CHAR1,X\$):: CALL CHAR(CHAR2,X\$) men COPY gör det bra mycket snabbare och med mindre minne.

"ROTATE",turns,char(ant)

Roterar ett eller flera tecken i 90 graders hack, 1 till 3 turns. 1 ger 90 graders vinkling, 2 ger 1/2 varv och 3 ger 3/4:s varvs vridning av tecknet eller tecknen.

"INVERT",char(ant)

Inverterar tecken, så bakgrundsfärg blir förgrund och vice versa.

"FLIP",char(ant)

Vänder upp och ner på ett antal tecken!

"MIRROR",char(ant)

Spegelvänder tecknen i sidled.

Disk Access:

"CAT"

Ger diskkatalogen på skärmen.

"PRO",DR,filnamn

Skrivskyddar filen i drive DR.

"UNPRO",DR,filnamn

Tar bort skyddet.

"RENAME"

Omdöpning av fil.

"RSEC",DR,sektor,\$1,\$2

Läser en sektor från disk till strängar 1 & 2. De 128 första byten tas i sträng 1, och de 128 sista i sträng 2.

"WSEC"

Skriver tillbaka från strängarna till disk

Text mode:

"TEXT"

Vips! In i text-mode. Bestäm färger med "SCREEN".

"PRINT",rad,kol,sträng

Skriv var som helst på skärmen i Text-mode

"GRAPH"

Vips tillbaka till grafik-läge.

Sträng-hantering:

"CAPS"

Ändrar alla små tecken till stora i en sträng.

REVRSE"

Vänder strängen bak och fram!

Det var en del av de 53 *mycket* bra hjälprutinerna i STAR. Detta måste vara en av de bättre hjälpmedlen till alla som skriver XB-program. En del rutiner är ju unika förbättringar i ett program.

Rekommenderas varmt till XBasic-skrivarna.

Får också högt betyg vad gäller manual, den är mycket bra skriven, och ger alla rutiner en ordentlig förklaring.

På disken finns också 9 testprogram, som använder en del rutiner.

Dessutom en fil som heter VDPUTIL2, vilken gör att Du

kan köra vanliga Basic-program i Extended. Det MERGE-as in i ett Basic-program.

Avslutningsvis vill Jag säga att jag har inte gjort något program ännu som använder rutinerna, men jag har sett testprogrammen. Och det räcker för mig för att säga att dessa rutiner är fantastiska!

Betygssättning:

Dokumentation	+++++
Användbarhet	+++++
Funktion	+++++
Totalt värde	+++++

Myarcs disk-kontroll kort

av Sören Bernle

Som nämndes i Anders Perssons Geneve artikel i PB 87-1, fungerade inte våra (Sven Lundgrens och mitt) Myarc kontrollkort tillsammans med Pascal, därtill hade Sven även andra problem med kortet.

Efter att ha berättat för Myarc, om våra problem med korten, både per telefon och brevledes, i det senare fallet med bifogad "inlägga" av Anders Persson, fick vi tillsänt oss, efter lång väntan, varsin ny EPROM-krets, som skulle skiftas ut mot den gamla på kortet.

Att skifta krets visade vara enkelt. Man behövde bara lyfta ut den gamla kretsen ur sin hållare och sätta i den nya, samt kontrollera att den blivit monterad på rätt håll och att alla benen kommit på plats.

Med den nya kretsen fungerar kortet helt klanderfritt. Att köra eller programmera i Pascal går utmärkt och Svens problem är som bortblåsta.

Fördelarna med Myarcs kontrollkort jämfört med TIs:

Enär Du som läser detta kanske vet mindre än jag och för att underlätta mitt skrivande, visar jag nedanstående tabell, som skildrar lagringsförmågan på en 5 1/4 tums diskett under olika betingelser

Beskrivning på engelska	Förk.	Drivverk	Utrymme
Single-Sided-Single-Density	SSSD	40 spårs	90 kb
Double-Sided-Single-Density	DSSD	40 spårs	180 kb
Double-Sided-Double-Density	DSDD	40 spårs	360 kb
Double-Sided-Quad-Density	DSQD	80 spårs	720 kb

Myarc kortet kan hantera disketter upp till DSDD, mot DSSD för TIs.

Det kort jag och Sven har i vår ägo kallas för quadcard och kan hantera disketter upp till DSQD, fast för att nå quad-density krävs det 80 spårs diskettmaskiner, vilket ingen av oss ännu har.

Jag hoppas att Du via ovanstående, snabbt har insett den största fördelen.

Angående dubbel densitet, så kan kortet hantera både 18 och det betydligt ovanligare 16 sektors formatet. Vad det är för bra med det senare begriper jag inte!

Därtill klarar kortet av att hantera 4 diskettmaskiner + RAMdisk mot, att TIs endast klarar 3 diskettmaskiner.

Utöver detta kan man i kommandomode eller i Basic, utnyttja följande 4 kommandon:

CALL DIR(#) utträttar det samma som Catalog kommandot i en diskmanager. Man kan alltså med hjälp av CALL DIR, se vilka filer som finns på disketten i den drive som #-tecknet blivit ersatt av, utan att behöva ladda en diskmanager och utan att det program som eventuellt finns i minnet blir påverkat.

Exempel: I det program som Du håller på att skriva, skall en fil anropas som finns på disketten i drive # 1, men Du minns inte filens exakta namn.

Gå då över i kommandomode om Du inte redan är i det och skriv CALL DIR(1) följt av <enter>, varpå diskettens innehåll blir listat på skärmen. När Du funnit vad Du sökt, kan Du fortsätta med programskrivningen där Du avbröt den. CALL DIR kan även ingå i ett program om man så vill, jag har dock själv inte kommit på något skäl till att låta det göra det.

3 kommandon som möjliggör körning av assemblerprogram från kommandomode eller i Basic.

CALL ILR utträttar exakt samma som CALL INIT i TIXB. CALL LR utträttar exakt samma som CALL LOAD i TI XB. CALL LLR utträttar exakt samma som CALL LINK i TI XB.

När modulbaserade program skall sparas via programmet CSAVE, för att senare kunna köra dem på Geneven, kommer två av dessa kommandon väl till pass.

LLR-kommandot utnyttjas inte här eftersom CSAVE är autostartande.

Efter att modulen med det program man vill spara är på plats i modulporten och disketten med CSAVE-programmet är placerad i drive # 1, förfar man enligt följande:

Välj TI BASIC från menyn.

Skriv CALL ILR följt av <enter>.

Skriv CALL LR("DSK1.CSAVE") följt av <enter>.

Efter en kort stund går CSAVE-programmet igång. Vad som skall ske vidare härifrån har jag beskrivit tidigare.

DISKMANAGERn som följer med kortet är diskettbaserad och vida överlägsen TIs DM2, vilket väl inte säger så mycket. Det är nog bättre att jämföra den med DM1000 ver 3.5. De innehåller ungefär samma kommandon, det som skiljer dem åt mest är, att man med Myarcs manager vid formateringen kan välja hur sektorerna inbördes skall vara arrangerade på spåret (se *Advanced Diagnostics* manualen).

Det var allt!

Produkter från Rave 99

Rave 99 Co
112 Rambling Road
Vernon, CT 06066
USA

Denna firma tillverkar vissa unika tillbehör till TI-99/4A. Bl a finns:

Keyboard Enhancement Model 99/105 kostar USD 200 (i USA) och USD 230 (till Sverige). Detta är ett fristående tangentbord som ersätter det inbyggda i TI-99/4A.

Speech Synth. Adapter Card för expansionboxen kostar USD 50 (USA) respektive 55,50 (Sverige)

Memory Enhancement Card MX01/64 kostar USD 200 respektive USD 215.

Memory Enhancement Card MX01/288 kostar USD 300 respektive USD 315.

Memory Enhancement Card MX01/544 kostar USD 400 respektive USD 415.

Dessa minnekort används som RAM-disk men har även:

8 kb non-DSR >4000 till >7FFF

8 kb minne >6000 till >7FFF

8 kb System DSR

8 kb User DSR

32 kb Main memory (expansionminne ??)

Det finns 17 hållare för 32 kb statiska RAM-chip dvs totalt 544 kbytes. På dessa priser tillkommer 5 % på både USA-priset och Sverige-priset.

Du som vill ha en kopia på RAVE 99-katalogen (8 sidor) kan sända ett frankerat svarskuvert för 100 g till:

Jan Alexandersson
Springarvägen 5, 3 tr
142 00 TRÅNGSUND

Geneve-rapport

av Sören Bernle

Genom ett havandeskap på 9 månader, är jag nu i ett ägandeskap av en GENEVE-dator!

Efter att detta ägandeskap varat i drygt två månaders tid, känner jag mig nu mogen att avge rapport.

Jag tänker här nedan berätta för Dig, det jag vet och kan om Geneven och vad den kan.

Att hårdvarubeskrivningen kommer att vara ytlig i dubbel bemärkelse har Du väl redan förstått!

Jasså inte??????

Följande ingår i köpet av en Geneve:

Datorkort.

Tangentbord.

Manual.

samt disketter innehållande:

MDOS

Myword

Multiplan (upgrade)

CSAVE (Cartridge SAVE program)

Myarc Disk Manager III ver 2.0

Advanced BASIC *

Pascal (runtime) *

* = Ännu ej levererade!

För att kunna köra Geneven krävs minst:

Expansionsbox.

Diskkontroll-kort.

Diskettmaskin (minst 1 st).

Analog RGB-monitor (som klarar 60 Hz).

Monitor-kabel.

Ytterligare hårdvara man kan behöva:

RS232-kort.

Printer.

Modem.

Mus.

Joysticks.

Prator.

Min uppkopplade utrustning består av:

Expansionsbox.

Geneve kort.

TI RS232-kort.

Myarc Quad Disk-kontroll-kort.

2 fullhöjds DSDD Tandon diskettmaskiner (i egen separat låda).

Philips RGB-Monitor CM 8533.

Taxan printer (parallell-kopplad).

Myarc Mus.

TI Joysticks.

Hemmagjorda kablar till monitor och printer.

Geneve-datorn och expansionsboxen

Datorkortet ser ut som tidigare beskrivits i PB 87-1 sid 8-9), förutom på två punkter.

Klockan med sitt batteri är nu på plats och överbygglings-trådarna är reducerade till 1 st.

Innan datorn placeras i boxen måste expansions-, minnes- och P-kodskortet plockas ut.

Skall den anslutas till en RGB-skärm, måste en "jumper" kopplas om, (gör man inte det, går det ingen röd signal ut och bilden blir verkligen rödfattig).

Datorn placeras sedan på expansionskortets tidigare plats, dvs utrymme nr 1. Därefter kopplas tangentbords-, monitor-, mus- och joysticks-kablarna in på sina respektive kontakter på dess baksida.

Mitt omdöme om Geneve-kortet

Negativt är:

1. Att videosignalen inte är anpassad till PAL-systemet (se under Monitor, längre ner i texten).

2. Att anslutningarna och själva kretskortet tillsammans sticker ut ca 9 cm utanför boxens bakkant. Risker för att

någon form av drullighet kan orsaka ett brott på kretskortet verkar på mig rätt stor. Två åtgärder bör nog vidtas.

A. Dragavlasta samtliga till kortet anslutna kablar på något stabilt ställe bakpå boxen.

B. Skydda det "utstickande" genom att placera boxen så att ingen kan "råka" komma åt det eller ännu bättre med något fast förstyvande ytterhölje.

3. Att kortet måste lyftas ur boxen vid eventuell transport. Skäl samma som ovan. Avsaknaden av fingerkrokar på kortfodralets ovansida. OK, det är en liten struntsak, men man saknar dem vid de tillfällen man ska lyfta ur kortet.

Positivt är:

1. Packningstätheten. Trots att kortfodralet håller samma mått som TI originalkortet, innehåller det så fantastiskt mycket mer. Bl a 640 kb RAM jämfört med TIs 48 kb RAM (konsol + minneskort).

2. Utrymmet i boxen har så att säga utökats med plats för ett kort till, (inget separat expansionskort).

3. Ingen "symaskinspedal" längre. Boxen står för all strömförsörjning.

4. Bara en brytare, boxens brytare, för att knäppa på och av datorn.

Tangentbordet är ett äggskalsfärgat så kallat PC/XT/AT tangentbord, av märket UNITEX K-155 och är tillverkat i Taiwan.

Det har formen av en 20 cm bred 52 cm lång tunn något lutande plast-planka som kan lutas något ytterligare via två utfällbara ben på undersidans bakkant.

Det är anslutet till Geneven via en DIN-kontakt och en ca 3 m lång spiralkabel (ca 4 mm i diameter).

På bakkanten av det finns en XT/AT-omkopplare som skall stå i läge XT för att fungera korrekt ihop med Geneven.

Det är försett med 95 tangenter fördelade i fyra olika funktionsblock.

10 st funktionstangenter på rad längst upp.

18 st numeriska tangenter längst till höger.

11 st markör-påverkande tangenter till vänster om de numeriska.

56 st "egentliga" tangentbordstangenter längst till vänster.

Därtill finns det längst upp till höger 3 gröna lysdioder för indikering av CAPS LOCK, NUM LOCK och SCROLL LOCK.

Mitt omdöme om tangentbordet

Negativt är:

1. Att tangenterna känns något sladdriga jämfört med TI-99/4A.

2. Tangenterna för åäö sitter förvisso mer samlade här än på TI-99/4A och var bokstavs versal och gemen finns på samma tangent, men de är placerade något i skymundan.

Positivt är:

1. Separata markör-tangenter.

2. Separata numeriska tangenter som på en räknedosa.

3. Separata funktionstangenter.

4. De speciella kommando-tangenterna som Print Screen, Page Up, Page Down, End, Ins, Del, Tab etc.



5. Ergonomiskt, betydligt bättre.
6. Flyttbart under arbetet, utan risk för datorkollaps.
7. "Brandslångs"-anslutningen från konsolen till boxen är borta.

Manualen. Består egentligen av tre manualer:

1. **THE MYARC 9640 manualen** (84 sid), beskriver på: 5 sidor datorsystemets hårdvara på en mycket elementär nivå.

4 sidor hur man kopplar upp datorsystemet.

6 sidor tangentbordets funktioner.

60 sidor Disk Operativ Systemet (DOS), mycket ingående och förklarande.

2. **Myarc Advanced BASIC-manualen** (254 sid), består till största delen (219 sid) av förklarande text till samtliga programinstruktioner, från ABS till WHILE. Var programinstruktion är fylligt och väl beskriven och ofta åtföljd av ett eller flera små programexempel.

Basic-manualen avslutas med 13 olika bilagor (25 sid). De tio första bilagorna skiljer sig knappast alls från vad som står att läsa i TIBX-manualen, de tre sista innehåller däremot något helt nytt, nämligen:

Ett diagram över de nio olika grafikmoderna.

Ett program som illustrerar mus-kommandon.

En tabell över de ASCII-koder som CALL KEY mode 6 ger.

3. **Technical Reference** (18 sid), uppfattas nog rätt tunn av Dig som begriper mer än jag.

Min totala avsaknad av kunskap om vad sker inne i en dator, har det goda med sig att jag slipper redogöra för vad som står på de första 17 sidorna.

Resten av manualen innehåller kopplingsanvisningar till kontakterna bak på datorkortet.

Mitt omdöme om manualen

Negativt är:

1. Det idiotiska lösblads-pärm systemet som amerikanerna använder! Manualen levereras i samma sorts hållförstörande och pappersförstörande pärm som E/A, Multiplan- eller TI-Writer-manualen levererades i.

2. Den något tunna tekniska manualen.

Positivt är:

1. Den håller nästan TI-standard, vilket väl får anses som ett gott betyg.

2. I basicmanualen finns vid var programinstruktion även hänvisningar till andra programinstruktioner som står i nära samband med den beskrivna programinstruktionen.

Exempel:

RETURN

Cross reference

GOSUB, ON GOSUB, ON ERROR

Musen som är äggskalsfärgad och treögd (3 tangenter) och har en alltför kort svans (läs anslutningskabel), är främst köpt för framtida behov.

Advanced Basic innehåller muskommandon, därtill uppges det att det utvecklas en hel del övrig mus-styrd programvara.

Tack vare en programsnutt som Jan Alexandersson inhämtat från MICROpendium och delgett mig, kan jag redan nu använda den i TIBX program.

Monitor. Eftersom datorn kan producera en skärm med 80 tecken i bredd eller med 256 olika färger, krävs det en monitor med hög upplösning för att upplevelsen skall bli både njut- och/eller läsbar.

Förvisso var jag lovad en dator som var anpassad till det europeiska PAL-systemet, vilket skulle medföra att jag skulle kunna använda en vanlig TV som skärm om den anslöts

via en RF-modulator. Upplösningen skulle dock inte bli något att yvas över.

Tyvärr visade det sig att datorns videoutgång inte fungerar tillsammans med en PAL TV.

I varje fall inte vid de försök som gjordes. Allt pekar på att datorn även på videoutgången kör med 60 Hz i bildfrekvens, vilket medför att bilden blir helt oläslig.

Jag bestämde mig för att skaffa en RGB-skärm, redan innan jag fått Geneven. Jag visste alltså inte då om problemet med videoutgången.

Jag valde en Philips CM 8533 monitor av följande skäl:

Den är högupplösande och klarar 60 Hz bildfrekvens.

Den har en inbyggd ljudled.

Den har ingångar för RGB-LIN+AUDIO, RGB-TTL, RGBI-TTL, vilket medför att jag troligen inte behöver köpa någon ny skärm om jag skulle få för mig att byta till något annat datorsystem.

Därtill klarar monitorn av CVBS vilket innebär att den går att använda som vanlig TV om man kopplar den till tex en videomaskin.

Jag har alltså datorn inkopplad på den analoga RGB-ingången och videoapparaten inkopplad på CVBS-ingången.

Dvs monitorn fungerar antingen som en data- eller video-monitor.

Hur ser då bilden ut?

Som datorskärm till Geneven är alla tecken (80 kolumner) knivskarpa, samtliga färger har hög briljans, vart pixel står helt stilla när interlacet är inkopplat, (vilket regleras från tangentbordet).

Som videomonitor är bilderna betydligt skarpare och bättre färgmättade än på en vanlig TV.

Nackdelar beroende på datorns bildfrekvens.

1. På fronten av monitorn finns en omkopplare för RGB/CVBS. Om datorn är igång och man väljer CVBS, så uppstår samma problem som när man försöker köra datorn på videoutgången fast här omvänt. Dvs monitorn kör i 60 Hz bildfrekvens medan videomaskinen kör i 50 Hz, bilden blir alltså helt oigenkännbar.

Ma o monitorn kan endast användas som videomonitor när datorn är stängd eller bortkopplad.

2. Placerar monitorn ovanpå expansionsboxen uppträder det störningar i form av en i sida gungande bild. Detta beror troligen på att det magnetiska 50 Hz-fältet som trafön i boxen alstrar, negativt påverkar monitorn som kör i 60 Hz.

Placerar man monitorn en bit från boxen, försvinner dessa för ögonen mycket obehagliga gungningar. Troligtvis går det att skärma av dessa magnetiska störningar med en 3-5 mm tjock järnplåt.

(Har provat med en 1 mm tjock bakplåt, blev bara något bättre).

Sakkunnig vid ovanstående monitor uppkopplingsförsök och konklusioner var Lennart Thelander.

Att jag här har nämnt andra användningsområden för en datamonitor beror på att den kostar en del (jag köpte min på rea för 2800 SEK, listpris ca 4000 SEK) och att det gäller att förklara den utgiften både för sig själv och de man eventuellt lever ihop med.

Monitor-kabeln består av en mjuk 8:a ledare med en 8:a polig DIN-kontakt för anslutning till datorn och en SCART-kontakt för anslutning till monitorn. Efter att ha rådfrågat Lennart Thelander, dristade jag mig till att löda på kontakterna själv och så det fungerade!

CALL MYARC



Vad kan Myarc-grejerna uträtta?

Innan TI-99/4A systemet plockas ner, är det kanske inte så dumt att överföra sina modulprogram till diskett som Geneven senare kan köra i TI-mode?

CSAVE är programmet som klarar av detta!

Detta program skall köras på TI-99/4A datorn.

Tillvägagångssättet under programexekveringen beror på vilket diskkontrollsystem datorn är utrustad med.

Diskkontrollsystemen kan delas upp i två grupper:

1. Myarc, Corcomp med MG EPROM och Horizon RAM-disk.

2. TI och Corcomp utan MG EPROM.

Det som skiljer grupperna åt är, att man med grupp 1 kan ladda och köra assemblerprogram från Basic, och att grupp 2 inte har den möjligheten.

Med diskkontrollsystem som tillhör grupp 1, sättes modulen, vars program man vill spara, in i konsolen. Efter att ur huvudmenyn ha valt Basic, laddas och körs assemblerprogrammet CSAVE.

Med de system som tillhör grupp 2, måste man först ladda CSAVE med någon av modulerna E/A eller TI-Writer via "option" 5 resp. 3. Därefter får man försiktigt skifta till den modul vars program man vill spara.

OBS! Modulens kontaktyta nr 1 måste vara övertjeppad för att förhindra ett "reset"-kommando till datorn.

Efter att programmet fått uppgifterna, om till vilken drive och under vilket namn modulprogrammet skall sparas. Så "simsalabim" efter en kort stund ligger det på diskett och väntar på att bli kört i Geneven.

Om TI-Extended Basic blivit sparad på en tom diskett under namnet TIXB, kommer disketten (beroende på att programmet är så stort så att det måste fördelas på flera filer), att innehålla 6 st filer med namnen TIXB, TIXB1, TIXB2, TIXB3, TIXB4, TIXB5.

Senare anropning av programmet sker förstås endast med filnamnet TIXB.

Alla program som jag överfört till diskett och testat har fungerat helt perfekt, förutom Schack-modulen som inte vill spela efter reglerna, (se VIDEO CHESS längre ner i texten).

Efter att samtliga modulprogram blivit sparade är det dags att koppla ned TI-99/4A:n och koppla upp Geneven. (Hur? se ovan!).

Stoppa in MDOS-disketten i drive #1, slå till strömmen till monitorn och boxen.

I UPPSTARTnings fasen sker följande:

Skärmen uppfylls av en simmande svan, målade i en mångfald färgnyanser.

Datorn checkar i några sekunder de interna systemen och laddar därefter DOSen. Så snart laddningen är klar är man inne i MDOS och skärmen innehåller följande:

Myarc Disk Operating System (MDOS)
Version 1.00

18:56:23 Saturday 12-19-87

A>

Skärmen är ställd i 40 kolumners mode och efter A står markören blinkandes.

Jag kommer i det följande att redogöra för vad MDOS ver 1.0 kan uträtta, och inte för vad manualen påstår!

DOSen kan hantera upp till 7 st separata diskdrive enheter, varav 2 st kan vara Horizon RAM-diskar, de anropas endast med en bokstav. A = DSK1, B = DSK2 ... G = DSK7. Vill man se vad som finns på disketten i DSK1, skriver man bara i kommandot DIR direkt efter "promptet" A> och trycker på enter.

Bokstaven framför "promptet" >, talar om vilken diskdrive DOSen anropar om den inte kan utföra kommandot internt.

Vill man rikta DIR-kommandot till en annan ansluten drive, ex vis. DSK2, skriver man så här DIR B:.

Vill man att det skall vara diskdrive B, dvs DSK2, som DOSen skall auto-anropa, ger man bara kommandot B: och vips, före "promptet" står nu B.

Som diskmanager tror jag DOSen innehåller alla kommandon man kan förvänta sig av en bra sådan! Den har

kommandon som för mig är helt nya, de utför följande:

Jämföra två disketter och se om de är helt identiska.

Titta på vilken textfil som helst, dvs DIS/VAR eller FIX från 1 till 80 filer.

Wildcard-tecken, för ett (?) eller för flera (*) tecken. Exemp. A> DIR ?K* visar på skärmen alla filer på disketten i DSK1 vars andra tecken i namnet är ett K oberoende av hur långt namnet är.

En DOS skall förstås vara något mer än en bra diskmanager och det är den här också fast det fattas en hel del.

Man kan skapa BATCH-filer, kanske inte så snygga men de fungerar. En batchfil kan innehålla vilket som helst av DOSens kommandon och få DOSen att exekvera dem i den ordning man bestämt.

Man kan förstås ställa klockan och ändra datum.

Rensa skärmen.

Ändra skärmen till 80 kolumners mode.

Få datorn att göra en paus.

Avbeställa ett tidigare begärt kommando.

Bläddra i kommandostacken. Dvs se vilka kommandon som tidigare blivit utförda och eventuellt utföra dem igen genom att bara trycka på enter när det önskade kommandot syns på skärmen.

Skriva ut korta meddelanden direkt på printern.

Sätta anropsnamn till printern.

Sätta anropsnamn till modem, plus en hel del andra DOS-handhavande kommandon.

Subdirectories möjligheten har jag inte kollat än. Den är väl främst tänkt att användas tillsammans med en hårdisk.

Vad som skall finnas enligt manualen men som ej är nåbara i MDOS ver 1.0 är följande:

Graphic Print = screendump.

Print Screen = textdump.

Scrolling the Screen = se det bort-"scrollade" igen.

Snygga BATCH filer.

Avsätta minnesarea till RAM-Disk funktion.

Avsätta minnesarea till SPOOL-funktion.

Ställa datorn i TI-99/4A MODE.

Ställa datorns körhastighet.

Kan jag då inte köra några program?

Jo det kan jag, men då måste först en GPL-tolk laddas, vilket åstadkommes genom att efter A> skriva GPL.

I och med att GPL:n har laddats har också ett 180 kb stort minnesutrymme avsatts som RAM-Disk med anropsnamnet DSK5.

Datorn lämnar nu MDOS och skärmen ser nu ut så här:

Myarc GPL Interpreter

LOADER 0.97 Speed=3

(c) Copyright 1987, Paul Charlton

Which cartridge do you want to load?

Press <enter> to begin

" F1 to protect >6000

" F2 to protect >7000

" F3 to clear cartridge space

" F4 to change GPL speed

Någon användning av F1, F2 och F3 har jag ännu ej haft!

Med F4 kan man välja mellan 5 olika hastigheter. Vald hastighet avspeglar sig på menyn, i siffran efter Speed.

Den lägsta hastigheten 1 = (jämförbar) med TI-99/4A.

Hastigheten ökar med ökat värde. Den snabbaste hastigheten är alltså 5, som är 3 1/4 ggr snabbare än TI-99/4A.

Trycker man på någon annan tangent än en F-tangent, raderas nedre hälften av skärmen ut och en markör blinkar under den 4:e textraden.

Här står man vid vägskalet, skall datorn köra i TI-Basic eller skall något annat program eller programspråk laddas?

Datorn kan i det här läget jämföras med TI-99/4A utan någon modul inpluggad. Dvs trycker man på enter kommer man in i TI-Basic som redan finns i "konsolen", (blev laddat tillsammans med GPL-tolken).

Vill man plugga in en modul, dvs ladda sparade moduler, är det här det skall ske. För att ladda exvis TI Extended-Basic sparad under namnet TIXB och denna diskett finns i i diskdrive 2, så skriv bara DSK2.TIXB och tryck enter.

Vissa program som MY-Word och ett gammalt Myarc Musdemo program som jag har, måste laddas och köras med E/A modulen "ipluggad".

Vill man byta modulprogram eller ändra körhastigheten, trycker man samtidigt på Control och de båda Shift tangenterna och vips är man tillbaka till GPL skärmen.

Diskettfiler. Alla filer skapade och/eller lästa av Geneven, har tillägget timme:minut:sekund månad/dag/år, både när de blev skapade och senast uppdaterade.

Gamla TI-99/4A filer har värdet noll på alla positionerna, för såvida de inte under körning på Geneven blivit uppdaterade eller sparade under ett filnamn som ej finns på disketten.

RAM-disk

I alla av ovanstående beskrivna program där man kan anropa en diskdrive, kan man också anropa RAM-disken med namnet DSK5.

Fördelen med en RAM-disk är snabbheten, den är mycket snabb.

Exemp. Att spara en 162 sektorer stor fil i DIS/VAR 80 format på en DSDD diskett, tar 75 sek i tid mot 12,2 sek på Geneve-RAM-disk, dvs mer än 6 ggr snabbare.

Nackdelen är det något begränsade minnesutrymmet, men kanske främst att innehållet inte är bestående om och när strömmen bryts till boxen.

Trots det senare är fördelarna betydande främst när man sysslar med compilerande eller assemblerande programvara.

Körningar i olika språk

TI Extended BASIC och TI BASIC

Frånsett möjligheten till olika exekveringshastigheter och effekter därav, uppför sig programmen som om de vore körda på TI-99/4A.

Fortfarande är skärmen 32 kolumner bred och antalet färger endast 16 st.

Åtkomstmöjligheterna är dock markant förändrade på fyra väsentliga punkter:

1. 7 st diskdrivesenheter kan anropas, även den "inbyggda" RAM-disken som i min nuvarande uppsättning anropas med DSK5.

2. Det ovanstående medför tyvärr att tillgången till minnesutrymme minskar i "konsolen" med 2088 bytes och i "minnesexpansionen" med 23 bytes. Därtill verkar datorn helt immun mot kommandot CALL FILES, den tar tack-samt emot kommandot men rättar sig inte efter det.

3. Ingen möjlighet ansluta bandspelare, varför allt man har på band måste kopieras över till diskett, innan man kopplar upp Geneven.

4. I nuläget, ingen möjlighet att ansluta pratorn. Innehåller programmet exvis subprogrammet CALL SAY, hoppar datorn, efter lång betänketid, över detta och går till nästa instruktion.

Samtliga program jag kört har fungerat klanderfritt (*). En del av dessa anropar hemsnickrade assemblerprogram.

Sättes exekveringshastigheten till 5, går programmen i så hög hastighet att det kan var svårt att hinna med.

(*) Ett undantag. Tangentbordsnedtryckningar tycks bli lagrade i en buffert som subprogrammet CALL KEY hämtar data ifrån. Trots att datorn tycks syssla med annat än tangentet, så är kommunikationen mellan tangentbord och bufferten aktiv ändå.

I ett litet tärningsspel (mot datorn) program jag gjort, kastar jag mina två tärningar via ett tryck på mellanslagstangenten. Skulle någon av tärningarna visa en sexa får jag inte tillgodoräkna mig de poäng som jag fått i denna omgång och det blir datorns tur att kasta.

Vill jag spara uppnådda poäng, trycker jag på enter och det blir datorns tur att kasta.

Om jag skulle hålla mellanslagstangenten nedtryckt en längre stund, tappar jag kontrollen över programmet, eftersom det via repetefunktionen har lagrats en massa space

tecken i bufferten, som CALL KEY måste beta av innan det når fram till det "enter" tecken jag önskar skall gälla. M a o datorn tycks spela helt själv utan behov av min närvaro, och jag blir en given förlorare.

Editor/assembler

Inga problem vare sig med att editera eller assemblera.

Forth

Jaha, nu får jag krypa till korset. Jag har TI-Forth men jag kan inte ett smack om det. Jag minns svårigen hur man startade det och andra Forthprogram på TI-99/4A.

Mina försök med att köra TI-Forth på Geneven gav följande resultat:

TI-Forth 2 lät sig laddas och verkade fungera.

Forth demo lyckades jag inte få igång.

FORIT lyckades jag få igång, men bilden såg inget vidare ut. Det blev ett stort svart censurstreck mitt på och tvärs över skärmen, som programmet i övrigt helt ignorerade. Dvs huvudskärm, kaleidoskop-programmet och ritfunktionerna, lät sig beskådas på de icke "censurerade" raderna.

Vid samtliga återhopp från Forth program till GPL-skärmen, läste sig datorn. Enda sättet att få igång den igen var att knäppa av och på strömmen till den.

Trots min stora okunskap i Forth, så pekar väl mina försök på att Geneven inte klarar av att köra TI-Forth program.

Till tröst för Forth-programmerarna får väl nämnas att DataBioTics sågs vara på gång med ett Forth program, Super-Super 4TH, för Geneven.

C

Oj, dags att krypa till korset igen. Trots att jag har Small C i min ägo, så har jag inte försökt köra det på Geneven, inte ens på TI-99/4A.

Enligt uppgift är dock C ett språk som Myarc satsar mycket på.

Advanced BASIC och Pascal

Ett skäl till att jag inte har provkört dessa har jag nämnt tidigare i den här rapporten, fler skäl har jag inte!

Körningar av TI-99/4A-program

Disk manager 2

Inga problem!

Terminal emulator 2

Tycks fungera! Eftersom jag ännu inte kan använda pratorn och inte äger något modem, kan jag inte testa dessa programmöjligheter, men för övrigt är allt sig likt.

TI-Writer

Inga problem.

Multiplan

Som tidigare nämnts, följer det med två nya Multiplan-filer vid köp av Geneven. Dessa utökar programmets data minne till 41 kb, samt ökar skärmbredden till 80 kolumner.

Programmet arbetar ca 4 ggr snabbare när det körs i Geneven än det gjorde tidigare.

Video Chess

Som tidigare nämnts fungerade inte detta program.

Men tack vare Jan Alexandersson, som från tidskriften MICROpendium inhämtat kunskaper om hur man får programmet att fungera, samt förmedlat dessa till mig, fungerar det nu helt perfekt.

Varför fungerade inte programmet?

Jo enligt uppgift använder programmet >8804 istället för >8800 när det skall läsa i VDP data registret. I TI-99/4A är >8804 en kopia av >8800, men ej i Geneven.

Alltså måste alla adresser med värdet >8804 ändras till >8800.

Detta utföres enkelt och lätt om man har ett diskkontrollkort med möjligheten att ladda och köra assemblerprogram från Basic.

Med hjälp av TI-Debugger ändrar man värdet på 14 ställen i programmet till >8800. Varefter man med hjälp av CSAVE-programmet sparar det ändrade programmet på disk.

Vadå CSAVE? Tidigare var ju sagt att det skulle köras på

TI-99/4A! Då kan man väl inte köra det på Geneven!

Jo det kan man, för Geneven är när GPLn är laddad en simulerad kopia av TI-99/4A.

Eftersom modulen med programmet man vill spara redan är "inpluggad" och i detta fall dess program något modifierat, är det bara att spara igång CSAVE.

En förutsättning är dock att diskkontrollsystemet tillhör tidigare nämnda grupp 1 (se under CSAVE ovan).

ASSEMBLER runtime

Jag har stött på ett program, som bär sig tokigt åt, Demon Attack. Det uppför sig ungefär som schack programmet. Inga problem att ladda och köra igång, men det följer inte reglerna. Kanske samma fel som i schack programmet?

Övriga av mig körda program har fungerat utan några problem.

Körningar av Myarc-program

My-Word

Är väl närmast att se som ett utökat TI-Writer program.

Vilka fördelar har då MY-Word jämfört med TI-Writer?

Jo följande:

En 56 kb stor textbuffert.

80 kolumner och 26 rader stor skärm.

De två extra raderna innehåller följande:

Rad 25: Tabulator linje med tabulator-koder och en markör som visar skärmmarkörens aktuella position på raden.

Rad 26: Ger upplysning om:

1. Aktuellt datum och tid.
2. Hur många procent av textbufferten som är ledig.
3. Den laddade textfilens namn och om den blivit ändrad sedan laddningen.
4. Åt vilket håll sökfunktionerna är riktade (uppåt eller nedåt), Control >AND>AM ger kommandot "ändra riktning".

TABulator koderna har utökats med B och C.

B alstrar ett beep, tänkt att varna för högermarginal.

C. All text till vänster om C blir mottagna som versaler oberoende av hur Caps Lock är ställd.

FindString funktionen har utökats. När den sökta strängen har hittats frågar den: "Stop at this string? Yes or Next". Genom att trycka på N fortsätter sökandet i dokumentet utan att man behöver anropa FS på nytt.

Sökfunktionerna letar upp den sökta strängen oberoende av om den består av versaler eller gemener. Sökfunktionerna har dessutom blivit utrustad med Wild-card karaktären (?), som kan stå för vilken karaktär som helst.

CTRL <R> får markören att hoppa till den högra marginalen.

CTRL <Q> får markören hoppa till dokumentets första karaktär i dess första rad.

FORMATTER programmet nås utan att lämna editorn, därtill kan källfilen anropas direkt från textbufferten och man slipper gå omvägen via diskett. Utöver detta kan man kolla innehållet på en diskett utan att lämna formateringsprogrammet.

Help. Det finns hjälpskrmar laddade i RAM, som kommer hypersnabbt vid anrop.

ProgramEditing detta tillval underlättar betydligt assembler programskrivning, efter som tabulatorn redan är ställd för att motsvara kraven Assembler programmet ställer. Därtill blir det versaler där det skall vara versaler oberoende av hur man skriver. Naturligvis är editorn ställd i word wrap mode.

SaveOptions ger möjligheten att lagra tabulator inställningarna så som man vill ha den vid ett senare anrop av MY-Word.

ViewFiles ger möjligheten att titta på DIS/VAR 80-filer utan att den text man arbetar med blir bortspolad.

Tyvärr saknas radnummer i det VF åstadkommer! Synd!! Det tillkommer en del andra nyheter men jag hoppar över dem här.

Positivt:

Hanteringsmässigt mycket TI-Writer liknande.

Tillgång till RAM-Disk.

För övrigt, se allt ovan.

Negativt:

Inga valmöjligheter vad gäller skärmkaraktärer. Dvs inga

Svenska, Tyska, Franska, Italienska eller Spanska tecken möjligheter.

Inga riktiga underslängar på gemenerna.

Mousez

Är ett mus-demoprogram som jag fick när jag hade Geneve-prototypen på "besök" i februari 1987.

Det laddas via E/A Load and Run. Enda sättet att stoppa programmet är att stänga av datorn.

Programbeskrivning:

Programmet är helt musorienterat dvs låter sig endast påverkas av musen.

Med musen kan man förstås styra markören samt via 2 av de 3 tangenterna ge olika kommandon.

Kommandon:

Pointer = en markör i formen av en pil.

Circles = slå ofyllda cirklar.

Rectangles = rita fyllda rektanglar.

Pencil = rita på frihand.

Rubber line = dra rät linje mellan två punkter.

Fill = fyll figur med färg.

File = spara på diskett. Fungerar ej datorn läser sig.

Skärmen är ställd i 256 x 212 pixels och har längst ned en palett med 256 olika färgnyanser, varifrån man med markören hämtar den färg man vill rita med.

Till skillnad mot i TI-99/4A mode, kan man här sätta var enskild pixel till vilken färg som helst, dvs de angränsande pixlarnas färg har ingen inverkan på den enskilda pixeln.

Någon nytta har jag inte av programmet, men det känns som att få sina 16 trubbiga vaxkritor utbytta mot 256 oljefärgstuber med åtföljande spetspensel och kalkborste. Och det är väldigt roligt att kladda på skärmen tycker jag och de andra barnen i familjen.

Myarc Disk Manager III ver 2.0

Kan bara köras på en Geneve utrustad med ett Myarc Diskkontroll Kort.

Är enligt uppgift den enda managern som via GPL-tolken klarar av att köras i hastigheten.

Jag har i skrivande stund inte hunnit testa hastigheten med klocka, men nog verkar den snabbare än tidigare. För övrigt verkar den sig helt lik tidigare versioner.

Dvs den uträttar något mer än DM 1000 v 3.5.

Problem

Klockan har en benägenhet, trots sin batteri uppbackning, att visa helt galna värden efter att en ny uppstart gjorts. Det verkar som om strömtillslaget till boxen orsakar Chicago i klocksystemet.

Monitor-interlacet hoppar ibland ur, bl a vid anrop av yttre enheter. Går förvisso lätt att koppla på igen, men det är mycket irriterande.

Lysdiödsindikeringen för Caps Lock är långtifrån tillförlitlig!

Den lyser eller är släckt när den skall vara tänd och ska den vara släckt, så kan den mycket väl vara tänd, men ibland är den släckt. (Ursäkta min MARXistiska skrivstil)

Inga problem

De problem som vi hade i februari i år, med att få Geneven att fungera tillsammans med ett kontrollkort från Cor-Comp, finns inte längre!

Önskan

Att utskrivningssättet av datumet skulle gå att ändra. Datorn skriver det på det anglosaxiska viset dvs månad-dag-år. Det blir lätt förbryllande när man som vi, är vana vid år-mån-dag eller dag-mån-år.

Att i My-Word ha tillgång till bl a svenska tecken.

Väntan

Gott om tålamod har visat sig vara bra att ha när det gäller Myarc. Jag tycker mig dock ana en minskad latens i kommunikationen med dem.

Vad jag närmast väntar på är version 1.4 av MDOSen.

Min förhoppning om den är att den är bättre avluskad och att den kan allt vad manualen påstår. Dvs:

Att tangentbord fungerar helt som det är tänkt.
Att klockan blir säker.
Att monitorinterfacet inte hoppar ur.
Att man kan ställa RAM-Disk och Spooler efter eget gottfinnande.
Att DOSen är mer homogen, dvs att man kan få laddat det man vill ha laddat utan en massa mellansteg.
Att man kan skapa snygga Batchfiler.
Därtill väntar jag förstås på Advanced Basic och Runtime Pascal.
Enligt ett personligt brev från Myarc, skall jag bara behöva vänta detta året (1987) ut. Fan trot!

Advanced BASIC

För att Du ska få någon hum om vad Advanced Basic erbjuder utöver TI-Extended Basic, har jag gjort en listning som troligtvis blir placerad någonstans utanför den här rapporten.

Det uppges att Advanced Basic skall vara 6 gånger snabbare än TI Extended-Basic.

Jag har ju, under en mycket kort och hektisk tid i februari 1987, haft möjlighet att köra Myarc Extended Basic II från vilket Advanced Basic blivit vidareutvecklat.

Det som imponerade mest på mig då var grafiken, den var fantastisk, sedd genom "TI-99/4A glasögon".

Pascal version 4.21

Med den runtime Pascal, som ingår i köpet, skall man enligt uppgift kunna köra Pascalprogram utvecklade på andra maskiner, som IBM PC, Apple etc, utan att behöva modifiera dem.

Ett krav är förstås att det egna programbiblioteket innehåller de subprogram som huvudprogrammet anropar.

Tänk Dig att Du har ett modem, och att det är som det sägs, flera tillgängliga databaser här i landet med fri tillgång till bl a Pascalprogram skapade på andra maskiner.

Gosse, snacka om vidgade vyer, det är annat än att bara se Geneve från ovan.

Programmeringsspråket Pascal, uppges snart vara tillgängligt.

Hela P-systemet utvecklas av *PECAN Systems*, USA. Till detta system skall även andra kompillerande språk än Pascal vara tillgängliga såsom Modula-2, BASIC och Fortran.

Ytterligare väntan

Enligt MICROpendium och Myarcs egen tidning CALL MY-ARC, är det en hel del på gång vad gäller utveckling av programvaror. Det är särskilt ett program som lockar mig, nämligen My-Word-Pro, ett ordbehandlingsprogram med bl a stavningskontroll samt mycket mera. Detta program skulle passa mig som hand i handske, speciellt när jag till äventyrs känner mig manad att skriva på engelska.

Ingen väntan

Rätt som det var kom det ett helt oannonserat brev från Jan Alexandersson. I detta brev fanns bland annat en diskett innehållande DM 1000 version 3.8.

Den är bl a gjord för att köras på Geneven och den fungerade helt perfekt.

Vad jag tycker är glädjande är att programmakare som inte är involverade i utvecklingen av mjukvara till Myarc, börjat skriva program för Geneven.

Hårdvaru utbyggnadsmöjligheter tillgängliga idag

RAM-minnet kan utökas till upp över 2 mb RAM, via 3 i boxen inpluggade minneskort à 512 kb vardera.

Hårddisk-kort med möjlighet att ansluta 3 st ST-506 kompatibla hårddiskar med upp till 134 mb lagringsmöjlighet vardera, samt upp till fyra 5 1/4 eller 3 tums diskettmaskiner från SSSD till DSQD.

Eget projekt på gång

Med benäget bistånd av klurighet och kunskap från Lennart Thelander, hoppas jag få pratorn inpluggad i boxen i ett eget kort. Materialet består av pratorn själv och ett från Digital-service inköpt elektroniskt defekt minneskort.

Kanske vi lyckas få Geneven säga "jiniiv".

Mitt totala omdöme

Formulerar jag nog bäst så här:

Efter att jag sparat mina moduler på skiva och kopplat upp GENEVEN, har jag ännu inte haft för avsikt att koppla upp TI-99/4A igen.

Detta kan förstås bero på att jag är ganska enahanda i mitt val av programspråk.

Vad kostar en Geneve?

Enligt sista brevet från Myarc är det föreslagna priset i butik 530 dollar.

Enligt samma brev påstår man att man inte säljer till kunderna direkt, utan man måste köpa genom återförsäljare.

Som närmsta återförsäljare utpekas:

Albert Visser
Legio Computer Center
Haagweg 169
2281 AJ RI JSW ILK
Nederländerna

Telefon 009-31-70995757

Herr Visser uppges ha generalagenturen för Europa.

Enligt en annons som han hade i Computer Kontakt Dec 87—Jan 88, kostar den 1245 Tyska Mark.

Grovt räknat, innebär detta att Herr Visser vill av tyskar ha 1300 SEK mer än Myarc föreslår som pris i USA.

I båda fallen ingår samma tangentbord som jag nämnt tidigare, ett s k enhanced keyboard.

Av MYARC föreslagna minutpriser (okt 87) i USA, på hårdvarutillbehör:

Hård-disk-kort	325 \$
512 kb minneskort	255 \$
Floppydisk-kort DSDD	170 \$
Dito DSQD	220 \$
Mus	50 \$

Om det skulle finnas några möjligheter att i fortsättningen direkthandla med Myarc, som jag gjort, vet jag i skrivande stund inte. Men det faktum att Sverige inte är med i EG är väl ett litet skäl och skulle det vara så att flera tänker köpa en Geneve på en gång så är väl det ett större skäl att få handla direkt.

Skulle det vara så, att minst fem datorer köpes samtidigt, tror jag det kan finnas möjligheter till direkt-köp. Detta grundar jag på tidigare Myarc uttalanden. Jag hjälper gärna till om någon vill ha min hjälp. Men att samordna pengar till köpet avser jag mig, redan innan jag blivit tillfrågad.

Det var allt!

Sören Bernle, tel 042/933 05

Ny katalog från Tigercub

Det har kommit en ny katalog på 9 sidor från:

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213
USA

Den innehåller ett mycket stort antal program i Basic och Extended Basic för dig som har grunddator och kassettbandspelare. Programmen kostar endast USD 2 per styck plus USD 3 per order till Sverige.

Det finns även tre flexskivor med SUB-program i MERGE-format. Varje sådan skiva innehåller minst 100 små programsnuttar och kostar USD 15 per styck.

Du som vill ha en kopia av katalogen kan sända frimärken 5 (rabattmärke eller brevporto) till:

Jan Alexandersson
Springarvägen 5, 3 tr
142 00 TRÅNGSUND

TI-74 BASIC

av Anders Persson

Det lär finnas många sätt att få en katt på, sägs det. Om det är sant vet jag inte — jag har aldrig provat. Men att det finns många sätt att förpacka BASIC på råder det inget tvivel om. Ett av dessa, som jag tittat lite närmare på, heter TI-74.

Tidigare nummer av tidningen har innehållit lite om TI-74 BASICALC, som den egentligen heter, men jag tänkte ändå helt kort rekapitulera maskinens grunder.

Som namnet antyder är det egentligen en hybrid mellan två olika ting. Å ena sidan är maskinen en fickdator med åtta kilobytes RAM och BASIC tolk i ROM. Å andra sidan är den en konventionell, icke-programmerbar, teknisk räknare med någorlunda normala prestanda. Genom att trycka på en tangent växlar man mellan de olika lägena. Som räknare är maskinen inte så mycket att orda om. Däremot tänkte jag berätta lite mer om dess egenskaper som dator.

Hårdvara

Uppbyggnaden av maskinen är mycket enkel. Tittar man inuti hittar man ett kretskort med CPU (en TMS 7000 i CMOS utförande), RAM, ROM och inget mer. På utsidan syns en anslutning för allehanda tillbehör samt, naturligtvis, en modulport. Texas Instruments har ju praktiskt taget uppfunnit programmodulen, så en sådan port måste ju finnas. Dessutom finns en display (en rad, 31 tecken) och ett tangentbord.

Mjukvara

En BASIC-tolk finns alltså inbyggd i ROM. Trots att TI gör så gott de kan för att glömma bort tidigare misstag med olika maskiner, tycks det vara svårt att även glömma bort deras programspråk. 74:ans BASIC är mycket lik Extended BASIC till 99:an och närmast identisk med den i CC-40. Och varför inte? Båda BASIC tolkarna är ju ganska innehållsrika och trevliga varianter. Att funktioner för att hantera sprites, färger, ljud, syntetiskt tal och joysticks saknas är naturligt, då 74:an inte har dessa möjligheter. Anrop av assemblerprogram låter sig tyvärr inte heller göras. Men i övrigt finns det mesta med, och även några små finesser som inte finns i X-BASIC.

Skärmen

Liksom i X-BASIC finns både PRINT och DISPLAY, där den senare kan förses med extra påhäng för styrning av utskriften. AT finns, men bara med ett argument. Det finns ju bara en möjlig rad att skriva på. Tyvärr saknas CALL CHAR, varför maskinen inte kan stava svenskan korrekt. Det finns visserligen en laddning tecken med koder större än 127, däribland ett antal japanska krume-lurer, men inte bokstaven "å". Med tanke på tyskarna finns faktiskt "ä" och "ö", så bära kan man göra, men inte på en bär...

För inmatning kan man skriva ACCEPT NULL, där man kan ange vilket värde som ska användas om användaren bara trycker på ENTER. Mycket praktiskt. Dessutom finns förutom den välkända CALL KEY även funktionen KEY\$, som stannar programmet tills en tangent trycks ner och sedan returnerar tecknet i fråga. Det finns också en PAUSE instruktion, som tillåter operatören att läsa vad som står på den enda skärmdelen innan något annat hinner skrivas där.

Matematik

När det gäller matematiska funktioner är det ovanligt väl sörjt. I räknarläge kan man välja tre olika vinkelmått för trigonometriska funktioner och även använda hyperboliska funktioner. Dessa möjligheter finns med i BASIC också. Kommandona DEG, RAD och GRAD väljer vinkelmått. Räkna kan man sedan göra med SIN, COS, TAN, SINH, COSH, TANH och samtliga inverser. Logaritmer finns med både tio och e som bas.

Sekundärenheter

Om man vill lagra sina alster på band måste man skaffa sig ett interface som tillhör. Det fungerar ungefär som det som är inbyggt i 99:an, men med vissa förbättringar. Datafiler förses med en speciell filslutsmarkering, varför EOF kan användas även med bandspelare. Det går också att bli av med ledtexterna som ges, om man inte vill ha dem, eller kanske vill göra egna. VERIFY är ett separat kommando, som tillåter kontroll av att lagringen gick som den skulle.

Fullständig filhantering finns inbyggd. TI kan själva leverera en liten termoskrivare, som också är batteri-driven och portabel. Mechatronic lär ha ett diskkontrollkort för den som vill lagra program och data lite snabbare. Men då tycker jag att maskinen är för liten för att motivera utrustning av det slaget.

Intressantare är väl då det utlovade RS232 interfacet. Med ett sådant kan man ju haka på en vanlig skrivare, och även koppla ihop en TI-74 med exempelvis en 99/4A.

Minne

Inbyggt finns alltså 8 Kbyte. Funktionen FRE talar om hur mycket som man kan använda, och hur stor plats programmet tar. När systemet tagit sitt finns det 7710 bytes kvar att programmera på. Räcker inte det kan man dubblera minnet med en extra minnesmodul.

Den går att utnyttja på två olika sätt. Om man har två olika program som man vill köra, där programmen var för sig ryms i maskinens interna minne, kan man låta det ena finnas i modulen när det andra används. Med kommandona CALL GET och CALL PUT kan man flytta program till och från modulen. Det går också att låta programmen i maskinen och modulen byta plats med varandra. Eftersom RAM-modulerna är försedda med ett litet batteri kan man skapa sig ett helt bibliotek med egna program i modulform. Kan dock endast rekommenderas för penningstarka individer...

Om nu ett program inte ryms inom 7710 bytes då? Ja, då får man skriva CALL ADDMEM. Maskinen betraktar sedan modulens och maskinens minnen som ett sammanhängande. 15902 bytes att programmera i blir resultatet. Intelligent nog raderar CALL ADDMEM inte det program som finns i maskinen. Det gör alltså inget om man inte i förväg tänkt på att programmet man skriver blir stort.

Det finns också några ROM-moduler som man kan stoppa in i stället för extra RAM minne. Bland annat finns programspråket Pascal, men för att få in det i den här lilla apparaten har TI fått modifiera det tills döden inträffat. Det påstås vara till för att man ska kunna skaffa sig en uppfattning om vad Pascal är för ett språk. Den som efter att ha tittat på just den här versionen drar slutsatsen att Pascal är ett trevligt språk lär vara mogen för stor sinnesundersökning. Inget att lägga pengar på!

Underprogram

Liksom i X-BASIC kan man använda namngivna underprogram med lokala variabler. En väsentlig skillnad är att lokala variabler är dynamiska, inte statiska. Vid anrop av ett underprogram allokeras och nollställs alla lokala variabler. När sedan SUBEND eller SUBEXIT exekveras, återlämnas minnesutrymmet igen. Så fungerar normalt exempelvis program skrivna i Pascal. 74:an tillåter dock inte rekursiva anrop av subprogram.

Instruktionen DEF är ganska begränsad i X-BASIC (se PB 84-3, sid 4—9). I 74:ans BASIC är den så begränsad att den inte finns alls.

Diverse

Felhanteringen är ännu mer omfattande än vad Extended BASIC kan erbjuda. Det går att hantera de olika fallen ERROR, WARNING och BREAK ungefär likadant, men det går också att betrakta både WARNING och BREAK som om det vore ERROR. Med ON ERROR kan man ju anropa en godtycklig rutin i händelse av fel. Det går varken med ON WARNING eller med ON BRE-

AK. Men genom att låtsas att ett ERROR inträffat går det alltså i alla fall. Ganska klurigt, men varför inte lika gärna tillåta konstruktionen ON WARNING 1580 (eller vilken rad man nu vill hoppa till)?

En praktisk detalj är att man kan skriva IF NUMERIC(IN\$) THEN AMOUNT=VAL(IN\$). Bättre att veta att det går innan man försöker med VAL än att behöva prova och se om det blev fel.

Sammanfattning

Vad ska man då ha en sådan här manick till? Du som kanske är på språng mycket i jobbet, behöver en teknisk räknare och dessutom vill kunna programmera den med tämligen avancerade program utan alltför stort besvär, vad väntar du på? Det finns förvisso fler maskiner av den här typen, men om du råkar ha en 99/4A också — inte helt osannolikt om du läser den här tidningen — har du dessutom fördelen att kunna det mesta om maskinens BASIC redan från början. Extra intressant blir det med ett RS232-interface, så att man exempelvis kan samla in data ute på fältet och sedan "tanka över" alltihop till 99:an, eller någon annan maskin, för vidare bearbetning.

Enhanced Display Package (EDP)

recenserat av Per Virving

Enhanced Display Package är ett freeware som gör användandet av 99:ans grafik betydligt lättare. Du kan få textmode, ny skärmfärg i kommando-mode, fönster, inverterad video, blinkande text med mera. Härligheten bygger på CALL LINK-satser etc. CALL LINK("MODE",2) ger textmode.

Programmet är inte som andra freeware, där källkoden följer med och man får ändra den om man vill (och kan). EDP laddas in i datorn med LOAD-programmet på disketten. En anmärkningsvärd sak är att LOAD innehåller rad nummer 0 (noll)!!!

Ett fönster öppnas med kommandot WINDOW och kan göras precis i vilken storlek man vill, hela skärmen eller bara ett tecken. Allting sker sedan i relation till fönstret, under förutsättning att man använder EDPs kommandon för in- och utmatning av text till skärmen. Texten i fönstret scrollas uppåt när man kommer längst ned i fönstret. När en rad scrollas ut ur fönstret kan man välja om den skall försvinna eller om man skall ha den kvar i en speciell buffert.

GTEXT är ett kommando som tillåter att man hämtar en rad bokstäver från skärmen utan att man trycker ENTER.

ACCEPT är mer utvecklat och man kan skriva 255 tecken och inte som den vanliga ACCEPT där man bara kan skriva tills cursorn kommer till andra sidan skärmen.

FLASH gör att vissa teckengrupper blinkar i de färger man anger. I EDP finns en klocka. Den går dock inte rätt!! Detta p g a att man i USA som bekant har 60 Hz i vägguttaget istället för 50 Hz som i Sverige. Till klockan finns ett alarm som tutar och blankar skärmen, det slutar tuta och man får tillbaka det man skrivit när man trycker på en tangent.

Förutom grafik-kommandona finns det en del kommandon för att ändra om i minnet, LOADR, LOADV, PEEKV, INTON/OFF. LOADR laddar ett register med ett värde, med LOADV kan man manipulera VDP-minnet. EDP använder interrupt-rutiner för att kontrollera blinkande text, klocka osv. Om man t ex har ett annat assembler-program som använder interrupt-rutiner kan man stänga av dem INTOFF. INTON sätter på EDPs interrupt-rutiner igen. Med READ och WRITE kan man spara en hel skärm på diskett. Alla teckendefinitioner, färger, inverterad text kommer med. Manualen är mycket utförlig (32 sidor) och innehåller en allmän

information om EDP och vissa saker man måste veta för att skriva egna program. Manualen innehåller dessutom utförliga beskrivningar om varje kommando.

Det finns naturligtvis mer kommandon än de jag beskrivit här.

EDP borde vara ett mycket kraftfullt hjälpmedel för dem som skriver egna program och om man ger sig tid att utnyttja dess möjligheter.

P.S. Någon som kan assembler bra skulle kanske kunna ta sig an uppgiften att ändra EDP:s klocka så att den går rätt på 50 Hz, eftersom många önskar sig en klocka som man visas på skärmen när som helst (en Utmaning!!!).

Variabel-överföring i Extended BASIC

av Fredrik Nilsson

Här är ett par små program som jag hoppas är intressanta. I Programbiten 86-4 så beskrevs ju några olika sätt att överföra variabler från ett Extended basic program till ett annat.

Om man inte har expansionsminne är det ju ganska trassligt (skriva på skärmen osv).

Men eftersom det finns delar i minnet som inte nollställs när man laddar in ett nytt program, nämligen teckendefinitionerna. Så finns det ett enkelt sätt att göra det, nämligen genom att göra om variabeln som man vill överföra till en sträng. Sedan stoppar man in strängen i någon charsats, lämpligen någon med ascii över 127 (128—143 nollställs inte ens av NEW). Det går bra med ascii under det också, men man riskerar att de återställs av t ex CALL CHARSET.

När nästa program är inladdat använder man sig helt enkelt av ordern CALL CHARPAT och får ut strängen, som sedan är enkel att göra om till en variabel igen.

De första två programmen demonstrerar detta.

Anledningen till att det första programmet frågar efter en sträng och inte en variabel, beror på att datorn som bekant skriver stora tal med tio-potens notation, och i char satserna skriver du bara med hex. Därför gillar inte datorn om du försöker stoppa in ett + i strängen. T ex om du skriver PRINT 10000000000 så får du 1.E+10 som svar, det undviks nu. I vanliga program där variablerna kanske överstiger den gräns där datorn börjar med notation blir det problem.

En idé kunde vara att man rensar bort punkten och pluset innan och sedan sätter tillbaka dom. Det är inte så svårt. Men om du tänker efter så blir variabler oftast inte så stora (om du nu inte skriver matematiska program).

Att det bara går att använda 0—9 och A—F medför stora problem om du vill överföra riktiga strängar. Ett sätt där är att låta två tecken i charsatsen motsvara vad dom egentligen är, nämligen ett hexadecimalt tal 0—FF, decimalt 0—255. Alltså ascii koden för ett tecken.

På det sättet kan man teoretiskt sett få in ca 3,5 strängar med 255 tecken i varje, om man använder alla charsatserna.

Det andra programmet är bara en liten demonstration vad man göra med sprites och lite cosinus och sinus.

Det går inte heller att skriva in decimaltal som du förstår. Men det kan ju avhjälpas genom att byta ut punkten mot t ex "P".

Tips och tricks:

Om du har alpha lock nedtryckt och håller t ex G nere så repeterar datorn G:et och om du sen trycker ned ctrl 2 eller 3 eller 4 så blir det plötsligt små bokstäver som skrivs ut. Trots alpha lock.

Just det ja, om du flyttar rad 935 till 925 i spelet HELIKOPTER som publicerades i 86-2, så slipper du börja om från det gröna varje gång du dör. Istället får du starta från det område du dog på, t ex du dör på vitt och då startar du från första vita skärmen.

```

100 !*****
110 !* DEMO 1.1 *
120 !* FLYTTNINGSDemo AV *
130 !* VARIABLER MELLAN *
140 !* 2 EXTENDED BASIC *
150 !* PROGRAM. *
160 !* *
170 !* AV FREDRIK NILSSON*
180 !* 1988 *
190 !*****
200 !
210 CALL CLEAR
220 !
230 DISPLAY AT(8,1)BEEP:" TAL 1:"
240 !
250 ACCEPT AT(8,10)SIZE(16)VALIDATE("1234567890"):TAL1$
260 !
270 DISPLAY AT(11,1)BEEP:" TAL 2:"
280 !
290 ACCEPT AT(11,10)SIZE(16)VALIDATE("1234567890"):TAL2$
300 !
310 !
320 TAL1$=RPT$("0",16-LEN(TAL1$))&TAL1$
330 TAL2$=RPT$("0",16-LEN(TAL2$))&TAL2$
340 !
350 CALL CHAR(128,TAL1$)
360 CALL CHAR(129,TAL2$)
370 !
380 !
390 CALL CLEAR
400 !
410 RUN "CS1"

```

```

100 !*****
110 !* DEMO 1.2 *
120 !*****
130 !
140 CALL CLEAR
150 !
160 CALL CHARPAT(128,TAL1$)
170 CALL CHARPAT(129,TAL2$)
180 !
190 !
200 TAL1=VAL(TAL1$)
210 TAL2=VAL(TAL2$)
220 !
230 !
240 DISPLAY AT(8,5)BEEP:"TAL 1=";TAL1
250 DISPLAY AT(10,5)BEEP:"TAL 2=";TAL2
260 !
270 !
280 END

```

```

100 !*****
110 !* SPRITE DEMO *
120 !* AV F.NILSSON *
130 !* 1988 *
140 !*****
150 CALL CLEAR :: CALL SCREEN(2):: DIM COSS(28),SINN(28)
160 FOR T=1 TO 14 :: CALL COLOR(T,16,2):: NEXT T
170 DISPLAY AT(12,11):"* DEMO *"
180 CALL CHAR(40,"3C7EFFFFFFF7E3C")
190 FOR SPRITE=1 TO 28
200 RADIANTER=360/28*SPRITE*PI/180
210 CALL SPRITE(#SPRITE,40,2,92+SIN(RADIANTER)*75,124+COS(RADIANTER)*75)
220 NEXT SPRITE
230 FARG=2
240 FOR GANG=1 TO 20
250 FOR SPRITE=1 TO 28
260 CALL COLOR(#SPRITE,16):: CALL COLOR(#SPRITE,FARG)
270 NEXT SPRITE
280 FARG=FARG+1 :: IF FARG=17 THEN FARG=1
290 NEXT GANG
300 CALL CLEAR :: CALL DELSPRITE(ALL):: CALL CHAR(40,"COC000000000000000000")
310 FOR STEG=1 TO 28
320 COSS(STEG)=COS(STEG*2/28*PI)
330 SINN(STEG)=SIN(STEG*2/28*PI)
340 NEXT STEG
350 SPRITE,STEG=1
360 CALL SPRITE(#SPRITE,40,5,92,124,COSS(STEG)*10,SINN(STEG)*10)
370 SPRITE=SPRITE+1 :: IF SPRITE=29 THEN SPRITE=1
380 STEG=STEG+1 :: IF STEG=29 THEN STEG=1
390 GOTO 360

```



Vad Myarc Advanced BASIC har som TI-99/4A Extended BASIC inte har

av Sören Bernle

MYARC ADVANCED BASIC är utökat med följande: Commands = c
Statements = s
Functions = f

MYARC ADVANCED BASIC förändringar,
som jämförda med
TI-99/4A EXTENDED BASIC är: Utökade = u
Olika = o
Borttagna = b

Vad?	Ti?	Namn:	Kommentar:
cs	u	ACCEPT AT	inputen kan avslutas med annan tangent än enter. VALIDATE tillåter två nya datatyper: ALPHA = All alphabetic characters. LALPHA = All lower-case alphabetic characters. även utökad med BLINK, CLIP, INVERT se DISPLAY A.
cs		BCOLOR	anger färgen på BLINK funktionen.
cs	u	BEEP	ger ett beep var som helst i programmet.
cs		BTIME	anger hastigheten på BLINK funktionen.
f		CDBL	ändrar ett tal till dubbel precision.
cs	u	CHAR	tillåter definition av alla tecken, ASCII 0-255.
cs	u	CHARSET	kan återställa bara vissa karaktärer.
c		CHDIR	ändra förvald diskdrive.
f		CINT	ändrar ett reelltal till heltal.
cs		CIRCLE	ritar en cirkel eller ellips.
cs	u	CLOSE	om inget filnummer anges stängs alla öppna filer.
cs		CLS	rensar ett aktivt fönster eller hela skärmen.
cs	u	COLOR	uträttar olika saker i olika grafikmoden.
c		COM	ändrar förvald kommunikationports namn.
f		CREAL	ändrar ett heltal till reelltal.
f		CSNG	ändrar ett tal till enkel precision.
cs		DATE	ange nytt datum.
f		DATE\$	skriver aktuellt datum.
cs		DCOLOR	anger färg på ritfunktionerna.
s	u	DEF	klarar upp till sju parametrar.
		DEFvartype:	
s		DEFINT	% = heltal.
s		DEFSNG	! = enkel precision, binärkodat.
s		DEFDBL	# = dubbel precision, binärkodat.
s		DEFSTR	\$ = sträng.
s		DEFREAL	é = dubbel precision, RADIX 99 kodat.
cs	o	DELETE	raderar programrader.
cs	u	DIM	accepterar namngivning med alla variabeltyperna.
cs	u	DISPLAY AT	har utökats med: BLINK som får texten att blinka. INVERT som får textens för- och bakgrundsfärger att bli omkastade. CLIP som hugger av strängen i DISPLAY AT-satsen istället för att låta den fortsätta på nästa rad.

S		DRAW	ritar och raderar linjer mellan två punkter.
S		DRAWTO	ritar och raderar linjer från "nuvarande" punkt.
CS	O	FILES	visar vilka filer en diskett innehåller.
S		FILL	fyller en sluten figur med färg.
f	O	FREESPACE	utför detsamma som TI-XB kommandot SIZE.
CS	U	GCHAR	ger i högupplösande mode, pixelns färgnummer.
CS		GRAPHICS	välja bland 9 olika grafikmoden.
CS	U	INIT	även avgöra hur stort assembler utrymme som skall reserveras.
f		HEX\$	omvandlar ett decimalt tal till ett hexadecimalt.
CS	U	CALL KEY	key-unit har utökats från 5 till 6 st.
CS		KEY	ändra funktionstangenternas kommandon.
CS		ON KEY	låter Dig via funktionstangenterna påverka programmet under körningen.
			Jmfr Ti-99/4A: <Fctn+4> BREAK!
CS		KILL	raderar diskett-filer, samma som DELETE i XB.
f		LEFT\$	skapar en ny sträng av källsträngens vänstra del.
		LLIST	listar ut på printer.
CS	U	LOAD	flera start adresser kan anges och i hexa-värde.
S		LPR	ger koordinaterna för sista berörda punkten efter ett grafikkommando.
CS		LPT	ändra förvalt printer anrop.
S		LTRACE	skriver exekverade radnummer via printer och låter skärmen vara intakt.
CS		MARGIN	skapar fönster på skärmen.
S		MEMSET	sätter alla elementen i en matris till samma önskade värde.
f		MOD	ger ett avrundat heltalsvärde av en divisions rest.
C		MOM	ändra förvald modem port.
		MOUSE	kommandon:
CS		MOUSE ON	
CS		MOUSE OFF	
CS		MOUSE STOP	
CS		ON MOUSE GOSUB	
CS		CALLS MKEY	
CS		CALL MKEYLAST	
CS		CALL MLOC	
S		CALL MREL	
S		CALL MOUSE	
CS		CALL MOUSEDRAW (ON)	
CS		CALL MOUSEDRAW (OFF)	
CS		CALL HIDEMOUSE	
S		CALL SEEMOUSE	
C	O	NEW	assembl-program laddade via CALL LOAD raderas ej.
C	U	NUMBER	ett felmeddelande visas om en tidigare skriven rad, med ett fel i, kommer upp på skärmen.
C	U	OLD	autostartande assemblerprogram kan köras. kan ladda DIS VAR 80 filer.
CS		OUT	sänder ett databyte till en "port".
C		PCM	skriver förvald kommunikationports namn.
CS	U	PEEK	flera startadresser kan anges och i hexavärde.
CS		PEEKV	läser adressinnehållet i VDPram.
C		PMD	skriver förvald modemports namn.
CS		POINT	ritar eller raderar en eller flera punkter på skärmen.

CS		POKEV	skriver in värden på VDPräm adresser.
C		PPR	skriver printerns anropsnamn.
C		PPT	enligt manualen samma som ovan?
C		PWD	skriver förvald diskdrives namn.
S		RECTANGLE	ritar rektanglar.
f		RIGHT\$	skapar en ny sträng av källsträngens högra del.
CS	u	SCREEN	sätter även förgrundsfärg i textmode.
C	b	SIZE	utbytt mot FREESPACE.
C		SPEED	avgör programmets exekveringshastighet.
CS	u	SPRITE	tillåter 32 st, vissa moden även 8 på samma rad.
S		SWAP	byter värde mellan två variabler.
f		TERMCHAR	ger ASCII-kod värdet av den tangent som fick programmet att fortsätta efter en input sats.
CS		TIME	ställer klockan.
f		TIME\$	läser klockan.
CS	b	TRACE	utbytt mot TRACE ON.
CS	u	TRACE ON	skriver på skärmen de exekverade programradsnummren.
CS		TRACE OFF	avbryter ovanstående kommando.
CS	b	UNTRACE	utbytt mot TRACE OFF.
f		VALHEX	omvandlar ett hexadecimalt tal till ett decimalt.
S		WEND	avslutar en WHILE loop.
S		WHILE	loopar så länge WHILE-förutsättningen är sann.

Därtill är ADVANCED BASIC ASCII-KODEN utökad till 256 tecken fördelade på 32 teckenuppsättningar (sets) enligt nedan.

Set	ASCII-KOD	Set	ASCII-KOD	Set	ASCII-KOD	Set	ASCII-KOD
29	0-7	5	64-71	13	128-135	21	192-199
30	8-15	6	72-79	14	136-143	22	200-207
31	16-23	7	80-87	15	144-151	23	208-215
0	24-31	8	88-95	16	152-159	24	216-223
1	32-39	9	96-103	17	160-167	25	224-231
2	40-47	10	104-111	18	168-175	26	232-239
3	48-55	11	112-119	19	176-183	27	240-247
4	56-63	12	120-127	20	184-191	28	248-255