

ALFA

SLANG

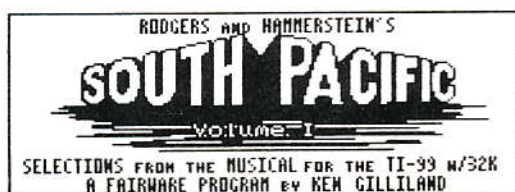
nittinian

BITEN

97-98

MATRIKEL
NUMMER

ALFA
SLANG



Innehåll

Redaktören . . .	2
Super Extended BASIC	3
DM 1000 — varianter	3
Karaktärsdanande älvödare	4-5
Nyheter från Tyskland	6
Tips för Pascal	6
Nya freeware	7
Sörens menyprogram	8
Matrikel	9-16
Alfaslang	12-13
MAX-RLE	17
Turbo Pascal till 99:an	18-20
Ordföranden . . .	21
Resetknapp till 99:an	21-22
TI-95 nytt	22
RAM-disk	23-24

ISSN 0281-1146

Redaktören . . .

I detta nummer

Innehållet är ganska blandat tycker jag, det finns program både i Extended BASIC, Pascal och assembler. Lite hårdvarutips också samt en del om tillbehör. Som Du nog märker har jag använt bilder som kommer från programmet MAX-RLE flitigt som utfyllnad i tidningen.

Medlemsmatrikeln

I detta nummer publicerar vi medlemsmatrikeln. De som skrivit sitt telefonnummer på inbetalningskortet får det också tryckt i matrikeln, precis som en del medlemmar hade begärt. Baktanken är förstås att man skall kunna kontakta andra medlemmar för att kunna utbyta erfarenheter. Använd denna möjlighet! Ett par felaktiga uppgifter kom med av misstag när matrikeln trycktes. Vi har målat över dem på de färdiga tidningarna.

Nyheter från Corcomp

En hel del nyheter presenteras fortfarande till 99:an. Senast är det Corcomp som kommit med ett par nya program. Det har bla tagit fram en modul som gör det möjligt att läsa och skriva disketter i IBM PC-format på 99:an. Haken är förstås att måste ha ett av Corcomps egna diskkontrollkort för att det ska vara möjligt. För en del kan det nog vara intressant att kunna skriva texter hemma och ta med disketten till en PC på jobbet (eller varför inte tvärtom).

95:an och 74:an

börjar vi skriva mer och mer om i tidningen samtidigt som det kommer mindre material om gamla 59:an. I nästa nummer återkommer vi med ett fylligare material om 95:an.

Internationella möten

Från en användargrupp i Bremen har vi fått en inbjudan till den "2:a internationella TI-träffen". Den första hölls i Köln förra hösten, och var enligt arrangörerna en stor framgång. Om någon har vägarna förbi Bremen den 10 oktober och är intresserad så går det bra att få mera information från föreningen.

Vår danska motsvarighet som ger ut "Computerbladet TI-eren", hade ett möte en helg i mitten av maj. Totalt besökte 175 personer mötet. Så många brukar vi inte vara på våra möten. Ett par möten i Stockholm är i alla fall inplanerade till hösten. Till det första (den 26 september) har Du redan fått inbjudan när denna tidning kommer ut. Till nästa möte återkommer vi.

Egen databas

En databas i föreningens regi har diskuterats länge, dock utan att något konkret hänt. Styrelsen har dock nyligen fattat beslut om att inrätta en sådan på prov. Vi kommer alltså att ha ett telefonnummer öppet för de som har modem till sina system. I databasen kommer att finnas en "brevlåda", där olika ämnen kan dryftas, program ska kunna hämtas därifrån mm. Vi håller nu på och undersöker olika alternativa lösningar och hoppas kunna återkomma under hösten med mera information.

Två nummer till

av Programbiten hoppas vi kunna ge ut under 1987. Det hänger dock på Dig om vi ska lyckas med det. Vi måste nämligen få in mera material till tidningen. Som Du kan läsa på annan plats så drösar det in ny freeware-program till föreningen. Själva har vi i styrelsen ingen chans att hinna gå igenom dem alla. Om Du åtar dig att skriva en "recension" till Programbiten så skickar vi programmet till Dig (på kassett eller diskett). Hör av Dig på föreningens adress. Som vanligt gäller: först till kvarn . . .

I redaktionen:

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

Föreningens och redaktionens adress:

Föreningen Programbiten
c/o Schibler
Wahlbergsgatan 9, nb
121 46 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 1 000 SEK per helsida, 500 SEK per halvsida. För lösblad som skickas med tidningen gäller 1 000 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:-



Super Extended BASIC

av Jan Alexandersson

Det finns nu på marknaden ett antal förbättrade versioner av Extended Basic:

- Mechatronic Extended Basic II Plus
- Myarc Extended Basic II
- MG GK Utility I
- Super Extended Basic

Mechatronic XB II PLUS kan du läsa mera om i Anders Perssons artikel i PB 86-5. Det du bör observera är att den kräver expansionsminne 32 kb om du vill använda alla grafikrutiner. Fristående 32k RAM som tar ström från datorn bör ej användas p g a att kombinationen tar för stor strömförbrukning. Grafikrutinerna måste initieras i kommandomod med CALL APESoft vilket betyder att du ej kan använda autoloadd med DSK1.LOAD tillsammans med grafikrutinerna. Myarc Extended Basic II kan endast användas tillsammans med Myarc RAM-disk. Programmet finns på disk men laddas med en egen modul. CALL VERSION ger 200 som svar.

GK Utility I från MG (f d Millers Graphics) kan endast användas tillsammans med GRAM Kracker men samma rutiner har tagits med i en helt ny modul kallad Super Extended Basic. Denna har ytterligare några kommandon som ej finns med på GK Utility I. Följande kommandon från GRAM Kracker (se PB 87-1) XB-CALLS finns med Super-XB:

- CALL NEW
- CALL BYE
- CALL CLSALL
- CALL CLOCK
- CALL CLOCKOFF
- CALL CAT

Följande nya kommandon från GK Utility I finns med Super-XB:

- LIST kan göras med valfri bredd
- CALL LOAD som POKE fungerar även utan expansionsminne
- RES kan göras även på del av programmet mellan önskade radnummer
- TRACE kan även skriva mot printer eller disk
- skärmfärgen är vit text på blå bakgrund i kommandomod men ändras till svart på cyan vid RUN
- äkta små bokstäver och cursor som ett streck
- auto-load av DSK1.LOAD kan stoppas genom att hålla tangenten nedtryckt
- vid editering av långa BASIC-rader kan markören hoppa mellan början och slut. Du kan även flytta markören upp och ned mellan skärmraderna för samma programrad. Du har alltså en äkta skärmeditor inom programraden. Samma finesser finns med INPUT och ACCEPT.
- COPY kopierar önskat antal programrader till andra radnummer
- DEL raderar önskat antal programrader
- MOVE flyttar önskat antal programrader och ändrar alla referenser till dessa rader i GOTO m m.
- CALL PEEKG läser av GRAM/GROM
- CALL POKEG skriver i GRAM
- CALL PEEKV läser i VDP-RAM
- CALL POKEV skriver i VDP-RAM
- CALL QUITON aktiverar QUIT
- CALL QUITOFF stänger av QUIT

Följande kommandon som fanns listade med objektskod (i form av ett Extended BASIC-program med CALL LOAD-satser) i Smart Programmer dec 86 finns med i Super-XB:

- CALL BEEP ger ljus ton
- CALL HONK ger mörk ton

- CALL STSPRT stoppar alla sprite-
- CALL GOSPRT möjliggör rörliga sprite
- CALL SCROFF stänger av skärmen
- CALL SCRON aktiverar skärm
- CALL COLORS ger alla COLOR-set samma färger

Följande kommandon är helt nya och finns endast i Super-XB:

- CALL ALL fyller hela skärmen med ett tecken
- CALL CHIMES ljud
- CALL GOSUB med variabel som radnummer
- CALL GOTO med variabel som radnummer
- CALL KEYS har validate av tangent
- CALL ALOCK undersöker om ALPHA LOCK är nedtryckt
- CALL SHIFT undersöker SHIFT
- CALL CTRL undersöker CTRL
- CALL FCTN undersöker FCTN
- CALL VERSION ger 120 (normal XB ger 110)

Super Extended BASIC säljs av Triton i USA. Denna firma exporterar inte till Sverige. *TexComp* har annonserat en modul för USD 60 med samma namn men enligt annonsen finns endast de kommandon som kommer från GRAM Kracker GK Utility I enligt ovan. Det är nog troligt att det är samma modul men att annonsen ej är fullständig.

Referenser

1. Mechatronic Extended Basic PB 86-5.
2. Mechatronic XB II PLUS *Micropendium* okt 85.
3. Mechatronic XB II PLUS *TI*MES* Nr 11 p. 54.
4. Myarc Extended Basic II *Micropendium* jan 87.
5. Myarc Extended Basic II *TI*MES* Nr 15 p. 45.
6. GK Utility *Micropendium* dec 86.
7. Seven new XB CALL's *Smart Programmer* dec 86.

Disk Manager 1000 v 3.5 — varianter

av Jan Alexandersson

Det finns tre olika varianter av DM 1000 version 3.5:

- Originalversion 33+30 sektorer
- RAM-diskversion 33+31 sektorer
- Funnelwebversion 33+29 sektorer

Den variant som vi distribuerat inom föreningen är RAM-diskversionen. Vi har inte lyckats få kontakt med *Ottawa Users Group* så DM 1000 har vi endast fått i andra hand. Från Stephen Shaw i England har vi lyckats få en kopia av originalversionen av DM 1000. MGR2-filen är här en sektor kortare. Den riktiga LOAD-filen som är en generell assembler-laddare finns med samt en textfil som beskriver vad som skiljer olika versioner av DM 1000.

Det finns små skillnader mellan de tre varianterna. Originalversionen kan inte anropa RAM-disk på CRU-adress > 1200, vilket de andra klarar. Om du har fått DM 1000 version 3.3 genom föreningen så klarar även denna CRU > 1200 eftersom den kommer från skivor från Horizon. Om du väljer ett disknummer som ej används t ex DSK5 så kommer RAM-diskversionen att skriva konstiga saker på skärmen. Disknamn blir ett filnamn på DSK4 och FREE=902 och USED=4634. Programmet hänger sig sedan i 10–20 sekun innan DISK ERROR kommer upp på skärmen. De övriga varianterna klämmer fram DISK ERROR omedelbart.

Funnelwebversionen som naturligtvis medföljer Funnelweb har endast återhopp till Funnelweb medan de övriga kan ladda in valfritt assemblerprogram.

Karaktärsdanande älvödare

av Sören Bernle

I PB 85-4 var jag förmåten nog, att anse mig själv som en lämplig vägledare i hur man kommer igång i E/A-programmering (på en mycket elementär nivå).

Några medlemmar har låtit mig förstå, att mina två artiklar och Anders Perssons omfattande utläggning i *Utmaningen* (PB 85-4), ej är tillräckligt förklarande för att de ska kunna förstå det allra enklaste E/A-program.

Min första tanke efter denna (svidande) kritik var: "Dom har inte gjort sig omaknet att läsa igenom E/A-manualen och lära sig de instruktioner och direktiv och subrutiner som förekommer i de olika artiklarna!"

Efter denna vettiga tanke kom genast en självvrannsan: "Har du glömt hur det var för dig själv Sören?, du läste ju manualen fram och tillbaka flera gånger och be- grepp ändå inte ett dugg!"

Nej jag har inte glömt!

När jag nu tänker tillbaka på och försöker analysera de svårigheter jag hade med att lära mig det lilla jag kan i assembler, upptäcker jag följande:

1. Jag visste inte om, att assembler-program exekve- ras i samma ordning som i BASIC, dvs att instruktioner- na utförs i samma ordning som de är listade.

För mig var det (hur tokigt det än låter) fullt möjligt att det fanns en annan och för mig helt oförståelig ord- ning, som avgjorde vad som skulle utföras och när!

2 A. Jag närmade mig assembler på samma sätt som när jag tidigare hade närmat mig BASIC. Vilket var mycket oturligt! Vad då oturligt?

Jo, om man jämför ett programs instruktioner med en ansamling av byggmaterial, så är BASIC-programmering som att bygga med legoklotsar medan assembler är att bygga med de plastmolekyler som legoklotsarna är bygg- da av.

Har man bara plastsmet att bygga med och man har den inställningen att man skall bygga en modell utan att först ha tillverkat byggmoduler har man minst sagt svårt att överblicka och genomföra hela projektet.

Att vägen till framgång i assembler-programmering går via subrutiner, har väl egentligen först nu i skrivande stund gått helt upp för mig.

2 B. Svårigheten att ta mig ur den psykologiska be- gränsning som BASIC håller mig i. Nedanstående pro- gram som ersätter en del BASIC-instruktioner är det första assembler-program som jag skrivit utan att först ha skrivit det i BASIC. Dvs mina programidéer är bygg- da av BASIC-legoklotsar. M a o, mina assemblerpro- gram, förutom det sista, är snarast att betrakta som upp- snabbade BASIC-program.

3. Den stora avsaknaden av minnestabeller i manua- len har varit ett stort hinder, men vissa publiceringar i PB har varit mig till stor hjälp.

4. Min absoluta okunskap om vad som hårdvarumässigt sker i datorn, är förstås belastande, men tydligen inte ett tillräckligt stort hinder för att inte kunna uträtta något alls i assembler.

Med förhoppning om att ovan sagda löser någon "knut" hos Dig, avslutar jag här min självvrannsan.

Karaktärsdanande älvödare

Min irritation över att Extended BASIC-subprogrammet CHARSET inte laddar in standard karaktärer med ASCII-värde över 95, samt att subrutinen DELSPRITE inte alltid får all sprite-aktivitet att upphöra, fick mig att knäpa ihop nedanstående lilla assembler-snutt.

Programmet uträttar följande:

— Samtliga standard karaktärer, från ASCII 32 t o m 127 återställs.

— ASCII-karaktärerna 128 t o m 143 nollställs.

— Samtliga karaktärer tilldelas färgen svart på trans- parent.

— Svenska tecken: ÉÄÖÅÜéäöåü, laddas in.

— Samtliga sprite "dödas".

— MAGNIFY sätts till ingen förstoring.

— QUIT-funktionen (FCTN =) inaktiveras.

CHRSET-programmet är främst tänkt att användas i ett meny-program, där det anropade programmet gör återhopp till meny-programmet, när Du och/eller pro- grammet så vill.

Applikations-exempel:

Extended BASIC-programmet LOAD, som eventuellt ge- nererar något välkomnande i bild och/eller ljud, avslutas med programraderna:

```
CALL INIT
CALL LOAD("DSK1.CHAR.SET")
RUN("DSK1.MENY")
```

Extended BASIC-programmet MENY startar med rader- na:

```
CALL INIT
CALL LINK("CHRSET")
```

övriga rader innehåller det egentliga MENY-program- met.

Varje program som anropas av MENY-programmet av- slutas med raden:

```
RUN(DSK1.MENY)
```

Med undantag av Anders Perssons galanta GPLLNK- rutin (vilken jag fått Anders muntliga tillstånd att låta publicera), och som jag själv inte begriper ett smack utav, så kan Du faktiskt förstå programmet, om Du bara ger Dig tid och har både E/A-manualen och sidorna 10—11 i PB 84-5 att tillgå, samt följande två kunskapstillägg som jag fått via en tillfällig tillgång till ett *Explorer*-pro- gram:

Extended BASIC använder endast 15 bytes i färgtabel- len, nämligen de på adresserna >80F—>81D.

De 28 stycken 4 bytes stora värdena i SPRITE AT- TRIBUTE TABLE har värdet >C0000000 som startvär- de.

```
***** X-BASIC CHARACTER RESET *****
*
* This subroutine:
*
* Restores the standard character patterns and colors
* for characters ASCII 32 through ASCII 143.
*
* Loads 5 swedish characters, in both upper and lower
* case.
*
* Stops all sprite activity. Disables the quit key.
*
* Thank's Anders Persson for the GPLLNK subroutine!
*
* 870401
*
* S. Bernle
* PL 280
* S-260 38 Kattarp Tel: 042-93305
*
*****
```

```
DEF CHRSET
*GPLWS EQU >83E0 IS IN THE GPLLNK ROUTINE
VDPWA EQU >8C02
VMBW EQU >2024
STATUS EQU >837C
FAC EQU >834A
BUFF BSS 264
MYWS BSS 32
```

* SWEDISH CHARACTER DATA

```
CHR64 DATA >000B,>7C40,>7B40,>407C
CHR91 DATA >0044,>3B44,>447C,>4444
DATA >0044,>7C44,>4444,>447C
DATA >0010,>3B44,>447C,>4444
DATA >0044,>0044,>4444,>447C
CHR96 DATA >000B,>107C,>407B,>407C
CHR123 DATA >0000,>443B,>447C,>4444
DATA >0000,>447C,>4444,>447C
DATA >0000,>103B,>447C,>4444
DATA >0000,>4400,>4444,>447C
```



```

* EXECUTION START
CHRSET LWPI MYWS
      CLR R5          USED AS EMPTY REGISTER

* LOAD STANDARD CHARACTER SET
      LI R3,>0400      LOAD CAPITAL CHARACTER SET
      MOV R3,@FAC      INTO VDP-RAM
      MOV R5,@STATUS
      BLWP @GPLLNK
      DATA >001B

      LI R3,>0600      LOAD LOWER-CASE CHARACTER SET
      MOV R3,@FAC      INTO VDP-RAM
      MOV R5,@STATUS
      BLWP @GPLLNK
      DATA >004A

      LI R1,BUFF       CLEAR THE BUFFER
      LI R2,132
      MOV R5,#R1+
      DEC R2
      JNE MORE1

      LI R0,>06FB      ERASE CHR 127 - 143 AND CLEAR
      LI R1,BUFF       SPRITE MOTION TABLE
      LI R2,264
      BLWP @VMBW

* SET COLORTABLE TO DEFAULT VALUE
      LI R1,BUFF       SET 16 BYTES TO >10
      LI R2,B          (i.e black on transparent)
      LI R3,>1010
      MOV R3,#R1+
      DEC R2
      JNE MORE2

      LI R0,>080F      MOVE 15 OF THIS BYTES INTO
      LI R1,BUFF       VDP-RAM COLORTABLE
      LI R2,15
      BLWP @VMBW

* LOAD 10 SWEDISH CHARACTERS
      LI R0,>0500      ASCII 64
      LI R1,CHR64
      LI R2,B
      BLWP @VMBW

      LI R0,>05DB      ASCII 91-94
      LI R1,CHR91
      LI R2,32
      BLWP @VMBW

      LI R0,>0600      ASCII 96
      LI R1,CHR96
      LI R2,B
      BLWP @VMBW

      LI R0,>06DB      ASCII 123-126
      LI R1,CHR123
      LI R2,32
      BLWP @VMBW

* STOP ALL SPRITE ACTIVITY
*
* (The sprite motion table is cleared above)
      LI R4,>C000      CREATE 28 TWO-WORDS
      LI R1,BUFF       CONTAINING >C0000000
      LI R2,28          (i.e. default value)
      MOV R4,#R1+
      MOV R5,#R1+
      DEC R2
      JNE MORE3

      LI R0,>0300      SET SPRITE ATTRIBUTE TABLE
      LI R1,BUFF       TO DEFAULT VALUE
      LI R2,112
      BLWP @VMBW

      LI R1,>E0B1      SET MAGNIFY FACTOR TO ZERO
      MOV R1,@>B3D4
      MOV R1,@VDPWA
      SWPB R1
      MOV R1,@VDPWA
      NOP

      MOV R5,@>B37A    SET HIGHEST SPRITE # IN
*                      AUTO-MOTION TO ZERO

* DISABLE THE QUIT KEY
      LI R1,>1000
      MOV R1,@>B3C2

```

```

* RETURN TO THE CALLING PROGRAM

```

```

      MOV R5,@STATUS
      LWPI GPLWS
      B @>0070

```

```

* XB GPLLNK ROUTINE * A-DATA

```

```

UTILWS EQU >2038
SUBSTK EQU >8373
GRMRA EQU >9802
GRMWA EQU >9C02
GPLWS EQU >83E0
PAD EQU >8300

```

```

GPLLNK DATA UTILWS,GPLPGM

```

```

GPLPGM MOV R5,@GRMRA,R1  SAVE GROM ADDRESS
      SWPB R1
      MOV R5,@GRMRA,R1
      SWPB R1
      AI R1,-3           BACK UP TO THE XML INSTRUCTION
      MOV R5,@SUBSTK,R2
      SRL R2,B
      AI R2,PAD          PUSH XML ADDRESS FOR RETURN
      INCT R2

      MOV R1,#R2
      SWPB R1
      MOV R1,@1(R2)
      SWPB R2
      MOV R2,@SUBSTK

      MOV R14+,@GRMWA SET UP ADDRESS TO CALL
      MOV R14+,@GRMWA
      MOV @>2000,R4      SAVE CURRENT XML LINK
      LI R3,GPLEND
      MOV R3,@>2000
      LWPI GPLWS
      B #R11

```

```

GPLEND LWPI UTILWS
      MOV R4,@>2000
      RTWP
      END

```

Ken Gilliland

av Jan Alexandersson

från Ken Gilliland i USA har det kommit 7 diskar med musik och grafik. Varje skiva har en snygg etikett som kan printas ut med MAX-RLE. Tre av skivorna har dessutom en extra stor bild för att sätta på skivhuvudet.

Alla skivorna kräver Extended BASIC och 32 kb minne för att fungera. Om speech finns kommer datorn att sjunga på fyra av skivorna även om dessa fungerar utan speech. Följande skivor finns nu i Programbanken:

- WAGNER (2 skivor) med grafik och musik
- STAR TREK med grafik och musik
- SOUTH PACIFIC med grafik, musik och sång
- PATSY CLINE med grafik, musik och sång
- MUSIC MAN med grafik, musik och sång
- WIZARD OF OZ med grafik, musik och sång

Sångprogrammen innehåller Text-to-Speech programmet från TI. Det finns risk att detta inte fungerar med t ex Myarc 128—512 kb minne.



Nyheter från Tyskland

av Jan Alexandersson

Det finns många säljare av tillbehör till TI-99/4A i Tyskland. Jag har skaffat kataloger från några av dessa. Dessa kataloger är endast enkla utskrivna sidor utan några längre förklaringar och bilder. Priserna är ej lika låga som i USA. Ett program som kostar USD 20 (c:a 200 SEK) kostar i Tyskland DEM 100 (c:a 400 SEK). Från listornas priser kan du dra bort tysk moms med 14 % men sedan tillkommer svensk moms vilket gör ett en D-mark kommer att kosta 4 kr.

1. ELEKTRONIK-SERVICE

Peter Kleinschmidt
Linning 37
D-4044 KAARST 2
Västtyskland

Denna firma tillverkar foliekorten till t ex GRAM-KARTE. Det finns många halvfabrikat för den som vill pula litet själv:

Dummy-Platine för expansionbox inkl. 5 V stabilisering och 4 drivkretsar 74LS244. Pris DEM 89.
Eprommer-Gerät för programmering av EPROM. Pris DEM 200.

Modulplatine pris DEM 19.

Modulkontakt 36-polig pris DEM 8.

Busskontakt 44-polig pris DEM 8.

GRAM-KARTE 128 kb pris DEM 598.

80 Zeichen Erweiterung med 192 kb RAM. Pris DEM 450.

Mechatronic Extended Basic II plus pris DEM 200.

Horizon RAM-disk DS/DD (1440 sektorer) bestyckad med 12 st 32 kbytes kretsar 58256P-12L + 1 st 8 kbytes 6264LP-15. Detta ger totalt 392 kbytes. Pris DEM 698.

Mechatronic operativsystem för Foundation 128 kb RAM-DISK. Pris DEM 69.

GPL-Assembler pris DEM 88.

TI 99/4A INTERN på engelska pris DEM 20.

2.

Wolfgang Riegert
Schlosshofstrasse 5
D-7324 RECHBERGHAUSEN
Västtyskland

TURBO-PASC'99 pris DEM 149.

Window Manager för TURBO-PASC'99 pris DEM 99.

Graphik Toolkit för TURBO-PASC'99 pris DEM 99.

Atronic CPS 99 expansionssystem med 32k, RS232/PIO, Diskkontroll DS/DD och en diskdrive DS/DD. Pris DEM 999.

3.

RAUSCH & HAUB
Berliner Freiheit 16
D-5300 BONN 1
Västtyskland

Realtidsklocka för inbyggnad i konsolen pris DEM 80.
Myarc Geneve 9640 pris DEM 1150.

4.

Programm-Service REIS
Bergstrasse 80
D-5564 BULLAY
Västtyskland

Atronic CPS 99 pris DEM 998.
MEGA-RAM pris DEM 1242.
GPL-Assembler pris DEM 98.

5.

albs-Alltronic G.Schmidt
Postfach 1130
D-7136 ÖTISHEIM
Västtyskland

Mechatronic Extended Basic II PLUS pris DEM 149.
Mechatronic 128 kb + Centronics pris DEM 200.
GRAM-Karte pris DEM 698.
EPROMMER pris DEM 200.
80 kolumnerskort pris DEM 450.

Tips för Pascal

av Anders Persson

Även om det är fullt möjligt att använda UCSD Pascal tillsammans med SS/SD disketter, kräver det en del eftertanke för att bli så lättarbetat som möjligt. Nedanstående fördelning av filer har visat sig göra det bästa möjliga av det begränsade utrymmet.

PASSYS1:

SYSTEM.CHARAC	2	24-Feb-85
SYSTEM.SYNTAX	14	2-Apr-84
SYSTEM.MISCINFO	1	1-Oct-85
SYSTEM.PASCAL	7	22-Oct-85
SYSTEM.FILER	34	13-Mar-81
SYSTEM.LIBRARY	66	26-Feb-86
SYSTEM.STARTUP	2	26-Feb-86

7/7 files, 48 unused, 48 in largest

PASSYS2:

SYSTEM.COMPIILER	99	27-Apr-82
SYSTEM.EDITOR	45	6-Nov-81
STARTUP.CODE	11	26-Feb-86

3/3 files, 19 unused, 13 in largest

Minst två drivverk krävs. PASSYS1: placeras i # 4, PASSYS2: i # 5.

Uppdelningen ovan lämnar maximalt med utrymme på *, dvs PASSYS1:, för att tillåta användandet av workfile. Även om man inte vill utnyttja någon workfile är det lämpligt att behålla filerna på samma sätt, och fortfarande utnyttja PASSYS1: för program under konstruktion. Om tre drivverk finns tillgängliga kan # 9: utnyttjas i sin helhet till käll- och objektkodsfiler.

Observera att SYSTEM.STARTUP är minimal för att tillåta en så stor workfile som möjligt. SYSTEM.STARTUP gör inget annat än startar PASSYS2:STARTUP.CODE. Den filen får då innehålla det som verkligen ska göras vid uppstart.

De som även vill skriva assemblerrutiner får skaffa sig tre drivverk *eller* kontrollkort och/eller drivverk med högre kapacitet *eller* byta diskett vid assemblering och länkning.

SYSTEM.LIBRARY i katalogerna ovan är bara 66 block långt. Ett större bibliotek inkräktar naturligtvis ytterligare på utrymmet, man kan å andra sidan reducera den nödvändiga storleken på själva programmet.

Skivorna ovan har enkla kataloger. Åtminstone PASSYS2: kan förväntas ha ett så statiskt innehåll att en kopia av disketten kan hålla sig identisk en längre tid. Däremot kan det nog vara lämpligt att offra fyra block på PASSYS1: och använda dubbel katalog där.

Att köra p-systemet med en enda SS/SD disk är rena skämtet. Det kan inte rekommenderas för någon annan än möjligen hängivna självplågare. Men av egen erfarenhet kan jag nämna att med fyra DS/DD skivor, vilket ger 2 880 block tillgängligt, är det en fröjd att köra Pascal!

Ny freeware

Creative Filing System version 7.0 av Mark Beck från juli 1987 är ett databasprogram som håller reda på register, dvs det är ett alternativ till *Personal Record Keeping* och *PR-BASE*. Det ryms på 3 skivor.

Fractal Explorer version 2.00 av Steve Langguth ritar kartliknande bilder i multicolor mode (Mandelbrot set). Det ryms på 2 skivor.

Knecht-1 och **Knecht-2** är två separata skivor med musik för Extended BASIC utan krav på expansionsminne.

Emusic preprocessor version 1.2 av Norm Sellers är ett assemblerhjälpmedel för att skriva musik i Extended BASIC. Du skriver musiken som vanliga BASIC-rader med ! följt av data så du behöver ej kunna något om assembler. Emusic är mycket avancerat men ändå lätt att använda. Programmet ryms på 2 skivor som även har flera färdigskrivna musikexempel. På skivan finns även *Systex* av Barry Boone som lagrar ett assemblerprogram tillsammans med Extended BASIC (hybridprogram).

Pulsar version 1.2 av Michael Amundsen består av ett stort antal (49 st) subprogram (källkod) i assembler som anropas med BL och har DATA-satser efter BL för att föra över variabler och konstanter. Det finns rutiner för flyttal, grafik, heltalsmatematik, tangenttryckningar, minneshantering, skärmrutiner och programflödestroll (IF THEN ELSE m m). Pulsar är endast för dig som skriver assemblerprogram men rutinernas namn och användning liknar vanlig BASIC-programmering mycket.

Star version 1.0 av Michael Riccio består av många (53 st) subprogram för Extended BASIC som anropas med CALL LINK. Det finns rutiner för teckenset, ljud, kasett, färger, POKEV, PEEKV, VDPREG, spara/ladda skärm, NEW, BYE, tangentkontroll (ALPHA LOCK, SHIFT, CTRL, FCTN), katalog, sektorhantering, textmod (40 tecken) och grafikmod.

Linker version 1 av R.A. Green gör om assemblerprogram i DIS/FIX 80-format till PROGRAM-format. Den kan även lösa ut referenser som t ex VMBW vilket gör att resultatet kan laddas av alla moduler (XB, EA, MM och TW) och resultatet blir mycket kompaktare och laddas även snabbare.

Enhanced Display Package version 2.1 från Pargon Computing har ett stort antal (33 st) CALL LINK-rutiner som anropas från Extended BASIC. Följande rutiner finns: ACCEPT, ALARM, BUFFER, CHAR, CHARPAT, CHRSET, CLOCK, CLEAR, COLOR, DISPLAY, FLASH, GCHAR, GTEXT, HCHAR, INTOFF, INTON, LOADR, LOADV, MODE, PEEKV, READ, RESET, SCREEN, SCROLL, SHOW, STEADY, STRING, TEXT, TIME, TXTPAT, VCHAR, WINDOW, och WRITE. Du kan använda alla rutiner i grafikmode med 32 tecken eller textmode med 40 tecken.

Disk Aid av Col Christensen är skrivet i Forth och kan analysera hur sektorerna används och även editera dessa.

Disk Utilities version 3.3-1 av John Birdwell kan jämföra skivor eller filer och skriva ut skillnaden. Du kan även söka upp en textsträng på hela skivan eller i en fil. Det ingår en sektoreditor och även möjlighet till att skriva kommentarer till varje fil som lagras i filhuvudet.

XB*Tools version 1.2 av Jim Swedlow liknar Smash och Programming Aids III. Du kan få en referenslista med radnummer för GOTO m m. Det går att få ett kompaktare XB-program genom att variabel- och konstantnamn byts ut mot en bokstav, konstanten 1 byts mot @, REM och ! tas bort och närliggande rader skrivs på en rad. *Namchange* byter variabelnamn och *Linemove* flyttar, ra-

derar eller gör RES på rader. *Datapress* slår ihop DATA-rader även om de är utspridda i programmet.

Basicsort av A.L. Dessoff har assemblerrutiner som anropas med CALL LINK från BASIC eller Extended BASIC.

TI Sings av Barb Berg för Terminal Emulator II gör att datorn kan sjunga med Speech Synthesizer.

Neatlist version 1.0 nu med 360 sektorer och 16 filer. Troligen inget nytt utom ett program som skriver ut manualen.

Screendump version 3.01 är en förbättrad version från juni 1985. Den är helt komplett inkl källkod och kommer direkt från Danny Michael juni 1987.

Superbug II version 1.0 inklusive manual på 40 sidor finns nu. Om Du beställer Superbug II får Du version 2.1 (*Obs!* ny version) på disk och manualen till version 1.0 utskrivna. Fullständig manual till version 2.0 (52 sidor) kan beställas från USA för USD 5 plus porto USD 3.

Funnelweb version 3.4 med datum MAY/14/87 på 2 skivor.

S/Foster musik och grafik med följande program: Gledyburk, Green/V&F, Hoedown, Take/Five, Wierd, Yesterday. Kräver 32K och XB. Tre av programmen fungerar även utan 32K.

Sörens menyprogram

På sid 4—5 publicerar vi Sörens program för att ladda in önskad teckenuppsättning. Hans rubrik "älvdödare . . ." gör ju att åtminstone undertecknad (redaktören) hajar till innan man fattar att det är en ordlek (sprite kan ju översättas med älvor eller andra övernaturliga väsen). Samtidigt med denna artikel kom från Sören några andra program också. Dels är det ett program som han lagrat som LOAD på disketten. Det laddas alltså in automatiskt vid uppstart med Extended BASIC. Programmet är det första som listas nedan. Det anropar i sin tur assemblerprogrammet CHAR.SET och XB-programmet MENY. Assemblerprogrammet måste förstås vara assemblerat först. Även den som inte har expansionsminne och diskenhet kan nog finna en del matnyttigt i Sörens sätt att använda sprites.

Nästa program vi listar är MENY-programmet som nämns ovan. Slutligen listar vi också ett program kallat PÅSK. Varför det heter så lär framgå om man knappar in det och kör. Det kräver Extended BASIC, men är i övrigt fristående från de andra programmen. Sista raden i det kan man ju hoppa över om man vill. Ytterligare ett alternativ finns i menyn: Sverige 1967. Det programmet tar vi dock inte med denna gång. Huvudsyftet var ju ändå att visa hur meny kan användas.

```
100 CALL CLEAR :: CALL SCREEN(2)
101 CALL SAY("WELL COME")
102 RANDOMIZE
105 CALL CHAR(95,"7E8188887E09897E")
110 CALL CHAR(129,RPT$("00000038447C44440000
007844784844",2)):: CALL CHAR(133,"00000
07C4078407C")
120 CALL SPRITE(#2,45,16,1,1,127,127):: CALL
SPRITE(#3,98,16,1,256,-128,127):: CALL
SPRITE(#4,97,16,192,1,127,-128)
130 CALL SPRITE(#5,115,16,256,192,-128,-128)
:: CALL SPRITE(#6,105,16,9,1,120,120)::
CALL SPRITE(#7,99,16,9,256,-120,120)
140 CALL SPRITE(#8,112,16,183,1,120,-120)::
CALL SPRITE(#9,114,16,183,256,-120,-120)
:: CALL SPRITE(#10,111,16,17,1,112,112)
```



```

150 CALL SPRITE(#11,103,16,17,256,-112,112):
: CALL SPRITE(#12,130,16,175,1,112,-112)
: CALL SPRITE(#13,129,16,175,256,-112,-
112)
160 CALL SPRITE(#14,109,16,25,1,104,104):: C
ALL SPRITE(#15,131,16,25,256,-104,104)::
CALL SPRITE(#16,118,16,167,1,104,-104)
170 CALL SPRITE(#17,83,16,167,256,-104,-104)
: CALL SPRITE(#19,66,16,33,1,96,96):: C
ALL SPRITE(#20,101,16,33,256,-96,96)
180 CALL SPRITE(#21,132,16,159,1,96,-96):: C
ALL SPRITE(#22,110,16,159,256,-96,-96)::
CALL SPRITE(#23,108,16,41,1,88,88)
190 CALL SPRITE(#24,133,16,41,256,-88,88)::
CALL SPRITE(#1,88,16,151,1,88,-88):: CAL
L SPRITE(#25,95,16,151,256,-88,-88)
195 CALL SPRITE(#18,46,16,41,1,88,88)
200 FOR I=1 TO 1200 :: NEXT I
210 CALL MOTION(#1,57,57,#2,-57,57,#3,57,-57
,#4,-57,-57,#5,49,49,#6,-49,49,#7,49,-49
,#8,-49,-49)
220 CALL MOTION(#9,30,30,#10,-30,30,#11,30,-
30,#12,-30,-30,#13,22,22,#14,-22,22,#15,
22,-22,#16,-22,-22)
230 CALL MOTION(#17,12,12,#18,-12,12,#19,12,
-12,#20,-12,-12,#21,4,4,#22,-4,4,#23,4,-
4,#24,-4,-4,#25,-4,-4)
240 FOR I=1 TO 25 :: NEXT I
270 FOR I=1 TO 14 :: READ A,B :: CALL PLACE(
I,A,B)
280 NEXT I
290 READ A,B
300 I=25 :: CALL PLACE(I,A,B)
320 FOR I=15 TO 24 :: READ A,B :: CALL PLACE
(I,A,B)
330 NEXT I
350 FOR I=1 TO 200 :: NEXT I
360 CALL INIT :: CALL LOAD("DSK1.CHAR_SET")
410 FOR I=1 TO 24 :: CALL PATTERN(#I,42):: C
ALL MOTION(#I,INT(RND*256)-128,INT(RND*2
56)-128):: NEXT I
420 FOR I=1 TO 50 :: NEXT I
430 FOR I=1 TO 24 :: CALL DELSPRITE(#I):: NE
XT I :: CALL MAGNIFY(2)
432 CALL SAY("PLEASE PRESS ANY KEY")
440 CALL KEY(0,K,S):: IF S=0 THEN 440
442 CALL SAY("JUST A1 SECOND")
450 RUN "DSK1.MENY"
500 DATA 76,94,66,104,56,114,46,124,56,134,6
6,144,76,154,96,94,106,104,116,114,126,1
24,116,134
510 DATA 106,144,96,154,86,124,146,119,146,1
29
520 DATA 156,89,164,99,172,109,180,119,180,1
29,172,139,164,149,156,159,00,00
530 DATA 125
1000 CALL MOTION(#I,0,0):: CALL LOCATE(#I,A,B
)
1010 FOR II=1 TO 20 :: NEXT II
1020 RETURN
4990 SUB PLACE(N,A,B)
5000 CALL POSITION(#N,R,C)
5010 CALL MOTION(#N,(A-R)/2,(B-C)/2)
5020 CALL COINC(#N,A,B,96,F):: IF F=0 THEN 50
20
5030 CALL POSITION(#N,R,C)
5040 CALL MOTION(#N,(A-R)/2,(B-C)/2)
5050 CALL COINC(#N,A,B,10,F):: IF F=0 THEN 50
50
5060 CALL POSITION(#N,R,C)
5070 CALL MOTION(#N,A-R,B-C)
5080 CALL COINC(#N,A,B,3,F):: IF F=0 THEN 500
0
5090 CALL POSITION(#N,R,C)
5100 CALL MOTION(#N,A-R,B-C)
5110 CALL COINC(#N,A,B,0,F)
5120 IF F=0 THEN 5010
5130 CALL MOTION(#N,0,0)
5140 CALL LOCATE(#N,A,B)
5150 SUBEND

```

```

100 GOTO 140
110 CALL KEY :: CALL LINK :: CALL SAY :: CAL
L SCREEN :: CALL CLEAR
120 K,S
130 !EP-
140 CALL CLEAR :: CALL SCREEN(16):: CALL LIN
K("CHRSET")
150 DISPLAY AT(1,10):"M E N Y"
160 DISPLAY AT(12,1):"1 -> PASK HALSNING"
170 DISPLAY AT(13,1):"2 -> I SVERIGE 1967"
180 DISPLAY AT(24,1):"0 -> AVSLUTA PROGRAMM
ET"
190 CALL SAY("MAKE YOUR CHOICE")
200 CALL KEY(0,K,S):: IF S=0 THEN 200
210 IF K<48 OR K>50 THEN 200 ELSE K=K-47
220 ON K GOSUB 230,240,250
230 DISPLAY AT(14,1)ERASE ALL:"SKRIV 'BYE' O
CH TRYCK ENTER!" :: BREAK
240 RUN "DSK1.PASK"
250 RUN "DSK1.SVERIGE"
260 END

```

```

1 REM SAVED AS "PASK"
2 REM RAD 4 TOM 8 = PRE SCAN
4 GOTO 100 :: CALL CLEAR :: CALL CHAR :: C
ALL SCREEN :: CALL COLOR :: CALL HCHAR ::
: CALL MAGNIFY :: CALL SPRITE :: CALL LO
CATE :: CALL MOTION :: CALL POSITION ::
CALL PATTERN :: CALL DELSPRITE :: RV ::
CV :: P :: PU :: Y :: X :: XX :: YY :: P
AT
8 !EP-
100 CALL CLEAR
110 CALL CHAR(96,"0000001F1E7C1E0F")
120 CALL CHAR(97,"0000003B07")
130 CALL CHAR(98,"0F1F3763C37B0703")
140 CALL CHAR(99,"C0E0F0F8F8FCFCFC")
150 CALL CHAR(100,"0303020100000001")
160 CALL CHAR(101,"FEFF0C0080408")
170 CALL CHAR(102,"8082B7FEFC46A9A8")
180 CALL CHAR(103,"804247FEFC444444")
190 CALL CHAR(104,"7FC098442312806")
200 CALL CHAR(112,"0141E17F3F629515")
210 CALL CHAR(113,"0142E27F3F222222")
220 CALL SCREEN(5)
230 RV=5 :: CV=-10
235 CALL COLOR(12,3,1)
240 CALL CHAR(120,RPT$( "F",16))
250 CALL HCHAR(22,1,120,96)
270 CALL MAGNIFY(1)
280 CALL SPRITE(#1,96,2,160,140,RV,CV)
290 CALL SPRITE(#2,97,2,167,133,RV,CV)
300 CALL SPRITE(#3,98,2,167,140,RV,CV)
310 CALL SPRITE(#4,99,2,167,147,RV,CV)
320 CALL SPRITE(#5,100,2,174,140,RV,CV)
330 CALL SPRITE(#6,101,2,174,147,RV,CV)
340 CALL SPRITE(#7,104,9,174,154,RV,CV)
350 CALL SPRITE(#8,103,2,162,1,0,1)
360 P=1
370 RV=10 :: CV=-20
375 PU=0
380 RV=RND*12-8 :: CV=RND*12-10
385 PU=PU+1
390 CALL MOTION(#1,RV,CV,#2,RV,CV,#3,RV,CV,#
4,RV,CV,#5,RV,CV,#6,RV,CV,#7,RV,CV)
400 CALL POSITION(#1,X,Y)
410 IF Y>240 THEN Y=240
420 IF Y<11 THEN Y=240
425 IF Y<11 THEN Y=240
430 IF X<11 THEN X=176
440 IF X=176 THEN Y=240
442 CALL POSITION(#8,XX,YY)
444 IF YY<=Y THEN 448
446 PAT=112 :: CALL MOTION(#8,0,-2):: GOTO 4
50
448 PAT=102 :: CALL MOTION(#8,0,2)
450 CALL LOCATE(#1,X,Y,#2,X+7,Y-7,#3,X+7,Y,#
4,X+7,Y+7,#5,X+14,Y,#6,X+14,Y+7,#7,X+14,
Y+14)
460 IF P=0 THEN P=1 ELSE P=0
470 CALL PATTERN(#8,PAT+P)
480 GOTO 380
490 CALL CLEAR
500 CALL DELSPRITE(ALL)
510 RUN "DSK1.MENY"

```


MAX-RLE

av Jan Alexandersson

Om du har en matris skrivare kan du också printa ut bilder i punktgrafik med speciella grafik-koder. Vi har tidigare beskrivit *SCREENDUMP* som gör att du kan få ut en exakt kopia av skärmen på din skrivare. Detta gäller om skrivaren har samma kontrollkoder för grafik som EPSON.

Det har nu dykt upp ett nytt "FREEWARE"-program av Travis Watford kallat *MAX-RLE*. RLE är en förkortning för Run Length Encoding. Detta är ett kompakt sätt att lagra bilder för punktgrafik. RLE-bilderna är lagrade i DIS/FIX 128 eller DIS/VAR 80. Dessa bilder kan printas ut av programmet till en EPSON-kompatibel printer.

För att göra det hela mera flexibelt kan *MAX-RLE* även läsa in grafik-filer som har skapats med *TI-ARTIST* eller *GRAPH-X* och printa ut dessa eller lagra tillbaka på skivan i något av dessa fyra format:

- DIS/FIX 128 RLE
- DIS/VAR 80 RLE
- GRAPH-X
- TI-ARTIST

Detta gör att du kan konvertera bilder mellan de fyra olika lagringsformaten. Det finns i programbanken två olika skivor:

MAX-RLE EA

Denna kan laddas med modulerna Editor/Assembler eller Mini Memory. Med FUNNELWEB kan du ladda in detta program även med Extended Basic. Du behöver dessutom 32k minne och disk drive. Följande bilder följer med skivan: E-PRESLEY, C-BRINKLEY, SNOOPY och STAN&OL-LIE. Manualen på 2 sidor finns endast på denna skiva.

MAX-RLE XB

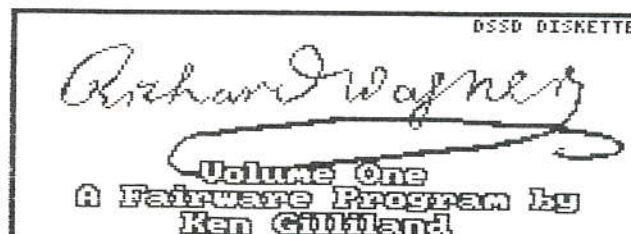
Denna kan laddas av modulerna Extended Basic, Editor/Assembler och TI-Writer. Även här behövs 32k minne och disk drive. Följande bilder följer med skivan: 3GIRLS/X, BLOOM/OPS3, CHRISDOS, DISNEY01, DRAGON&EGG, STARTREK08, SUMMERNITE och VAMP.

Bilderna på den första skivan kan användas med XB-laddaren på den andra skivan. Alla bilder visas först på skärmen innan du bestämmer om de skall printas ut. I de fall GRAPH-X och TI-Artist har många färger så kan du även se dessa på skärmen.

Du kan även använda *MAX-RLE* för att printa ut bilder på de skivor som kommer med musik från Ken Gilliland.

MAX-RLE fungerar mycket bra till vad den skall göra. Det finns dock några småfel i programmet. Första menyn blir svart text på grön med EA eller MM respektive vit text på mörkblå med cyan skärm för Extended Basic. När andra menyn kommer upp blir hela skärmen blå med vit text. Markören (cursor) saknas för alla laddare utom EA Load and Run och MM Load and Run. Om du försöker använda QUIT så kommer underliga saker att hända. Även om färgbalkarna syns blir det inget äkta reset. Du måste använda strömbrytaren.

Bilderna på denna sida är alla utskrivna från *MAX-RLE*.



Turbo Pascal till TI-99/4A

av Mikael Nordlin

Den 16 maj i år var det svensk premiär för ett nytt programspråk till 99:an. På ett medlemsmöte i Stockholm hade jag då nöjet att visa TURBO-PASC'99 för Programbitens medlemmar.

Handbok

Jag hade köpt programmet från en tysk postorderfirma, CSV Riegert, men bara haft tillgång till programmet ett par dagar. På mötet demonstrerade jag kompilatorn genom att skriva, kompilera och köra några enkla program såsom primtalsberäkning med iterativ resp rekursiv metod.

För ca 500 kr över postgirot fick jag snabbt en diskett (SS/SD) och ett häfte på tyska, omfattande ca 100 icke-paginerade A5-sidor. Häfte är nog en mer adekvat benämning än den tyska översättningen "användarhandbok". Häftet vänder sig till den som redan kan Pascal, och nöjer sig med att redogöra för denna Pascal-dialekts elementa och fördefinierade funktioner. Någon 99-specifik teknisk information om hur 99:ans minneskarta används eller hur parametrar överförs till funktioner/procedurer ges inte. Häftet är limmat i ryggen och förefaller vara fotokopierat.

Kopieringsskydd

Disketten visade sig vara kopieringsskyddad genom att de sista spårens sektorer numrerats, inte från 0-8, utan från 18-28. Diskcontrollern tycker *inte* om att läsa sådana sektorer, och kopiering med DMII, Forth eller DM1000 misslyckas. Enbart kopiering av övriga sektorer medger att editorn startas som vanligt, men programmet kraschar vid kompilering. Pascal-kompilatorn (jag fick version 2.0) har alltså, precis som TURBO-Pascal till MS-DOS och CP/M en inbyggd editor, varom mera nedan. Jag anser att kopieringsskydd enbart är av ondo även för den legitime användaren, genom att personlig back-up förhindras. Eftersom jag inte har tillgång till något kopieringsprogram som kopierar hela spår (sådana lär finnas till 99:an, de nämnda kopieringsmetoderna kopierar en diskett sektor för sektor) försökte jag att kopiera disketten på en PC med några av de kopieringsprogram som finns till denna datorfamilj. Det fungerade endast tillsammans med "Option Board", som är ett kopieringsprogram till PC med egen diskcontroller på ett kort. Kopian fungerar utmärkt, men är förstås såsom en exakt kopia av originalet kopieringsskyddad.

Editorn

Programmet är utvecklat i Tyskland, vilket dock inte märks vid handhavandet. Menyer och kommandon är på engelska och lättbegripliga. Editorn är snabb och lättarbetad, mycket tack vare att den liknar den för oss 99-användare kända Editor/Assemblers editor såvitt gäller kommandostruktur. Editorn utnyttjar 80-teckens bredd, varav 40 visas samtidigt på skärmen. Precis som Editor/Assembler "scroller" editorn automatiskt i sidled. Den läser och skriver också i DIS/VAR 80 format. Källkoden lagras i minnet på ett effektivt sätt genom att de reserverade Pascal-orden lagras som en byte. På diskett lagras dock texten som ren ASCII, varför kompilatorn kan drivas såväl med den inbyggda editorn som med annan valfri DIS/VAR 80 editor, t ex TI-WRITER, Editor/Assembler eller MITEC. Som en särskild finess varnar editorn, vid avslut, om källkoden inte har lagrats på diskett. Genom att kompilatorn hela tiden finns i minnet begränsas tillgängligt minnesutrymme för källkod. Det rymmer ändå drygt 15 K källkod samtidigt i minnet, vilket med hänsyn till den effektiva lagringen av källkod som jag nämnde ovan, får anses vara helt tillräckligt. Skulle det utrymme ändå inte räcka för något större program, går det att lösa genom att kompilera funktioner/procedurer separat, som sedan anges vara externa till huvudprogrammet. Huvudprogrammet och tilläggsrutinerna kan sedan länkas ihop till en körbar Program-Image-fil, se mera om den medföljande länkaren nedan. Från editorn

kan man direkt övergå till kompilering. Vid kompileringsfel, t ex p g a att man glömt någon viktig detalj som ett semikolon, återgår kompilatorn till editorn med cursorn på det ställe där felet upptäcktes. Felmeddelanden ges i form av en kod, som man får slå upp i användarhäftet. Att felmeddelanden inte ges i klartext får antas vara en eftergift åt 99:ans begränsade minne. I allmänhet ser dock, i vart fall den med Pascal någorlunda väl insatte, genast vilket fel som har uppstått bara genom att titta på cursorns placering i källkoden. Felet kan då snabbt rättas till och kompilering påbörjas på nytt. Eftersom editor och kompilator finns i minnet samtidigt krävs inte någon tidsödande växling mellan editor och kompilator med åtföljande disk-access. Editorn ger också automatisk indragning (indentering) vid radbyte. Därigenom placeras cursorn i den nya raden omedelbart under det yttersta vänstra ordet på raden ovanför, vilket underlättar lättläst, strukturerad källkod. Kopiering, radering och flyttning av rader sker genom att blockmarkörer placeras i texten. Av denna anledning behövs alltså inte några radnummer och dessa visas inte heller i texten. Man kan dock genom att ange radnummer hoppa till valfri rad i källkoden. Om fel uppstår först vid körning av det färdiga programmet ges besked om radnummer och kod för felet. Tyvärr saknar editorn möjlighet att erhålla en filkatalog som i TI-WRITER.

Kompilatorn

Från editorn går man alltså direkt till kompilering. Denna kan ske till diskettfil eller, vid programutveckling i testsyfte, utan kodgenerering. I sistnämnda fall sker kompileringen blixtn snabbt och därigenom fås felkontroll med eventuella fel markerade i källkoden med angivande av felkod. Då kompilering sker till diskett, med kodgenerering, tar det hela förstås något längre tid beroende på den något slöa diskethantering med TI:s diskcontrollerkort (som jag har, Corcomps t ex lär vara snabbare). Kompilatorn genererar assemblerkällkod, som förefaller vara tämligen effektiv. Därefter måste alltså resultatet gå igenom assemblern. Inte heller efter detta ytterligare steg har man körbar kod, eftersom alla fördefinierade funktioner/procedurer finns i en särskild fil, som måste länkas ihop med det Pascal-program som skall köras.

Länkaren

Ett eget länkarprogram medföljer. Detta länkar ihop användarens assemblerade filer med varandra och med de biblioteksrutiner (funktioner/procedurer) som programmet använder. Dessa finns lagrade i en medföljande biblioteksfil (i Program-Image-format för snabb laddning) och laddas in automatiskt i erforderlig mån. Även länkaren har felhantering, men i klartext. Länkaren kan ladda in såväl komprimerade som okomprimerade program- eller biblioteksfiler. Länkaren har en bug i det att endast filnamn om längst nio tecken accepteras.

Egenheter

TURBO-PASC'99 utgör en egen Pascal-dialekt, som i många stycken liknar UCSD Pascal. Således finns datatypen STRING. Inom klammer måste anges strängens maxlängd (högst 255). Flera UCSD strängfunktioner finns, men ofta med andra namn, liknande TI-BASIC. Flyttal finns, med alternativt 4, 6 eller 8 bytes precision. Vid tilldelning av värden till variabler och också i flera funktioner sker automatisk datatypsanpassning i viss mån. Datatypen CHAR (en byte) saknas, och får ersättas av STRING[1] som fordrar två byte liksom INT.

TYPE, SET OF samt RECORD saknas, liksom dynamisk stackhantering (MARK/HEAP/RELEASE). ARRAY har alltid ett undre värde av 0 och deklarerats t ex ARRAY[25] OF REAL[6]. Strängar anges inom citationstecken som i BASIC. Möjlighet saknas att lägga in assemblerkod inline, vilket i och för sig är i enlighet med standard Pascal, men ändå en brist när tillgängliga rutiner är alltför begränsade som t ex för grafik.


```

CONST
  gamma=32000;
  p=10;
  alpha=0.1;
  n=0.425;
  male=1;

VAR
  theta,
  wo,
  wf,
  intake,
  kb,
  ka,
  b,
  height,
  age: REAL[4];
  sex: INTEGER;
  answer: STRING[3];

FUNCTION odd(i: INTEGER): BOOLEAN;
BEGIN
  IF i MOD 2 = 0
    THEN odd:=false
    ELSE odd:=true
END;

FUNCTION find(s: STRING[21]): REAL[4];
VAR r: REAL[4];
BEGIN
  write(s, ">>"); readln(r); find:=r
END;

FUNCTION sum: REAL[4];
VAR
  s: REAL[4];
  j: INTEGER;

FUNCTION f(j: INTEGER): REAL[4];

FUNCTION w(j: INTEGER): REAL[4];
BEGIN
  w:=wo-(j/p)*(wo-wf)
END;

BEGIN {f}
  f:=1/((1-alpha)*intake-(ka*w(j)+kb*exp(n*ln(w(j)))));
END;

BEGIN {sum}
  s:=f(0)+f(p);
  FOR j:=1 TO p-1 DO
    IF odd(j)
      THEN
        s:=s+4*f(j)
      ELSE
        s:=s+2*f(j);
  sum:=s
END;

BEGIN {diet}
  writeln;
  writeln("Welcome to the diet!");
  age:=find("Age");
  sex:=cri(find("Sex (Male=1)"));
  height:=find("Height in metres");
  IF sex=male
    THEN
      b:=173.8-0.5195*age
    ELSE
      b:=156.36-0.3636*age;
  kb:=4.8*b*exp(0.725*ln(height));
  REPEAT
    writeln;
    writeln("Sample Activity Coefficients:");
    writeln("Sedentary 34, light 40, moderate 53,");
    writeln("vigorous 74, severe 113");
    writeln;
    ka:=find("Activity coefficient");
    wo:=find("Initial weight, kg");
    wf:=find("Final weight, kg");
    intake:=find("Daily food intake, kJ");
    theta:=(gamma*(wf-wo)/(3*p))*sum;
    writeln;
    writeln("Time for this weight change is ", cri(theta), " days.");
    write("Another? ");
    readln(answer)
  UNTIL (seg(answer,1,1)="n") OR (seg(answer,1,1)="N")
END.

```


Erforderlig utrustning

32K expansionsminne. Minst en disk-drive. Editor/Assembler. **Funnelweb** kan köra såväl editor/kompilator som länkare, men inte de länkade programmen.

Tillbehör

Som tillbehör kan man på diskett köpa biblioteksfiler som möjliggör grafik- (Graphics Toolkit) och fönsterhantering (Windows Manager). Dessa disketter kostar 99 DM vardera inkl bruksanvisning. Dessutom finns en utökad användarhandbok (59 DM) om 300 sidor som skall innehålla det som jag ovan efterlyste i det medföljande häftet. Dessa tillbehör har jag inte provat.

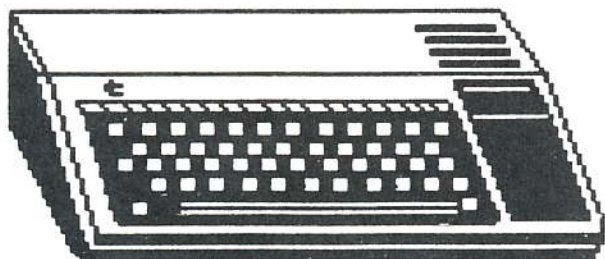
Sammanfattning

Enligt mina prov blir Pascal-programmen rent av snabbare än motsvarande program skrivna i C99 (vilket hittills varit det snabbaste högnivåspråket till 99:an). Det existerande flyttalsbiblioteket i C är ju begränsat till de fyra räkneseiten (medan Pascal har fullständig flyttalshantering), varför jämförelse får ske på heltalsbasis. Denna fördel för Pascal-kompilatorn kan dock i många situationer, i vart fall för den som kan assembler, i C uppvägas av den enkla möjligheten att skriva tidskritiska avsnitt i assembler. Detta går inte i Pascal.

TURBO-PASC'99 är en utomordentligt lättarbetad kompilator. Jag antar att den blir ett utmärkt hjälpmedel för den som vill övergå till ett snabbt och strukturerat högnivåspråk från BASIC, och inte kan och inte heller vill lära sig assembler. Den inneboende styrkan i Pascal som språk är att man enkelt kan koda och beräkna komplicerade matematiska uttryck, t ex i tekniska eller fysiska sammanhang.

Programexempel

Programexemplet på föregående sida är hämtat från en artikel av David A Mundie i "The BYTE book of Pascal", 1979. Små ändringar har gjorts från UCSD Pascal till TURBO-PASC'99. Programmet gör inte så mycket nytta om man inte vill gå ned i vikt. Dags att lära Pascal, Claes?



Annonser

Säljes:

Byggsats till Horizon RAM-disk

Ingvar Andréasson

☎ 040/19 35 26 efter 19.00

TI Disk-kontrollkort

2 (ev 3) diskdrivar

Sören Bernle

☎ 042/933 05

P-kodkort med programvara och manualer

Mikael Jonsson

☎ 046/11 97 00 efter 19.00

Ordföranden ...

Du medlem skriv och berätta vad du använder din TI-59 eller TI-99 till. Vi i styrelsen har lättare att jobba om vi vet att någon läser det som står i tidningen. Speciellt bör du som inte hittar det du önskar i tidningen höra av dig. PROGRAMBITEN är en medlemstidning och detta betyder att det är medlemmarna som skall skriva i tidningen med tips och program som andra har nytta av. Det är inte meningen att styrelsemedlemmar och våra fasta skribenter skall skriva allt. Vi behöver artiklar om allt men även korta notiser som inte behöver vara långa eller märkvärdiga.

Finns det någon medlem som åtar sig ett begränsat område och kan skriva regelbundet om något av följande:

TI-74

TI-95

TI-99/4A BASIC

Mini Memory

Musik på TI-99/4A

Speech

Assembler

GRAM

Modem

Byggen av egen hårdvara

Test och rescension av ny FREEWARE

Finns det någon medlem som har TI-74? Hör av dig och berätta vad du använder den till och vad du vill läsa i PROGRAMBITEN om den.

Vi har fått tag i ett mycket stort antal FREEWARE till TI-99/4A. Dessa program är ofta mycket avancerade och minst lika bra som kommersiella program. Detta betyder att det tar en viss tid innan man satt sig in i alla finesser. Vi i styrelsen har inte tid eller intresse att gå igenom och skriva om alla program. Jag vill dock säga att alla är mycket bra och ingen som beställer ett sådant program bör bli besviken. Så gott som alla program är skrivna i Assembler även om Extended Basic ibland används som styrprogram.

Jag uppmanar dig dock att sända en slant (USD 5-10) till författarna till till dessa FREEWARE. Speciellt tack till följande författare som har varit mycket vänliga och hjälpsamma utöver vad man normalt kan begära:

- Tony och Will McGovern (Funnelweb, Colist och Disk Hacker)

- R. A. Green (Macroassembler och Linker)

- Edgar Dohmann (Superbug II)

- Clint Pulley (C99)

- Ken Gilliland (Startrek, Wagner, South Pacific, Patsy Cline, Music Man och Wizard of Oz)

En genomgång av beställningar av FREEWARE under första halvåret 1987 visar följande popularitetslista:

1. Funnelweb
2. PR-BASE
3. Rapid Scroll
4. C99
5. TK-Writer
6. TI-FORTH

Speciellt tråkigt är att ingen har beställt SUPERBUG II som är ett mycket bra program. Vi har nu manualen till version 1.0 som du får tillsammans med skivan till version 2.1.

Disk Manager 1000 och Funnelweb är de FREEWARE du i första hand bör beställa. Ingen ägare till TI-99/4A med expansionsbox bör vara utan dessa program.

Jan Alexandersson

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

Resetknapp till 99:an

av Anders Törnkvist

1 Allmänt om reset av datorn

Reset av datorn kan på TI 99 göras på 2 sätt:

- Tryck FCTN QUIT. Detta fungerar dock enbart om datorn reagerar på tangenttryckningar
- slå av spänningen och slå på igen. Det fungerar ju alltid, men...

Att slå av och på datorn många gånger är inte bra, alla delar slits av det och risken att datorn går sönder ökar kraftigt. Lösningen är att bygga in en RESET-knapp i datorn, så här kan du göra det.

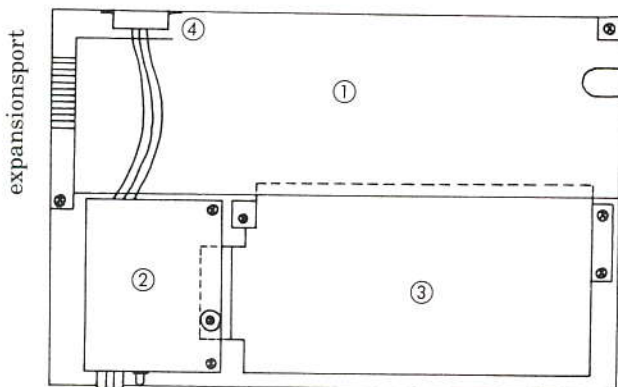
2 Material som du behöver

- 1 Återfjädrande tryckknapp
- 1 Motstånd på ca 100 ohm
- ca 0,5 m två-ledares el-tråd tex högtalarkabel
- Isoler-tejp eller krympslang
- Lödtenn

3 Verktyg som du behöver

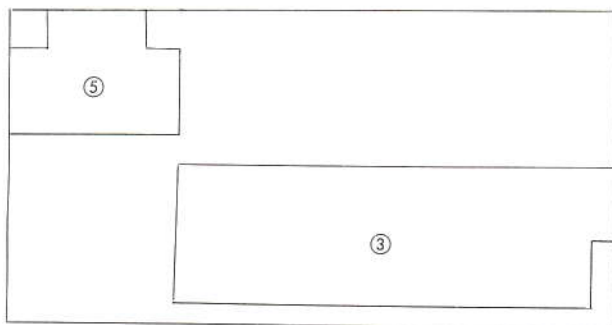
- Kryss-spårs-mejsel
- Liten kniv eller skalpell
- Ohm-meter
- Lödkolv
- Liten tång eller pincett
- Avbitartång

4 Figurer



Figur 1 Datorn med botten avtagen

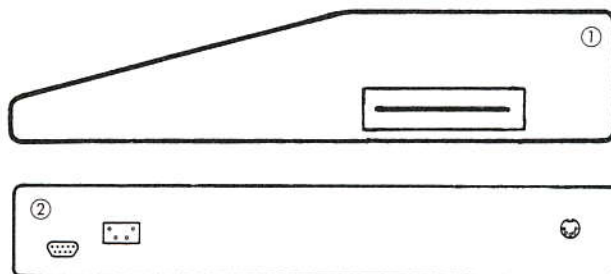
- 1 CPU-del (kretskort i plåtlåda) (3 skruvar)
- 2 Kraft-kort (2 skruvar)
- 3 Tangentbords-kort (4 skruvar)
- 4 Kraft-kontakt (ligger i slitsar i höljet)



Figur 2 Datorn med CPU och kraft borttagna

- 3 Tangentbords-kort (4 skruvar)
- 5 Botten till modul-facket (2 skruvar)

Figur 3 Lämpliga platser för RESET-knapp

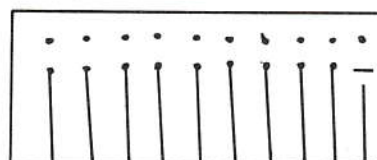


① höger sida bakre övre hörnet

② baktill vänstra övre hörnet

På plats två passar en rund kontakt sämre pga spjälorna

Figur 4 MODUL-kontakt sedd bakifrån



här löds andra
tråden på

kapas

Observera att kortet sticks ned i en kontakt, därför måste trådarna lödas ganska högt upp.

5 Isär-tagning av konsolen

- Tag bort eventuell modul och lossa alla anslutningar.
- Tag bort strömbrytaren (haka i bakom med något och dra framåt).
- Vänd konsolen uppochner och skruva av botten (7 skruvar).
- Lossa kraftkortet (2 skruvar) (se fig 1), dela på sladden till CPU-delen (det finns en slits i kontakten där man kan få tag).
- Lossa skruvarna som håller CPU-delen (3 stycken) och lyft i bakre kanten.
- Lossa kontakten mellan CPU-delen och tangentbords-kortet.
- Lossa botten till modul-facket (2 eller 3 skruvar) (se fig 2).

6 Montering av tryckknappen

- Ta upp ett hål för knappen på lämpligt ställe (se fig 3).
- Kolla att knappen passar (det kan bli knepigt senare).
- Kolla med ohm-metern vilka stift på kontakten som ligger ihop då den är tryckt men ej då den är släppt.
- Korta av benen på motståndet till ca 10 mm och löd fast det på ena stiftet (längre ben kanske inte får plats).
- Löd fast trådarna i ena ändan av el-sladden på motståndet resp andra stiftet
- Kolla kretsen genom att mäta i andra ändan av sladden, 100 ohm om knappen är intryckt, oändligt annars.
- OBS!** Sladden bör vara ganska lång, då blir det lättare att sätta ihop och ta isär konsolen senare.
- Sätt fast knappen i hålet
- Ta loss kontakten till modul-uttaget genom att dra uppåt.
- Skär av ledaren längst till vänster på baksidan så nära stiftet som möjligt.
- Löd fast tråden till motståndet under kapningen och den andra tråden på fjärde stiftet från höger i undre raden (löd så nära kapning resp stift som möjligt och gör lödningarna små annars går det ej att sätta tillbaka kontakten)

7 Rengör alla kontakter med sprit eller kontaktspray

8 Sätt ihop konsolen

- Sätt fast botten på modulfacket
- Sätt fast modul-kontakten på CPU-delen
- Sätt ihop kontakten mellan CPU-delen och tangentbordskortet
- Lägg ned CPU-delen medan du stoppar in sladden till RESET-knappen i utrymmet vid knappen (sladden får inte komma in i modul-facket)
- Skruva fast CPU-delen
- Sätt ihop kontakten mellan kraft-kortet och CPU-delen.
- Sätt fast kraft-kortet (sladden till el-kontakten ska ligga under kortet)
- Kolla att strömbrytarkontakten kommit rätt.
- Sätt tillbaks botten och strömbrytaren, sätt i alla anslutningar

9 Prova

Slå på datorn och ta fram huvudmenyn.

Tryck på RESET-knappen så ska startbilden komma upp igen.

Stoppa i och ta ut en modul några gånger, endast någon enda gång ska en reset ske, den beror i så fall på att programmet har sparat ur (ibland kan man byta modul utan att programmet stannar, man måste vara bestämd).

Varning!

Ett byggprojekt som detta bör Du bara genomföra om Du har tidigare erfarenhet av elektronikbyggsatser el dyl. Vi kan inte ta något ansvar för fel som uppstår på grund av ev felaktigheter i denna beskrivning eller missförstånd.

Pressgrannar

RD Computing är en kanadensisk tidning som ges ut varje månad. Utgivaren RYTE Data är egentligen en postorderfirma som specialiserat sig på att sälja program och tillbehör till 99:an. Tyngdpunkten i innehållet ligger på hårdvara. Tidningen innehåller ofta byggprojekt, både för konsol och expansionsbox. För den som är intresserad av sånt är nog RD Computing den bästa källa som finns för närvarande.

Tidningen brukar vara på 20 sidor. En prenumeration kostar 21 USA-dollar för 12 nummer med flygpost.

Adressen är:

R/D Computing
Ryte Data
210 Mountain Street
Haliburton
Ontario
K0M 1S0
Canada

TI-95 nytt

Det har börjat röra sig en del kring TI:s nya programmerbara räknare TI-95 Procalc. TI har själva börjat ge ut ett nyhetsblad kallat *Programmable Calculator News*. Den handlar om TI-74 (se PB 87-1) och TI-95. Det första numret presenterar 95:an och handlar en del om BASIC-tolken på 74:an. Det nämns bl a att den har likheter med den BASIC som finns på TI:s minidatorer i 990-serien (att den är närmast identisk med Extended BASIC på 99-an nämns inte). Den som äger en 74:a eller 95:a och vill ha bladet hemskickad i fortsättningen kan skriva en rad till:

The Editor
Programmable Calculator News
P. O. Box 53
Lubbock
TX 79408
USA



En annan tidning om TI:s programmerbara räknare, *TI PPC Notes*, är nu inne på 12:e årgången. I första numret skriver man om framgångsrika försök att överföra program mellan två olika 95:or genom att skicka kassetter eller ROM-klotsar. Enligt artikeln fungerar det mycket bättre än på gamla 59:an.

Mest intressant för den som vill veta hur 95:an fungerar är ett brev som kom från *Robert Prins* i Holland strax innan tidningen gick till tryck. Han beskriver hur man kommer åt en meny med självtester till 95:an, något som inte finns beskrivet i manualen. Han skriver också om hur man dumpar ut delar av minnet från ROM till RAM. En översättning av Prins artikel hoppas vi ha klar till nästa nummer av *Programbiten*.

Enligt Prins har TI i Holland utlovat ett serie-interface till slutet av oktober.

Digitalservice har flyttat

Digitalservice, firman som specialiserat sig på att sälja och serva räknare och portföljdatorer från Texas Instruments, har flyttat. De återfinns nu på Ringvägen 151 i Stockholm. Det är bara något kvarter från deras gamla lokaler, men de nya ligger mycket bättre. Telefon är samma som tidigare, 41 60 52.

Rättelse

Sören Bernles program i förra numret av PB (sid 22—24) blev tyvärr monterat i felaktig ordning, och därmed helt galet. Så här ska det vara: Det stycke på sidan 23 som börjar efter kommentaren * PERFORMS THE I/O ... med raden JNE LOOP och hela sidan ner ska skjutas in efter raden DEC R3 Has all been printed out? ovanför labeln LOOP på sidan 22.

Jag ber om ursäkt för det missförstånd som denna klantiga montering kan ha orsakat.

— Redaktören

\$2.00

VOLUME 2 - 1987
VER 20
FEBRUARY

R/D COMPUTING
Dedicated to TI 99/4A and 9900 Computer Systems

Ryte
Data.....

210 MOUNTAIN STREET,
HALIBURTON, ONTARIO K0M 1S0
(705) 457-2774

RAM-disk

av Jan Alexandersson

1. Huvudfunktioner

En RAM-disk består av ett stort RAM-minne som placeras på ett kort i expansionsboxen eller direkt till datorn i högra kanten. Det adresseras på samma sätt som en vanlig disk drive. RAM-disken kan användas för att lagra PROGRAM-filer eller DATA-filer men kan även lagra FORTH-skärmar eller adresseras med en sektoreditor som tex *Disk-Patch* eller *Advanced Diagnostics*.

2. Horizon 192 kb RAM-disk

Horizon RAM-disk är ett öppet kort till expansionsboxen. Det har 192 kb CMOS-RAM uppdelat på 24 st 6264LP-15 kapslar. Det finns även en mindre version med 104 kb RAM. Kortet är försett med laddningsbara batterier som laddas när expansionsboxen är på. Detta gör att operativsystem och filer finns kvar även nästa gång du sätter på datorn. Jag köpte 192 kb-versionen färdigbyggd från USA. Den kostade USD 215 inklusive flygfrakt. Med kortet kommer 3 fulla flexskivor och en 40 sidor tjock manual.

3. CRU-adresser

Innan man börjar använda ett expansionskort av detta slag så måste man ställa in CRU-adressen. Det är viktigt att alla kort du använder har olika adresser. Här följer en lista över av mig kända kort:

	HEX
Mechatronic 80 kolumners kort	1000
Myarc RAM-disk	1000
Disk Controller (TI, Corcomp, Myarc)	1100
RS232/PIO nr 1	1300
Mechatronic 128k+Centronics	1400
RS232/PIO nr 2	1500
TI Thermal Printer	1800
Triple Tech	1D00
Foundation RAM-disk	1E00
P-code kort för Pascal	1F00
GRAM-karte	1000-1F00 (16)
Horizon RAM-disk	1000-1700 (8)
Corcomp RAM-disk	1000 och 1400 (2)

4. Operativsystemet

Hela operativsystemet finns på disk och laddas in från skivan till RAM-disken. Även källkoden till detta operativsystem finns på skivorna. Det finns således möjlighet att ändra för den som kan och vill. Operativsystemet tar ca 8kb.

5. Disk manager 1000

Horizon använder DM 1000 som följer med på en av skivorna. Denna disk manager används normalt för att initiera RAM-disken men det går även bra att använda DM 2.

6. Nya CALL från BASIC

RAM-disken har nio nya CALL-rutiner som kan anropas från kommandomod i BASIC och Extended Basic och även i ett körbart program i BASIC (ej XB):

- CALL DN ändrar RAM-disk till valfritt nummer DSK1-DSK6
- CALL MS ändrar maximala antalet sektorer som normalt är 360 eller 720
- CALL WO skrivskyddar RAM-disken vilket motsvarar att tejpa över slitsen på en vanlig disk
- CALL WF tar bort skrivskyddet
- CALL EX startar masinkodsprogram på angiven adress
- CALL CO aktiverar RAM-disken
- CALL CF stänger av RAM-disken

- CALL DM laddar in och startar DM 1000
- CALL NF bestämmer antalet fysiska diskdrivar.

7. XBCALL

Det är inte möjligt för ett Extended Basic program att hitta CALL-rutiner på kort i expansionsboxen men den letar efter DSR-rutiner i boxen. Den enda möjligheten är att anropa dessa CALL-rutiner via CALL LINK. Genom att skriva DELETE "XBCALL" kommer dessa rutiner att laddas in till det låga 8 kb minnet. DM 1000 kan sedan anropas med CALL LINK("DM") i ett XB-program.

8. Skriva egna CALL-rutiner

Det finns 4 kb minne reserverat för annat än operativsystemet. Där finns bla de inbyggda CALL-rutinerna. Det finns dock ledigt minnesutrymme för att lägga till egna rutiner. Du kan skriva egna CALL som kan anropas från BASIC eller Extended Basic. I manualen finns ett komplett exempel med källkod.

9. SAVE, OLD, OPEN och CLOSE

RAM-disken kan anropas för att spara och ladda in BASIC-program med OLD och SAVE. Datafiler kan öppnas och stängas på normalt sätt med OPEN och CLOSE. RAM-disken har det nummer du själv har valt mellan 1 och 6. Du kan alltid ändra numret med CALL DN.

10. Editor/Assembler och TI-Writer

Om du lägger hjälpfilerna EDIT1, ASSM1-2, EDITA1-2 och FORMA1-2 på RAM-disken kan dessa laddas in mycket snabbt. Det tar mindre än 1 sekund med RAM-disk och ca 10 sekunder med vanlig diskdrive. Om du har EA- eller TIW-modulen måste RAM-disken döpas till "DSK1." medan du kan välja att kalla den tex "DSK3." om du har GRAM (GRAM Kracker, Maximem eller GRAM-karte). Även om du döper RAM-disken till "DSK1." kan du använda en skiva i den fysiska driven nr 1 om du anropar den med skivnamn tex DSK.SKIVNAMN.FILNAMN.

11. FORTH

Forth kan användas på normalt sätt med RAM-disk men det är nog lämpligt att döpa om RAM-disken till "DSK1." innan du använder FORTH.

12. Pascal mm

Enligt uppgifter i manualen kan PASCAL med P-CODE kortet ej användas tillsammans med RAM-disken. Vissa andra program med speciell användning av disken tex tidiga versioner av *TI-Artist* fungerar inte med RAM-disk.

13. Disponering av minnet

Horizon RAM-disk har adresser avsedda för DSR-rutiner mellan >4000->5FFF. Kortet kopplas in till dessa adresser genom att CRU-biten för kortet sätts till 1. Det är därför viktigt att alla kort har olika CRU-adresser. De första 6 kb av detta (>4000->57FF) används av operativsystemet. De resterande 2 kb (>5800->5FFF) bank-switchas mellan 93 olika bankar kallade RACK (RAM TRACK) numrerade 0-92.

RACK 0-89 innehåller de 720 sektorerna på vardera 256 bytes. Eftersom en RACK har 2 kb kommer det att rymmas 8 sektorer på en RACK. RACK 0 har således sektor 0-7. RACK 90 har de resterande 2 kb till operativsystemet. RACK 91 innehåller CALL-rutiner och RACK 92 innehåller ett komplett eget teckenset för DM 1000 och DELETE "XBCALL" med de rutiner som anropas med CALL LINK från Extended Basic.

14. Buggar

RAM-disken har fungerat mycket bra utan några som helst problem med batteri och minne. Jag har ännu ej behövt initiera RAM-disken på nytt. Om du trixar med operativsystemet så att det kraschar är det fullt möjligt att ladda in det på nytt utan att initiera sektorerna. Alla filer finns kvar även om du laddar in operativsystemet på nytt.

Det finns dock en liten bug i CALL LINK("DM") som laddas med DELETE "XBCALL". Detta kommando fungerar inte eftersom källkoden modifierats vid olika tillfällen. Källkoden i PARTD skall ändras så att INPT får EQU >4638. I stället för att ändra i RAM-disken kan anropet göras så här:

```
100 CALL INIT
110 DELETE "XBCALL"
120 CALL LOAD(10007, 56, " ", 10055, 56)
130 CALL LINK("DM")
```

Det finns ytterligare ett litet problem med CALL DM eller CALL LINK("DM") om du har reserverat en stor del av övre delen av VDP-minnet med tex CALL FILES(7). DM 1000 kommer att laddas in men när du anropar tex DSK1 kommer datorn att låsa sig med lysdioden tänd på diskkontrollkortet. Samma problem fås med CALL P(2007) om du har Personal Record Keeping och CALL APESoft med Mechatronic Extended Basic II PLUS.

15. Bygga själv

Du kan även bygga kortet själv genom att köpa ett tomt foliekort med flexskivor och manual för USD 53. Vid beställning av fem kort samtidigt är priset USD 45 men då måste du själv kopiera manualen och skivorna. Peter Odelryd och fyra andra medlemmar i föreningen har byggt egna kort. Du bör ta kontakt med någon av dessa innan du beställer byggsats på egen hand. Det är nog enklare att bygga ett kort med 104 kb eftersom endast 13 platser finns på kortet för RAM-chip. De resterande 11 kretsarna monteras ovanpå de först monterade så att benen löds ihop.

16. Foundation, Myarc och Corcomp

De övriga RAM-diskar som finns att köpa idag skiljer sig från Horizon på två punkter. De är ej försedda med inbyggda batterier på kortet utan de måste normalt initieras på nytt varje gång du sätter på datorn. Vissa kort kan förses med en yttre strömkälla som ansluts till nätet. Den andra skillnaden är att de övriga korten även innehåller det normala expansionsminnet 32 kb. Detta innebär att du bara kan ha ett sådant kort eftersom det bara får finnas ett minne på dessa adresser. Dessutom kan de när de ej används som RAM-disk switcha 32k i flera bankar vilket gör att man kan ha flera assemblerprogram som man hoppar till och från. Detta switchande av 32 kb-bankar görs för några av dessa kort med interrupt-rutiner. Det finns då stora risker att ett avancerat assemblerprogram av misstag switchar bort sig själv. Om du har ett sådant kort skriv gärna och berätta hur bank-switchningen går till och om det finns problem med några program.

Foundation 128k och **Mechatronic** fristående 128k + Centronics är vad jag kan förstå ej några äkta RAM-diskar utan kan lagra ett begränsat antal filer på ett primitivt sätt.

Myarc 128k/256/512k kan användas för upp till 400 kb (1600 sektorer) som RAM-disk. Det resterande minnesutrymmet används som printbuffert (print spooler) eller som RAM-minne för Myarc Extended Basic II som tar 128 kb av minnet.

Corcomp 256k eller 512k kan använda hela 480 kb som RAM-disk. Eftersom TI endast har lämnat plats i skivhuvudet för 1600 sektorer används tre sektorer för skivhuvudet. Detta gör att du endast kan använda Corcomps egen disk manager.



17. Många samtidiga RAM-diskar

Med Horizon kan du samtidigt använda flera RAM-diskar så länge det finns lediga kortplatser och CRU-adresser.

18. Senaste nytt från Tyskland

Elektronik-Service i Västtyskland har kommit med en ombyggd version av Horizon med 392 kb RAM och 1440 sektorer. Mer om detta i kommande nummer av tidningen.

19. PROM från DOS

Disk Only Software i USA säljer en EPROM till Horizon RAM-disk för USD 25. Den klarar av max 256 kb RAM. RAM-disken kan adresseras som DSK1-DSK6 eller RD. Det finns ett antal nya CALL och möjlighet att fixa saker vid powerup. Denna EPROM frigör 14 sektorer. Enligt rek-lambroschyren är operativsystemet baserat på Myarc RAM Disk vad det nu skall betyda.

Referenser:

1. *Programbiten* 86-4.13: Bygg din egen RAM-DISK
2. *Programbiten* 85-4.09: Foundation 128 k
3. *Micropendium* juli 86: Horizon 192k review
4. *Micropendium* sept 85: Myarc 128k review
5. *Micropendium* aug 86: FORTH and Myarc 128k
6. *Micropendium* okt 86: Corcomp 512k review
7. *Computer Kontakt* 6/86: Mechatronic 128k + Centronics
8. *Micropendium* jan 85: Foundation 128k review
9. *Programbiten* 86-2.04: Att hitta på en skiva

