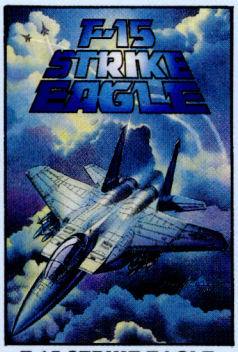
Dansk
RUN
Commodore-magasin
2
Nov./Dec. 84
Kr. 19,85

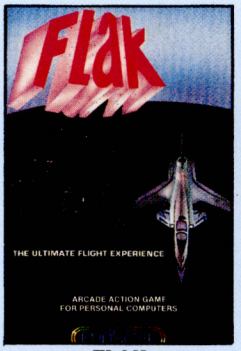
NATO COMMANDER
CBM Kass: **198,-** disk: **278,-**



F-15 STRIKE EAGLE
SP: **198,-** CBM Kass: **298,-**
disk: **298,-**

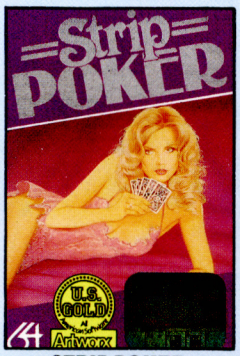
SOLO FLIGHT
SP: **198,-** CBM Kass: **298,-**
disk: **298,-**

SPITFIRE ACE
CBM Kass: **198,-** disk: **278,-**

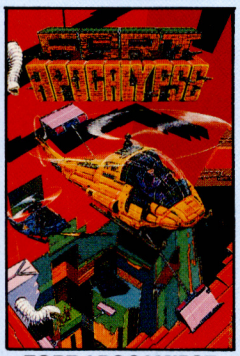


FLAK
CBM Kass: **198,-** disk: **298,-**

SNOKIE
CBM Kass: **198,-** disk: **278,-**

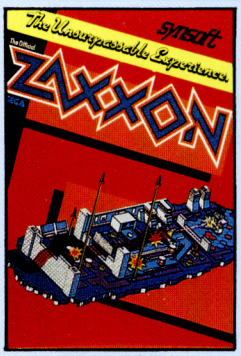


STRIP POKER
SP: **175,-** CBM Kass: **198,-**
disk: **278,-**

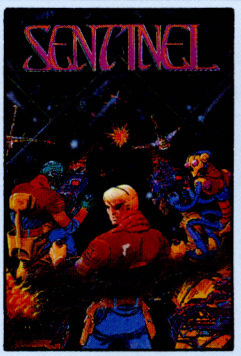


FORT APOCALYPSE
SP: **175,-** CBM Kass: **198,-**
disk: **278,-**

BLUE MAX
SP: **175,-** CBM Kass: **198,-**
disk: **278,-**



ZAXXON
SP: **175,-** CBM Kass: **198,-**
disk: **278,-**



SENTINEL
CBM Kass: **198,-** disk: **278,-**

DRELBS
CBM Kass: **198,-** disk: **278,-**

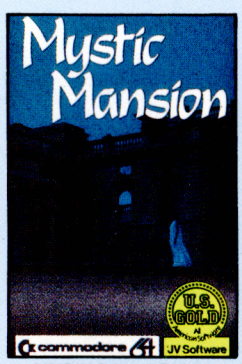
TWILIGHT APS.

Flintholm Alle 26, 2000 København F.
Tlf.:01-880734

GOLD U.S. GOLD All American Software RUSH!

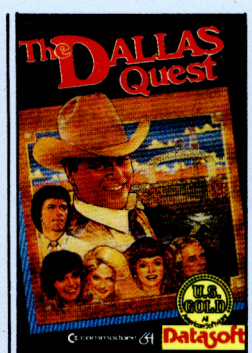
NB: Kun salg til videreforhandler.

DET BEDSTE SOFTWARE FRA DANMARKS BEDSTE DISTRIBUTØR!

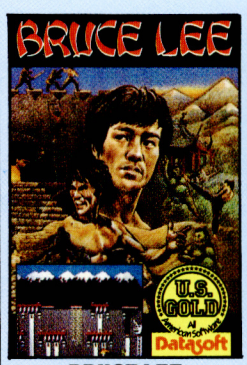


MYSTIC MANSION
CBM Kass: **198,-** disk: **278,-**

BEACH-HEAD
SP: **175,-** CBM Kass: **198,-**
disk: **278,-**

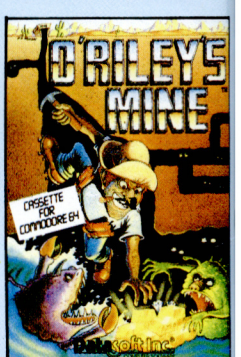


DALLAS
CBM disk: **298,-**

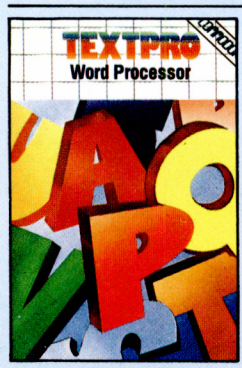


BRUCE LEE
SP: **175,-** CBM Kass: **198,-**
disk: **298,-**

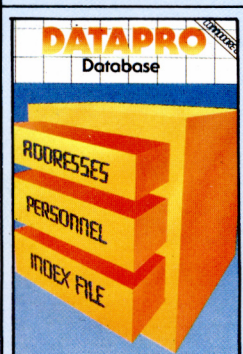
POOYAN
CBM Kass: **198,-** disk: **278,-**



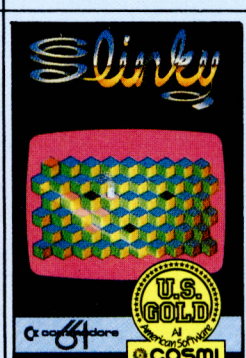
O'RILEY'S MINE
CBM Kass: **198,-** disk: **278,-**



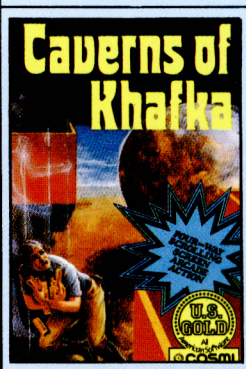
TEXTPRO
CBM Kass: **198,-** disk: **278,-**



DATAPRO
CBM Kass: **198,-** disk: **278,-**

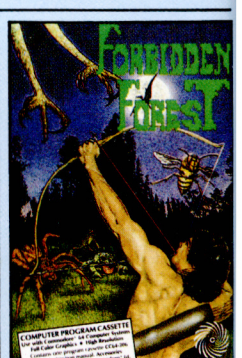


SLINKY
CBM Kass: **186,-** disk: **278,-**



CAVERNS OF KHAFKA
CBM Kass: **186,-** disk: **278,-**

AZTEC CHALLENGE
CBM Kass: **186,-** disk: **278,-**



FORBIDDEN FOREST
CBM Kass: **186,-** disk: **278,-**

RUN

NR. 2 - 1984

Artikler

Teledata
En praktisk udnyttelse af Commodore 64 10

De brændes uden varme
Hvad er EPROM egentlig 16

Messerapporter
Rapport fra Frankfurt og Herning 22

PASCAL
et anderledes programmeringssprog 49

Anmeldelser og tests

LOGO
Gennemgang af Commodores LOGO 6

Programanmeldelser 17

PLUS/4 og C16
Jim Butterfield anmelder for RUN 23

COMAL 80
Et godt programmeringssprog til C64 32

Spreadsheet
Stor gennemgang af populære programmer 38

PETSPEED
Compileren der speeder BASIC op 40

Boganmeldelser 58

Tips, tricks og vejledning

Læserbreve og elementære spørgsmål 8

VIC-20 routiner 30

Sekventielle filer – del 2 31

MERGE
En nyttig routine til VIC-20 og C64 41

F-tasterne
Hvordan benyttes de 50

Programmer

Tipsprogram 11

Simons Basic
Flere hjælperoutiner 12

Disk og drømme
Hjælpeprogram til diskettestationen 26

RUN Amok
Rettelser og korrektioner 64

Indtastningsvejledning og fortegnelse over alle programmer 64

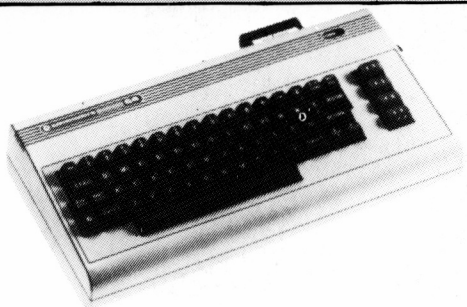
Udgiver: Computerworld Danmark A/S. Ansvarshavende redaktør: Jørgen Jørgensen. Fagredaktion: Robert Ch. Noya, Flemming Lerbæk, Robin Sagar, Axel Bang, Steen Schmeltzer. Direktion: Preben Engell (adm. direktør). Annoncechef: Leif Rasmussen. Marketing: René Koefod. Bladsekretær: Grith Axel. Abonnement: Dorthe Christensen. Telefon: (01) 12 34 11. Telex: 37 566 CWDAN. Distribution: Dansk Centralagentur. Sats/tryk: J. H. Schultz A/S, København. Oplag: 22.000. RUN udkommer 2 gange i 1984 og 6 gange i 1985.

RUN er et medlem under CW-Communications Inc., verdens største udgiver af dataorienteret information. Gruppen udgiver 52 computer-publikationer i 21 industrilande. 9 millioner læser én eller flere af gruppens publikationer hver måned. Medlemmerne af gruppen er:
Argentina: Computerworld/Argentina. Australien: Computerworld, Australian Micro Computer Magazine, Australian PC World and Directories. Brasilien: DataNews, Micro Mundo. Canada: Computerworld Canada, PC World Canada. Danmark: Computerworld Danmark, Micro Verden, Buyer's Guide, RUN. England: Computer News, Computer Management, Computer Business Europe. Finland: Mikro. Frankrig: Le Monde Informatique, Golden (Apple), OPC (IBM. Holland: Computerworld Benelux, Micro/Info. Indien: Dataquest. Italien: Computerworld Italia. Japan: Computerworld Japan, PersoCom World. Kina: China Computerworld. Mexico: Computerworld/Mexico, Compu/Mundo. Norge: Computerworld Norge, Mikrodata/PC. MikroData. Saudi Arabien: Saudi Computerworld. Singapore: The Asian Computerworld. Spanien: Computerworld Espana, MicroSistemas, Commodore World. Sverige: ComputerSweden, MikroDatorn, Person Datorn, PC World. Tyskland: ComputerWoche, MicroComputerWelt, PC Welt, Software Markt, CW Edition/Seminar, Computer Business, RUN. USA: Computerworld, Computerworld on Communications, Hot CoCo, In-Cider, InfoWorld, jr., MacWorld, MICRO MARKETWORLD, Microcomputing, PC World, PC Jr. World, RUN, 73 Magazine, 80-Micro.

RUN
Commodore-magasin
Gammel Strand 50 - 1202 København K
Tlf. 01 - 12 34 11

Administrative opgaver på Commodore 64

Forhandler søges



PROGRAMSPECIFIKATION TIL MCH-FINANS-BOGHOLDERI

1200 posteringer
300 konti
Resultatopgørelse & balance
Årsafslutning
Transaktionsjournal
Rettelser af poster
Periodeafslutning
Automatisk momspostering
Udskrift af kontogrupper og enkelt konto
Udskrift af debitorer og kreditorer
Transjournal med grand total
Pris for MCH-finansbogholderi

Kun kr. **2.095,-**

(Det billigste og bedste Commodore 64, nu 40 % hurtigere).

NYT · NYT · NYT · NYT

GOLDEN TOOLS

SM KIT 64

Programmeringshjælp til Basic og Assemblerprogrammering som **Merge, Find, Renumber, Trace, Dump, Assembler og Disassembler Floppy Monitor** burde være hjælpeprogrammet til alle dem, der programmerer på Commodore 64. Uden programmet spilder du meget tid - alt for meget.

Kr. **995,-**

ISM 64

Indexsekentiel Database til Commodore 64. Op til 40 nøgler, variabel tekstlængde og 10 ISM kartoteker til at hente alle mulige relevante data. Kr. . . . **995,-**

FAKSYS VERSION 1.0

Commodore 64 faktureringsystem. Systemet er integreret (kører sammen) med COMFINANS systemet fra MCH-DATA. FAKSYS er et enkelt og let brugbart system til mindre og mellemstore virksomheder, der ikke ønsker at hoppe på EDB-toget med en investering på 30.000 kr. eller mere.

Systemkapacitet:
- 250 kunder
- 250 varer
- 750 posteringer pr. periode
- Udskrift af debitorer
- Faktura
- Kreditnota
- Posteringsjournal
- Debitorstatus
- Kontoudtog m/renter
- Prisliste
- Labels
- Salgsstatistik
- Integreret med COMFINANS
- Med eller uden fortrykt blanket
- Renteminimum

Minimum konfiguration:
- Commodore 64
- VIC 1541 diskettestation
- Printer MPS 801 (forslag)
- Fjernsyn eller monitor

Systemet leveres på diskette med en udførlig dansk brugervejledning.

Pris incl. moms . . . **2.595,-**

Med systemet følger 12 mdr.'s gratis opdatering, såfremt nye versioner af systemet fremkommer gennem forbedringer.
FAKSYS er klar til levering primo november.

CBM 64 + diskettestation 1541 + MPS 801 og
Comfinans og Faksys
Spar 2.000,- . . kr. **13.985,-**

Comfinans alene
Spar 1.400,- **11.985,-**

Faksys alene
Spar 1.600,- **12.285,-**

Vi kan bare li' at være de bedste

MCH

80 tegn/grafikkort til CBM 64 med digitalt ur, tekstbehandling og kalkulationsprogram. Mulighed for blandet grafik og tekst i farve, med fastfrysning af øverste linie og indstilbar linieafstand.
Pris kr. **1.995,-**

TILBUD PÅ **3M** DISKETTER FRA SCOTCH

Single side single density
v/1 stk. **36,95** v/10 stk. **32,95**
Single side double density
v/1 stk. **45,00** v/10 stk. **39,95**
Double side single density
v/1 stk. **45,00** v/10 stk. **39,95**
Double side double density
v/1 stk. **51,95** v/10 stk. **45,95**

MANNESMANN TALLY PRINTER MT80

Den bedste af sin klasse (pris sammenholdt med ydelse), 80 tegn i sekundet, logisk søgning, virkelig god til grafisk fremstilling, bruger firkantede nåle istedet for runde nåle, derfor velegnet til tekstbehandling.

Normalpris centronics parallel kr. 7.195,-

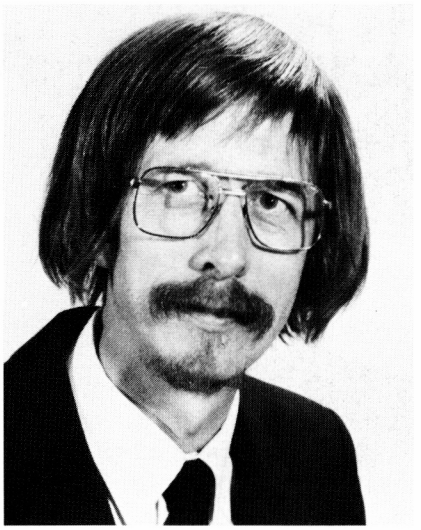
Vor pris lige nu kr. . 5.195,-

Normalpris RS232C kr. 8.695,-

Vor pris lige nu kr. . 6.695,-

MØLLEPLADSEN 3 • DK-6100 HADERSLEV • TELEFON 04-53 17 71

Den svære start



Al begyndelse er svær. Det er svært at lære sin computer at kende og det er svært at lave et blad, som alle er tilfredse med. De fleste læsere har været virkelig tilfredse, men enkelte synes, vi ikke er kritiske nok. Sandheden er den, at vi ikke vil bruge vores kostbare spalteplads til at rakke produkter ned. Vi vil i stedet informere om, hvad produkterne kan og hvorledes de fungerer. Hvis et produkt kan det, der loves, må læserne ud fra en saglig gennemgang selv vurdere, om det pågældende produkt har interesse eller ej. Vi vil ikke sætte os til dommere over, hvad folk skal købe.

Vi havde håbet at kunne præsentere flere af Commodores nyheder i dette nummer, men der har desværre været en del forsinkelser, så vi må alle vente i spænding lidt endnu. Om forholdet til Commodore kan i øvrigt siges, at vi har et positivt samarbejde, uden hvilket et blad som RUN ikke kan være »up to date«. Omvendt er Commodore naturligvis også interesseret i RUN, men vi kan ikke kraftigt nok pointere, at forholdet er helt uafhængigt, men hviler på en gensidig respekt.

Vores printprogram har spillet os et puds. Vi vil i programlistningerne konvertere alle de specielle grafiske tegn til et mere forståeligt sprog. Det lykkedes ganske vist også, men programmet har ikke taget højde for alt. Der har derfor været et par fejl i nogle af vore programmer. Det har sikkert givet et par søvnløse nætter rundt omkring, men det har givet os en god kontakt med vore læsere, som har givet os mange tip til emner eller programmer, de kunne tænke sig i RUN. Vi siger tak for de mange gode ideer og tak for de mange pæne ord vi har fået med på vejen.

Vi har fået mange læserbreve – flere end vi kan besvare her i første omgang. Vi beder derfor indsenderne have tålmodighed, medens vi graver os gennem bunken. Vi har også fået programmer fra læserne, men vi er interesseret i mange flere. Derimod har brugerklubberne svigtet os i første omgang. Flere læsere vil gerne i kontakt med andre brugere, og derfor ville det være rart, om de forskellige klubber ville komme frem fra busken, så læserne kunne finde adresserne her i bladet.

Alt i alt har tiden siden første nummer gjort os mere overbevist end nogensinde om, at der er et stort behov for et blad som RUN.

Jørgen Jørgensen
Redaktør

LOGO - mere end Turtle Grafik

Af Robin Sagar

Mange Commodore-ejere håber sikkert, at deres maskine kan anvendes i uddannelsesmæssig sammenhæng, men udover muligheden for at lære Basic og måske Assembler, har de fleste nok været lidt skuffede. Det er beklageligt, at det meste af den gode software til 64'eren er spil og til en vis grad »business«-software, hvor loading og opstartsprocedurerne hurtigt bliver en rutinesag, når de først er lært. Det er derfor glædeligt, at LOGO-indlæringssproget nu er tilgængeligt på Commodore 64 i Danmark, hvilket betyder, at der nu endelig er et computersprog, som er lettere at lære end Basic, og som kan »vokse« med brugeren.

LOGO er egentlig et biprodukt fra forskningen i »kunstig intelligens« (Artificial Intelligence), og er udviklet af Seymour Papert, og hans kolleger på MIT (Massachusetts Institute of Technology). LOGOs hovedformål er, at give brugeren kommandoen over computeren. Dette står i stærk modsætning til ideen bag »Computer Assisted Learning« (CAL), hvor det er computeren, der har kommandoen over brugeren. Et eksempel på en CAL-rutine er den velkendte type, med et spørgsmål- og svar-program, hvor eleven bliver belønnet, når svaret er rigtigt, og samtidig bliver præsenteret for et nyt sprøgsmål. Hvis svaret derimod er forkert, bliver spørgsmålet gentaget, indtil eleven har fundet det rigtige svar.

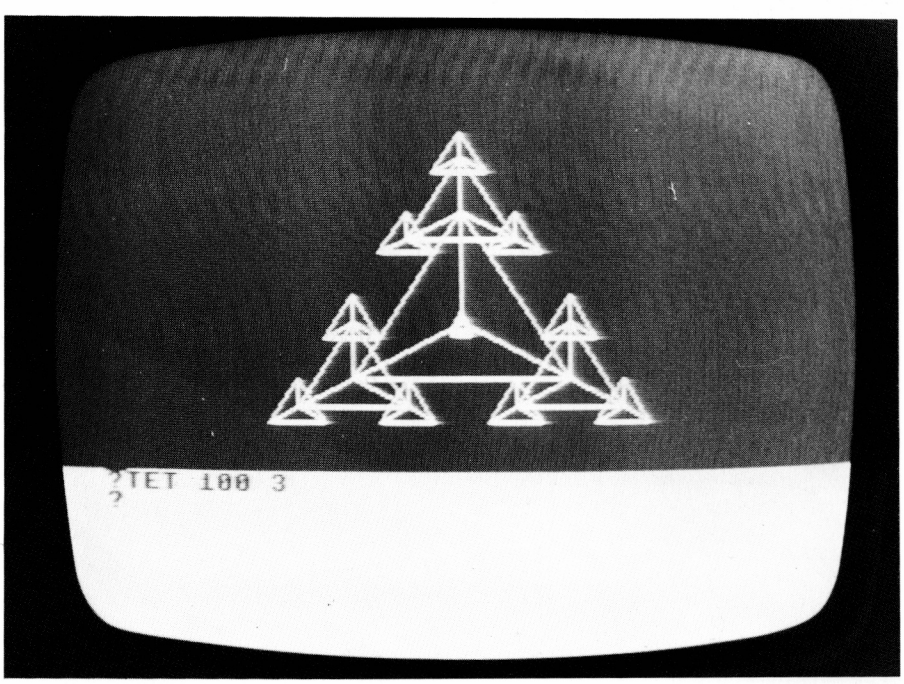
Papert og hans meningsfæller forkaster denne form for kontrolleret indlæring, og de udviklede LOGO-sproget ud fra den grundtanke, at et barn skulle være i stand til at bruge systemet. Det var fra starten hensigten, at sproget skulle kunne anvendes til kreativ indlæring i skoler og – med den store stigning i antallet af hjemmecomputere – også i hjemmet.

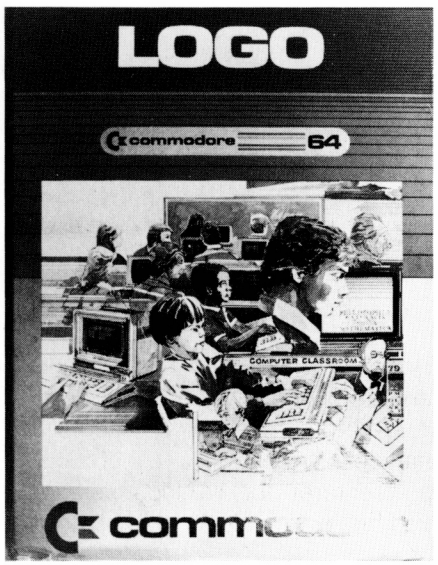
LOGO startede som et list-behandlingssprog, og benyttede mange af ideerne fra LISP og PROLOG, som er to andre Artificial Intelligence-sprog. I den oprindelige version, indeholdt LOGO ikke den senere verdensberømte »Skildpadde«-grafik. Denne tidligste version var udviklet til brug på Apple II og TI 99/4. Appleversionen blev derefter ændret til en type af LOGO, som idag kører på Sinclair Spectrum under navnet LCSI LOGO, mens en anden version, der hedder Terripin LOGO og produceres af Terripin Inc., dannede grundlag for den version, som nu kan fås fra Commodore.

Systemet med Commodore-LOGO indeholder to disketter, en engelsk tutorial/guide på over 400 tæt skrevne sider, samt en dansk håndbog og introduktion på 134 sider. Den ene diskette er LOGO-disketten, der er kopi-beskyttet, og som indeholder selve LOGO-sproget. Den anden er en LOGO »Utilities«-(hjælpe)-diskette, som indeholder forskellige færdigskrevne programmer og eksempler.

For at forstå, hvordan og til hvad LOGO kan bruges, kan man med fordel se på strukturen i den medfølgende engelske vejledning. Denne manual er inddelt i kapitler med følgende overskrifter: Grafik, Talbehandling, Ord og Lister, Sprites og Musik. Herudover er der nogle tillæg, som bl.a. indeholder et kapitel om Assembler-sproget. Kapitlet knytter sig til det assembler-program, der ligger på hjælpedisketten. Meningen med strukturen i vejledningen er, at brugeren skal følge denne når han/hun skal lære sig LOGO-sproget. Det er naturligt, at f.eks. et barn vil starte med at tegne, derefter begynde at bruge tal og bogstaver, for senere at anvende det allerede tillærte, til at lave en liste eller et brugbart program.

Det første kapitel i vejledningen handler om grafik. Her bliver brugeren præsenteret for »skildpadden«, der ved hjælp af enkle instruktioner, som f.eks. »forward«, »back«, »right« og »left« kan bruges til at tegne figurer på skræmen. For at komme i »grafik-tilstanden« (grafik-mode), bruges kommandoen »draw«, som opdeler skærmen i to dele. Den største del bruges af skildpadden, og de nederste fem linjer anvendes til indtastning af instruktioner til skildpadden. Når ordren er indtastet, og der er trykket Return, udfører skildpadden straks kommandoen,





som f.eks. kan bestå i, at dreje sig et bestemt antal grader eller at tegne en linje.

Vejledningen starter med at vise, hvordan der tegnes firkanter og trekanter, og forklarer derefter hvordan, man ved at kombinere disse figurer, kan tegne omridset af f.eks. et hus med vinduer.

Når brugeren har lært, at flytte skildpadden rundt på skærmen, kommer turen til at lære, hvordan man opbygger procedurer, og hvordan man ved at ændre en smule på en procedure og gentage den flere gange, kan tegne et stort antal forskellige og meget specielle figurer. Hjælpedisketten støtter gennemgangen af det grafiske afsnit, med en del procedurer, til at tegne figurer med krumninger. Vigtigere er det måske, at disketten indeholder en procedure, der giver børn, som endnu ikke kan læse, mulighed for at bruge LOGO.

Desuden findes en procedure, der sætter brugeren i stand til at tegne på skærmen ved hjælp af et joystick.

Det næste kapitel i vejledningen er logisk nok, talbehandling. I afsnittet om grafik, er brugeren blevet indført i anvendelsen af tal og elementær beregning. Der bygges nu videre på dette grundlag, ved først at forklare hvordan LOGO anvender tal, og siden ved at introducere brugeren for begreberne tilfældige tal, sinus, cosinus og tangens. Igennem tilegnelsen af LOGO kan f.eks. en skoleelev således få en rimelig god forståelse for mange matematiske begreber.

Det nærmeste LOGO kommer på egentlig Artificial Intelligence er i brugen af ord og lister. På disse områder ligger LOGO tæt på LISP-sproget.

Ord og lister anvendes for at gøre LOGO til et interaktivt sprog. Dette er gjort for at få det til at se ud som om, programmet kan tænke. Herved kan der skabes mere spændende situationer og effekter. For at illustrere disse faciliteter, er der et eventyrspil på hjælpedisketten.

Commodore-LOGO indeholder også afsnit om Sprites og musik, hvorved man kan udnytte de meget store muligheder på 64'eren. Yderligere indeholder et af vejledningens tillæg et Assembler, der fuldstændigt er skrevet i LOGO.

LOGO vil være til nytte for de personer, der ønsker at give deres børn flere uddannelsesmuligheder uden for skolen.

Desuden vil LOGO kunne anvendes af skoler i forbindelse med undervisningen i beregning, kalkulation og i brugen af logik. Det skal dog pointeres, at LOGO ikke skal ses som et selvstændigt og løsrevet værktøj, men kun bør bruges som et element i en større undervisningssammenhæng. LOGO kan således betragtes som en nøgle til at åbne en af dørene til fremtiden, idet det kan give et barn en tidlig og let forståelig introduktion til EDB-baseret informationsbehandling.

Der kan kun kritiseres få ting ved Commodore-versionen af LOGO, men det skal dog nævnes, at den engelske vejledning tilsyneladende ikke har som formål at blive læst og forstået af LOGO-brugeren, f.eks. et barn. Hvis denne vejledning derimod har været tænkt til den voksne læser, virker den nærmest, som om den taler ned til læseren. Dette gælder også den danske håndbog, hovedsagelig fordi den er en direkte oversættelse af dele, af den engelske vejledning.

Få mere ud af din COMMODORE

The Commodore 64 Handbook *Peter Lupton & Frazer Robinson*	**kr 130,–**
The Commodore 64 Program Book *Vince Apps*	**kr 108,–**
Sprite Graphics for the Commodore 64 TM *Sally Greenwood Larsen*	**kr 322,–**
Using the Commodore 64 in the Home *Hank Librach & Bill L. Behrendt*	**kr 214,–**
Microwars on the Commodore 64 *Humphrey Walwyn*	**kr 130,–**
Brainteasers for the Commodore 64 *Genevieve Ludinsky*	**kr 130,–**
Commodore 64 - Getting the Most From It *Tim Onosko*	**kr 180,–**
LOGO Programming *Anne Moller*	**kr 152,–**
Music and Sound for the Commodore 64 *Bill L. Behrendt*	**kr 300,–**
Commodore 64. BASIC Programming & Applications *Goldstein & Mosher*	**kr 186,–**
Brain Games for Kids and Adults Using the Commodore 64 *J. Stephenson*	**kr 139,–**
The Advanced Commodore 64 Handbook *Peter Lupton & Frazer Robinson*	**kr 152,–**

BESØG VOR NYE BOGLADE MED DET STORE UDVALG AF DATALITTERATUR

svensk-norsk bogimport a.s. st. kongensgade 59, 1264 københavn k, telefon 01 14 26 66

LÆSERBREVE

Rolf Hjorth fra København S, spørger, om man kan anvende såvel et farvefjernsyn som et s/h (eller en monokrom monitor) på samme tid med sin 64'er.

I teorien er der ikke noget til hinder for, at man kan spænde begge apparater på sin computer. Den lille skærm på f.eks. SX64 er nok en anelse for lille til f.eks. tekstbehandling – men så sætter man blot en større monitor eller et farvefjernsyn til SX's videoudtag, hvorefter man reelt har to video-»billeder« at kigge på. Jeg tror kun, at man vil få problemer, hvis kablerne til de forskellige TV-apparater er meget støjfølsomme, eller de to apparater kan genere hinanden med RF-udstråling. Prøv dig frem, Rolf. Du kan i hvert fald ikke brænde computeren af ...!

Videre spørger Rolf, om man kan anvende flere indstiksmoduler på samme tid – f.eks. med et motherboard.

Det kan man teoretisk set godt, når blot man ikke spænder mere på ekspansionsstikket, end det kan strømforsyne (750 mA). I praksis volder det hele dog flere problemer.

Mange indstiksmoduler indeholder autostart-rutine, og udover at de »fylder« området fra $8000 og op, kan de i opstartfasen placere »interne« programdele mange steder i 64 – f.eks. i $C000 og opad. Hvis to sådanne moduler forsøger at lægge beslag på det samme lagerområde, opstår der betydelige problemer, som brugeren ikke selv vil kunne løse. De forskellige producenter af indstiksmoduler, er normalt ikke flinke til at anføre, hvilke lagerområder, deres programmer inddrager.

Der er en stribe moduler på vej med f.eks. »programmer's aid«, toolkits, turbo tape og lignende – disse moduler er efter deres natur, beregnet til at fungere sammen med de programmer, som vi selv skriver – derfor er disse moduler normalt godt beskrevet med hensyn til deres brug af lager. Sådanne moduler vil i mange tilfælde kunne fungere sammen med andre programmoduler.

Om brugen af funktionstasterne, henviser vi til artiklen andetsteds i bladet.

Grafik PA MPS-801

Der er mange brugere af Commodores MPS-801 grafiske printer, som spørger om, hvordan man definerer sine egne tegn, og bruger dem til f.eks. brevlogos og lignende.

Teknikken bag de brugerdefinerede tegn, er grundigt beskrevet i brugsvejledningen, men til gengæld er det ikke særligt godt beskrevet, hvordan man styrer linieskift og så videre. Jeg har altid fundet det en god idé at samle sine tegndata i en strengvariabel, og så »montere« en CHR$(13) eller en CHR$(10) i variablen, når der skulle udføres en vognretur.

Hvis man lægger sine tegndata i form af data, kan man klare sig med følgende model:

```
For X = 0 TO 6
READ A
A$ = A$ + CHR$(A)
NEXT
A$ = A$ + CHR$(vognretur)
```

Hvis det brugerdefinerede felt består af flere tegn, som skal printes ved siden af hinanden, læser man blot alle tegndata-sættene, samler dem i A$ efter ovenstående model – hvorefter man klistrer en CHR$(vognretur) bagpå – om man anvender CHR$(10) eller CHR$(13) eller eventuelt dem begge to, er op til brugeren selv. Det betaler sig at eksperimentere, men jeg tror, at den teknik, jeg har beskrevet herover, i det lange løb er den enkleste.

Fra Lars Krogager i Køge, har vi modtaget følgende kommentar til vores program om printposition:

Lad mig først sige, at jeg synes det er et godt og interessant blad, I har startet for disse spændende maskiner.

Jeg har med interesse læst jeres notits om PRINT positioner ved hjælp af POKE værdierne 211 og 214. Da jeg synes det er en besværlig udvej, bruger jeg altid SPC(X).

SPC(X) kan indeholde værdier op til max. 255. Så med højest 4 SPC(X) kan man printe over hele skærmen. Det lille program, I viste som demonstration, har jeg lavet herunder. Desuden har jeg føjet et PRINT i nederste linie af skærmen til.

```
10 PRINT''(CLR/HOME)''SPC(255)
SPC(160)''TEST''SPC(79)''TEST''
20 PRINT SPC(255)SPC(255)''H
(HOME)''
```

Linie 10 udfører jeres test, og linie 20 printer ''H'' i nederste linie og går derefter »hjem« for at undgå »oprulning«.

Fra flere læsere har vi modtaget forespørgsler om, hvordan man »åbner en list« i færdigkøbte programmer.

I den anledning kan vi sige, at når et program er beskyttet på den måde, er det for at hindre andre i at misbruge programmet. Selvfølgelig kan sikkerhedsforanstaltningerne brydes, men vi hverken kan, eller vil, give oplysninger fra os i den retning.

Piratkopiering af programmer tager vi meget skarpt afstand fra.

Elementære spørgsmål

Når man første gang står med en computer i hånden, er der en masse elementære spørgsmål, der helt naturligt trænger sig på - spørgsmål, som brugervejledningen går ud fra, man kender svaret på, og derfor ikke kan få sig selv til at stille. Vi vil her og i kommende numre prøve at besvare nogle af disse spørgsmål.

Hvorfor fungerer kommandoen INPUT ikke i direct mode?
Sv.: INPUT-kommandoen er beregnet til at afbryde et kørende program for at give en variabel en anden værdi. Dermed er der mulighed for en »dialog« mellem brugeren og programmet. Men når der ikke kører noget program, tjener det jo ikke noget formål. Derfor kommer der en ILLEGAL DIRECT ERROR melding.

Kan jeg ændre den hastighed som cursor'en blinker med, og som printeren arbejder med?
Sv.: Det er svært at føre det ud fra Basic, fordi man skal ændre varigheden af processorens »interrupt«, dvs. den afbrydelse computeren foretager, normalt 60 gange pr. sek., f.eks. for at få cursor'en til at blinke. Den eneste fornuftige måde at lave en sådan ændring på, er ved programmering i maskinsprog.

I nogle programpakker, man kan købe, fungerer »RUN/STOP« og »RESTORE« ikke. Kan jeg selv programmere den slags?
Sv.: For at kunne gøre det, må du nøjagtig vide hvad der sker ved »RUN/STOP« og »RESTORE«. Disse to taster stiller computeren tilbage til udgangsstillingen. Farverne bliver reset, sprites, grafik og lyd afbrydes og programmer går i stå.

Mere nøjagtigt er det følgende der sker: så snart computeren får »RUN/STOP« og »RESTORE«-signalet (det kaldes også NMI), springer den til et underprogram i operativsystemet, der ligger i adresserne 792 og 793. Du behøver kun at ændre disse to pegere med POKE 792,134: POKE 793,234 og afbrydelsen er klaret.

I håndbogen er der tale om ASCII- og skærmkoder. Hvad er forskellen?
Sv.: ASCII står for American Standard Code for Information Interchange. Denne kode er standardiseret, som navnet siger. Dvs. at 65 i næsten alle computere betyder A og 90 Z. Denne kode, hvor alle bogstaver og alle styretegn har deres bestemte talkode, bruges hovedsageligt til oplagring og overføring af data. Med CHR$(X) kan man betegne det tegn, eller den funktion, der svarer til dette tal.

F.eks. står CHR$(13) for »RETURN«. Hos Commodore har skærmkoderne derimod kun betydning, i den bestemte form de har. De står for de tal der skal skrives i skærmlageret, for at afbilde et bestemt tegn på skærmen. Der er der ingen styretegn, derfor har inverterede tegn en anden skærmkode end normale tegn.

Fjerner man plads i hukommelsen, når man stikker et programmodul ind?
Sv.: Det kommer an på modulet. Hukommelsesarealet »forsvinder« ikke under nogen omstændigheder, det er bare ikke til rådighed. Når du bruger Basic-udvidelser eller hjælpeprogrammer, er der f.eks. mindre hukommelsesareal til rådighed end uden det pågældende modul. Ved andre programmer som spil og brugerprogrammer, har man slet ingen hukommelse til rådighed, fordi disse programmer starter selv, og man kommer ikke ud af dem. Derfor har man ikke noget at bruge resten af hukommelsen til. Nogle moduler, som interface moduler, optager iøvrigt kun meget lidt eller slet ingen hukommelsesplads.

Kan man tilslutte 2 Datasetter til Commodore 64?
Sv.: Nej, det er der ingen mulighed for. På CBM-computerne som CBM 4000 eller 8000 var der 2 kassettetilslutninger, for at gøre det lettere at kopiere data og programmer. Men på 64'eren er der kun 1 stik til kassetterekorderen. Teoretisk kunne man forestille sig en hardwareudvidelse, f.eks. i form af en omskifter på kassetteporten, der kan styres med et signal, der kunne komme fra bruger-porten, således at man kan skifte mellem 2 kassetteoptagere. Men en opstilling af den slags har jeg aldrig set.

Hvad er forskellen mellem en Joystisk og en Paddle?
Sv.: Joysticks er styrepinde, som der kan tilsluttes to af til Commodore 64. De afgiver et retningssignal, som »op«, »ned«, »venstre« eller »højre«, og desuden er der en aftrækkertast.

Paddles er derimod nærmest et rat. Der er indbygget en skydemodstand, hvis modstandsværdi ændres når du drejer på rattet, således at der kommer mere eller mindre spænding til computeren. Dermed er også virkemåden givet og forskellen mellem de to former for spiludstyr. Paddles giver en signalværdi fra sig, som i bedste fald går fra 0 til 255 (ikke alle kan komme op på så høje værdier). Derfor er paddles særlig gode til at styre ketchers (i computer-tennis osv.), eller biler og lignende, hvor det kommer an på præcis styring af bevægelser frem og tilbage.

Kan der bruges forskellige DATA-linier for subrutine og hovedprogram?
Sv.: I DATA-linierne skriver man de værdier, som i løbet af programmet læses ind med READ-kommandoen. Maskinen gør det strengt i rækkefølge. Ved første READ indlæses den første DATA-værdi, lige meget hvor den ligger i programmet. Der skelnes altså ikke mellem subrutine og hovedprogram i den forbindelse.

Kan man i en subrutine bruge samme variabler som i hovedprogrammet, eller gør de forskel på subrutine og hovedprogram?
Det gør de ikke. De har samme værdi lige meget hvor i programmet de findes. Støder maskinen på en CLR eller NEW slettes samtidig alle variabler.

RUN
Commodore-magasin

RUN er navnet på Danmarks første uafhængige magasin om Commodore computere.
RUN bringer nyheder, tests af hardware og software, litteratur m.m. alt med relation til Commodore. RUN vil indeholde mange programmer.
RUN udkommer 10. september og 10. november i år, og hver anden måned i 1985.
RUN fås i kioskerne
kr. 19,85

Abonnement 6 numre kr. 98;
Benyt bestillingskupon på bagsiden.

Computerworld

Computerworld er nyhedsmediet for datanyheder til det danske marked.
Computerworld henvender sig til dem, der ønsker at holde sig ajour med udvikling, marked, uddannelse og meget mere – alt med relation til anvendelse af computere.
Computerworld udkommer 24 gange pr. år.
Som abonnent på Computerworld modtager De som bonus Computerworld Buyers Guide, altomfattende håndbog for edb-købere, uden beregning (værdi kr. 210,- + moms).
Årsabonnement 24 numre

kr. 215;

+ moms kr. 47,30 i alt kr. 262,30
(Prøveabonnement 6 måneder kr. 98,-)
Benyt bestillingskupon på bagsiden

Micro Verden

Micro Verden er bladet for brugere af personlige computere.
Micro Verden tester og anmelder personlige computere, programmer samt alle former for tilbehør. Micro Verden er en meget saglig og objektiv informationskilde for brugere og kommende brugere af personlige computere.
Micro Verden udkommer 11 gange pr. år og sælges i landets kiosker. kr. 24,50.

Årsabonnement kr. 205;
+ moms kr. 45,10 = 250,10.

(Prøveabonnement 5 numre kr. 98,-)
Benyt bestillingskupon på bagsiden.

Computerworld Danmark A/S
Gammel Strand 50
1202 København K.

01·12 34 11

JA!

Jeg (vi) bestiller herved abonnement 6 numre kr. 98;-

Måned:_____

Firma:_____

Navn:_____

Adresse:_____

Postnr.:_____ By:_____

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret Computerworld betaler portoen

778

Reserveret postvæsenet

JA!

Jeg (vi) bestiller herved
☐ Årsabonnement – 24 numre
☐ Prøveabonnement – 6 måneder

Måned:_____

Firma:_____

Navn:_____

Adresse:_____

Postnr.:_____ By:_____

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret Computerworld betaler portoen

778

Reserveret postvæsenet

JA!

Jeg (vi) bestiller herved
☐ Årsabonnement
☐ Prøveabonnement – 5 måneder

Måned:_____

Firma:_____

Navn:_____

Adresse:_____

Postnr.:_____ By:_____

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret Computerworld betaler portoen

778

Reserveret postvæsenet

Computerworld

64-eren ideel til Teledata

Af Flemming Lerbæk

- Jeg hørte tilfældigt en dag, at Commodore 64 kunne bruges i forbindelse med Teledata. Derfor faldt valget på den - et valg jeg ikke siden har fortrudt.
Sådan siger Knud Leo Larsen, der er indehaver af Larsen-Vinhobby A/S. Firmaet har til huse i Tommerup Stationsby ca. 18 kilometer fra Odense.

Larsen-Vinhobby var det første firma på Fyn, der blev tilsluttet Teledata. Siden er flere kommet til. Det er – selvfølgelig – de store banker osv. Men hvad har fået Leo Larsen til at vælge denne mulighed?

– Jeg tager mig til daglig af den overordnede styring på virksomheden. I den forbindelse blev jeg hurtig opmærksom på telestyrelsernes tilbud om teledata, og fik anskaffet mig en af deres terminaler.

– Den var dyr i anskaffelse, og mangler nogen af de muligheder, jeg er interesseret i. Jeg forhørte mig, om der var andre muligheder på markedet. Der blev peget på CBM 64 og et lille teledata-modul fra HANDIC. Med det og Teledatas modem er jeg nu i stand til at arbejde meget mere effektivt med systemet.

Leo Larsen kan gemme 24 sider information, i sin computer ad gangen. Hvis der skal spares, kan forbindelsen til Teledata afbrydes og siderne gemmes i ro og mag. Har man en diskstation ved siden af, er der gode muligheder for at gemme de oplysninger, man har hentet ud af det omfattende Teledata system. For øjeblikket er der ca. 45.628 sider med information i systemet. Gyldendals to binds leksikon er netop kommet med.

– Jeg bruger systemet på mange måder, men som leverandør til systemet, er det lidt trist at se, at der endnu kun er få brugere af systemet.

– På 30 sider giver jeg selv en slags vinleksikon. Folk kan hente oplysninger om alt, hvad der hører til mit produkt, Treugersvinen.

Leo Larsen har god forstand på de ting, der oplyses i de 30 sider teledata-information. I 35 år har han været i vinbranchen. Dels som vinkyper, men i de sidste 15 år i selvstændigt firma.

Firmaet tæller en halv snes ansatte, der er 12 millioner i årlig omsætning, og 75 pct. af vinen eksporteres til Norge, Sverige og England.

– Den viden jeg giver fra mig i Teledata, er oplysninger på mit eget produkt, og der er muligheder for at bestille via Teledata. Oven i købet med en lille rabat.

Men Larsen er først og fremmest forretningsmand.

– Jeg har ikke spor begreb om at programmere. Jeg har ikke haft tid til at sætte mig ind i det. Men 64'eren åbner muligheder, som jeg meget gerne vil sætte mig ind i. Ud over Teledata bruger jeg foreløbig kun denne maskine som skakpartner.

Leo Larsen fik telefolkene til at hjælpe sig med udformningen af de første sider information i systemet. Men nu er han selv i stand til at programmere – tegne/layoute og rette i siderne – netop fordi han har sin hjemmecomputer.

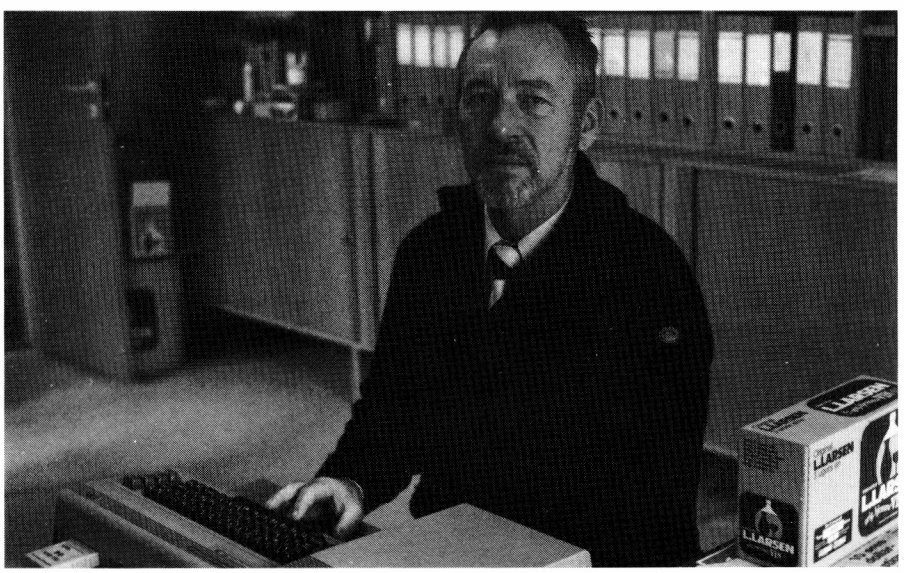
Det er iøvrigt langt fra den eneste EDB-maskine, han og hans firma betjener sig af. På privatadressen står desuden en METRIC med en meget stor floppy. Den bruges til administrative og planlægningsmæssige opgaver. I kæmpe skemaer justeres data automatisk, hvis blot en af de mange faktorer f.eks. kostprisen på sukker går op eller ned.

En lidt mindre METRIC bruges til tekstbehandling, og til den er der pladelager.

– 64'eren er imidlertid meget velegnet til Teledata, og det er min plan, at jeg, når systemerne er lidt mere udbyggede, også vil koble mig på en lignende database i England.

Hvornår det sker, er et spørgsmål om økonomi. Teledata tager 1148 kroner for de 30 sider i kvartalet, fra leverandøren. Det koster 400 kroner i oprettelse, og 150 kroner i kvartalet for at være på systemet, og 50 øre i minuttet at se med.

– Jeg kan se, at der er en fremtid i Teledata. Men det er en teknik, som kan forbedres i forbindelse med det såkaldte hybridnet. Der er næppe plads i det gamle net, til alle potentielle brugere. Der er imidlertid ingen tvivl om, at man kan finde mig på hybridnettet, så snart det er muligt, siger Leo Larsen.



Leo Larsen, Larsens Vinhobby A/S med sin Commodore 64'er.
– Den er ideel til netop denne opgave.

TIPSPROGRAM

Spændingen er udløst. Ugens 13'er er en kendsgerning, og nu skal man i gang med at undersøge, hvor mange rigtige man selv har. Tipper man mange rækker, vil det ofte være et ret stort arbejde, og har man en Commodore 64 stående, vil man nok spørge sig selv, hvorfor den ikke lige så godt kan lave noget af det sure arbejde.

Tipsprogrammet er primært lavet for at gennemgå et stort antal rækker for eventuelle gevinster. Tipper man, som mange gør, de samme rækker hver uge, behøver man blot at indtaste rækkerne een gang og gemme dem på bånd eller diskette.

Tipper man forskellige rækker hver uge, er besparelsen måske knap så stor, idet man så skal indtaste nye rækker hver uge.

Programmet arbejder kun med enkeltrækker, og man er derfor nødt til at lave et lille konverteringsprogram, hvis man tipser systemtips. Det er i det hele taget sådan, at programmet her nemt kan indlæse en file, der er skrevet på et eventuelt andet program, blot man opbygger den på samme måde som i dette program. Det vil sige, at man skal opbygge sine rækker i en strengvariabel med 13 tegn, og man kan næsten ikke opbygge et tipsprogram anderledes.

Derudover er programmet her i stand til selv at lave tilfældige rækker ud fra en forudbestemt procentuelt sandsynlig tegnfordeling. Programmet vil ikke holde sig 100% til denne tegnfordeling, men statistisk vil den ligge tæt på, med de afvigelser og variationer, der i sidste ende måske alligevel kan give gevinsten.

Programmet er opbygget med en menu, hvor man har 8 valgmuligheder. Inden man når så langt, skal man lige indtaste, hvor mange rækker man skal tipse. I det her viste program er der sat et maksimum på 500 rækker, men dette kan nemt ændres ved linie 130 at ændre DIMRK$-kommandoen.

Vælger man punkt 1 i menuen, skal man selv indtaste sine rækker. Er man kommet til at lave fejl, kan man vælge punkt 2, der giver mulighed for at rette i en eller flere rækker, ligesom man også her kan få printet sin række ud på skærmen. Punkt 3 vil danne nogle tilfældige rækker ud fra en besvarelse om tegnfordeling. Under punkt 4 indtaster man 13'neren, hvorefter programmet går i gang med at gennemgå rækkerne for eventuelle gevinster. Man kan i et lille »vindue« øverst til venstre på skærmen se, hvor langt den er nået.

Eventuelle gevinstrækker vil så til sidst blive skrevet ud med det pågældende rækkenummer ud for den pågældende gevinst.

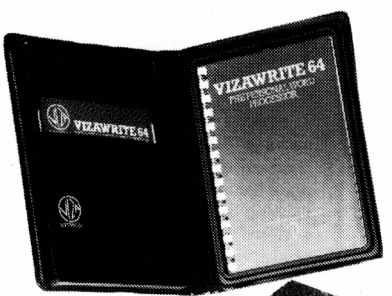
Endelig kan man under punkterne 5-8 gemme eller loade sine rækker, på enten tape eller diskette.

Programmet kan nemt udbygges eller sammenbygges med andre tipsprogrammer, og VIC-20 ejere skulle heller ikke have besvær med at tilpasse det til deres maskiner.

Programlistning side 42

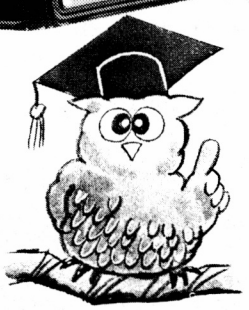
Kom og se det store udvalg

Vi har et af Danmarks største udvalg i bøger, blade og software til Commodore 64.
Ta' f.eks. 4 forskellige kvalitets-spil til kun kr. 149,-!
Få tilsendt vores software-katalog!!



VizaWrite 64

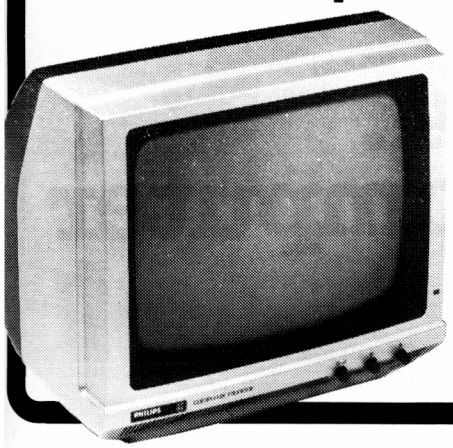
Brug Commodore 64 som professionelt tekstbehandlingsanlæg med VizaWrite. Arbejder med 80 tegn pr. linje og har indbyggede danske karakterer. Leveres på både modul og diskette med softloaded interfaces til skrivemaskiner.
Disketteversion: **1248,-**



SUMMER GAMES 64

Efterårets store hit! De olympiske lege med 8 dicipliner. Et action/strategi-spil med utrolig fornem grafik og bevægelse. Skal ses
Disketteversion kun kr. 248,-
Kassetteversion (2 kassetter) kun kr. **175,-**

Philips V7001



PHILIPS V 7001 sort/grøn monitorskærm Computer-monitor 12" med lyd. Leveres med gratis ledning, lige til at sætte på din Commodore 64/VIC-20.

1395,-

Vi sender over hele landet...

Åbningstider:
Mandag-torsdag: 10.00-17.30
Fredag: 10.00-19.00
Lørdag: 10.00-12.00

Tlf. 06 · 13 20 55

Computer-Butikken
Akademisk Boghandel
Vestergade 58A . 8000 Århus C

Saligheder med Simon

Af Flemming Lerbæk

Efter sidste nummers gennemgang af Simons Basic's mange muligheder, når det gælder programmering og omgangen med sprites, kan vi denne gang tage fat på karakter-defineringer og grafikken.

Simon Basic byder på ikke færre end 35 kommandoer, der kan henføres til grafik og skærm-manipulering. Derimod klares nydefinering af karakterer med blot tre meget effektive kommandoer.

Lad os se lidt nærmere på de tre, nemlig MEM, DESIGN og A.

MEM

Der sker noget i din Commodore 64, når ordren MEM effektueres. Det kan ske både direkte og via et program, som vi senere skal se på.

Fem ting skal man være opmærksom på sker, når MEM bruges.

1. For det første forskydes eller rettere kopieres tegndatasættet fra ROM $D000 (53248) til RAM $E000 (57344). Kan det nu passe, vil nogen måske sige. For er det ikke i det område, computeren gemmer sine data til grafik? Jo, det er rigtig nok. Derfor kan man heller ikke både arbejde med et nyt tegnsæt *og* grafikskærm samtidig i SIMON Basic.
2. For det andet får computeren besked på, at den nu skal søge sine karakterdata på adressen 57344 og fire K frem.
3. Tekst- og billedskærm lagerpladsen forskydes til $CC00 (52224).
4. Skærmen bliver blank.
5. Arbejder man med sprites skal man tillægge 49152, idet sprite-data forskubbes. Derfor har sprigeprogrammet, som vi tidligere har præsenteret, også benyttet disse adresser.

   Dvs. at der er benyttet sprites med numrene fra 192 og opefter.

   I øvrigt kommer man ud af MEM-tilstanden på to måder. Enten ved at brug NRM eller CSET0.

DESIGN

Under gennemgangen af Sprites havde vi også brug for DESIGN-kommandoen. Denne gang skal vi blot skrive DESIGN 2.

A

Og det samme gælder sidste kommando i trioen A. Husk igen, at linjenumrene skal være på samme ciffertal.

Nye tegn

Et dansk karaktersæt er altså ingen luksus for en ejer af Simons Basic. Det klares meget let, som det er vist i programmet »karakterer«. Programmet kaldes som de øvrige via den første menu. Trykkes 3 kommer der kort tid efter oplysninger om F-tasternes belægning.

Og mere enkelt kan det næppe være. Man slår en tast an, og computeren spørges, hvilken tast der skal nydefineres. Tag f.eks. A. Det tastes ind og RETURN slås an. Straks er tegnet omdefineret til de informationer, der ligger i programmets linjer fra 60767 til 60774.

Vil man nydefinere i forhold til disse data, indtastes blot et nyt B-mønster i linjerne og den sag er klar.

Ikke blot A-et kan omdefineres. Også de tegn, der fremkommer ved at bruge SHIFT eller Commodore logoet sammen med A kan omdefineres.

Om selve programmet skal der også falde et par kommentarer. I linjerne 60700-60720 defineres Æ og Å. Det er tasterne og der laves om. Derefter kommer KEY-tildelingen og de forskellige titel-sider.

Fra linje 60745 begynder så de forskellige små programrutiner, der er knyttet til KEY-ordrene, der effektueres via F-tasterne.

Frem til 60777 gennemføres så nydefineringen efter eget valg. I linje 60763 forekommer PEEK (52990). Det er det sted på skærmen, hvor man indtaster det tegn, der skal modificeres. Man får skærmkoden ud, og det er netop dette tal, man har brug for i DESIGN-sammenhængen.

Til slut følger det sædvanlige menuprogram.

Og husk at indføje et REM i første linje i hvert af de viste programmer, ellers går der kuk i gennemløbet, som beskrevet i sidste nummer af RUN. Når indtastningsfasen er overstået slettes REM igen i den endelige version.

Grafik

Grafikprogrammet benytter nogle få af de 35 mulige skærm og grafik-kommandoer, Simon Basic byder på. Der er et overflod af valgmuligheder. Det var svært at finde netop de 16 kommandoer, der skulle tilkendes F-tasterne. Jeg valgte de kommandoer, der efter min mening oftest er brug for.

Men de 35 muligheder byder på mange andre muligheder, som jeg ikke her vil kommentere. Der kommer i næste nummer en gennemgang – eller nærmere oplistning og en meget kort brugervejledning – på alle 114 kommandoer i Simon Basic.

Når programmet startes på vanlig vis fra hovedmenuen, kommer der et lille vue på, hvad grafikken rummer af muligheder. Hele herligheden laves i linjerne 60818-60821.

Der er tale om brug af REC, der laver rektangler og TEXT, der gør det muligt at skrive tekst på en højopløsnings grafisk skærm.

Fra linje 60830 begynder så de små rutiner, som igen knytter sig til KEY ordrene via F-tasterne. Når en tast anslåes kommer den tilknyttede kommando frem. Det sker med en kort forklaring på dens opbygning.

Linjen forsynes med et linjenummer osv. som der foreslås på skærmen. Lad os gennemgå et eksempel: Vi vil lave et lille program, der skal skrive RUN i en ramme midt på skærmen. Derefter tegnes en anden figur.

For overhovedet at kunne arbejde på en grafisk skærm, skrives 10 HIRES 0,1. Det skulle give en hvid baggrund til en sort tegning.

Vi vil nu først tegne rektanglen. Og vi taster F 2. Skærmen spørger nu, om vi vil vælge mellem HIRES eller MULTICOLOUR. Men da vi vil arbejde i et program, og allerede har lavet en programlinje, ser vi stort på tilbudet og kaster i stedet et blik på de næste linjer.

Her vises kommandoen REC med alle dens facetter. For det første skal der et linjenummer på. Det er i vort tilfælde 20. Curseren køres op i den linje, hvor en REC kommando står alene. De 20 indtastes og curseren bevæges hen til X. Det er angivelsen af x-koordinaten for øverste venstre hjørne i et rektangel. Vi skriver f.eks. 100 samt et komma. Og tilsvarende med Y, som vi vil bestemme til 80, som indtastes samt et komma.

A og B angiver rektangelens udstrækning i henholdsvis vandret og lodret retning fra øverste venstre hjørne. Vi taster f.eks. 100, 40 ind og sidste faktor skal være 1, hvis vi vil se resultatet. Et nul ville slette i stedet.

Vi går nu ned og placerer curseren over CALL MODUS. RETURN tastes, og der spørges, om vi vil arbejde direkte dvs. se resultatet af vore anstrengelser straks eller arbejde videre i et program. Vi vælger den 2. mulighed og 2 tastes. Nu kommer der en liste over det hidtidige program.

Næste ordre, vi vil have i programmet, er TEXT. Vi taster F 11 (altså logo og F 3), og straks er vi i samme situation som ovenfor. Udfyld skemaet, og linje 30 er klar. Det kan ske med TEXT120,85,»RUN«,1,3,24. De to sidste værdier er en forstørrelses-faktor til skriften, og de 24 er angivelsen af, at vi ønsker tredobbelt »luft« mellem bogstaverne.

For at vi kan ser herligheden laves en lille sløjfe 100 GOTO100 og vi taster RUN. Jo, det virker. Sløjfen kvittes med et tast på RUN-STOP knappen.

Endelig kommer vi til DRAW-kommandoen, og vi taster F 8, så fremkommer et skema igen. Det er efter min mening et skoleeksempel på, hvordan man kan udbygge informationerne om de enkelte kommandoer, hvis man har lyst. Men det kan man muntre sig med selv.

DRAW»65789«,250,50,1 blev det til. Men, hov, stop! Der kommer ikke andet end en »prik« frem på skærmen ved gennemkørslen. Forklaringen er enkel. Når en ting tegnes med en enhed op, en højre, en ned og en op, sker der ikke meget. Derfor kaldes ROT-kommandoen. Den giver mulighed for at forstørre et objekt. Jeg vælger ROT 0,40, hvilket giver nul rotation og 40 ganges forstørrelse. Og straks er alt i orden.

Som sagt er der rig mulighed for at udbygge det her viste hjælpeprogram. Men jeg håber, du har fået fat i ideen og grundholdningen, som jeg allerede skitserede den sidst. Simon Basic har 114 kommandoer ekstra. Ingen vil vel påstå, at de kan dem alle. hvorfor så ikke lade computeren arbejde for dig og vise vejen i form af de rigtige stikord.

I næste nummer af »RUN« vil vi afslutte denne lille serie omkring Simon Basic. Det sker med et lille hjælpeprogram om lyd, samt lidt om nogle af de øvrige kommandoer. Og samtidig kommer det lovede skema, der i kort form fortæller om anvendelsen af alle 114 kommandoer. Endelig kommer der et par programeksempler, der skal vise noget af dem kraft og spændstighed Commodore 64 med Simon Basic byder på.

GRAFIK

```
60800 POKE45,PEEK(174):POKE46,PEEK(175):
      CLR
60801 PROC GRAFIK
60802 KEY1,"CALL COLOUR"+CHR$(13)
60803 KEY2,"CALL REC"+CHR$(13)
60804 KEY3,"NRM"+CHR$(13)
60805 KEY4,"CALL PLOT"+CHR$(13)
60806 KEY5,"CALL LINE"+CHR$(13)
60807 KEY6,"CALL CIRCLE"+CHR$(13)
60808 KEY7,"CALL PAINT"+CHR$(13)
60809 KEY8,"CALL DRAW"+CHR$(13)
60810 KEY9,"CALL ROT"+CHR$(13)
60811 KEY10,"CSET 0"
60812 KEY11,"CALL TEXT"+CHR$(13)
60813 KEY12,"CALL MOVE"+CHR$(13)
60814 KEY13,"CALL SCROL"+CHR$(13)
60815 KEY14,"CALL SCRSV"+CHR$(13)
60816 KEY15,"CALL READSCRSV"+CHR$(13)
60817 KEY16,"LOW COL 1,2,3"
60818 HIRES3,4:FORX=1TO99STEP2
60819 REC X,X,X,X,1:NEXTX:Y=99:Z=99
60820 FORX=99TO198STEP2:REC X,Z,Y,Y,1:Y=
      Y-2:Z=Z+2:NEXTX
60821 TEXT 215,150,*2]*2]RAFIK",1,3,16
60822 PAUSE2:PRINT"[CLR/HOME]":DISPLAY:P
      RINT"[CRSR/DOWN*2]KLAR":PRINT"[CRS
      R/DOWN*2]CALL MENU[CRSR/UP*3]":END
60823 PROC MUHI
60824 NRM:PRINT"[CLR/HOME]":COLOUR6,5:CE
      NTRE"[CRSR/DOWN]VÆLG OM DU SKAL AR
      BEJDE"
60825 PRINT
60826 CENTRE"I HIRES ELLER MULTICOLOUR?"
60827 PRINT:PRINT"[CRSR/DOWN][YEL]HIRES
      PC,SB"
60828 PRINT"MULTI F1,F2,F3[BLK]"
60829 END PROC
60830 PROC COLOUR
60831 EXEC MUHI
60832 PRINT"[CRSR/DOWN]LIN COLOURSC,BO"
60833 PRINT"---+------+-+--"
60834 PRINT"    [YEL]COLOUR[BLK]  ,":PRI
      NT"[CRSR/DOWN*2]CALL MODUS[CRSR/UP
      *3]":END
60835 PROC MODUS
60836 CENTRE"[CRSR/DOWN]DIREKTE (1) ELLE
      R PROGRAM (2)"
60837 PAUSE1:POKE198,0:WAIT203,63:GETA$
60838 IFA$="2"THEN LIST0-59000
60839 END
60840 END PROC
60841 PROC REC
60842 EXEC MUHI
60843 PRINT"[CRSR/DOWN]LIN REC  X,  Y,
      A,  B,1"
60844 PRINT"----+---+--+---+---+---+-"
60845 PRINT"     [YEL]REC[BLK]":PRINT"[C
      RSR/DOWN*2]CALL MODUS[CRSR/UP*3]":
      END
60846 PROC PLOT
60847 EXEC MUHI
60848 PRINT"[CRSR/DOWN]LIN  PLOT  X,  Y,
      1"
60849 PRINT"----+----+--+---+-"
60850 PRINT"     [YEL]PLOT[BLK]":PRINT"[
      CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]"
      :END
60851 PROC LINE
60852 EXEC MUHI
60853 PRINT"[CRSR/DOWN]LIN  LINE   X,  Y
      , X1, Y1,1"
60854 PRINT"----+----+---+---+---+---+-"
60855 PRINT"     [YEL]LINE[BLK]":PRINT"[
      CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]"
      :END
60856 PROC CIRCLE
60857 EXEC MUHI
60858 PRINT"[CRSR/DOWN]LIN  CIRCLE   X,
       Y, XR, YR,1"
```

```
60859 PRINT"----+------+---+---+---+---+
      -"
60860 PRINT"     [YEL]CIRCLE[BLK]":PRINT
      "[CRSR/DOWN*2]CALL MODUS[CRSR/UP*3
      ]":END
60861 PROC PAINT
60862 EXEC MUHI
60863 PRINT"[CRSR/DOWN]LIN  PAINT   X,
      Y,1"
60864 PRINT"----+-----+---+---+-"
60865 PRINT"     [YEL]PAINT[BLK]":PRINT"
      [CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]
      ":END
60866 PROC DRAW
60867 EXEC MUHI
60868 PRINT"[CRSR/DOWN]     1/6       9=
       AFBRYD"
60869 PRINT"       [SHIFT/B]"
60870 PRINT"3/8 [SHIFT/**2][SHIFT/+][SHI
      FT/**2] 0/5  N=  -TEGN/TEGN"
60871 PRINT"       [SHIFT/-]"
60872 PRINT"     2/7       4= OP"
60873 PRINT"[CRSR/DOWN]LIN  DRAW";CHR$(3
      4);"NNNN";CHR$(34);"  X, Y,1"
60874 PRINT"----+----+-----+--+--+-"
60875 PRINT"     [YEL]DRAW[BLK]";CHR$(34
      ):PRINT"[CRSR/DOWN*2]CALL MODUS[CR
      SR/UP*3]":END
60876 PROC ROT
60877 EXEC MUHI
60878 PRINT"[CRSR/DOWN]LIN  ROT V,S   [Y
      EL]V[BLK]INKEL=0-7  [YEL]S[BLK]T0R
      ELSE=1-?"
60879 PRINT"----+---+-+-"
60880 PRINT"     [YEL]ROT ,[BLK]":PRINT"
      [CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]
      ":END
60881 PROC TEXT
60882 EXEC MUHI
60883 PRINT"[CRSR/DOWN]LIN  TEXT   X,  Y
      ,";CHR$(34);"TEKST";CHR$(34);",1,S
      ,B"
60884 PRINT"----+----+---+---+-------+-+
      -+-"
60885 PRINT"     [YEL]TEXT[BLK]":PRINT"[
      CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]"
      :END
60886 PROC MOVE
60887 EXEC MUHI
60888 PRINT"[CRSR/DOWN]LIN  MOVE R,C,B,L
      ,DR,DC"
60889 PRINT"----+----+-+-+-+-+--+--"
60890 PRINT"     [YEL]MOVE[BLK]":PRINT"[
      CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]"
      :END
60891 PROC SCROL
60892 EXEC MUHI
60893 PRINT"[CRSR/DOWN]LIN  RETNING W EL
      . B SR,SC,EC,R"
60894 PRINT"----+-------+-+--+--+--+--"
60895 PRINT"     [YEL]RIGHT[BLK]":PRINT"
      [CRSR/DOWN*2]CALL MODUS[CRSR/UP*3]
      ":END
60896 PROC SCRSV
60897 EXEC MUHI
60898 PRINT"[CRSR/DOWN]LIN  SCRSV 2,8,2,
      ";CHR$(34);"NAVN,S,W";CHR$(34)
60899 PRINT"----+-----+-+-+-+-----+-+-+"
60900 PRINT"     [YEL]SCRSV 2,8,2,";CHR$
      (34);"    ,S,W";CHR$(34);"[BLK]"
60901 PRINT"[CRSR/DOWN*2]CALL MODUS[CRSR
      /UP*3]":END
60902 PROC READSCRSV
60903 EXEC MUHI
60904 PRINT"[CRSR/DOWN]LIN  SCRSV 2,8,2,
      ";CHR$(34);"NAVN,S,R";CHR$(34)
60905 PRINT"----+-----+-+-+-+-----+-+-+"
60906 PRINT"     [YEL]SCRSV 2,8,2,";CHR$
      (34);"    ,S,R";CHR$(34);"[BLK]"
60907 PRINT"[CRSR/DOWN*2]CALL MODUS[CRSR
      /UP*3]":END
60908 PROC MENU
60909 PRINT"[CLR/HOME]":N%=20:FORI=1TON%
      :READA$(I):IFA$(I)="+"THENCALL LIS
      TEMENU
60910 PRINTI;". ";A$(I)
60911 NEXT
60912 PROC LISTEMENU
60913 PRINT"[CRSR/DOWN]V[LG ET NUMMER";
60914 WAIT203,63:GETN%:PRINTN%:IFN%=7THE
      NPOKE198,0:END
60915 LOADA$(N%),8
60916 DATAPROGRAMMERING,SPRITES,KARAKTER
      ER,GRAFIK,LYD,ANDET,STOP,+
60917 END
```

KONTROLSUM FOR GRAFIK

```
60800 200    60801 105    60802 175
60803 182    60804 142    60805 29
60806 7      60807 146    60808 93
60809 16     60810 216    60811 40
60812 81     60813 68     60814 145
60815 160    60816 189    60817 173
60818 223    60819 80     60820 52
60821 195    60822 59     60823 232
60824 236    60825 153    60826 1
60827 78     60828 216    60829 152
60830 51     60831 234    60832 24
60833 122    60834 53     60835 19
60836 245    60837 0      60838 83
60839 128    60840 152    60841 33
60842 234    60843 224    60844 54
60845 239    60846 27     60847 234
60848 10     60849 255    60850 84
60851 28     60852 234    60853 190
60854 144    60855 61     60856 91
60857 234    60858 138    60859 234
60860 199    60861 142    60862 234
60863 103    60864 89     60865 145
60866 35     60867 234    60868 82
60869 95     60870 233    60871 122
60872 37     60873 131    60874 222
60875 253    60876 34     60877 234
60878 189    60879 243    60880 86
60881 136    60882 234    60883 70
60884 66     60885 90     60886 40
60887 234    60888 132    60889 218
60890 76     60891 56     60892 234
60893 153    60894 99     60895 147
60896 134    60897 234    60898 10
60899 140    60900 112    60901 48
60902 13     60903 234    60904 5
60905 140    60906 107    60907 48
60908 234    60909 70     60910 233
60911 130    60912 202    60913 26
60914 103    60915 32     60916 33
60917 128
```

KARAKTERER

```
60700 POKE45,PEEK(174):POKE46,PEEK(175):
      CLR
60701 REM***TEGN***
60702 MEM
60703 DESIGN2,$E000+27*8
60704 @ ..BBBBBB
60705 @ .B.B....
60706 @ B..B....
60707 @ BBBBBB..
60708 @ B..B....
60709 @ B..B....
60710 @ B..BBBBB
60711 @ ........
60712 DESIGN2,$E000+29*8
60713 @ ..BBB...
60714 @ ..B.B...
60715 @ ..BBB...
60716 @ .B...B..
60717 @ BBBBBBB.
60718 @ BB...BB.
60719 @ BB...BB.
60720 @ ........
60721 REM***TEGNDEFINERING***
60722 KEY1,"LIST 0000-[CRSR/LEFT*5]"
60723 KEY2,"FIND NAVN[CRSR/LEFT*4]"
60724 KEY3,"DISPLAY"+CHR$(13)
60725 KEY4,"TRACE10"+CHR$(13)
60726 KEY5,"CALL KA"+CHR$(13)
60727 KEY6,"EXEC LABEL[CRSR/LEFT*5]"
60728 KEY7,"PRINT$0000[CRSR/LEFT*4]"
60729 KEY8,"?%00000000"
60730 KEY9,"DUMP"+CHR$(13)
60731 KEY11,"TRACE0"+CHR$(13)
60732 KEY13,"PROC NAVN[CRSR/LEFT*4]"
60733 KEY15,"CALL NY LINJENR"+CHR$(13)
60734 KEY10,"MERGE PR,1[CRSR/LEFT*4]"
60735 KEY12,"RETRACE"+CHR$(13)
60736 KEY14,"END PROC"+CHR$(13)
60737 KEY16,"?P[SHIFT/E](00000)[CRSR/LEF
      T*5]"
60738 COLOUR5,5:PRINT"[CLR/HOME][CRSR/DO
      WN]DU HAR FØLGENDE I F 1-16:[CRSR/
      DOWN]":DISPLAY
60739 FLASH7,10
60740 PRINT"[CRSR/DOWN*2]KLAR S] [RVS/ON
      ][YEL]TAST[RVS/OFF][BLK]"
60741 WAIT203,63
60742 PRINT"[CLR/HOME]":PRINT AT(8,12)"K
      LAR TIL TEGNDEFINERING":FLASH0,10:
      BFLASH10,2,6
60743 PAUSE2:BFLASH0:OFF:COLOUR6,14
60744 DELAY120:CALL SPRITES
60745 PROC NY LINJENR
60746 PRINT"[CRSR/DOWN]NYE LINJENUMRE NØ
      DVENDIGE HVIS [RVS/ON]RENUMBER[RVS
      /OFF]"
60747 PRINT"BRUGES"
60748 PRINT"[CRSR/DOWN]ON FINDES I LINJE
      RNE:":PRINT"FINDON"
60749 PRINT"[CRSR/DOWN*3]GOTO FINDES I L
      INJERNE:":PRINT"FINDGOTO"
60750 PRINT"[CRSR/DOWN*3]GOSUB FINDES I
      LINJERNE:":PRINT"FINDGOSUB"
60751 PRINT"[CRSR/DOWN*3]GET FINDES I LI
      NJERNE:":PRINT"FINDGET"
60752 PRINT"[CRSR/DOWN*3]RENUMBER0,10"
60753 END
60754 PROC SPRITES
60755 TRACE0:COLOUR14,6
60756 PROC KA
60757 PRINT"[CLR/HOME][CRSR/DOWN*12]DU K
      AN NY-DEFINERE";
60758 PRINT" ALLE TEGN OG          BOGSTA
      VER UDEN PROBLEMER"
60759 PRINT"[CRSR/DOWN]HVIS F[RDIG S] SK
      RIV 'STOP'"
60760 PROC TEGN
60761 INPUT"[CRSR/DOWN*3]TEGN";A$
60762 IFA$="STOP"THENCALL MENU
60763 TE=PEEK(52990)
60764 PROC RET
60765 PRINT"[CRSR/DOWN]RET OG TRYK [RVS/
      ON][YEL]RETURN[RVS/OFF][BLK]"
60766 DESIGN2,$E000+8*TE
60767 @BBBBBBBB
60768 @BBBBBBBB
60769 @........
60770 @BBBBBBBB
60771 @BBBBBBBB
60772 @........
60773 @BBBBBBBB
60774 @BBBBBBBB
60775 PRINT"[CRSR/DOWN*11]CALL KA"
60776 PRINT"CALL MENU[CRSR/UP*13]":LIST6
      0767-60774:END
60777 END PROC
60778 PROC MENU
60779 PRINT"[CLR/HOME]":N%=20:FORI=1TON%
      :READA$(I):IFA$(I)="+"THENCALL LIS
      TEMENU
60780 PRINTI;". ";A$(I)
60781 NEXT
60782 PROC LISTEMENU
60783 PRINT"[CRSR/DOWN]V[LG ET NUMMER";
60784 WAIT203,63:GETN%:PRINTN%:IFN%=7THE
      NPOKE198,0:END
60785 LOADA$(N%),8
60786 DATAPROGRAMMERING,SPRITES,KARAKTER
      ER,GRAFIK,LYD,ANDET,STOP,+
60787 END PROC
60788 END
```

KONTROLSUM FOR KARAKTERER

60700 200	60701 185	60702 222
60703 20	60704 72	60705 248
60706 248	60707 72	60708 248
60709 248	60710 72	60711 208
60712 22	60713 12	60714 248
60715 12	60716 248	60717 92
60718 32	60719 32	60720 208
60721 148	60722 211	60723 98
60724 183	60725 114	60726 107
60727 52	60728 100	60729 100
60730 221	60731 111	60732 167
60733 233	60734 172	60735 215
60736 254	60737 85	60738 194
60739 71	60740 39	60741 188
60742 212	60743 108	60744 76
60745 104	60746 85	60747 165
60748 83	60749 173	60750 59
60751 251	60752 45	60753 128
60754 223	60755 92	60756 65
60757 64	60758 0	60759 66
60760 227	60761 202	60762 190
60763 103	60764 160	60765 203
60766 68	60767 80	60768 80
60769 176	60770 80	60771 80
60772 176	60773 80	60774 80
60775 96	60776 247	60777 152
60778 234	60779 70	60780 233
60781 130	60782 202	60783 26
60784 103	60785 32	60786 33
60787 152	60788 128	

De brændes uden varme

EPROM'er er computerens langtidshukommelse. Hvad bruges de til? Hvordan virker de?

EPROM står for Erasable, Programable Read-Only-Memory, dvs. sletbar, programmerbar kun-læse hukommelse. For brugeren betyder det, at han kan bruge EPROM'en som en hukommelseschip, hvor data ikke forsvinder, når strømforsyningen bliver slukket, og hvor indholdet kan ændres efter behag.

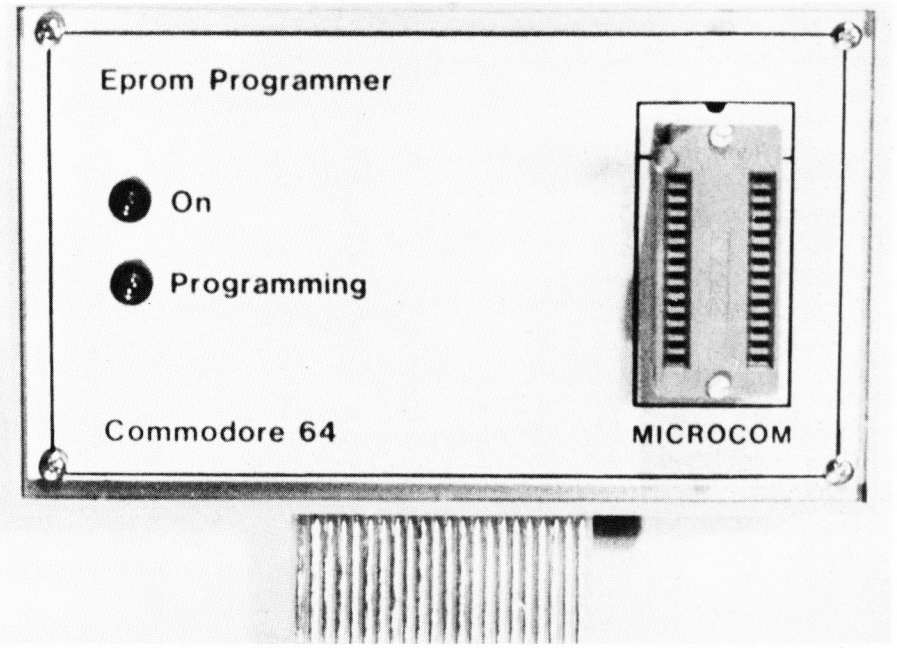
Man kan forestille sig oplagringen af data i en EPROM på den måde, at hver lagerplads, hvor der gemmes en bit, består af en lille celle, der kan fyldes op med elektriske ladninger. Når der slettes en bit svarer det til, at denne elektriske ladning fjernes fra cellen, og cellens tilstand svarer derefter til et logisk »1«.

Fylder man nu cellerne op med elektriske ladninger ved hjælp af strømforsyningen, ændrer deres elektriske potentiale sig, og de slettede bits bliver nu til indprogrammerede bits, svarende til logisk »0'er«.

For at de indprogrammerede bits ikke skal forsvinde, dvs. for at cellerne ikke aflades, er de beklædt med et isolerende lag, som ladningerne kun kan passere, i en bestemt retning.

Den spænding der skal til, for at presse ladningerne ind i en lagercelle, kaldes programmeringsspændingen. For de normale EPROM'er der fås i handelen, er denne spænding ca. 21 til 25 volt. Opladningen sker forholdsvis langsomt, og derfor skal programmeringsspændingen vare tilstrækkeligt længe til, at et passende antal ladninger kan trænge ind i cellen. I fagets jargon kaldes det at »brænde« EPROM'en.

Hvordan kan man så igen slette EPROM'en? For at forstå det, må man kende til egenskaberne af cellernes isolerende lag. Når dette lag udsættes for ultraviolet lys, dvs. lys med korte bølgelængder, bliver dette lag ledende i begge retninger, og ladningerne kan slippe ud af cellerne. Derfor er EPROM'ens plasthylster forsynet med et »vindue«. Gennem dette vindue kan UV-lys lukkes ind, når man vil slette EPROM'en. Sollys og lys fra almindelige glødelamper indeholder lidt UV-stråler. Derfor dækkes EPROM'ens vindue normalt til, således at EPROM'en ikke i utide får »hukommelsestab«, og glemmer sine data. Ønsker man at slette indholdet, er en højfjeldssol velegnet, men sollys kan også bruges. I så fald tager det nok flere timer, eller en hel dag, før du kan være sikker på, at hele indholdet er slettet. Efter sletningen er hver bit igen sat til logisk »1«. I en EPROM kan man ved »brænding« kun frembringe nuller. Det kan man til enhver tid gøre. Et-tallerne derimod kan kun produceres gennem sletning, hvor samtlige bits ændres.



En ny EPROM brænder vil være i handelen omkring 15/11 til en pris af kr. 1.495,00. Derudover vil hvert modul koste omkring kr. 200,00.

Programmeringen

Antallet af bits i EPROM'en fremgår af typebetegnelsen. En type 2716 indeholder f.eks. 16K bits, svarende til 2K bytes. En EPROM fungerer »byte-orienteret«, dvs., at man altid programmerer 8 bit ad gangen. Hver byte har sin egen adresse. Ved programmeringen får EPROM'en via adresseledningerne at vide, hvilke bytes der skal programmeres. Derefter skiftes den ved hjælp af skrive/læse ledningerne til stillingen »skrive«. Dataledningerne fortæller derefter hver celle, om indholdet skal ændres til logisk »0«, eller blive ved med at være logisk »1«.

Når adresse- og dataledningerne er tilkoblet, tilsluttes programmeringsspændingen, og EPROM'en skifter til stillingen »programmering«.

Data-, adresse- og programmeringsspændingerne skal holde sig stabile i 50 millisekunder, for at de 8 bit, hver byte består af, kan blive »brændt«. Derefter afbrydes programmeringsspændingen, og EPROM'en sættes i stillingen »læse« for at teste den nybrændte byte, dvs. sammenligne den med originalens bit-mønster. Stemmer de sammen, går man videre til næste byte, der brændes efter samme procedure.

Programanmeldelser

Spitfire Ace

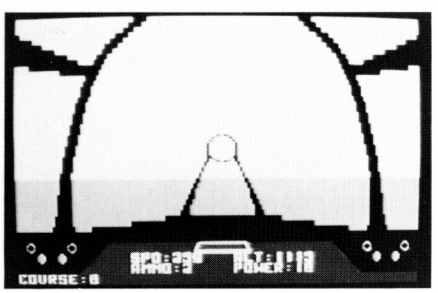
Vi er tilbage i anden verdenskrig, og du er pilot i en Spitfire i kamp mod tyskerne. Skærmen viser udsynet fra cockpittet med nogle af de vigtigste instrumentoplysninger: hastighed, kurs, højde, motorkraft og ammunition. Desuden er der et lille spejl, så du om dagen kan se, hvad der foregår bag dig.

Spillet kræver mindst ét joystick. Det ene til at styre og affyre skud med, og det andet til bl.a. at regulere motorkraften med. Har du kun et, kan de sidste funktioner styres fra tastaturet.

Spillet er ikke så ligetil at mestre. Du kan (bortset fra sværhedsgrader) vælge mellem 14 forskellige scener fra anden verdenskrig, f.eks. Dunkrik, London, D-dag m.v. Computeren stiller dig forskellige opgaver, som du skal løse. Du kan risikere både dag- og natflyvninger og du har ikke megen ammunition til at løse din mission.

Det er ret svært at ramme fjenden, og han har desuden den kedelige vane, at han også beskyder dig – og han skyder godt!

Det er også noget af en udfordring at lære at flyve ordentligt. Der er forskellige undvigemanøvrer, der er gode at kunne, og dem lærer man ikke på to minutter.



Konklusion
Da Spitfire Ace er et spil med mange forskellige scener, kan det holde dig beskæftiget i lang tid, hvis du vil mestre det til bunds. I længden bliver det dog lidt ensformigt at arbejde med.

Både grafik og lyd er udmærket, men spillet er nok for kompliceret til mindre børn.

Brugsanvisningen er desværre kun på engelsk, og kunne være bedre.

Prisen, der ligger på kr. 198,00 for kassetteversionen og kr. 278,00 for disketteversionen, må siges at stå i rimeligt forhold til kvaliteten.

Vurdering:	*Meget god*	*God*	*Rimelig god*	*Ringe*
Instruktion	☐	☒	☐	☐
Præstation	☐	☒	☐	☐
Betjeningsvejledning	☐	☐	☒	☐
Grafik	☐	☒	☐	☐
Lyd	☐	☒	☐	☐
Variation	☒	☐	☐	☐
Fængslende	☐	☒	☐	☐
Pris/kvalitetsforhold	☐	☒	☐	☐

Turbo 64

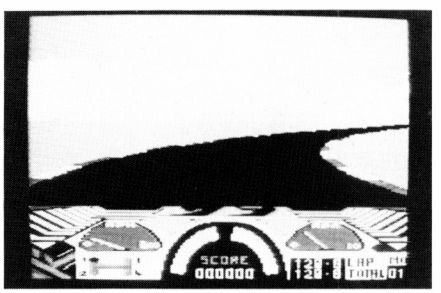
I Turbo 64 sidder du bag rattet i et formel 1 racerløb. Din bil kan køre stærkt, og din modstander er tiden. Ved første øjekast et spil af de mere traditionelle.

Når først du kommer i gang, må du erkende, at Turbo 64 alligevl er noget ud over det sædvanlige. Med en grafik, der er helt i klasse for sig, drøner du ad snoede veje gennem et meget realistisk bjerglandskab. Lyden er også helt i top, så det er meget svært, ikke at blive grebet af spillet.

Spillet rummer følgende faciliteter. Op til ni spillere kan konkurrere ad gangen. Du kan vælge mellem 2 forskellige »baner«, og du kan vælge, hvor mange omgange banen skal gennemkøres. Du kan derudover vælge mellem automatisk gearskift og manuelt gearskift. I sidstnævnte tilfælde virker joystickets fireknap som kobling, og de enkelte gear (1-4) vælges med joysticket. De har samme placering på joysticket som på et »rigtigt« gear.

Farten regulerer du ved at skubbe joysticket enten lige frem eller tilbage, og retningen dirigeres selvfølgelig ved, at skubbe til enten højre eller venstre. Bilen retter selv op, når joysticket står i midten. Det hele lyder rigtig nemt – og det er det også, så længe bilen snegler sig af sted. Når den begynder at køre stærkere, får du en anden opfattelse – og så bliver du glad for, at usynlige hænder sætter bilen uskadt tilbage på vejen, når du har smadret den!!!

Der gøres mange ting for at komme piratkopiering til livs. Her har man forsøgt en pudsig ting, idet man beder om at få indtastet fire farver (ikke de samme hver gang) i henhold til et farvekodekort, der følger med. Det sikrer selvfølgelig ingenting, men det gør det mere besværligt, for man kan ikke ved en fotokopi skelne farverne. Jeg vil ønske, man en dag finder et effektivt sikrimgssytem.



Konklusion
Turbo 64 er et fantastisk spændende og realistisk racerspil ud over det sædvanlige. Brugsanvisningen er kun på engelsk, men da instruktionerne er få, har det ikke den store betydning i dette tilfælde.

Absolut anbefalelsesværdigt.

Vurdering:	*Meget god*	*God*	*Rimelig god*	*Ringe*
Instruktion	☐	☒	☐	☐
Præstation	☐	☒	☐	☐
Betjeningsvejledning	☒	☐	☐	☐
Grafik	☒	☐	☐	☐
Lyd	☒	☐	☐	☐
Variation	☐	☒	☐	☐
Fængslende	☒	☐	☐	☐
Pris/kvalitetsforhold	☒	☐	☐	☐

Hjælpekommandoer til VIC-20

Hjælpeprogrammer til VIC-20 og C-64 er altid en god investering, hvis man vil programmere selv. Der kan hurtigt spares flere timers arbejde, hvis man har de rigtige hjælpeprogrammer til sin rådighed.

Hjælpeprogrammer til VIC-20 findes der ikke mange af, men Commodores eget (VIC-1212) dækker stort set de mest almindelige behov.

Programmet ligger i et programmodul, og kan således ikke være nemmere at have med at gøre. Hjælpeprogrammet giver eksempelvis følgende nye kommandoer:

RENUMBER tildeler nye linienumre med et vilkårligt spring mellem de enkelte linienumre. Funktionen er 100%, hvilket vil sige, at linienumre, der følger f.eks. GOTO-kommandoen, også ændres automatisk.

MERGE-funktionen er også bedre end man ser det mange gange, idet et program, der blandes med et andet, ikke lægges i forlængelse af det første, men virkelig flettes sammen i overensstemmelse med linienumrene.

Med kommandoen FIND kan man i programmet finde en bestemt kommando, udtryk eller streng, og med kommandoen CHANGE kan man ændre samme.

DELETE-kommandoen vil slette en blok af linienumre, og AUTO vil automatisk skrive linienumre, når man programmere.

Med TRACE, STEP og OFF kommandoerne kan man følge med i programafviklingen, i et lille »vindue« øverst til højre på skærmen.

Desuden kan man definere funktionstasterne, få udskrevet variabler, få vist »Syntax error« fejl, » liste baglæns« med mere.

For den seriøse programmør, er her stillet et virkelig godt hjælpemiddel til rådighed.

Pris kr. 115,-.

For VIC-20:

Sargon II Chess

Der dukker ikke megen ny software til VIC-20 frem på markedet. Det er naturligvis beklageligt, men blandt den software, der findes, er der en del, der fortjener at blive trukket frem i lyset.

Skakspillet Sargon II hører til blandt disse, og skønt det ikke er på højde med de bedste til Commodore 64, er der alligevel her tale om et program, der kan give megen god underholdning og en god udfordring for de fleste.

Spillet findes på et indstiksmodul, og kræver således ikke nogen udvidelse af VIC-20's hukommelse. Grafikken er udmærket, selv om jeg personlig foretrækker et »rigtigt« skakbræt stående ved siden af computeren.

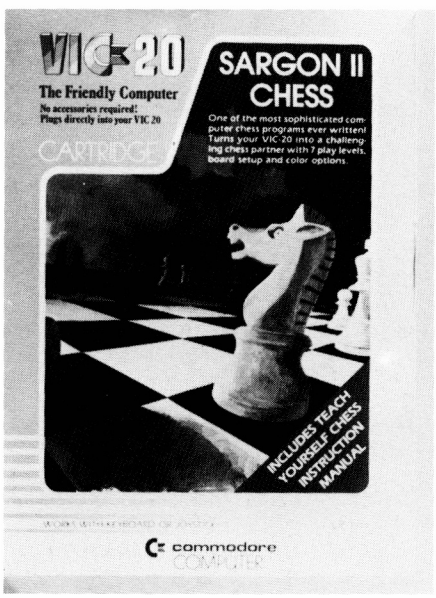
Man kan vælge mellem 7 forskellige spilleniveauer. På det laveste trækker computeren med det samme, og på det højeste bruger den op til 4 timers »betænkningstid« mellem hvert træk. Man kan som hovedregel gå ud fra, at sværhedsgradens nummer angiver det antal træk, som Sargon II »tænker fremad«.

Man kan flytte brikkerne ved hjælp af tastaturet, eller man kan flytte dem ved hjælp af et joystick. Er man på »bar bund«, kan man bede om hjælp fra computeren til at foretage sit næste træk.

Man kan så følge rådet eller lade være. Man kan også gå tilbage i spillet og gøre trækkene om, ligesom man kan starte fra en hvilken som helst opstilling. Man har altså her mulighed for at løse skakopgaver ved hjælp af computeren.

Det skal også fremhæves, at brugervejledningen er på dansk og langt over middel. Ud over selve vejledningen findes en kortfattet introduktion til selve skakspillet, så selv begynderen kan være med her.

Alt i alt må jeg sige, at Sargon II til kr. 95,00 er et af de bedre spil til VIC-20.



Vurdering:	*Meget god*	*God*	*Rimelig god*	*Ringe*
Instruktion	[X]	[ ]	[ ]	[ ]
Præstation	[ ]	[X]	[ ]	[ ]
Betjeningsvejledning	[X]	[ ]	[ ]	[ ]
Grafik	[ ]	[X]	[ ]	[ ]
Lyd	[ ]	[ ]	[X]	[ ]
Variation	[X]	[ ]	[ ]	[ ]
Fængslende	[X]	[ ]	[ ]	[ ]
Pris/kvalitetsforhold	[X]	[ ]	[ ]	[ ]

EXTENDED Basic

Af Flemming Lerbæk

Næsten alle vil på et eller andet tidspunkt gerne udnytte deres 64'er til mere end legetøj eller »spillemaskine«. Og når det bliver aktuelt, melder der sig et problem for de fleste. Commodores noget begrænsede Basic-version. Det har et utal af software-fabrikanter forlængst erkendt. Vi skal her se på et af dem, nemlig Extended Basic fra Bug software house.

»Lav din Commodore 64 om til en kraftfuld maskine«, står der udenpå den lille pap-boks båndet kommer i. Og det lyder jo lovende. Men kan det holde for en nærmere undersøgelse? Det prøver vi at finde ud af her.

Mange computerfans er i stand til at læse og forstå engelsk. Hele atmosfæren omkring hjemmecomputere, og naturligvis også for de store mini-og macrocomputere, er gennemsyret af engelsk/amerikansk sprogbrug. Men det er vel et spørgsmål om man kan forlange, at en nybagt hjemmecomputerejer skal kunne engelsk. Derfor finder jeg det som en utilgivelig fejl, hvis en software sælges uden dansk brugervejledning. Og da især, hvis man

som her, betaler næsten 350 kroner for sin cassette. Extended Basic er kun forsynet med en engelsk manual.

27 nye kommandoer

Når man til daglig styrer sin 64'er gennem computerjunglens mange snarer, kunne man godt tænke sig, at det gik lidt hurtigere. Al den PEEK'en og POKE'en er uendelig langsom. I mange situationer savnes netop denne eller hin kommando, som man har set andre computeres programlister være udstyret med. Med Extended Basic får man 27 nye brigger at flytte rundt med. De er fornuftigt inddelt i afsnit.

Alle kommandoer er til at bruge sammen med Commodores eget basic. Men man skal være opmærksom på, at hvis man gemmer sit program væk på disk eller bånd, skal man pånyindlade Extended Basic, hvis der har været slukket for maskinen i mellemtiden. Ellers får man en række uforståelige tal-bogstaver på de steder, hvor de »nye« kommandoer stod, når der genindlæses fra disken eller båndet.

Og nu vi er ved indlæsningen. Programmet kommer som nævnt på bånd. Og det er dræbende langsomt indlæst. Først 3½ minut hvor der indlæses en del af programmet. Så skal man gennem en »pinefuld« langsom præsentation af kommandoerne. Fire undergrupper er det inddelt i, og dem præsenteres man for, *hver* gang der indlæses. Hvorfor?

Det ser meget godt ud første gang. Men tiende gang? Nej, vel? Da vil man bare i gang – hurtigt. Fem minutter og ca. 40 sekunder tog det mig at indlæse programmet fra cassetten.

De 27 kommandoer man præsenteres for – hver gang – er:

GRAFIK:
GRn, MODEn,COLOURn1
(,n2)(,n3)(,n4)(,n5),USEn,SETSC,
SCOLORn1,n2,PLOTn1,n2,DRAWn1,
n2TOn3,n4TOn5,n6,TYPEn1,n2,n3,n4
,string.

SPRITES:
SHADEn, SPRITEn,colour,priority,
expansion,shape,SPRITEn,-1,
SMOVEn,x,y,COLLIDEn,C%,
CSPRITEc1,c2

LYD:
SOUNDn,a,d,s,r,w(,p),sr, PLAY T,
»xxxxx«,»xxxx«,»xxxx«,K,VOLume,
filter frequency,filter mode,FILTER
resonance,external,f3,f2,f1

ANDRE KOMMANDOER:
STICKn,a%,PADDLEn,a%,PENX%,
Y%,POINTx,y,a%,OLD,NUMBERn1,
n2,FIND/streng/(,n1-n2).

Herlighederne:

Som de fleste vil kunne genkende blot ved at læse de 27 kommandoer, er der tale om netop de fleste af de lækkerier, der kan forsøde en sur tilværelse for programmøren derhjemme i dagligstuen.

Når jeg før oplyste, at Extended Basic kan bruges sammen med Commodore basic, er det en sandhed med modifikationer. En meget alvorlig *fejl* ved det udvidede basic er nemlig, at CBM 64 melder SYNTAX ERROR ved enhver brug af Commodore-kommandoen OPEN.

Jeg vil tro, at denne alvorlige brist hænger sammen med Extended Basic's måde at identificere sine kommando-ord på. Der findes en OLD-kommando, der redder et program, som man er »kommet til« at fjerne med NEW. Formentlig ser den på de første par bogstaver, og beslutter sig så for, om den kan godkende eller vil forkaste kommandoen. Ihvertfald godkender den uden tøven følgende pussighed:

200 OLDN3,6:CMD3:LIST

Men kun som en OLD kommando og svarer med READY.

Det er altså udelukket at liste programmer på en plotter/printer. I det følgende skal jeg vise nogle få at 27 kommandoer og deres lette anvendelse.

GR2 lader os vælge een af tre mulige skærm-tilstande (modes). COLOUR farvelægger alt på skærmen, og den kan bruges både i højopløsnings- og multicolour tilstandene. USE sætter den farve hvormed der plottes eller tegnes. Og DRAW er en hurtig og præcis tegnekommando.

En sprite defineres med SHAPE. I den engelske manual er der iøvrigt her – som overalt i bogen – sparet grundigt på eksemplerne. Der er dog vist en spritedefinering. Men den er forkert. Der er kun angivet 23 enheder mellem »anførselstegnene« der SKAL være 24 for at tingene lykkes.

Med SPRITE kaldes den enkelte sprite frem i den ønskede størrelse, farve osv.

Lydmuligheder

SOUND kommandoen giver adgang til at styre såvel hvilken af de tre stemmer, der skal i brug, attack, decay, sustain og releasse, samt waveform, eventuelt pulse og endelig synkroniseringen. Læs mere om disse begreber i CBM manualen side 81-85.

Lyd er generelt svært for de fleste. Dette program letter meget på brugen. Man slipper for at holde rede på en masse POKE's osv. Men teorien bag lydfrembringelsen slipper man ikke for at skulle forstå.

Der er desuden en meget mærkelig fejl i min version af Extended Bacis. Jeg kan, som det ses af programlisten, kun have én række »noder« i hver PLAY-linie. Ifølge den – dårlige – manual skal der kunne stå mindst tre, og CBM 64 skal vel at mærke spille dem, men det gjorde den ikke. Et par timers eksperimenteren resulterede i et par tomme strenge omkring den, som programmet »gider« læse.

Andre kommandoer

I afsnittet om sprites blev ordren STICK brugt. Den leverer værdierne for joystick-bevægelserne.

På lignende måde aftastes med PADDLE og PEN kommandoerne. Point giver dig farverne under et givet punkt på skærmen.

Endelig er der de uundværlige OLD, NUMBER og FIND kommandoer. Enhver programmør vil få brug for OLD kommandoen på et tidspunkt.

NUMBER er i virkeligheden en ægte RENUMBER. Ægte vil sige, at alle adresser laves om f.eks. på GOSUBS, GOTO, ON og naturligvis på programlinierne.

FIND er også overordentlig nyttig. Her kan søges på hele programmet eller dele af det.

En anden behagelig overraskelse var den mulighed for at forstørre bogstaver, der bydes på i forbindelse med den grafiske skærm. Modsat f.eks. i SIMON BASIC, kan man her se bogstaverne udviddet i begge retninger, hvis det ønskes.

Og endelig til sidst et lille råd til kommende eller nuværende brugere af programhjælpen som Extended Basic byder på. Har man fået sit program skrevet, så det ikke af sig selv kommer tilbage til tekst-mode, kan man blot skrive en ugyldig ordre f.eks. UPS og den grafiske skærm afløses af tekstskærmen og udskriften »SYNTAX ERROR«.

Konklusion

Extended Basic vil for mange være et godt køb. Men det lider af alvorlige mangler og fejl, som f.eks. den svigtende mulighed for at bruge basicordren OPEN sammen med programmet. Man får 27 velvalgte nye kommandoer for de ca. 350 kroner selv om jeg savner grafiske kommandoer som CIRCLE, PAINT, ROTATE. Som programmør savnes TRACE, MERGE, KEY og AUTO.

HVEM GI'R DIG FREM

TIDENS SOFTWARE?

Commodore!
Billedet viser bare et lille udpluk af programmer til Commodore 64, verdens mest populære hjemmecomputer.

Commodore er faktisk det computerfirma i verden, der har udviklet flest programmer overhovedet, ca. 2.000 ialt - foreløbig!

I realiteten findes der et program til enhver opgave, du kan tænke dig - plus en hel del andre. Og der kommer stadig nye til - der står Commodore på fremtidens software.

Commodore

Fordi fremtiden forlængst er begyndt.

International Commodore messe i Frankfurt

Commodores udstillinger er næsten uden undtagelse altid en stor publikumssucces, og messen i Frankfurt var da heller ingen undtagelse, den fandt sted i Frankfurt i dagene 5/9 – 8/8 1984. I løbet af de tre dage messen varede, blev den besøgt af ca. 20000 mennesker, hvilket var omkring 50% mere end man havde regnet med.

Småt med nyheder

Hvis man var kommet til Frankfurt for at finde nyheder, blev man sørgeligt skuffet, for der var så at sige ingen af betydning. På Commodores egen stand kunne man bese de nye maskiner 116, C-16 og Plus/4.

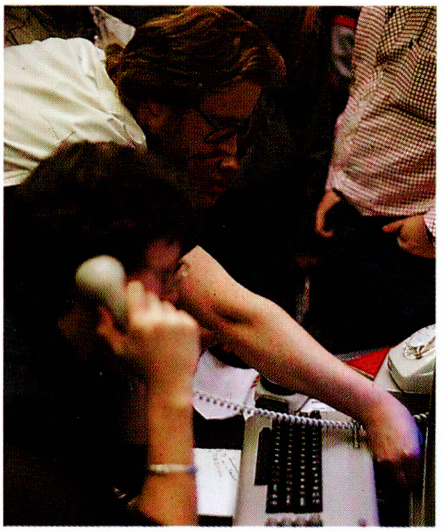
Førstnævnte er kun produceret i 24000 eksemplarer, og disse vil blive solgt i Tyskland, idet produktionen af denne maskine er stoppet. C-16 var i brugsfærdig stand, hvorimod den indbyggede software til Plus/4 endnu ikke var færdig.

Workshops var en succes

Der var i hallen indrettet 6 workshops, hvor man dagen igennem kunne blive orienteret om forskellige programmeringsproblemer, samt få demonstreret de nyheder der fandtes. Jeg overværede to af foredragene, et om musikprogrammering af Commodore 64 og et, hvor Commodore-guruen Jim Butterfield for en tæt pakket tilhørerskare fortalte om, og demonstrerede de nye maskiner Plus/4 og C-16. Det skinnede igennem, at selv Jim Butterfields kendskab til de nye maskiner, endnu kun var ret overfladisk. Foredragene var dog udmærkede, men var til en vis grad hæmmet af et højt støjniveau på grund af en dårlig lydafskærmning. Det var synd for både tilhørere og foredragsholdere.

Bedre musikudnyttelse

For de musikinteresserede var der dog lidt nyt at hente. Der blev præsenteret flere keyboards, der direkte kunne tilsluttes computeren. Det er unægtelig mere tiltalende at spille på rigtige tangenter end på et skrivemaskinetastatur. Det medfølgende software til disse keyboards gav også spændende muligheder for at udnytte computerens fremragende lydmuligheder fuldt ud.



Øvrige indtryk

Programmeringssproget LOGO, der nu findes på diskette, blev præsenteret, hvorimod det danskudviklede COMAL 80 modul, der på alle punkter er LOGO langt overlegen, ikke var at finde nogetsteds.

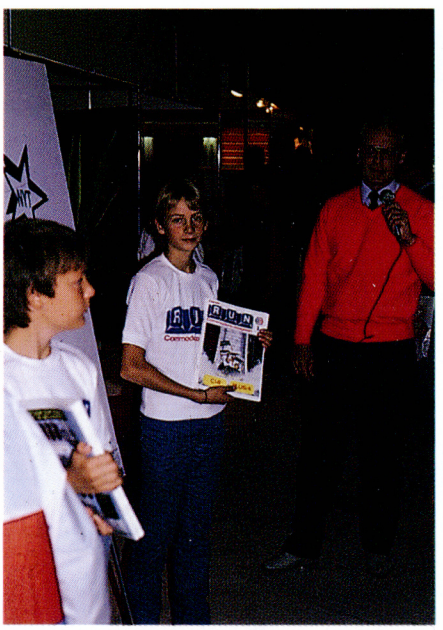
Som nævnt var det også småt med nyheder på softwaresiden. På spillesiden var der ingen programmer, jeg hæftede mig særligt ved. Der var nye programmer, men kun variationer over eksisterende. Er man mon ved at nå grænsen for de optimale spillemuligheder?

Alle de tyske computerblade var repræsenteret, og her hæftede man sig især ved den kendsgerning, at næsten alle havde kastet sig ud i lanceringen af software.

Man havde placeret den nye Plus/4 på den »professionelle« afdeling af Commodores stand. Det siger lidt om, hvilket publikum man sigter efter med den nye Plus/4. Det var da også her, der var størst publikumsinteresse.

Telefonmodem havde stor interesse blandt det unge publikum. Der var en speciel stand, hvor man i praksis selv kunne prøve at kalde databaser via telefonnettet. Der ligger ganske givet en fremtid på dette område.

Hvis man var kommet til Frankfurt for at finde nyheder, var helhedsindtrykket alt i alt ret skuffende.



HI-messe

I midten af september, blev der holdt en stor messe i Herning, hvor bl.a. fandtes en stor afdeling for kontor og data.

Commodore var repræsenteret med en stor stand og vort nye blad blev præsenteret, og der blev udloddet flere gevinster.

For Commodore-fans var der imidlertid ikke noget nyt at hente på Commodores stand. Alle de nye maskiner var ganske vist udstillet, men låst inde.

Tilstrømningen til messen var stor, og Commodore samlede alligevel stor interesse, selv om mange gik forgæves efter nyhederne.

Jim Butterfiels anmelder for RUN:

Vi præsenterer Plus/4 og C-16

Den kendte Commodore ekspert, forfatteren Jim Butterfield, giver her en oversigt over, de to nye computere på Commodore's leveringsprogram: Plus/4 og C-16. Her kan du læse, hvad de kan.

Her er de så: Commodore's to nye hjemmecomputere – Plus/4 og dens lillebror, C-16. De to modeller ligner ikke hinanden ret meget: de ser ret forskellige ud, de har også forskellig hukommelseskapacitet, og Plus/4 har et kommunikationsinterface som C-16 ikke har, ellers arbejder de cirka på samme måde, bortset fra at Plus/4 er født med 4 indbyggede programpakker.

Udseende

Plus/4 har et moderne, strømliniet design. Den er ganske kompakt og kileformet, men tastaturet er alligevel i fuld størrelse, med cursor tasterne i den fikse rudefacon nederst til højre på tastaturet. Funktionstasterne sidder på en separat række øverst på tastaturenheden.

C-16 ligner mere VIC-20 eller C-64, men med mørkegråt kabinet. Cursortasterne sidder her i den øverste række. Funktionstasterne sidder samme sted som på VIC-20 og C-64.

Der er samme antal funktionstaster som på VIC-20 og C-64, men der er to ting der er anderledes. For det første er der en funktionstast markeret med *help*. Hvis du får en fejlmelding mens du kører et Basic program, behøver du kun at trykke på help-tasten for at se, hvilken linie der er fejl i, og hvor programmet gik i stå. For det andet er de fleste programmer, der er ved at blive udviklet til de to nye modeller, forberedt til help-tasten og giver gode råd og vejledning, når du bruger den.

Funktionstasterne er defineret på forhånd, f.eks. viser F3, når du trykker på den, disk directory på skærmen. Hver tast kan omdefineres med en ganske enkel kommando. Hvis du f.eks. skriver KEY 3, »MERRY XMAS«, vil denne tast blive printet hver gang du trykker F3 ind. Alle taster kan omdefineres i et program eller ved hjælp af direkte kommandoer.

Der er også en *reset* knap ved siden af on/off omskifteren, så du kan restarte maskinen uden at skulle afbryde strømmen.

Plus/4 leveres med indbygget software: Commodore's 3-Plus-1 pakke som indeholder integreret tekstbehandling, et regneark, databasehåndtering og grafik. F1 tasten aktiverer tekstbehandlingen, og derfra kan du kalde de øvrige programmer.

Der er også en cartridge port til at stikke programmoduler ind i. Den ligner C-64 modul-stik, men er ikke kompatibel med denne.

Programmering

Med hensyn til programmering er den største forskel mellem C-64/VIC-20 og de nye maskiner den, at sidstnævnte har en kraftigere Basic, der er udvidet med mange nye kommandoer. Der er strukturerede kommandoer der gør programløkker mere fleksible, og If..Then.. Else og Print Using kommandoer, så du hurtigere og nemmere får rapporter printet ud. Grafik kommandoerne Box, Circle, Paint, Draw og Color gør det legende let at fremtrylle diagrammer og kurver.

Hvis du har lyst til at skrive dine egne programmer, har du mange editeringskommandoer til din rådighed som Delete, Renumber og Auto for automatisk nummerering af linierne. Restore kommandoen giver en linie et bestemt linienummer tilbage.

Til fejlretning af programmerne er der en Trace-funktion, der kan kaldes med Tron-kommandoen, og en errortrap der kaldes med Trap-kommandoen.

Basic'en har masser af arbejdsareal at boltre sig i. Selv om Plus/4's RAM ikke er større end C-64'ens, (64K, selvfølgelig!), er det frie RAM-areal meget større. I stedet for en skrabet 38911 bytes workspace, har Plus/4 over 60K bytes frit RAM til Basic'en. Der er mange programmer der ikke udnytter de ekstra 20K, men de er nu meget rare at ha', når du skal bruge dem.

Begge modeller har en god maskinsprog monitor, TED-MON, indbygget. Indtast MONITOR fra Basic for at aktivere TED-MON. De kommandoer du kan bruge, er Assemble, Compare, Disassemble, Fill, Go, Hunt, Load, Memory, Registrers, Save, Transfer og Verify. Du kan også ændre hukommelses- og register display. Monitorprogrammet ligner mange af de eksisterende pakker for 64'eren, som f.eks. SUPERMON. Hvis du er vant til at arbejde i maskinsprog, er der en hel del, du vil nikke genkendende til.

Lige en bemærkning til maskinsproget: selv om mikroprocessorchippen nu hedder 7501, har den nøjagtig samme instruktioner, som de velkendte chips 6502 og 6510, og du vil finde de samme velkendte funktioner på plads, som JSR CHROUT (eller $FFD2) for output af et tegn, præcis som i de eksisterende maskiner.

Commodore 16 er en udmærket computer for begyndere, og den er fuldstændig kompatibel med Plus/4.

Arkitektur

De nye maskiner har en mere avanceret arkitektur. Det er den der gør, at der er mere RAM-kapacitet til rådighed end tidligere. Hemmeligheden ligger i processorchippens og memorybussens arkitektur. Memorysystemet ved hele tiden, om processoren er i gang med at hente en instruktion, eller med at udføre en instruktion (det er de to muligheder der er). Den ordner det på den måde, at instruktionen hentes fra en slags hukommelse, mens data hentes fra – eller afleveres til – en hel anden slags hukommelse, selv om de to hukommelses-accesoperationer bruger samme adresse!

I Commodore 64 blev trafikken styret af en »færdsels-betjent« i form af en programmerbar integreret kreds, en såkaldt PLA-(Programmable Logic Array) kreds. Den får anmodninger om adgang til en bestemt hukommelsesadresse, og på grundlag af hvor henvendelsen kommer fra, plus andre overvejelser afgør den, hvilken lagerchip der skal aktiveres. I de nye modeller, er denne funktion henlagt i en mere avanceret chip, kaldet TED, som ikke blot tager sig af lagerarkitekturen, men desuden passer de fleste in/output funktioner, timingen, skærmdisplayet og lyden.

Plus/4 har også en såkaldt ACIA-chip, som kan klare mere avancerede kommunikationsfunktioner end både VIC-20 og Commodore 64. Til gengæld har C-16 ingen kommunikationsmuligheder.

Ydre enheder

De nye maskiner har samme serielle bus som Commodore 64, og alle eksisterende serielle bus periferi-enheder – primært disk og printer – fungerer derfor uden ændring.

Der er interface til kassette, som i princippet arbejder på samme måde som tidligere Commodore modeller. I praksis er kassette interfacet dog ikke kompatibelt med tidligere modeller, da det er beregnet til en lavere læse/skrive hastighed. Stikket er også anderledes.

Der er porte til joysticks, men stikkene er ændret til små, runde konnektorer, s.k. skærmkonnektorer. De ligner stikkene til video og serieltbus interface, men de er mindre. Der er et lignende lille stik til kassettebåndoptageren.

Grunden til, at man er gået over til de nye stik er, at de ikke giver forstyrrelser i radio-frekvensområdet; derfor er det nu lettere for Commodore, at opnå typegodkendelse fra myndighederne. Der kommer sikkert snart overgangsstik.

Der er en hurtig diskettestation på vej; på udstillinger er den blevet vist med betegnelsen SFS 481, men når den kommer til Danmark, har den nok fået et andet navn og modelnummer.

Den nye station ser stort set ud som 1541'en, men data-overføringen sker meget hurtigere.

Commodore 16 er en god maskine for begyndere, og den er i høj grad kompitabel med Plus/4.

Plusser og minusser

De nye modeller har ikke alle de mange faciliteter Commodore 64 kunne tilbyde. Den fyldige lyd med Attack, Decay, Sustain og Release mulighederne i 64'eren er erstattet med 2 kanaler med firkantpulser i de nye maskiner. Sætter du pris på smuk, synthesizerlignende musik, vil du sikkert foretrække 64'eren.

Også »sprites«, de bevægelige figurer fra 64'eren, er væk. Derfor bliver det svært at overføre programmer med »skyderi« fra 64 til Plus/4 og C-16. Opløsningen er stadig god, og farveudvalget er større end nogensinde, men animering er en besværlig affære at programmere – figuren skal slettes et sted på skærmen, og genopbygges et andet sted.

En mere teknisk detalje, som du vil savne i de nye modeller, er muligheden for at udnytte ledige »huller« i hukommelsen. Basic'en i 64 levnede 20K frit memory-areal til behagelig afbenyttelse for den snedige programmør: til definering af nye skærmbilleder, tegnsæt, eller en alternativ Basic i RAM. Den slags bliver mere besværligt med de nye modeller – hele RAM'en er til rådighed for Basic'en, og det kræver en del arbejde, at inddrage noget af den.

Commodore kalder Plus/4 og C-16 for »produktivitets maskiner«; med udvidet Basic, indbygget software og brugervenlige funktioner, kan du klare opgaverne hurtigt og effektivt.

De er også gode til at lære programmering.

Når du vil ha' god lyd, og morsomme sprites, foretrækker du sikkert C-64. For mere seriøse databehandlingsopgaver, som erhvervsgrafik – f.eks. »lagkage-diagrammer« i op til 16 farver og 8 lysstyrker – er Plus/4 og C-16 sagen.

Hvad med kompatibiliteten med C-64? Den har samme 40 tegn × 25 linier skærm, samme opad-kompatible Basic, men Poke-sætningerne skal ændres; maskinsproget er det samme, men adresserne skal oversættes. Basic'en vil altså ikke være fremmed for C-64 brugere, men den er ikke helt kompatibel for alle eksisterende programmer. Hvad programmodulerne (cartridges) angår, er der ikke store chancer for at de passer.

Commodore 16 er en god, billig maskine for nybegyndere. Den er i høj grad kompatibel med storebror Plus/4. Begge modeller er velegnede for programmører med »seriøse« anvendelser, fordi de hurtigt vil kunne skrive effektive programmer på dem.

Tabel 1 - Specifikationer

	Plus/4	C-16
RAM	64K	16K
ROM	32K	32K
Bytes fri til Basic	60671	11519
RS-232 kommunikations interface	ja	nej
Indbygget software	ja	nej
Pris	ca.	ca.
	Begge modeller	
Serielt interface (disk, printer)	ja	
Kassette-interface (ikke 64-kompatinelt)	ja	
Skærm (tegn × linier)	40 × 25	
Opløsning i billedpunkter	200 × 320	
Multicolor (MCM)	ja	
Extended color (ECM)	ja	
Lyd	2 kanaler, firkant pulser	
Cursor	4 taster	
Reset omskifter	ja	
Skærmfarver	15, hver med 8 nuancer	

Tabel 2.

Sammenligning mellem Plus/4 og C-16 med Commodore 64

Fordele af Commodore 64:
Bedre lyd og musik
Nem programmering af bevægende figurer med sprites
Hurtigere kassettebånd
Enkel arkitektur

Fordele af Plus/4 og C-16:
Kraftigt udvidet Basic med:
Diskette kommandoer
Print Using kommando
Strukturerede sætninger
Error Trap, Tracing og Renumber
Kraftigere kommunikationsport (kun Plus/4)
Separate cursortaster
Større farveudvalg
Vinduer på skærmen, blinkende tegn
Ingen ventepauser for »garbage collection«.

Tabel 3.

Myter om de nye computere

Myte nr. 1: Plus/4 og C-16 er skrabede udgaver af Commodore 64.
Sandheden: De er en hel ny slags computere, og de er ikke beregnet til at erstatte 64'eren. Hver model har sine særlige fordele. Plus/4 og C-16 er specielt egnet til nem programmering af nyttige brugerprogrammer.

Myte nr. 2: Commodore 64 vil udgå af handelen.
Sandheden: 64'eren går ikke ud af produktionen. VIC-20 salget påvirkes måske af den billige C-16, men VIC'ens kommunikationsmuligheder gør, at den stadig vil kunne sælges et stykke tid. C-16 har ikke noget tilsvarende.

Myte nr. 3: Plus/4's og C-16's programmer er ikke kompatible med C-64 og VIC-20.
Sandheden: Basic'en på de nye maskiner er den samme, den er bare udvidet. Basic programmer fra de tidligere modeller kan loades, og kan i de fleste tilfælde køre. C-64 programmer skal dog ændres først for at blive af med Peek og Poke kommandoer, og erstatte dem med de tilsvarende kommandoer i udvidet Basic. Maskinprogrammer er de samme som før, blot skal adresserne ændres, og den mere avancerede programmør, skal sætte sig ind i arkitekturen for de nye modeller.

Myte nr. 4: De nye maskiner er sværere at programmere.
Sandheden: Basicprogrammerne er lettere at skrive, fordi der er flere kommandoer og større arbejdsareal til rådighed på Plus/4. Programmering i maskinsprog er omtrent det samme som før, eller den kan kræve mere forståelse af arkitekturen, det kommer an på, hvor kompliceret programmet er.

Myte nr. 5: Der er ikke noget marked for de nye modeller, eller også går det ud over C-64 salget.
Sandheden: Det må selvfølgelig vise sig, hvordan køberne reagerer. Plus/4 og C-16 kan bruges til mange ting. Mange, som ellers ville have købt en 64, vil købe en af de nye computere i stedet. Til gengæld vil der være en del nye markeder. På grund af deres lette programmering, flere skærmfarver og andre muligheder og de nye programmer, vil Plus/4 og C-16 være meget attraktive for brugere inden for små virksomheder, og til undervisning.

Med udvidet Basic og indbygget software klarer Plus/4 og C-16 en masse opgaver let og elegant.

Tabel 4.

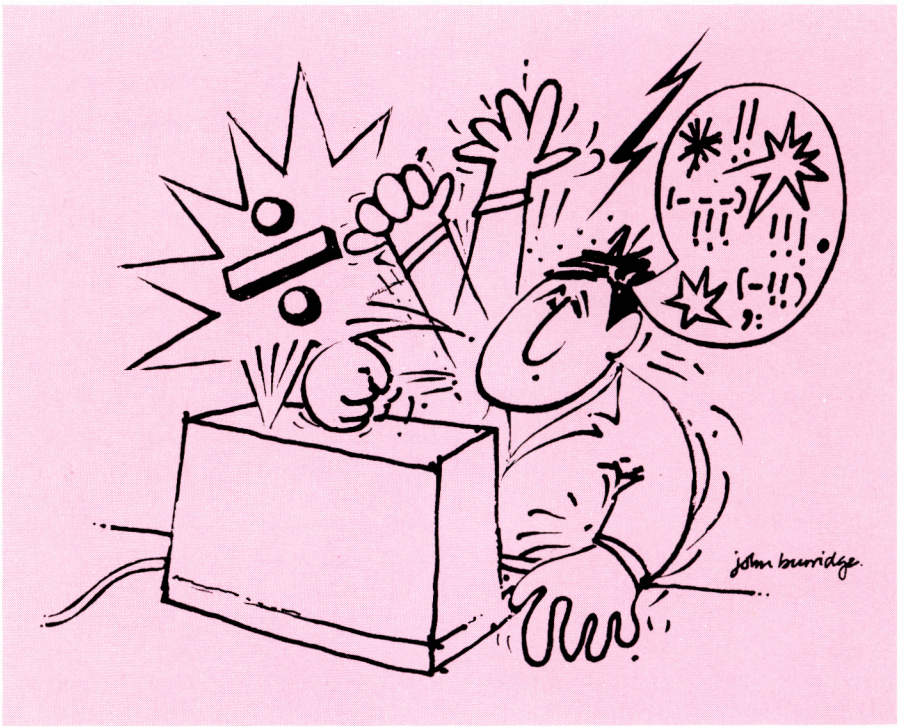
De vigtigste nøgleord i Plus/4 og C-16 Basic. De ord der ikke forekommer på Commodore 64 står i fed.

Almindelig Basic - I/O kommandoer - Grafik- og lydkommandoer - Hjælpefunktioner - Funktioner (de samme som på C-64, desuden følgende):

Alm. BASIC		I/O Kommandoer	Grafik og lydkommandoer	Hjælpekommandoer	Funktionskommandoer (de samme som C-64, plus nedenstående)
CLR	**PRINT ...USING**	LOAD	**BOX**	AUTO	**DEC**
NEW	**PUDEF**	SAVE	**CHAR**	DELETE	**HEX$**
CMD	READ	VERIFY	**CIRCLE**	HELP	**ERR$**
CONT	DATA	**DLOAD**	**COLOR**	KEY	**INSTR$**
DIM	RESTORE	DSAVE	**GRAPHIC**	TRON	**RCLR**
LET	**RESTORE**	**DIRECTORY**	**SSHAPE**	TROFF	**RDOT**
LIST	**TRAP**	**BACKUP**	**GSHAPE**	MONITOR	**RGR**
REM	**RESUME**	**COLLECT**	**LOCATE**	RENUMBER	**RLUM**
RUN	**DO**	**COPY**	**PAINT**		**JOY**
STOP	**LOOP**	**HEADER**	**SCALE**		**DS**
END	**UNTIL**	**RENAME**	**SOUND**		**DS$**
CONT	**WHILE**	**SCRATCH**	**VOLUME**		**PI**
SYS	**EXIT**				
WAIT	**FOR**				
OPEN	**TO**				
CLOSE	**STEP**				
GETKEY	**NEXT**				
GET	**IF**				
GET	**THEN**				
INPUT	**ELSE**				
INPUT	GOTO				
PRINT	ON..GOTO				
PRINT	GOSUB				
PRINT USING	ON..GOSUB				

Når disk og drømme er bristet

Diskdata gi'r dig alt, hvad der er værd at vide om disketternes indhold – både med VIC 20 og CBM 64.



En aften sad jeg og puslede med det sprite-program, der er gengivet i sidste nummer af RUN. Forsigtig som jeg er – og som alle programmører bør være – gemte jeg nu og da en ny version af programmet i takt med, at jeg ændrede i det.

Da kaffen for længst var blevet kold, besluttede jeg, at nu ville jeg stoppe.

SAVE »XSPRITE35«,8 tastede jeg på min 64'er.

Diskettestationen gav sig til at snurre. Alt var tilsyneladende i orden.

Da jeg næste dag tastede: LOAD »XSPRITE35«,8 startede diskettestationen godt nok, men standsede meget hurtigt. Blodtrykket steg, og en LIST kommando afslørede fejlen. Kun en meget lille del af programmet var kommet på disken, som det skulle. Resten var skrevet på et eller andet fremmedartet programmeringssprog, som jeg ikke forstod et kuk af.

En DIR«$ kommando fra Simon Basic afslørede, at programmet skulle findes på disken i fuld udstrækning. I hvert fald var det rigtige antal blokke belagt på disken.

Det blev starten til det program, jeg kalder Diskdata. For det gik op for mig, at der var kommandoer til genoplivning af et program, som jeg ikke havde. Derfor gav jeg mig til at søge oplysninger om Floppyens virkemåde. Det gik hurtigt op for mig, at der var muligheder for at genoplive programmet – hvis jeg ikke forsøgte at tvinge andre programmer ind på samme diskette.

Redningen

Nøglen til redningen ligger i diskens indholdsfortegnelse, den såkaldte »Directory«. Indtil dette øjeblik, havde jeg kun brugt min diskettestation til at lagre og genkalde programmer med SAVE og LOAD. Nu måtte jeg sætte mig ind i de mere avancerede kommandoer, som gav adgang til det enkelte spor og sektor. Ja, helt ned til den enkelte byte og dens manipulering. Det blev til et par dages læsning og forsøg, inden den var trængt ind. Men så åbnede endnu en vej sig, den direkte adgang til hver enkelt byte på disken.

Det kan lyde indviklet, når man læser den noget kortfattede brugervejledning, der følger med diskettestationen. Og det er da heller ikke nødvendigt at forstå hverken den eller de forklaringer til programmet, der kommer i det følgende, for at bruge Diskdata-programmet.

Diskdata giver imidlertid adgang til meget vigtige oplysninger og manipuleringsmuligheder, som de fleste næppe på forhånd har haft adgang til, og slet ikke i så nem en form som her. Tast programmet omhyggeligt ind. Gem en version på én disk. Indlæs så en version mere på en helt frisk disk. Hvis noget går galt under afviklingen har man ikke ødelagt andet end det program, man er ved at arbejde med på den friske disk, og kan tage den version, man gemte på en anden disk, frem igen til videre arbejde indtil alt fungerer.

Når programmet kører, kan man teste det på den »friske« disk ved at læse otte-ti versioner ind og maniculere med dem. Først derefter vil jeg råde til, at man går i gang med sine disks, som indeholder andre programmer.

Så går det løs

Diskdata byder på følgende muligheder:

1. visning af navnene på samtlige files
2. oplysning om følgende på files: åben, lukket, belagte blokke, længden af en file, side sektorer, antallet af records og datablokke i en relativ file, begyndelses adresse på en programfile, antallet af frie blokke, og antallet af belagte blokke for alle typer af files.
3. oplysningerne kan skrives til papir
4. en file kan genetableres
5. en file kan sikres mod »Scratchning« altså sletning
6. en sikret file kan frigives
7. en file kan slettes – også hvis den er sikret.

Programmet er menustyret, og kan altså betjenes uden at man på forhånd ved ret meget om, det der sker. Men jeg vil alligevel kort beskrive et handlingsforløb for at få forklaret den begrebsverden, som Diskdata bruger.

Når programmet startes med RUN skrives program-titel, og der spørges, om man vil se alle files på disken. Hvis man ved besked om, hvilken file man vil manipulere med tastes alt andet end J.

Tastes derimod J, vises alle files på disken. Også de som er slettet, eller »Deleted«, men endnu ikke overskrevet af en ny file. Der sker nemli' det, at filerne ikke slettes i egentlig forstand, når man bruger kommandoer: OPEN 1,8,15»S:navn«: CLOSE 1.

Der er kun tale om, at det allerførste byte i directory'en (indholdsfortegnel-

sen) nul-stilles frem for, at den indeholder oplysning om, hvilken filetype – sequentiel, program, user eller relativ – der er tale om. På side 56 i din brugermanual til diskettestationen kan du i skemaform se, hvilke oplysninger directory'en indeholder på hver enkelt file.

Skærmen spørger nu hvilken file, man søger oplysninger på. Indtast et af navnene, og du får en oversigt over status på dels disken som sådan, og på det program, der er kaldt. Du kan også kalde programmer som et slettet, men endnu ikke overskrevet. Altså: du kan kalde alle de programmer, som kommer frem i fileoversigten.

Der spørges, om du vil have oplysningerne over på papir. Programmet er indrettet til en printer/plotter, men kan laves om til en almindelig printer i linje 807, som så skal lyde:

```
807 IF SV$ = »J«THENOPEN1,4:R =
    1:GOSUB680:R = 0
```

Hvis du ikke ønsker en udskrift, tastes alt andet end J. Der spørges om du vil manipulere videre med filen. Hvis ja, så tast J. I modsat fald stilles spørgsmålet, om du vil slutte eller have fat i en anden file.

Hvis svaret er J, får du nu endnu en menu frem på skærmen. Her spørges, om filen skal genetableres, sikres, frigives eller slettes.

Hvis du vælger 1, genetableres filen. Det sker ved, at der spørges om hvilken filetype, den tidligere var. Var den slettede file f.eks. en programfile, trykkes på 2. Nu sørger programmet for at genskabe filen ved at sætte bit 7 og 1 til »1« således at bitmønsteret i byte 2 ser således ud: 1000 0010. Det svarer til 130 i decimaltal.

Havde vi i stedet bedt om den anden mulighed ved at trykke på 2, havde vi sikret en file mod at blive slettet ved en scratch-kommando. Det sker ved at sætte bit 6 til 1. Bitmønsteret ville så være (med en programfile): 1010 0010. Hvis du forsøger dig med en file, som allerede er sikret meddeles det på skærmen, og der sker intet.

3. mulighed betyder, at en sikret file trods alt kan slettes. Det kan du ellers ikke med de normale kommandoer, som er til rådighed i diskette-betjeningen! Her sættes bit 6 igen til 0.

Trykkes 4 åbner man mulighed for at slette en file. Det sker med Scratch-kommandoen. Men først undersøges om det er en file, som er sikret. Er det tilfældet, beder programmet dig om at overveje, om du også vil frigive denne file. Trykkes J, klapper fælden og filen frigives. Så spørges endnu engang, om du vil slette filen. Er svaret J, slettes den.

Selve programmet

Til slut vil jeg lige kommentere programmets enkelte afsnit. Det bliver kun ganske kort, idet jeg vil henvise folk, der ønsker at vide mere om de anvendte kommandoer til brugervejledningen, der fulgte med VIC 1541 diskettestationen. De finder oplysningerne fra side 24-41, samt skemaerne på de sidste sider i vejledningen.

Som tidligere bemærket, kan programmet også løbe i samarbejde med en VIC 20. Den eneste forskel er de rettelser, der skal foretages i første linje angående farvevalg. Hvis man igen vil have rød skærm, skal der i stedet stå:

```
10 CLR:POKE36879,42:PRINTCHR$
   (158);PRINT$(147);
```

20-90 programstart og de første valg. Hvis programnavnet er længere end 16 bogstaver må man vælge om.

100-240 indeholder et liste/søgeprogram, der udlister en eller alle files. Også de, som har betegnelsen DELETED, altså slettede. På den måde kan man se, om en redning også er mulig endnu. Man starter med at se i spor 18 (TR) sektor 1 (SP) med B-R-kommandoen læses direkte over kanal 2, det første byte, spor 18, sektor 1. Det er linje 110.

Linje 120 sætter en pointer til samme byte, og i 130 sørger man for, at en eventuel »tom« streng ikke får en fejludskrift til følge. I linje 160 sættes pointeren (B-P) til det sted, hvor man får oplyst programnavnet. Der er otte navne i hver sektor i spor 18. Det fundne navn udlæses via endnu en FOR... NEXT løkke i 170. Og i 240 husker man pænt at lukke pænt efter sig.

270-805 er de linjer, hvor man finder ud af filetype, om filen er lukket eller åben, længde osv.

Princippet for at hente disse oplysninger er det samme som beskrevet ovenfor. Man sætter i 270 en B-P-kommando til den byte, man ønsker oplysninger på. (X«32 + 2) er stedet, hvor filetypen udlæses. X opsumeres i takt med, at der skiftes programnavn idet, hver filesdagbog i directory'en fylder 32 byte.

I linje 390 peges på det sted, hvor programstart er angivet, og det omregnes til decimal (normalt altid 2049 for en programfile).

I 440 gælder det record-længden. For at få programlængden må man ind og læse i selve programmet – altså det program, som man henter oplfsninger på. Og det sker i linje 476.

I 540 går man ind i BAM-oplysningerne for at se, hvor meget plads der er brugt og derfor også, hvor meget der er tilbage.

I 660-805 udskrives resultaterne af den research, der er foretaget alt efter hvilken filetype, der er tale om. Se skemaet her i RUN.

820-990 er afsnittet, hvor et program genskabes som file. Det sker igen ved at gå ind og pege på det sted, man ønsker at manipulere og så OR-re bit-mønsteret ind og genindlæse (B-W) dvs. U2 i 970 via bufferen.

1000-1090 her sikres en file og det sker på præcis samme måde med bitmanipulering af bit 6.

1100-1190 her frigives en sikret file.

1200-1260 her slettes en file.

Det kan synes som en ordentlig mundfuld. Men det er ikke nødvendigt at sætte sig ind i alle disse kommandoers virkemåde for at få programmet til at køre. Tast det ind præcis, som det står – og du er kørende.

Jeg fik blodtrykket i orden igen. Fandt »XSPRITE35« frem igen. Det viste sig nemlig at jeg havde slettet nogle programmer fordi, der ikke var mere plads på disken, som jeg brugte til programudvikling.

Diskdata tilbyder:

	Filetyper				
	DEL	SEQ	PRG	USR	REL
Filen lukket	★	★	★	★	★
Filen åben	★	★	★	★	★
Antal belagte blokke	★	★	★	★	★
Recordlængde	–	–	–	–	★
Side-sector-blokke	–	–	–	–	★
Datablokke	–	–	–	–	★
Records	–	–	–	–	★
Begyndelsesadresse	–	–	★	–	–
Filen genetableres	★	★	★	★	★
Filen sikres	★	★	★	★	★
Filen frigives	★	★	★	★	★
Filen slettes	★	★	★	★	★

DISKDATA

```
10 CLR:POKE53280,2:POKE53281,2:PRINTCHR$
   (158);CHR$(147);
20 PRINTTAB(6);"**********************"
30 PRINTTAB(6);"VIS ALLE FILEPARAMETRE"
40 PRINTTAB(6);"**********************"
50 PRINT"[CRSR/DOWN*2]VIL DU SE SAMTLIGE
   FILE-NAVNE? [RVS/ON]J[RVS/OFF] ";
60 WAIT203,63:GETTA$:IFTA$="J"THENFL=1:P
   RINTTA$:GOSUB100
70 FL=0:PRINTTA$:PRINT"HVILKEN FILE SØGE
   S ";:INPUTFI$
80 IFLEN(FI$)<=16THEN100
90 PRINT"[CRSR/DOWN]FILENAVNET SKAL VÆRE
   UNDER 16 BOGSTAVER!":GOTO70
100 OPEN15,8,15,"I0":OPEN2,8,2,"#":TR=18
   :SP=1
110 PRINT#15,"B-R";2;0;TR;SP
115 TT=TR:SS=SP
120 PRINT#15,"B-P";2;0
130 GET#2,X$:IFX$=""THENX$=CHR$(0)
140 TR=ASC(X$):GET#2,X$:IFX$=""THENX$=CH
   R$(0)
150 SP=ASC(X$)
160 FORX=0TO7:PRINT#15,"B-P";2;X*32+5:FF
   $=""
170 FORY=0TO15:GET#2,X$:IFX$=""THENX$=CH
   R$(0)
175 IFASC(X$)=160THEN190
180 FF$=FF$+X$
190 NEXTY
195 IFFI$=FF$THEN260
200 IF FL=1THENPRINT FF$
210 NEXT X
220 IFTR=0THEN240
230 GOTO 110
240 CLOSE2:CLOSE15
250 IF FL=0THENPRINT"FILENAVNET FINDES I
   KKE PAA DISKEN!":GOTO50
260 IF R=1 THEN RETURN
265 IF FL=1THEN RETURN
270 PRINT#15,"B-P";2;X*32+2:GET#2,X$:IFX
   $=""THENX$=CHR$(0)
280 TY=ASC(X$):IF R=1 THEN RETURN
285 FT=TYAND15
290 IFFT=0THENFT$="DELETED"
300 IFFT=1THENFT$="SEQUENTIAL"
310 IFFT=2THENFT$="PROGRAM"
320 IFFT=3THENFT$="USER"
330 IFFT=4THENFT$="RELATIV"
340 IFFT>4THENPRINT"UGYLDIG FILETYPE":PR
   INT:GOTO50
350 IFTYAND128THENGE$="JA":GOTO370
360 GE$="NEJ"
370 IFTYAND64THENSA$="JA":GOTO390
380 SA$="NEJ"
390 PRINT#15,"B-P";2;X*32+30:GET#2,X$:IF
   X$=""THENX$=CHR$(0)
400 LB=ASC(X$)
410 GET#2,X$:IFX$=""THENX$=CHR$(0)
420 HB=ASC(X$)*256
430 BL=LB+HB:IFFT<>4THEN460
440 PRINT#15,"B-P";2;X*32+23:GET#2,X$:IF
   X$=""THENX$=CHR$(0)
450 RL=ASC(X$):IFFT<>2THEN540
460 PRINT#15,"B-P";2;X*32+3:GET#2,X$:IFX
   $=""THENX$=CHR$(0)
470 DT=ASC(X$)
475 GET#2,X$:IFX$=""THENX$=CHR$(0)
476 DS=ASC(X$):OPEN3,8,3,"#":PRINT#15,"B
   -R";3;0;DT;DS
480 PRINT#15,"B-P";3;2
490 GET#3,X$:IFX$=""THENX$=CHR$(0)
500 LB=ASC(X$)
510 GET#3,X$:IFX$=""THENX$=CHR$(0)
520 HB=ASC(X$)*256
530 AA=HB+LB:CLOSE3
540 PRINT#15,"B-R";2;0;18;0
550 BF=0
560 FORI=4TO140STEP4:IFI=72THEN590
570 PRINT#15,"B-P";2;I:GET#2,X$:IFX$=""T
   HENX$=CHR$(0)
580 BF=ASC(X$)+BF
590 NEXT
600 BB=664-BF
610 IFFT<>4THEN640
620 BS=BL/121:IFBS<>INT(BS)THENBS=INT(BS
   )+1
630 RC=INT((BL-BS)*254)/RL)
640 PRINT"[CLR/HOME]SKÆRM ELLER PLOTTER
   (S/P) ";:WAIT203,63:GETSV$:PRINTSV$
650 RE$=CHR$(18):RA$=CHR$(146):IFSV$<>"S
   "ANDSV$<>"P"THEN640
660 OPEN1,3
670 IFSV$="P"THENOPEN1,6
680 PRINT#1,"FILEN ";RE$;FI$;RA$;"'S PAR
   AMETRE      "
690 PRINT#1,"-------------------------
   ------------"
700 PRINT#1,"FILETYPE:                ";RE$
   ;FT$;RA$:PRINT#1
710 PRINT#1,"FILEN LUKKET:            ";RE$
   ;GE$;RA$:PRINT#1
720 PRINT#1,"FILEN BESKYTTET:         ";RE$
   ;SA$;RA$:PRINT#1
730 PRINT#1,"BELAGTE BLOKKE:          ";RE$
   ;BL;RA$:PRINT#1
740 IFFT<>4THEN790
750 PRINT#1,"RECORD LÆNGDE:           ";RE$
   ;RL;RA$:PRINT#1
760 PRINT#1,"SIDE-SECTOR BLOKKE:      ";RE$
   ;BS;RA$:PRINT#1
770 PRINT#1,"DATA BLOKKE:             ";RE$
   ;BL-BS;RA$:PRINT#1
780 PRINT#1,"RECORDS:                 ";RE$
   ;RC;RA$:PRINT#1
790 IFFT=2THENPRINT#1,"BEGYNDELSES-ADRES
   SE:   ";RE$;AA;RA$:PRINT#1
800 PRINT#1,"FRIE BLOKKE I DISKEN:  ";RE$
   ;BF;RA$:PRINT#1
804 PRINT#1,"BELAGTE BLOKKE I DISK:";RE$
   ;BB;RA$:PRINT#1
805 IFR=1THENRETURN
806 CLOSE1:PRINT"[CRSR/DOWN]FILEDATA TIL
    PAPIR [RVS/ON]J[RVS/OFF]? ":WAIT203
   ,63:GETSV$:PRINTSV$
807 IFSV$="J"THENOPEN1,6:R=1:GOSUB680:R=
   0
808 FORT1=1TO250:NEXTT1
809 PRINT"ØNSKER DU AT ARBEJDE VIDERE ME
   D FILEN [RVS/ON]J[RVS/OFF]":WAIT203,
   63:GETSV$
810 PRINTSV$
811 IFSV$<>"J"THEN890
820 PRINT"[CLR/HOME][CRSR/DOWN*2]DER ER
   FØLGENDE MULIGHEDER:"
824 PRINT"[CRSR/DOWN]1. FILEN GENETABLER
   ES"
826 PRINT"[CRSR/DOWN]2. FILEN SIKRES"
828 PRINT"[CRSR/DOWN]3. FILEN FRIGIVES (
   SIKREDE FILES)"
830 PRINT"[CRSR/DOWN]4. FILEN SLETTES":C
   LOSE1:CLOSE2:CLOSE15
870 GETSV$:IFSV$=""THEN870
880 IFSV$="1"THENGOSUB900
882 IFSV$="2"THENGOSUB1000
884 IFSV$="3"THENGOSUB1100
886 IFSV$="4"THENGOSUB1200
890 PRINT"[CRSR/DOWN*2]NY FILE J/N?"
```

```
892 GETSU$:IFSU$=""ORSU$<> J"ANDSU$<>"N"
    THEN892
894 IF SU$="J"THEN10
896 CLOSE1:CLOSE2:CLOSE3:CLOSE15:END
900 IFFT<>0THENRETURN
910 OPEN1,3:PRINT#1,"[CLR/HOME][CRSR/DOW
    N]HVILKEN FILETYPE HAR ";FI$;" V¿RET
    :"
920 PRINT#1,"1. SEQUENTIEL"
921 PRINT#1,"2. PROGRAM"
922 PRINT#1,"3. USER"
923 PRINT#1,"4. RELATIV"
924 GETSU$:IFSU$=""ORSU$<>"1"ANDSU$<>"2"
    ANDSU$<>"3"ANDSU$<>"4"THEN924
925 FT=VAL(SU$):R=1:GOSUB100:R=0
930 IFFT=1THENTY=(TY OR 129)
932 IFFT=2THENTY=(TY OR 130)
934 IFFT=3THENTY=(TY OR 131)
936 IFFT=4THENTY=(TY OR 132)
940 PRINT#15,"B-R";2;0;TT;SS
950 PRINT#15,"B-P";2;X*32+2
960 PRINT#2,CHR$(TY);
970 PRINT#15,"U2";2;0;TT;SS
980 PRINT#15,"I0":CLOSE1:CLOSE2:CLOSE15
990 PRINT"FILEN GENSKABT":RETURN
1000 IFFT=0THENPRINT"[CRSR/DOWN]FILEN ER
     SLETTET - SKAL GENETABLERES":RETUR
    N
1010 R=1:GOSUB100:R=0
1020 IF(TYAND64)=0THEN1030
1025 PRINT"[CRSR/DOWN]FILEN ER ALLEREDE
    SIKRET":GOTO808
1030 TY=(TY OR 64)
1040 PRINT#15,"B-R";2;0;TT;SS
1050 PRINT#15,"B-P";2;X*32+2
1060 PRINT#2,CHR$(TY);
1070 PRINT#15,"U2";2;0;TT;SS
1080 PRINT"[CRSR/DOWN]FILEN ER SIKRET NU
    !"
1090 CLOSE1:CLOSE2:CLOSE15:RETURN
1100 IFFT=0THENPRINT"[CRSR/DOWN]FILEN ER
     SLETTET - SKAL GENETABLERES":RETUR
    N
1110 R=1:GOSUB100:R=0
1120 IF(TYAND64)=64THEN1130
1125 PRINT"[CRSR/DOWN]FILEN ER ALLEREDE
    FRIGIVET":OPEN1,3:GOTO808
1130 TY=(TY AND 255-64)
1140 PRINT#15,"B-R";2;0;TT;SS
1150 PRINT#15,"B-P";2;X*32+2
1160 PRINT#2,CHR$(TY);
1170 PRINT#15,"U2";2;0;TT;SS
1180 PRINT"[CRSR/DOWN]FILEN ER FRIGIVET!
    "
1185 IFR1=1THEN RETURN
1190 CLOSE1:CLOSE2:CLOSE15:RETURN
1200 IFFT=0THENPRINT"[CRSR/DOWN]FILEN ER
     SLETTET":RETURN
1210 R=1:GOSUB100:R=0
1220 IF(TYAND64)=0THEN1230
1225 PRINT"[CRSR/DOWN]FILEN ER SIKRET -
    SKAL DEN FRIGIVES [RVS/ON]J[RVS/OFF
    ]":WAIT203,63:GETAN$:PRINTAN$
1227 IFAN$="J"THENR1=1:GOSUB1130:R1=0
1230 PRINT"FASTHOLDER DU SLETNINGEN?[SHI
    FT/SPACE]":WAIT203,63:GETAN$:PRINTA
    N$
1240 IFAN$<>"J"THENRETURN
1250 PRINT#15,"S:"+FI$
1260 PRINT"[CRSR/DOWN]FILEN ER SLETTET":
    RETURN
```

KONTROLSUM FOR DISKDATA

10	178	20	182	30	43
40	182	50	94	60	14
70	183	80	86	90	203
100	255	110	198	115	53
120	5	130	242	140	23
150	232	160	205	170	242
175	168	180	56	190	219
195	223	200	48	210	250
220	80	230	59	240	18
250	197	260	85	265	117
270	70	280	129	285	14
290	89	300	94	310	124
320	164	330	125	340	32
350	183	360	131	370	144
380	139	390	119	400	211
410	242	420	24	430	58
440	121	450	24	460	71
470	221	475	242	476	37
480	8	490	243	500	211
510	243	520	24	530	3
540	22	550	106	560	52
570	74	580	255	590	130
600	9	610	254	620	211
630	39	640	195	650	134
660	47	670	119	680	23
690	65	700	89	710	199
720	94	730	211	740	4
750	220	760	136	770	153
780	0	790	116	800	134
804	157	805	245	806	131
807	179	808	101	809	26
810	102	811	146	820	169
824	108	826	205	828	103
830	121	870	60	880	76
882	117	884	119	886	121
890	12	892	33	894	192
896	228	900	238	910	88
920	183	921	209	922	249
923	210	924	151	925	100
930	152	932	145	934	147
936	149	940	203	950	26
960	246	970	145	980	62
990	130	1000	247	1010	251
1020	239	1025	198	1030	183
1040	203	1050	26	1060	246
1070	145	1080	233	1090	229
1100	247	1110	251	1120	42
1125	189	1130	253	1140	203
1150	26	1160	246	1170	145
1180	180	1185	70	1190	229
1200	206	1210	251	1220	241
1225	43	1227	182	1230	130
1240	101	1250	88	1260	32

»VIC-20 Routiner«

Af Jan Brøndum

Hvis du er en af de 20.500 heldige ejere af en VIC-20, vil vi i dette nr. af RUN bringe nogle gode routiner til 20'eren.

I stedet for at slukke for din 20'er, hvis du vil have slettet alt i hukommelsen, kan du skrive SYS 64824 eller SYS 64802.

En af de routiner man virkelig mangler på en 20'er, er en ON ERROR GOTO, hvilket er en kommando, der gør, at hvis der er fejl i dit program, springes automatisk til en linie, som du bestemmer efter eget valg. Her er mit bud på en ON ERROR GOTO til 20'eren.

I linie 10 fortæller du maskinen, hvilken linie den skal gå til, hvis der er fejl i programmet.

```
10 INPUT "LINIENR. ";1
20 C$ = "GOTO" + STR$ (1) + CHR$
   (13)
25 D = LEN (C$)
30 POKE 198,D: FOR V = 1 TO D:
   POKE 630 + V, ASC(MID$(C$,V)):
   NEXT V
```

De næste linier poker linienummeret ind i programmet.

Som du nok har lagt mærke til ved mange cardridge spil, starter skærmbilledet med at befinde sig langt uden for skærmen. Ved hjælp af cursortasterne, kan du placere det, hvor du har lyst til. Denne lille routine kan selvfølgelig også bruges i basicprogrammer. Du starter med at POKE den vandrette position ind i location 36864.

Hvilket betyder, at du siger: POKE 36864, position (denne er normalt 12).

Det samme gælder for den lodrette position, forskellen er bare, at denne har en anden location, nemlig: 36865. Dennes position er normalt 37.

En af de ting man ser i mange programmer, er at run/stop og restore tasterne er sat ud af funktion. Det vil sige, at du ikke kan stoppe programmet, når først du har startet det.

Hvis du får lyst til at bruge denne routine i dit basic program, så er den her:

```
1 POKE 45,240: POKE 46,26:
POKE 788,194: POKE 37150,3
```

Hvis du f.eks. har en 8k ram udvidelse, og taster et program ind der kræver 3k ram udvidelse, har du nok opdaget at programmet for det meste ikke virker på en 8k ram fordi, adresserne på de 2 udvidelser er forskellige. Men dette kan der også rådes bod på med en lille routine, der får maskinen til at tro at nogle chips faktisk ikke er, hvor de er, hvilket vil sige, at hvis du f.eks. har en 8k ram, kan du få maskinen til at tro, at der kun er 3k ram. I det følgende skema, ser du helt til venstre hvor megen hukommelse, du vil have maskinen til at tro, der er. Til højre står der, hvad du skal gøre for at få lige præcis den hukommelsesstørrelse.

Standard	POKE 642,16: POKE 644,30: POKE 648,30: SYS 64824
+ 3k ram	POKE 642,4: POKE 644,30: POKE 648,30: SYS 64824
+ 8k ram	POKE 642,18: POKE 644,64: POKE 648,16: SYS 64824
+ 16k ram	POKE 642,18: POKE 644,96: POKE 648,16: SYS 64824
+ 24k ram	POKE 642,18: POKE 644,128: POKE 648,16: SYS 64824

Mange VIC-20 ejere har i stedet for at købe en 3k rams udvidelse, købt en super expander, da den både har højopløsning og 3k ram ekstra. I stedet for at pille superexpanderen ud, kan man så skrive SYS 64850, og maskinen vil tro, at der ikke er nogen superexpander. Hvis du på et tidspunkt får lyst til at slå den til igen, kan dette også lade sig gøre, ved at skrive: SYS 41031.

Har du lavet et program, og vil holde programmet for dig selv, så du kun selv kan gå ind og rette i det, så brug denne poke til at disassemble list routinen med: POKE 775,0.

Hvis du får lyst til at lave om i dit program, kan du få list commandoen til at virke igen med: POKE 775,199.

Commodore Plus/4 compatibel med C-64?

Med det væld af fremragende software, der findes til Commodore 64, er Commodore (med rette) blevet stærkt kritiseret for, at denne ikke kan anvendes til den nye Plus/4. Ligeledes er mange C-64-ejere, der påtænker skift til Plus/4, utilfreds med, at de ikke kan anvende den software, der allerede er investeret i.

I den anledning udtaler Jim Dionne, marketingschef hos Commodore i USA, at man påtænker at fremstille et motherboard til Plus/4, der gør denne compatibel (forenelig) med C-64, herunder også med sprites og lydmuligheder.

– Hvis ønsket virkelig er til stede, vil vi meget hurtigt finde ud af det, og vi vil også være i stand til at reagere meget hurtigt, udtaler Jim Dionne.

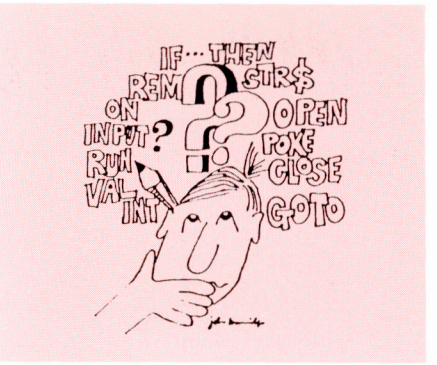
Muligheden lurer som sagt lige om hjørnet, og de tekniske muligheder er også til stede.

Hvis Commodore ikke selv gør det, skulle det ikke undre, om andre vil udnytte disse muligheder.

Det bliver i hvert fald spændende at se, hvordan det går.

Sekventielle filer – del 2

Af Jørgen Jørgensen



Vi så i sidste nummer, hvorledes man kan gemme data på bånd eller diskette. Vi gjorde det ved at lave en såkaldt sekventiel file, hvor den vigtigste regel er, at den skal indlæses nøjagtigt som den blev skrevet. LOAD og SAVE routinen er i princippet opbygget fuldstændig ens.

Mere om DIM-funktionen

I sidste nummer dimensionerede vi variabler i en enkelt dimension. Det vil sige, at den pågældende variabel kun bestemmes ved et nummer, f.eks.

AB(X)

hvor X er nummeret vi tildeler variablen.

En variabel kan imidlertid også have flere dimensioner. Hvis vi skriver

DIM AB(12,100)

eller genrelt

DIM AB(x,y)

kan vi forestille os variabler lagt ud i felter på et stykke kvadreret papir, hvor X svarer til kolonnerne og Y til de vandrette rækker. Hvert felt får så hver sin X,Y koordinat, i hvilken man kan lægge den værdi eller tekst, man vil tildele den enkelte variabel.

Flerdimensionerede variabler er imidlertid meget pladskrævende og i det lige viste eksempel vil computeren reservere plads til

(12 + 1)«(100 + 1)

altså i alt 1313 variabler. At vi skal lægge en til X,Y værdien, skyldes at »nullet« også skal tælles med.

Man kan også gå ud i tre eller flere dimensioner, men så slipper hukommelsen hurtigt op.

Den flerdimensionerede file

Når man skal gemme en sådan flerdimensioneret file på bånd eller diskette, er vi nødt til at lægge flere FOR... NEXT-løkker »inden i hinanden«.

Ellers er princippet det samme. I det nævnte eksempel vil programmet se således ud, hvis vi ville gemme på diskette:

```
10 OPEN1,8,2,"@Ø:NAVN,S,W"
20 FOR X = 0 TO 12
30 FOR Y = 0 TO 100
40 PRINT# 1, AB(X,Y)
50 NEXT Y
60 NEXT X
70 CLOSE 1
```

I linie 10 står S'et for sekventiel og W'et for write. "O0" tilkendegiver, at vi vil »skrive oven i« en allerede eksisterende file af samme navn. Det problem eksisterer ikke for kassettebåndoptageren. Den vil altid »skrive oven i«.

Man skal i linierne 50 og 60 passe på ikke at bytte om på X og Y. Man slipper nemmest over det ved blot at skrive:

NEXT:NEXT eller NEXT Y,X

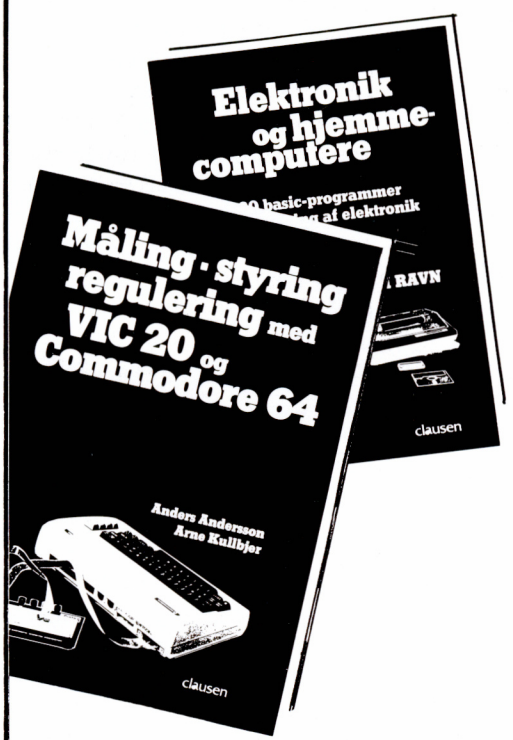
Skal filen indlæses, vil det tilsvarende program se sådan ud:

```
10 OPEN1,8,2,"NAVN,S,R"
20 FOR X = 0 TO 12
30 FOR Y = 0 TO 100
40 INPUT #1,AB(X,Y)
50 NEXT:NEXT:CLOSE 1
```

I linie 10 står R'et for read og S'et igen for sekventiel.

Hvis man skal gemme på bånd, er OPEN funktionen beskrevet i sidste nummer.

Få mere ud af din hjemmecomputer



Elektronik og hjemmecomputere

20 programmer til de elektronikinteresserede, der laver deres egne konstruktioner. Programmer med bl.a. Ohms lov, transistor- og operationsforstærkere, aktive filtre og svingningskredse. Programmerne er udviklet til Commodore 64 og VIC-20 men antallet af ordrer og instruktioner er holdt nede på et minimum, så programmerne let kan oversættes til andre dialekter i Basic.
Til hvert program findes en forklarende tekst, et diagram samt kopi af skærmbillederne ved input og output.
72 sider, kr. 65,00

Måling, styring og regulering med VIC-20 og Commodore 64.

Praktisk betonet bog der giver præcise anvisninger på, hvordan VIC-20 og Commodore 64 bruges til bl.a. at styre temperatur, tælle, måle frekvens, styre motorer, styre lys – alt helt automatisk efter fastlagte tidsskemaer.
Bogen giver en kort introduktion til datakommunikation ved måle-, styrings- og reguleringsproblemer samt mange ideer der er lette at omsætte i praksis.
104 sider, kr. 86,50

clausen bøger
(01) 29 44 22

791

COMAL 80

Af Steen Schmeltzer

Synes du også, at BASIC programmer til tider virker uoverskuelige? Er du træt af at lave subrutiner hele tiden? Mangler du programmørhjælp, mens du arbejder? Så synes jeg, du skulle prøve Commodor's COMAL 80 indstiksmodul. Det vil selvfølgelig kræve, at du skal lære et nyt programmeringssprog at kende. Selv om det måske vil være svært i begyndelsen, synes jeg, du skulle prøve alligevel. Du vil ikke blive skuffet.

COMAL, hvad er det?

COMAL blev udviklet her i Danmark midt i 70'erne af bl.a. Børge R. Christensen. Der er lidt BASIC og lidt PASCAL i COMAL. Men sproget har helt sin egen karakter og er uden tvivl det nye sprog til hjemmecomputere og skolebrug. I hvert fald indtil der kommer et, der er endnu bedre. Selv om det nu nok bliver svært at lave. For folk som mig selv, der først har lært BASIC, kan det nok blive sværere at lære COMAL 80, men er man i stand til at adskille de 2 sprog, vil man også nyde de store og indlysende fordele COMAL har.

COMAL er et struktureret sprog, der samtidigt er INTERAKTIVT. Og hvad er så det? For at gøre det kort, checker COMAL de linier, du indtaster, lige så snart du taster RETURN. Der er altså ikke noget med at skulle rette en masse irriterende SYNTAX ERROR fejl, når man tror, man skal til at se resultatet af sine anstrengelser. Samtidig med det kan man udføre en SCAN, inden programmet bliver RUNNET.

SCAN checker samtlige strukturer og rapporterer alle fejl, der måtte forekomme som ikke er SYNTAX ERRORS.

Der har været flere versioner af COMAL på diskette, men denne udgave er version 2.0 og det vil være denne, der fremover vil blive refereret til i denne artikel. Når computeren bliver tændt med indstiksmodulet isat, er der 30714 bytes fri.

Hvad får man for pengene?

Først og fremmest selve sproget, men også flere gode underprogrammer og hjælpeprogrammer. COMAL 80 indeholder 11 såkaldte PAKKER eller subrutiner. Nogle af disse kan benyttes direkte, deriblandt TURTLE & GRAFIK. Her er en hurtig oversigt.

ENGLISH/anvend engelske fejlmeddelelser
DANSK/anvend danske fejlmeddelelser
SYSTEM/konfigurationsrutiner
GRAPHICS/x-y grafik
TURTLE/LOGO grafik
SPRITES/siger sig selv
FRONT/brugerdefinerbart tegnesæt
SOUND/lyd og partiturrutiner
PADDLES/Paddlerutiner
LOYSTICK/Joystickrutiner
LIGHTPEN/lyspenrutiner

Udover dette overflødighedshorn er der også indbygget programmørhjælp:
RENUM ændrer samtlige eller et forudbestemt antal linier.
AUTO giver dig mulighed for at få automatisk tilskrivning af linienummer, mens du programmerer.
DEL sletter de linier eller områder, du angiver.
FIND gennemsøger dele af eller hele programmet efter det ønskede udtryk eller ord.
CAT viser diskettens katalog uden at gribe forstyrrende ind i, hvad der allerede måtte befinde sig i computeren.
STATUS viser diskettens nuværende fejlstatus.
DELETE fjerner programmer og filer fra disketten.

Herudover findes flere nyttige kommandoer, såsom PRINT AT, der sætter dig i stand til at skrive, hvor du vil på skærmen, uden at skulle bruge alle disse cursortegn. CURSOR, som tillader at anbringe cursoren overalt på skærmen på samme lette måde, hvilket især er nyttigt ved input kommandoer.

Der er også mulighed for at håndtere fejl, der måtte komme under programafviklingen. Man kan nemlig bestemme, hvad der skal ske, hvis noget trods alt går galt. Det kan f.eks. være en skærmmeddelelse. Derefter vil programmet fortsætte uden tab af data.

Hvis man bruger mange DATA i programmerne, kan man få programmet til at RESTORE et bestemt sted i DATA-sætningerne, som kun skal defineres med ETIKETTE (navn).

COMAL's store styrke ligger nemlig i opbygningen af programmet, hvor man i BASIC er bundet af linienummer og subrutiner, er man næsten frit stillet i COMAL. Man definerer sine subrutiner med et navn, og henviser derefter blot til navnet i resten af programmet. Man kan bruge navne og variabler på op til 16 tegn eller cifre, dog skal de starte med et bogstav. COMAL ser nemlig ikke på linienummer, men på ETIKETTER/NAVN.

Der følger en håndbog med indstiksmodulet, der ikke skal betragtes som en lærebog, men som en opslagsbog. Alle udtryk og deres SYNTAX er her beskrevet i alfabetisk orden. Der er desuden flere programeksempler, som også er på den medfølgende diskette. Jeg synes, håndbogen kunne være bedre – især afsnittet der beskriver »PAKKERNE«. Når der nu er så meget spændende, er det meget skuffende, at det ikke er dokumenteret så godt, at man straks kan få noget ud af det. De eneste, der straks er klar til brug, er TURTLE & GRAFIK.

Vi må håbe, at der snart kommer en bog, der beskæftiger sig indgående med disse emner. Jeg kunne også godt tænke mig en beskrivelse af, hvordan man får GRAFIK og LOGO udskrevet på printer.

Der må være en eller anden idiotsikring i indstiksmodulet. I hvert fald bliver følgende lille programstump automatisk ændret til COMAL, efter man har skrevet LIST. Indtast den nøjagtig som den står og tast så
LIST.

```
0010 for t = 1 to 2500
0020 next t
```

TURTLE & GRAFIK

TURTLE burde nu være så kendt, at

yderligere forklaring ikke tjener noget formål.

GRAFIK er derimod noget nyt i COMAL til 64'eren. GRARIK pakken arbejder ud fra et normalt x-y kordinatsystem. Kommandoer som DRAWTO (x,y) og CIRCLE (x,y,radius) er indkluderet.

Man kan også bestemme højden og bredden på de bogstaver og tal man vil bruge i grafikken. Desuden kan man bestemme, hvad vej (op-ned-højre-venstre) bogstaverne skal skrives.

Ved højopløsningsgrafik har man 320*200 punkter til rådighed. Skærmen bliver delt op i 40*25 blokke på 8*8 punkter, og man kan have 2 farver indenfor hvert punkt.

Ved flerfarvegrafik har man 160*200 punkter til rådighed. Skærmen bliver delt op som nævnt ovenfor. Men nu er hver blok delt op i 8*4 punkter, og kan derfor indeholde 4 farver.

Når man kan bestemme, hvor man vil have sine ting til at stå, er det også muligt at lave en »lysavisagtig« tekst scrolling over skærmen, op eller ned, højre eller venstre. I det hele taget åbner grafiTpakken uanede muligheder for udfoldelse. Kun fantasien sætter grænserne.

Konklusion

Der er ingen tvivl om, at COMAL 80 er et stærkt og godt sprog, gennemtænkt og overskueligt.

Tager man prisen i betragtning (vejl. butikspris 2095,- kr.), er det jo nok ikke noget, man kaster sig over lige med det samme.

Men når og hvis man ikke vil bruge det mere »simple« BASIC, er COMAL 80 det bedste alternativ, der i mine øjne findes.

Vi håber fra redaktionens side, at komme tilbage med mere om dette emne.

HVAD INTERESSERER DU DIG FOR, NÅR DU IKKE HAR NÆSEN NEDE I COMPUTEREN?

Musik. Film. Idoler. Romantik. Spændende og dramatiske historier.
Det finder du hver måned i Vi Unge. Samt en masse andet.

I HVERT NUMMER EN OVERRASKELSE:
- **En eller flere plakater**
- **Måske en GIGANT-plakat**
- **Måske en lodseddel**
- **Måske en plade indlagt**
- **Måske ??**

Der sker hele tiden noget i Vi Unge.

VIL DU VÆRE MED? DET ER LET:
Skriv dit navn på kuponen og send den i lukket kuvert til Vi Unge, Hellerupvej 66, 2900 Hellerup.

Du kan vælge mellem et halvt eller et helt års abonnement.

EKSTRA

Som en ekstra service sender vi dig decembernummeret gratis, hvis vi har din bestillingskupon i hænde senest den 30. november.

Kupon
Sendes i lukket kuvert til Vi Unge, Hellerupvej 66, 2900 Hellerup.

JA ... jeg synes det er et godt tilbud. Jeg bestiller hermed et halvt års abonnement (fra januar til og med juni).
Prisen er kun 66 kr. ..... ☐
Abonnement for hele 1985. Det koster kun 130 kr. ..... ☐

NAVN _____

ADRESSE _____

POSTNR. _____ BY _____

(Vedlæg gerne check — medmindre du foretrækker at betale på giro — så sender vi girokort sammen med decembernummeret).

```
3 REM*** TOTAL MUSIC 64 ***
5 GOSUB12000
10 CLR:     REM VOICE 1 NOTES
15 DIM HI(200),LO(200),LN$(200):REM TOTA
   L NUMBER OF NOTES POSSIBLE IS 200
20 POKE53281,1:POKE53280,1:PRINTCHR$(144
   )CHR$(147)
30 UP$=CHR$(145):LF$=CHR$(157)
40 FORS=54272 TO 54296:POKES,0:NEXT
50 SV=54296:SH=54273:SL=54272:SA=54277:S
   S=54278:SW=54276
100 PRINT"VOLUME (1-15)";CHR$(32)CHR$(32
   )"15"
101 PRINTUP$;:FORX=1TO13:PRINTCHR$(29);:
   NEXTX
102 INPUTV1$
105 V1=VAL(V1$)
110 IFV1<1ORV1>15THENGOSUB7000
114 IFV1<1ORV1>15THENPRINTUP$UP$UP$:GOTO
   100
125 FORX=1TO22:PRINTCHR$(32);:NEXT:PRINT
200 PRINT"WAVEFORM"
202 PRINTTAB(10)CHR$(18)"T";CHR$(146);"R
   IANGULAR"
204 PRINTTAB(10)CHR$(18)"S";CHR$(146);"A
   WTOOTH"
206 PRINTTAB(10)CHR$(18)"P";CHR$(146);"U
   LSE"
208 PRINTTAB(10)CHR$(18)"N";CHR$(146);"O
   ISE";CHR$(32)CHR$(32)"T"
209 PRINTUP$;:FORX=1TO15:PRINTCHR$(29);:
   NEXT:INPUTW1$
210 IFW1$="T"THENW1=17:PRINT:GOTO300
212 IFW1$="S"THENW1=33:PRINT:GOTO300
214 IFW1$="P"THENW1=65:PRINT:GOTO300
216 IFW1$="N"THENW1=129:PRINT:GOTO300
220 IFW1$<>"N"THENGOSUB7000
226 IFW1$<>"N"THEN FORX=1TO6:PRINTUP$;:N
   EXT:PRINT:GOTO200
300 PRINT"ATTACK/DECAY (0-240)";CHR$(32)
   CHR$(32)"0"
302 PRINTUP$;:FORX=1TO20:PRINTCHR$(29);:
   NEXTX:INPUTA1$
310 A1=VAL(A1$)
320 IFA1<0ORA1>240THENGOSUB7000
330 IFA1<0ORA1>240THENPRINTUP$UP$:GOTO30
   0
400 PRINT"SUSTAIN/RELEASE (0-240)";CHR$(
   32)CHR$(32)"240"
402 PRINTUP$;:FORX=1TO23:PRINTCHR$(29);:
   NEXTX:INPUTS1$
410 S1=VAL(S1$)
420 IFS1<0ORS1>240THEN GOSUB7000
430 IFS1<0ORS1>240THENPRINTUP$UP$:GOTO40
   0
450 PRINT:PRINT"DURATION OF NOTE";CHR$(3
   2)CHR$(32)"128"
451 PRINTUP$;:FORX=1TO16:PRINTCHR$(29);:
   NEXTX:INPUTD1$
452 IFLEFT$(D1$,1)<"0"ORLEFT$(D1$,1)>"9"
   THENGOSUB7000
454 IFLEFT$(D1$,1)<"0"ORLEFT$(D1$,1)>"9"
   THENPRINTUP$UP$UP$:GOTO450
456 D1=VAL(D1$):PRINT
458 FORX=1TO400:NEXTX
460 PRINTCHR$(147)CHR$(18)"F3"CHR$(146)"
    PRINT & PLAY";
461 PRINTTAB(24)CHR$(18)"F7"CHR$(146)" R
   ESTART"
462 PRINTCHR$(18)" P"CHR$(146)" PAUSE";
463 PRINTSPC(12)CHR$(18)"RETURN"CHR$(146
   )" LINE FEED"
464 PRINTCHR$(18)" "CHR$(94)CHR$(146)" E
   RASE ALL";
465 PRINTSPC(6)CHR$(18)"INST/DEL"CHR$(14
   6)" ERASE NOTE" :PRINT:PRINT
467 FORMD=0TO39:POKE1624+MD,67:POKE55896
   +MD,12:NEXTMD
480 GETA$:IFA$=""THEN480
499 REM **** A$ SELECTION FOR NOTES-
                          OCTAVE 4
500 IFA$=CHR$(65)THEN HI=28:LO=49:GOTO70
   0: REM  A
510 IFA$=CHR$(193)THEN HI=29:LO=223:A$="
   A#":GOTO700:REM  A#
520 IFA$=CHR$(66)THEN HI=15:LO=210:GOTO7
   00: REM  B ** NOTE OCTAVE 3 NOT 4 **
   *
530 IFA$=CHR$(67)THEN HI=16:LO=195:GOTO7
   00: REM  C
540 IFA$=CHR$(195)THEN HI=17:LO=195:A$="
   C#":GOTO700: REM  C#
550 IFA$=CHR$(68)THEN HI=18:LO=209:GOTO7
   00: REM  D
560 IFA$=CHR$(196)THEN HI=19:LO=239:A$="
   D#":GOTO700:REM  D#
570 IFA$=CHR$(69)THEN HI=21:LO=31:GOTO70
   0:  REM  E
580 IFA$=CHR$(70)THEN HI=22:LO=96:GOTO70
   0: REM  F
590 IFA$=CHR$(198)THEN HI=23:LO=181:A$="
   F#":GOTO700: REM  F#
600 IFA$=CHR$(71)THEN HI=25:LO=30:GOTO70
   0: REM  G
610 IFA$=CHR$(199)THEN HI=26:LO=156:A$="
   G#":GOTO700: REM  G#
620 IFA$=CHR$(80)THEN HI=0:LO=0:A$="P":G
   OTO700: REM  PAUSE
635 IFA$=CHR$(134)THEN9000:REM F3 GOTO
   PUTTING DATA INTO ARRAYS
640 IFA$=CHR$(136)THEN10:REM F7 RESTART
645 IFA$=CHR$(13)THENPRINTUP$CHR$(13):GO
   TO480:REM RETURN- LINE FEED
650 IFA$=CHR$(20)THENHI=0:LO=0:PRINTCHR$
   (157)CHR$(157)CHR$(32);:GOTO480
651 REM LINE 650 -INST/DEL ERASE SINGLE
   NOTE
660 IFA$=CHR$(94)THENGOTO460: REM UP ARR
   OW SYMBOL CLEAN UP SCREEN
690 IFA$<>CHR$(80)THEN HI=0:LO=0:GOTO480
700 POKESH,HI:POKESL,LO:POKESA,A1:POKESS
   ,S1:POKESW,W1:POKESV,V1
740 FORX=1TOD1:NEXTX
750 PRINTA$;" ";:POKESV,16:POKESW,W1-1
800 GOTO480
1999 REM *** PLAY THE TUNE ***
2000 IFN=0 THEN RETURN
2005 FORX=54272 TO 54296:POKES,0:NEXTX
2010 FORNN=0TON-1:POKESH,HI(NN):POKESL,L
   O(NN)
2020 POKESA,A1:POKESS,S1:POKESW,W1:POKES
   V,V1
2030 FORX=1TOD1:NEXTX
2040 NEXTNN:POKESV,16:POKESW,W1-1
2050 RETURN
2999 REM *** PRINT NOTES USING LETTERS *
   ******
3000 PRINT
3002 QT$="NOTES BY LETTER":GOSUB10000:FO
   RX=1TO500:NEXTX
3005 QT$="TURN PRINTER ON":GOSUB 10000:F
   ORX=1TO500:NEXTX
3010 QT$="GIVE TUNE A NAME":GOSUB10000:P
   RINTUP$:PRINTTAB(16);:INPUTAW$
```

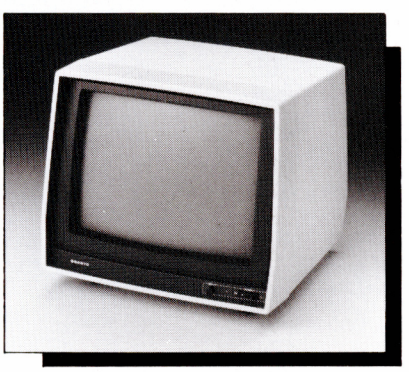
Hurra for den store forskel

Er du den lykkelige ejer af en Commodore 64 eller Atari 600? Så har Sanyo alle tiders tilbud til dig: En helt nyudviklet 14'' farve data monitor, den eneste på markedet, som kan tilsluttes direkte til din computer. Med et stabilt billede og helt skarp tekstgengivelse, så du kan arbejde i timevis uden at blive træt i øjnene. Med klare farver og semi-professionel grafik fremstilling.
Ring eller skriv, så sender vi brochure.

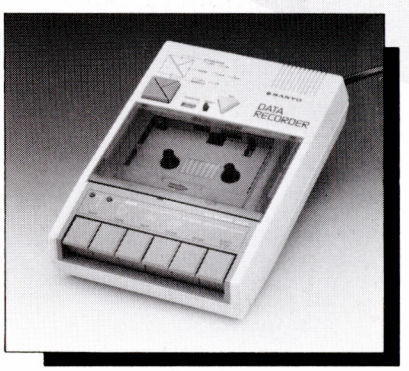
Tekniske data:
Horisontal opløsning: 340 punkter. 64×25 karakterer. Horisontal frekvens 15.750 Hz. PAL-composite-video indgang. Omskifter til grøn tekstdisplay, audioindgang og indbygget højttaler. Model 3195 C har separat chrominance, luminance og audioindgang via phonostik. Model 3185 A har PAL-video og audioindgang via phonostik og RGB analog indgang via 21-benet SCARTstik. Typisk butikspris 3.999,- incl. moms.

Stort udvalg i data udstyr, f.eks.:

12'' data monitor *med 15 MHz billedopløsning og antireflexbehandlet billedskærm, der giver det ''rigtige'' databillede. DM-2112 grøn grafik. DM-2212 gul grafik. Typisk butikspris* ***1.499,-*** *incl. moms.*



12'' professionel data monitor *med 20MHz billedopløsning der giver et perfekt, skarpt billede af både tekst og grafik. DM-8112 grøn grafik. DM-8212 gul grafik. Typisk butikspris* ***2.499,-*** *incl. moms.*



DR 201 data kassetterecorder *er specielt udviklet til databrug. Bl.a. er den forsynet med faseskiftknap og automatisk programsøger. Fast udgangsniveau på 1 og 3 volt. Typisk butikspris* ***799,-*** *incl. moms.*

SANYO

Sanyo Denmark K/S, Literbuen 11, 2740 Skovlunde.
Tlf. (02) 84 29 99

Ted Bates Retail

```
3020 OPEN1,4
3030 PRINT#1,AW$:PRINT#1
3040 FORNN=0TON-1:PRINT#1,LN$(NN);:PRINT
     #1," ";:NEXTNN
3050 PRINT#1:PRINT#1,"TOTAL NOTE COUNT="
     ;NN;"PLUS -1,-1"
3090 PRINT#1:CLOSE1:GOTO9200
3095 PRINTUP$;:FORX=0TO39:PRINTCHR$(32);
     :NEXTX:PRINTUP$UP$UP$:PRINT
3100 QT$="FINISHED":GOSUB10000:GOTO9200
7000 QT$="NOT A LEGAL CHOICE":GOSUB10000
     :FORX=1TO50:NEXTX
7010 PRINTUP$:FORX=1TO30:PRINTCHR$(32);:
     FORY=1TO25:NEXTY:NEXTX:RETURN
7999 REM *** DATA LINE ONLY PRINT ***
8000 PRINT
8002 QT$="PRINT DATA ONLY":GOSUB10000:FO
     RX=1TO500:NEXTX
8005 QT$="TURN PRINTER ON":GOSUB10000:FO
     RX=1TO500:NEXTX
8010 QT$="GIVE TUNE A NAME":GOSUB10000:P
     RINTUP$:PRINTTAB(16);:INPUTAW$
8020 OPEN1,4
8030 PRINT#1,AW$:PRINT#1
8040 FORNN=0TON-1:PRINT#1,HI(NN);",";LO(
     NN);",";:NEXTNN:PRINT#1,"-1,-1"
8045 PRINT#1:PRINT#1,"TOTAL NOTE COUNT="
     ;NN*2;"PLUS -1,-1"
8050 PRINT#1:CLOSE1:RETURN
8499 REM *** TOTAL PROGRAM PRINT ***
8500 PRINT
8510 QT$="PRINT ROUTINE":GOSUB10000:FORX
     =1TO500:NEXTX
8520 QT$="TURN PRINTER ON":GOSUB10000:FO
     RX=1TO500:NEXTX
8530 QT$="GIVE TUNE A NAME":GOSUB10000:P
     RINTUP$:PRINTTAB(16);:INPUTAW$
8550 OPEN1,4
8560 PRINT#1,AW$
8565 PRINT#1
8570 PRINT#1,"100 FOR S=54272 TO 54296 :
     POKE S,0 : NEXT S"
8580 PRINT#1,"110 READ SH,SL"
8590 PRINT#1,"120 IF SH=-1 THEN 180"
8600 PRINT#1,"130 POKE 54273,SH : POKE 5
     4272,SL"
8610 PRINT#1,"140 POKE 54277,";A1;": POK
     E 54278,";S1;": POKE 54276,";W1
8620 PRINT#1,"150 POKE 54296,";U1
8630 PRINT#1,"160 FOR X=1 TO ";D1;": NEX
     T X"
8640 PRINT#1,"170 GOTO 110"
8645 PRINT#1,"180 POKE 54296,0 : POKE 54
     276,";W1-1
8650 PRINT#1,"200 DATA ";
8660 FORNN=0TON-1:PRINT#1,HI(NN);",";LO(
     NN);",";:NEXTNN:PRINT#1,"-1,-1"
8690 PRINT#1:PRINT#1,"TOTAL NOTE COUNT="
     ;NN*2;"PLUS -1,-1"
8700 PRINT#1:CLOSE1:GOTO9200:REM BACK FO
     R ANOTHER A$ **************
8999 REM *** PUT DATA INTO ARRAYS ***
9000 PRINTCHR$(19);:FORX=0TO199:PRINTCHR
     $(32);:NEXTX
9005 PRINTCHR$(19);
9010 PRINTCHR$(18)"F1"CHR$(146)" DATA ON
     LY";
9011 PRINTTAB(15)CHR$(18)"F2"CHR$(146)"
     TOTAL PRINT"
9020 PRINTUP$;CHR$(18)"F3"CHR$(146)" PRA
     CTISE";
9022 PRINTSPC(4)CHR$(18)"F4"CHR$(146)" P
     RINT NOTES"
9025 PRINTCHR$(18)"F5"CHR$(146)" PLAY TU
     NE";
9030 PRINTTAB(15)CHR$(18)"F6"CHR$(146)"
     DURATION"
9031 PRINTUP$;CHR$(18)"F7"CHR$(146)" RES
     TART"
9032 PRINTCHR$(18)" P"CHR$(146)" PAUSE";
9035 PRINTSPC(7)CHR$(18)" "CHR$(94)CHR$(
     146)" ERASE ALL"
9040 PRINTCHR$(19):FORX=1TO11:PRINT:NEXT
     X:REM POSITION LINE ON GREY LINE
9100 PRINT"INPUT NOTES INTO ARRAY ROUTIN
     E";UP$
9150 A$=CHR$(94):GOTO9345: REM GET SCREE
     N AND N SET UP PROPERLY
9200 GETA$:IFA$=""THEN9200
9210 IFA$=CHR$(65)THEN HI(N)=28:LO(N)=49
     :GOTO9400: REM  A
9220 IFA$=CHR$(193)THEN HI(N)=29:LO(N)=2
     23:A$="A#":GOTO9400: REM  A#
9230 IFA$=CHR$(66)THEN HI(N)=15:LO(N)=21
     0:GOTO9400: REM  B *NOTE OCTAVE 3 N
     OT 4*
9240 IFA$=CHR$(67)THEN HI(N)=16:LO(N)=19
     5:GOTO9400: REM  C
9250 IFA$=CHR$(195)THEN HI(N)=17:LO(N)=1
     95:A$="C#":GOTO9400: REM  C#
9260 IFA$=CHR$(68)THEN HI(N)=18:LO(N)=20
     9:GOTO9400: REM  D
9270 IFA$=CHR$(196)THEN HI(N)=19:LO(N)=2
     39:A$="D#":GOTO9400: REM  D#
9280 IFA$=CHR$(69)THEN HI(N)=21:LO(N)=31
     :GOTO9400: REM  E
9290 IFA$=CHR$(70)THEN HI(N)=22:LO(N)=96
     :GOTO9400: REM  F
9300 IFA$=CHR$(198)THEN HI(N)=23:LO(N)=1
     81:A$="F#":GOTO9400: REM  F#
9310 IFA$=CHR$(71)THEN HI(N)=25:LO(N)=30
     :GOTO9400: REM  G
9320 IFA$=CHR$(199)THEN HI(N)=26:LO(N)=1
     56:A$="G#":GOTO9400: REM  G#
9330 IFA$=CHR$(80)THEN HI(N)=0:LO(N)=0:G
     OTO9400: REM  P  PAUSE
9335 IFA$=CHR$(133)THENGOSUB8000:GOTO920
     0: REM F1 PRINT DATA ONLY
9340 IFA$=CHR$(137)THENGOSUB8500:GOTO920
     0: REM F2 PRINT ROUTINE
9341 IFA$=CHR$(134)THENPRINTCHR$(147):N=
     0:GOTO458: REM F3 TO PRACTISE PLAY
9342 IFA$=CHR$(135)ANDN=0THEN9200
9343 IFA$=CHR$(135)THENGOSUB2000:GOTO920
     0: REM F5 PLAY THE TUNE
9344 IFA$=CHR$(136)THEN10: REM F7 START
     ALL OVER
9345 IFA$=CHR$(94)THEN10090
9346 IFA$=CHR$(138)THENGOTO3000: REM F4
     - PRINT NOTES USING LETTERS
9347 IFA$=CHR$(139)THENPRINT:PRINT"DURAT
     ION";:INPUT D1:GOTO9200: REM F6 DUR
     ATION
9350 IFA$<>CHR$(80)THENGOTO9200
9400 POKESH,HI(N):POKESL,LO(N):POKESA,A1
     :POKESS,S1:POKESW,W1:POKESV,V1
9410 FORX=1TOD1:NEXTX
9420 PRINTA$;" ";:POKESV,16:POKESW,W1-1:
     LN$(N)=A$: REM PUT LN - LIST NOTES
9440 N=N+1:GOTO9200: REM INCREASE ARRAY
     COUNT & GO BACK FOR ANOTHER NOTE
9999 REM *** THIS IS SCROLL TYPE ROUTINE
     ******************
10000 FORI=0TOLEN(QT$)
10010 POKESV,7:POKESH,126:POKESL,6:POKES
      A,120:POKESS,30:POKESW,33
10020 PRINTLEFT$(QT$,I+1):POKESV,0:POKES
      W,32
10030 PRINTUP$UP$:FORX=1TO40:NEXTX:NEXTI
10040 FORS=54272TO54296:POKES,0:NEXTS
```

```
10050 FORX=1TO100:NEXTX:RETURN
10089 REM *** CLEAN UP ARRAY SCREEN ****
      ************
10090 ER=N+29: REM ER = ERASE NOTES IN A
      RRAY
10100 FORMD=0TOER:POKE1664+MD,160:POKE16
      64+MD+1,62
10105 POKE55936+MD,1:POKE55936+MD+1,2:NE
      XTMD
10110 PRINTCHR$(19):FORX=1TO12:PRINT:NEX
      TX
10120 N=0:GOTO9200
11999 REM *** TITLE SCREEN ************
      ************
12000 PRINTCHR$(147)CHR$(5):POKE53281,0:
      POKE53280,0
12002 FORS=54272TO54296:POKES,0:NEXT
12004 SU=54296:SH=54273:SL=54272:SA=5427
      7:SS=54278:SW=54276:UP$=CHR$(145)
12010 FORX=1TO5:PRINT:NEXTX
12020 PRINT"TOTAL MUSIC 64":PRINT
12030 QT$="1984":GOSUB10000:PRINT:PRINT
12040 QT$="BY GARY FIELDS":GOSUB10000:FO
      RX=1TO500:NEXTX
12050 PRINTUP$UP$UP$:QT$="VOICE 1 / OCTA
      VE 4":GOSUB10000:PRINT:PRINT
12060 QT$="THIS AID CONVERTS SOUND TO DA
      TA":GOSUB10000:PRINT:PRINT
12065 QT$="F1 - DATA ONLY PRINT":GOSUB10
      000:PRINT
12066 QT$="F2 - TOTAL PROGRAM W/DATA":GO
      SUB10000:PRINT
12067 QT$="F4 - CONVERTS DATA TO NOTES":
      GOSUB10000:PRINT
12070 FORX=1TO500:NEXTX
12080 PRINTCHR$(147)
12090 PRINTCHR$(18)" LETTER =NOTE
      :PRINT
12100 PRINT"       A = A    "
12102 PRINT"       B = B    "
12104 PRINT"       C = C    "
12106 PRINT"SHIFT/C = C#   "
12108 PRINT"       D = D    "
12110 PRINT"SHIFT/D = D#   "
12112 PRINT"       E = E    "
12114 PRINT"       F = F    "
12116 PRINT"SHIFT/F = F#   "
12118 PRINT"       G = G    "
12120 PRINT"SHIFT/G = G#   "
12122 PRINT"       P = PAUSE":PRINT
12124 PRINTCHR$(18)"OTHER KEYS ARE EMPTY
      ":FORX=0TO39:PRINTCHR$(45);:NEXTX
12125 PRINT"UP TO 200 NOTES CAN BE CONST
      RUCTED"
12128 QT$="NOTES CAN BE ":GOSUB10000:PRI
      NT
12129 QT$="PLAYED AS NOTES":GOSUB10000:F
      ORX=1TO100:NEXTX
12132 QT$="PRINTED AS DATA":GOSUB10000:F
      ORX=1TO100:NEXTX
12134 QT$="LISTED AS NOTES":GOSUB10000:F
      ORX=1TO100:NEXTX
12136 QT$="PRINTED AS A PROGRAM INCLUDIN
      G DATA":GOSUB10000:PRINT
12200 PRINT"(ANY KEY) TO BEGIN":
12210 GETA$:IFA$=""THEN12210
12500 RETURN : REM **** BACK TO MAIN PRO
      GRAM *********
```

KONTROLSUM FOR TOTAL MUSIC

```
3     122   5     128   10    117
15    52    20    95    30    132
40    180   50    67    100   201
101   64    102   48    105   250
110   63    114   51    125   222
200   68    202   241   204   113
206   129   208   67    209   85
210   243   212   240   214   242
216   33    220   40    226   69
300   55    302   147   310   208
320   68    330   113   400   167
402   168   410   244   420   136
430   150   450   53    451   155
452   169   454   165   456   169
458   8     460   6     461   73
462   158   463   227   464   190
465   72    467   111   480   105
499   149   500   81    510   170
520   119   530   132   540   213
550   132   560   185   570   105
580   78    590   214   600   71
610   222   620   249   635   110
640   7     645   116   650   65
651   233   660   104   690   36
700   172   740   233   750   240
800   37    1999  126   2000  48
2005  17    2010  7     2020  76
2030  233   2040  54    2050  142
2999  178   3000  153   3002  238
3005  36    3010  248   3020  48
3030  180   3040  8     3050  158
3090  98    3095  66    3100  79
7000  44    7010  221   7999  92
8000  153   8002  227   8005  4
8010  248   8020  48    8030  180
8040  129   8045  124   8050  156
8499  84    8500  153   8510  141
8520  4     8530  248   8550  48
8560  177   8565  201   8570  246
8580  141   8590  217   8600  160
8610  91    8620  54    8630  224
8640  220   8645  67    8650  96
8660  129   8690  124   8700  91
8999  106   9000  49    9005  86
9010  163   9011  80    9020  136
9022  40    9025  189   9030  137
9031  27    9032  158   9035  137
9040  0     9100  89    9150  204
9200  152   9210  197   9220  62
9230  45    9240  248   9250  73
9260  248   9270  77    9280  189
9290  194   9300  74    9310  187
9320  82    9330  24    9335  164
9340  88    9341  42    9342  164
9343  14    9344  229   9345  200
9346  88    9347  60    9350  207
9400  234   9410  233   9420  172
9440  13    9999  162   10000 45
10010 221   10020 173   10030 66
10040 199   10050 205   10089 222
10090 61    10100 27    10105 217
10110 255   10120 190   11999 19
12000 249   12002 116   12004 170
12010 124   12020 95    12030 243
12040 127   12050 151   12060 125
12065 87    12066 171   12067 75
12070 9     12080 77    12090 116
12100 252   12102 254   12104 0
12106 240   12108 2     12110 242
12112 4     12114 6     12116 246
12118 8     12120 248   12122 187
12124 139   12125 167   12128 140
12129 210   12132 186   12134 216
12136 164   12200 161   12210 195
12500 64
```

Elektroniske regneark (spreadsheet) til 64'eren

Af Robin Sagar

Et elektronisk regneark (spread-sheet) er EDB-udgaven af papir- og blyantmetoden, som bankfolk, regnskabsfolk og investeringsanalytikere anvender når de kalkulerer budgetter og udarbejder forecasts og investeringsplaner.

Den nemmeste måde at forstå det elektroniske regneark og dets måde at fungere på, er ved at forestille sig et stykke papir inddelt i rækker og kolonner/søjler, hver med en given label eller anden form for identifikation, henholdsvis yderst til venstre i hver række og øverst i hver kolonne. Papiret/arket består nu af et vist antal rubrikker, som herefter vil blive kaldt »celler«. I hver celle kan der indtastes beskrivelser (tekst) eller tal (regneudtryk/formler).

Ideen i regnearket er, at man i den øverste række og den yderste venstre kolonne indtaster tekst, der beskriver indholdet i de øvrige rubrikker ned over arket/papiret.

I disse rubrikker kan der indtastes tal og/eller regneudtryk af næsten enhver art. Ved så at lægge til, trække fra, gange osv. på tværs af rækkerne og ned gennem kolonnerne danner der sig et billede af f.eks. indkomst og udgifter. Dette er simpelt nok, men det der gør arbejdet med »det manuelle regneark« (papir og blyant) kedeligt og tidskrævende er, at forandre tallene når de aktuelle tal har ændret sig eller hvis man ønsker at undersøge, hvad der vil ske, hvis en eller flere parametre ændres. Med det manuelle system er det nødvendigt at viske gamle tal ud, skrive nye ned og så regne hele papiret eller dele af det om igen.

Det er netop i denne forbindelse (ændringer i faktorer/parametre), at det elektroniske regneark kommer til sin ret.

Når først det originale regneark er dannet på computeren, er det nemt at foretage ændringer. Alt hvad der skal til, er at flytte cursoren hen til den rubrik, der skal ændres. Her tastes nye tal ind, og computeren foretager de nye beregninger i løbet af få sekunder. Der findes i dag forskellige regnearkssystemer, der kan køre på Commodore 64. De giver alle mulighed for mindst et regneark på 16002 celler fordelt på 63 kolonner og 254 rækker.

De fleste af systemerne indeholder endvidere et sæt af standard hjælpefunktioner, til at printe og lagre modellerne på bånd eller diskette. De almindeligste funktioner (som næsten alle systemer indeholder) er vist i figur 1, og nogle af disse forklares her:

»Replicate«-funktionen bruges til at kopiere indholdet af en celle eller gruppe af celler over i en anden celle eller gruppe af celler. I visse systemer kan denne funktion også bruges til at kopiere formler til andre dele af regnearket. Dette kan være meget tidsbesparende.

Der vil normalt være en »goto« – eller »jump«-funktion, som gør det nemt og hurtigt at bevæge sig fra den ene ende af regnearket til den anden. I de mere avancerede systemer, er der endvidere en »find«-funktion, som kan bruges til hurtigt at finde bestemte ord og formler.

En anden nyttig funktion i nogle systemer er, muligheden for at ændre regnearkets farve.

Måske en af de vigtigste funktioner, som visse systemer har, er »insert«- og »delete«-funktionerne, som dog fungerer forskelligt fra system til system. Når de fungerer bedst, giver de brugeren mulighed for, at ændre regnearket uden at lave om på alle de formler, der har betydning for den »nye opgave«. Personligt tror jeg, at »insert« og »delete« som oftest er meget tidsbesparende, f.eks. når man finder på at forbedre, flytte eller tilføje ting, til et allerede opstillet regneark.

En anden funktion jeg finder brugbar, er »process on/off«-om-skifteren, som kan bruges til at starte og stoppe en igangværende omkalkulering. Når der ændres i større regneark, kan det ofte være irriterende at vente de fem eller seks sekunder, som de fleste omkalkuleringer tager. Ved at benytte »process off« kan de nye tal indtastes, og alle omkalkuleringer foretages derefter på én gang.

Alle de regneark-systemer som jeg har undersøgt, har endvidere en »help«-funktion, som er meget anvendelig, men i nogle systemer ikke særlig avanceret. De forskellige systemers hjælpefunktioner varierer lige fra et enkelt skærmbillede med en oversigt over kommandoer, til en nøje beskrivelse af den pågældende funktion, man var ved at bruge, da man tastede »help«. En oversigt over de undersøgte systemer incl. priser er vist i fig. 2 og en kort beskrivelse af hver enkelt følger nedenfor:

CalcResult Easy og CalcResult Advanced

Disse to systemer er tilgængelige fra det svenske firma Handic. Easy-versionen er et cartridge-baseret system, som giver et standard regneark på 16002 celler, mens Advanced-udgaven er diskette-baseret, som kan give op til 32 siders regneark eller, hvad der svarer til over en million celler. De grundlæggende kommandoer er de samme for begge versioner, og regneark udviklet på Easy-versionen kan genbruges på Advanced-versionen. Forskellen er, at Advanced-versionen foruden flere kommandoer er mere fleksibelt.

En af de gode funktioner som er fælles for begge systemer, er muligheden for at bruge funktionstasterne, hvor F3 er »goto«, F6 skærmbilled-print, med F7 opnås CalcResult-funktioner og F8 bruges til at rette linien eller cellen, hvor cursoren står. Der er endvidere mulighed for lagring på enten tape eller diskette, samt at lave flerfarvede regneark.

Omnicalc

Dette er det første af to systemer fra Hesware. Omnical er et diskette-baseret system, som indeholder både regnearks-systemet Calc 64 og Plot 64 til grafik. Begge systemer er menu-styret, og man kan gå tilbage i menuen på et vilkårligt tidspunkt ved at bruge .

Plot 64 kan danne lagkagediagrammer, søjlediagrammer, kurver eller grafer enten fra direkte dannede filer, eller ved filer overført fra Calc 64.

Calc 64 indeholder alle standardfunktionerne, og rummer herudover mulighed for at danne flere regneark.

Desuden findes funktionerne »protect«, som gør det muligt, specielt at beskytte indholdet af en række eller kolonne og »invisible«, som jeg mener, er til stor hjælp. Med denne funktion er det muligt at skjule mellemregninger, som det kan være praktisk at have på selve regnearket, men som man ikke ønsker printet. Printede resultater fra Omnicalc er altid kun 80 tegn i bredden, men ved at give en speciel printkommando kan man specificere op til tre serier af kolonner, således at den første kolonne, på hvert stykke papir er beskrivelseskolonnen, selvom det originale regneark er meget bredere.

En ting, der virker irriterende i Omnicalc er, at man skal selv give, hver række man forventer at skulle bruge, en identifikation selvom start-skærmbilledet i sig selv indeholder en nummerering af rækkerne fra 1 til 255.

Tabel 1.

Funktion	CalcResult Advanced	CalcResult Easy	Plan Manager	Omnicalc	Mikrosoft Multiplan
Add (Row/Col)	★	★	★	★	★
Blank -,,-	★	★	★		★
Copy -,,-	★	★		★	★
Delete -,,-	★	★	★	★	★
Find (Id)	★	★			★
Help	★	★	★	★	★
Insert (Row/Col)	★	★	★	★	★
Invisible				★	
Goto	★	★	★	★	★
Move (Row/Col)	★	★	★		★
Name					★
Process on/off	★	★		★	★
Protect	★		★		★
Replicate	★	★	★	★	★
Split	★				★
Window	★				★
Multiple Sheets	★		★		★

Tabel 2.

Spreadsheet	Firma	Udsalgspris
CalcResult Advanced	Handic	1998,00
CalcResult Easy	Handic	998,00
Plan Manager	G. A. Import	1950,00
Omnical	Hesvare	1185,00
Microsoft Multiplan	Hesware	998,00

Microsoft Multiplan

Dette system kaldes ofte for »Rolls Royce«-modellen inden for regnearkssystemer til CBM 64. Dette system er i familie med det berømte VisiCalc, som oprindeligt blev udviklet til brug på Apple og IBM PC. Det er et meget imponerende system, som foruden en meget omfattende hjælpefunktion, har funktioner, der kan kæde flere regneark sammen, ændre regnearkets farve og omforme lagrede regneark fra Commodore ASCII til standard ASCII, således at ens regneark også kan køre på andre computere.

Hele software-pakken består af et A5-ringbind, som indeholder en velskrevet og illustreret brugervejledning/manual på ca. 430 sider. Desuden er der et oversigtskort og en skabelon til funktionstasterne.

Udover standardfunktionerne som tidligere er beskrevet, har Multiplan faciliteter, så man selv individuelt kan bestemme kolonnebredderne, og til at fastholde titlerne på skærmen. Dette betyder, at selv om man arbejder i den tiende kolonne, kan man stadig se beskrivelsen for den række, man er ved at taste tal ind i.

At udføre beregninger er meget enkelt. Når man har tastet »=« (for at markere starten på en ny formel), placeres cursoren i den første celle, som skal indgå i formlen. Derefter trykker man »+«,»–«,»*« eller »/«, hvorefter cursoren automatisk rykker tilbage til den celle, som skal indeholde resultatet af formlen. Derefter flyttes cursoren til den næste celle i formlen. Dette gentages, indtil formlen er færdig. Herefter trykkes return. På alle stadier af formel-beregning vises opbygningen af formlen på nederste linie af skærmen. Multiplan er endvidere det eneste af de afprøvede systemer, som giver brugeren mulighed for at udskrive de formler, der indgår i et regneark, hvilket kan være til stor hjælp, når man senere opbygger nye regneark.

En af de store nyskabelser ved Multiplan er muligheden for at opbygge formler ved hjælp af navne. Dette gøres ved at give den ønskede række eller kolonne eller celle et navn (v.h.a. »name«-funktionen). Herefter opstilles en formel ved f.eks. at sige:

TOTAL = INDTÆGTER-UDGIFTER.

Multiplan har også en såkaldt »window«-funktion, som kan bruges til at »indramme« en større eller mindre del af regnearket. Denne funktion er specielt en fordel i forbindelse med store regneark, idet man ved at bruge kommandoen: »next window« meget hurtigt kan bevæge sig fra et »window« til et andet, uden at skulle bruge cursoren eller »goto«-kommandoen.

Multiplan er også et af de eneste regneark-systemer, der tillader flere regneark, og i modsætning til de fleste andre systemer, der har denne facilitet, giver Multiplan mulighed for at opbygge et »master«-regneark foruden et antal underordnede ark, hvorfra man kan overføre rækker og kolonner til »masteren« ved hjælp af den før omtalte »name«-kommando og ved at bruge »external copy«-funktionen.

Jeg mener, dette virkelig er tegn på et avanceret regneark-system, som ikke kun har de personlige brugere som målgruppe, men som også er beregnet til anvendelse i små og mellemstore virksomheder.

Et andet tegn på Multiplans »professionalisme« er at det muliggør udskrift med op til 160 tegn (i bredden).

At forsøge at finde frem til det bedste regneark-system er næsten umuligt, idet hvert system har sine gode og dårlige sider, alt afhængig af brugerens ønsker og behov.

Multiplan er helt afgjort det bedste af de systemer, jeg har afprøvet, men for mange vil det måske være for avanceret. Hvis man skal vælge et system, må man først forsøge at afdække sine behov og gøre sig klart, hvor meget man vil betale, og derefter rådføre sig med den lokale computer-forretning.

PETSPEED 84

Lunter dit dødbringende raketmissil af sted med 20 km i timen? Tager det et kvarter at få sorteret dine data? Kort sagt – kører dit BASIC program for langsomt?

Umiddelbart ville du få det råd at sætte dig ind i maskinkodeprogrammering, men det må nok erkendes, at det ikke er alle, der har tid og lyst til det.

Løsningen er måske så at anskaffe PETSPEED 64 – compileren, der (ifølge annoncerne) kan få dit BASIC-program til at køre op til 40 gange hurtigere.

Hvad er en compiler?

En compiler er et program, der »oversætter« et såkaldt SOURCE PROGRAM, der i dette tilfælde er et BASIC-program, til et OBJECT PROGRAM d.v.s. et nyt program, som compileren fremstiller. Som regel er Object programmet skrevet i maskinkode, men PETSPEED 64 anvender en lidt anden teknik, der gør, at slutresultatet ikke bliver så hurtigt som maskinkode, men alligevel opnås en væsentlig hastighedsforøgelse.

Hvad består programmet af?

PETSPEED 64 findes kun i diskette-version, og koster 565 kr. Programmet består af en diskette, en brugervejledning (engelsk) og en såkaldt »dongle«, der er et lille modul, der anbringes i stikket til joystick 2 (nærmest afbryderen). Modulet er naturligvis kun lavet for at vanskeliggøre »piratkopiering«.

Ud over selve programmet, skal der bruges 4 nye disketter, så det er en yderligere udgiftspost, man er nødt til at budgettere med.

Brugervejledningen

– Jeg glæder mig til RUN nr. 2 udkommer. Jeg har købt PETSPEED-programmet, og jeg kan ikke engelsk. Forretningen, hvor jeg købte det, kan ikke hjælpe mig, og jeg aner ikke hvordan jeg skal bruge det!

Ovenstående nødråb er ikke opdigtet. Det er desværre et helt autentisk referat af en telefonsamtale med en læser, og desværre er det et helt generelt problem.

Jeg kan ikke forstå, hvorfor man ikke oversætter brugervejledninger eller i det mindste laver en kortfattet dansk vejledning. Jeg tror ikke på, at det er fordi der ikke er penge til det! Bortset fra sproget er brugervejledningen fyldestgørende, og efter et par gennemlæsninger var meningen sivet ind.

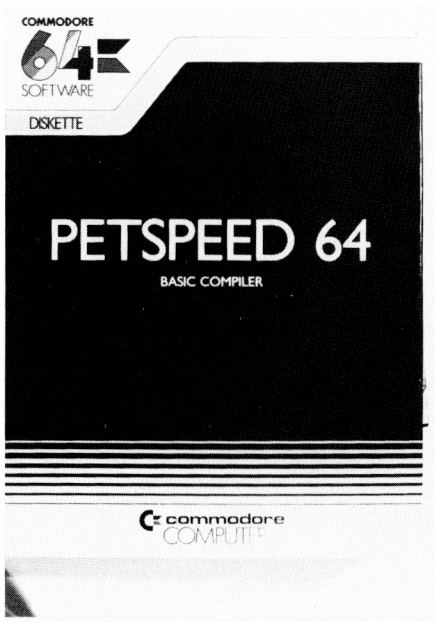
Opstarten

Man starter selvfølgelig med at formattere sine 4 hjælpedisketter. De benævnes henholdsvis PETSPEED MASTER, PETSPEED WORK, UTILITIES MASTER og UTILITIES WORK. De to sidstnævnte er til nogle hjælpeprogrammer og er ikke strengt nødvendige i første omgang. Selve den diskette, der følger med programmet benævnes SYSTEM DISK.

Man begynder med at LOADe »Backup«,8 og efter en RUN bliver man præsenteret for den menu, der bruges til at lave de 4 hjælpedisketter.

Man starter med at lave en PETSPEED MASTER. Det sker ved at indlæse programmer fra SYSTEM disken til computeren og derfra til MASTER disken. Programmet fortæller hele tiden, hvilken diskette, man skal anbringe i diskettestationen. Man hiver disketter ud og ind af diskettestationen – det er ikke svært, men det tager sin tid. Derefter skal der indlæses fra MASTER disken til WORK disken og det tager lige så lang tid. Samme fremgangsmåde bruges fra SYSTEM DISK til UTILITIES MASTER og så igen fra UTILITIES MASTER til UTILITIES WORK.

Man kan roligt afsætte tre kvarter til hele denne procedure, men det er kun en engangsforestilling.



BASIC-programmet

Der stilles nogle krav til det program, man ønsker speedet op. Det skal for det første være et »rent« BASIC-program. Det skal være »afluset« og fri for fejl. Det skal også være skrevet i Commodore »standard« BASIC, d.v.s. at programmer, der er skrevet i f.eks. SIMONS BASIC, ikke kan anvendes. Der er nogle ganske få andre begrænsninger som f.eks. en dimension må ikke være over to dimensioner (f.eks. DIM TT(25,100)) og dimensioneringen skal ske med tal og ikke med en variabel.

Programmet må heller ikke være ret meget over 20Kbytes, hvis man »kun« har en enkelt diskettestation, men ellers kan man bare køre løs.

Selve proceduren

Man indlæser sit BASIC-program på PETSPEED WORK disketten og programnavnet må ikke overstige 12 karakterer.

Derefter loader man »PETSPEED«,8 (fra samme diskette) og man taster RUN. Derefter besvarer man spørgsmål om, hvilken disettestation man har, og hvad programmet hedder.

Så kører programmet – og det gør det i omkring 10 minutter. Man kan følge med på skærmen, hvor langt programmet er nået. Det skal gennem PASS 1-4 og i hvert tilfælde vises det aktuelle linienummer, programmet er nået til. Hvis man ikke ved, hvad »PASS« betyder, fortæller Webster Computerordbog, at »PASS er en

komplet cyklus i udførelsen af et program«.

Kører programmet fejlfrit igennem, og det gør det, hvis vores BASIC-program (source-program) er i orden, bliver slutresultatet et såkaldt OBJECT-program, der hedder det samme som det oprindelige program med tilføjelsen ».WOW« (f.eks. »TEST.WOW«).

Dette program kan LOAD'es og SAVE's som et ganske almindeligt BASIC-program og startes på samme måde med kommandoen RUN.

Slutresultatet

Er det et lille program, du har kørt igennem, fylder det væsentlige mere end før. Er det derimod et stort program, fylder det mindre.

Programmet virker fuldstændig som det oprindelige program. Blot kører det væsentligt hurtigere.

Jeg har som forsøg kørt tipsprogrammet, der er vist andetsteds i bladet, gennem compileren og har opnået følgende hastighedsforøgelse.

Før PETSPEED tager det omkring 4 min. 19 sec. at danne 500 tilfældige rækker. Efter PETSPEED tager det kun 49 sec. Endnu mere markant bliver forskellen, hvis man ser på, hvor lang tid det tager at gennemgå 500 rækker for evt. gevinst. Før 3 min. 19 sec. – efter kun 19 sec.!!!

Det er virkelig en forøgelse af hastigheden, der er til at tage at føle på. I tilgift får man en vis beskyttelse af sine programmer, idet de ikke længere kan listes ud.

Alt i alt synes jeg, det er et effektivt program, der udfylder det gab, der er mellem det langsomme BASIC-program og det komplicerede maskinkodeprogram.

Til VIC-20 og C-64:

MERGE-routinen

Når man indlæser en program i computern, vil et eventuelt eksisterende program blive slettet – uanset om man har tastet NEW eller ej.

Hvis man indlæser et program UDEN at slette det eksisterende, kaldes det at »blande« eller MERGE de to programmer. Nogle computere har denne funktion som en standardkommando i deres BASIC, men hos Commodore er man nødt til at anvende et lille hjælpeprogram.

Nu findes der i princippet to typer MERGE-programmer. Den ene type lægger det sidst indlæste program i forlængelse af de(t) foregående og kræver således, at det sidst indlæste program har de højeste linienumre. Den anden type er lidt mere kompliceret og er i stand til, så at sige, at »flette« de to programmer ind i hinanden og kræver således ikke andet end, at de to programmer har forskellige linienumre. Det her viste program tilhøre den første type.

Merge til C-64

Det er mest praktisk at lave MERGE-programmer som maskinkodeprogrammer, da man kan indlæse programmet, så snart man har tændt for computeren og være nogenlunde sikker på, at det bliver liggende, indtil man slukker igen.

Et maskinkodeprogram bliver nemlig ikke slettet af en NEW-kommando. Programmet her er et maskinkodeprogram, der bliver indlæst af et BASIC-program.

Der er flere steder at lægge programmet ind i hukommelsen, men et praktisk sted at lægge det, er i det ubenyttede område fra 49152-53247, der ikke benyttes til BASIC-programmer. Det her viste program benytter området fra 49152-49230.

Når man har indtastet og gemt programmet på normal vis, gør man følgende:

1. Indtast RUN og tryk på RETURN. Der sker tilsyneladende ingenting, men computeren lægger lynhurtigt maskinkodeprogrammet på plads. Selv om du taster NEW, bliver det liggende.
2. Indtast eller indlæs det program med de laveste linienumre.
3. Indtast SYS 49152 »PROGRAMNAVN« – Hvor »PROGRAMNAVN« er navnet på det program, der skal indlæses efter det første. (SYS 49152»PROGRAMNAVN«,8 – hvis du bruger disketter).

... og til VIC-20

Programmet og fremgangsmåden adskiller sig ikke meget fra det foregående. Den eneste forskel ligger kun i, hvor i hukommelsen man placerer programmet. I linie 90 »stjæler« vi således lidt af BASIC-hukommelsen for at få plads til vores lille program.

VIC-programmet er skrevet til en VIC uden udvidet hukommelse. Hvis du har en VIC med SK udvidelse, skal du i linie 100 skrive S = 16300 istedet for S = 7601. Tilsvarende bruger man programmet ved at skrive SYS 16300 (eller 7601) »PROGRAMNAVN«.

Programlistning side 65

TIPSPROGRAM 1

```
100 REM**********************************
110 REM TIPSPROGRAM 1
120 REM**********************************
130 DIMRK$(500):DIMG4(40):PRINT"[CLR/HOM
    E]":X=10:Y=8:GOSUB60000:PRINT"******
    **********"
140 X=10:Y=9:GOSUB60000:PRINT"* TIPSPROG
    RAM 1 *"
150 X=10:Y=10:GOSUB60000:PRINT"*********
    ********"
160 X=10:Y=12:GOSUB60000:PRINT"[RVS/ON]
    ET RUN PROGRAM  [RVS/OFF]"
170 FORX=0TO2500:NEXT:PRINT"[CLR/HOME]"
180 X=6:Y=10:GOSUB60000:INPUT"HVOR MANGE
     RAEKKER IALT";TT$
185 TT=VAL(TT$)
190 IFVAL(TT$)<1 OR VAL(TT$)>500 THEN PR
    INT"[CLR/HOME]":GOTO180
200 PRINT"[CLR/HOME]":X=14:Y=2:GOSUB6000
    0:PRINT"[RVS/ON]  M E N U  "
210 X=10:Y=5:GOSUB60000:PRINT"1. INDTAST
     RAEKKER"
220 X=10:Y=7:GOSUB60000:PRINT"2. RETTELS
    ER/PRINT"
230 X=10:Y=9:GOSUB60000:PRINT"3. TILF. R
    AEKKER"
240 X=10:Y=11:GOSUB60000:PRINT"4. INDTAS
    T 13'EREN"
250 X=10:Y=13:GOSUB60000:PRINT"5. SAVE P
    AA TAPE"
260 X=10:Y=15:GOSUB60000:PRINT"6. LOAD F
    RA TAPE"
280 X=10:Y=17:GOSUB60000:PRINT"7. SAVE P
    AA DISKETTE"
290 X=10:Y=19:GOSUB60000:PRINT"8. LOAD F
    RA DISKETTE"
300 X=13:Y=23:GOSUB60000:PRINT"[RVS/ON]T
    RYK TAST 1-8[RVS/OFF]";
310 GETM$:IFM$=""THEN310
320 IFM$="1"THEN1000
330 IFM$="2"THEN2000
340 IFM$="3"THEN3000
350 IFM$="4"THEN4000
360 IFM$="5"THEN5000
370 IFM$="6"THEN6000
380 IFM$="7"THEN7000
390 IFM$="8"THEN8000
400 GOTO200
1000 REM**********************************
1010 REM INDTAST RAEKKER
1020 REM**********************************
1030 FORI=1TOTT
1040 Y$=STR$(I):S=LEN(Y$)-1
1050 PRINT"[CLR/HOME]":X=4:Y=9:GOSUB6000
     0:PRINT"[RVS/ON]KAMP NR........."SPC
     (S)"[RVS/ON]1234567890123"
1060 X=4:Y=11:GOSUB60000:PRINT"INDTAST R
     K."I;:INPUTRK$(I)
1070 IFLEN(RK$(I))<>13THEN1060
1080 FORTE=1TO13:TE$=MID$(RK$(I),TE,1)
1090 IFTE$<>"1"ANDTE$<>"2"ANDTE$<>"X"THE
     N1060
1100 NEXT:NEXT:GOTO200
2000 REM**********************************
2010 REM RETTELSER/PRINT
2020 REM**********************************
2030 PRINT"[CLR/HOME]":X=10:Y=3:GOSUB600
     00:PRINT"[RVS/ON]     M E M U     "
2040 X=10:Y=9:GOSUB60000:PRINT"1. PRINT
     RAEKKER"
2060 X=10:Y=12:GOSUB60000:PRINT"2. RETTE
     LSER"
2065 X=10:Y=15:GOSUB60000:PRINT"3. HOVED
     MENU"
2070 X=10:Y=20:GOSUB60000:PRINT"[RVS/ON]
       TRYK TAST 1-3  [RVS/OFF]";
2080 GETM$:IFM$=""THEN2080
2090 IFM$="1"THEN 2200
2100 IFM$="2"THEN 2500
2110 IFM$="3"THEN 200
2120 GOTO2000
2200 REM**********************************
2210 REM PRINT PAA SKAERM
2220 REM**********************************
2230 PRINT"[CLR/HOME]":KP=1:FORI=0TO(INT
     (TT/30)*30)STEP30
2231 X=SC+2:Y=3:GOSUB60000:PRINT"[RVS/ON
     ]NR."KP"[RVS/OFF]        [RVS/ON]NR.
     "KP+1"[RVS/OFF]        [RVS/ON]NR."K
P+2
2240 FORII=1TO13
2250 FORRK=1TO30
2254 IFRK<=10THENSC=0
2255 IFRK>10THENSC=3
2256 IFRK>20THENSC=6
2260 X=RK+SC:Y=II+4:GOSUB60000:PRINTMID$
     (RK$(RK+I),II,1)
2270 NEXTRK,II:IFM$="2"THENGOSUB60200
2280 PRINT"[CRSR/DOWN*5] TRYK EN TAST"
2290 GETM$:IFM$=""THEN2290
2300 PRINT"[CLR/HOME]":KP=KP+3:SC=0:NEXT
     I
2310 GOTO2000
2500 REM**********************************
2510 REM RETTELSER
2520 REM**********************************
2530 PRINT"[CLR/HOME]":X=8:Y=10:GOSUB600
     00:INPUT"HVILKEN RAEKKE RETTES";ER$
2540 IFVAL(ER$)<1ORVAL(ER$)>500THEN2530
2550 ER=VAL(ER$):X=14:Y=14:GOSUB60000:PR
     INT"[RVS/ON]"RK$(ER)
2560 X=12:Y=14:GOSUB60000:INPUTRK$(ER)
2562 IFLEN(RK$(ER))<>13THEN2550
2564 FORTE=1TO13:TE$=MID$(RK$(ER),TE,1)
2566 IFTE$<>"1"ANDTE$<>"2"ANDTE$<>"X"THE
     N2550
2568 NEXT
2570 X=10:Y=16:GOSUB60000:PRINT"FLERE RE
     TTELSER J/N";:GETQ$
2580 GETQ$:IFQ$=""THEN2580
2600 IFQ$="J"THEN2530
2610 IFQ$="N"THEN200
2620 GOTO2570
3000 REM**********************************
3010 REM BEREGN TILFAELDIGE RAEKKER
3020 REM**********************************
3030 PRINT"[CLR/HOME]":X=3:Y=8:GOSUB6000
     0:PRINT"ANGIV SANDSYNLIG TEGNFORDEL
     ING I %"
3040 X=8:Y=12:GOSUB60000:INPUT"INDTAST 1
     'TAL I %";R1
3050 X=8:Y=14:GOSUB60000:INPUT"INDTAST X
     'ER  I %";R2
3060 X=8:Y=16:GOSUB60000:INPUT"INDTAST 2
     'TAL I %";R3
3070 IFR1+R2+R3<>100THEN3000
3072 FORI=1TOTT:RK$(I)="":PRINT"[HOME][R
     VS/ON]"I"[RVS/OFF]"
3074 FORTE=1TO13
3080 TF=INT(RND(0)*100)+1
3090 IFTF<=R1THENRK$(I)=RK$(I)+"1"
3100 IFTF>R1ANDTF<=R1+R2THENRK$(I)=RK$(I
     )+"X"
3110 IFTF>R1+R2THENRK$(I)=RK$(I)+"2"
3120 NEXTTE,I:GOTO200
4000 REM**********************************
4010 REM INDTAST 13'EREN
4020 REM**********************************
```

```
4030 PRINT"[CLR/HOME]":X=4:Y=9:GOSUB6000
     0:PRINT"[RVS/ON]KAMP NR........"SPC
     (2)"[RVS/ON]1234567890123"
4040 X=4:Y=11:GOSUB60000:PRINT"INDTAST 1
     3'EREN";:INPUTRK$(0)
4050 IFLEN(RK$(0))<>13THEN4040
4060 FORTE=1TO13:TE$=MID$(RK$(0),TE,1)
4070 IFTE$<>"1"ANDTE$<>"2"ANDTE$<>"X"THE
     N4040
4080 NEXT
4100 E=0:F=0:G=0:H=0
4110 FORC=1TOTT:SU=0:PRINT"[HOME][RVS/ON
     ]"C"[RVS/OFF]"
4120 FORD=1TO13
4130 TE$=MID$(RK$(C),D,1)
4140 GE$=MID$(RK$(0),D,1)
4150 IFTE$=GE$THENSU=SU+1
4160 NEXT
4170 IFSU=13THENG1(E)=C:E=E+1
4180 IFSU=12THENG2(F)=C:F=F+1
4190 IFSU=11THENG3(G)=C:G=G+1
4200 IFSU=10THENG4(H)=C:H=H+1
4210 NEXT
4220 PRINT"[CLR/HOME][RVS/ON] RAEKKENR.M
     ED 13 RIGTIGE:[RVS/OFF]"
4230 FORX=0TO10
4240 IFG1(X)<>0THENPRINTG1(X),
4250 NEXT
4260 PRINT
4270 PRINT"[RVS/ON] RAEKKENR.MED 12 RIGT
     IGE:[RVS/OFF]"
4280 FORX=0TO10
4290 IFG2(X)<>0THENPRINTG2(X),
4300 NEXT
4310 PRINT
4320 PRINT"[RVS/ON] RAEKKENR.MED 11 RIGT
     IGE:[RVS/OFF]"
4330 FORX=0TO10
4340 IFG3(X)<>0THENPRINTG3(X),
4350 NEXT
4360 PRINT
4370 PRINT"[RVS/ON] RAEKKENR.MED 10 RIGT
     IGE:[RVS/OFF]"
4380 FORX=0TO40
4390 IFG4(X)<>0THENPRINTG4(X),
4400 NEXT
4410 END
5000 REM**********************************
5010 REM SAVE PAA TAPE
5020 REM**********************************
5025 PRINT"[CLR/HOME]"
5030 OPEN1,1,1,"TIPSFILE"
5040 FORI=1TOTT:PRINT#1,RK$(I):NEXT
5050 CLOSE1:GOTO200
6000 REM**********************************
6010 REM LOAD FRA TAPE
6020 REM**********************************
6025 PRINT"[CLR/HOME]"
6030 OPEN1,1,0,"TIPSFILE"
6040 FORI=1TOTT:INPUT#1,RK$(I):NEXT
6050 CLOSE1:GOTO200
7000 REM**********************************
7010 REM SAVE PAA DISKETTE
7020 REM**********************************
7030 OPEN1,8,2,"@0:TIPSFILE,S,W"
7040 FORI=1TOTT:PRINT#1,RK$(I):NEXT
7050 CLOSE1:GOTO200
8000 REM**********************************
8010 REM LOAD FRA DISKETTE
8020 REM**********************************
8030 OPEN1,8,2,"0:TIPSFILE,S,R"
8040 FORI=1TOTT:INPUT#1,RK$(I):NEXT
8050 CLOSE1:GOTO200
```

KONTROLSUM FOR TIPSPROGRAM 1

```
100   207   110   88    120   207
130   135   140   250   150   175
160   209   170   140   180   227
185   60    190   28    200   139
210   116   220   177   230   192
240   84    250   214   260   209
280   15    290   10    300   39
310   121   320   139   330   141
340   143   350   145   360   147
370   149   380   151   390   153
400   27    1000  165   1010  235
1020  165   1030  249   1040  57
1050  192   1060  89    1070  48
1080  70    1090  65    1100  147
2000  165   2010  37    2020  165
2030  71    2040  238   2060  33
2065  22    2070  156   2080  175
2090  174   2100  178   2110  126
2120  75    2200  165   2210  17
2220  165   2230  220   2231  92
2240  254   2250  8     2254  13
2255  92    2256  96    2260  4
2270  103   2280  171   2290  178
2300  38    2310  75    2500  165
2510  105   2520  165   2530  35
2540  120   2550  252   2560  60
2562  131   2564  148   2566  70
2568  130   2570  165   2580  188
2600  177   2610  125   2620  87
3000  165   3010  193   3020  165
3030  95    3040  134   3050  134
3060  141   3070  202   3072  223
3074  5     3080  166   3090  59
3100  13    3110  182   3120  229
4000  165   4010  155   4020  165
4030  159   4040  225   4050  24
4060  45    4070  66    4080  130
4100  80    4110  12    4120  176
4130  172   4140  140   4150  46
4160  130   4170  68    4180  71
4190  74    4200  77    4210  130
4220  116   4230  192   4240  205
4250  130   4260  153   4270  224
4280  192   4290  207   4300  130
4310  153   4320  223   4330  192
4340  209   4350  130   4360  153
4370  222   4380  195   4390  211
4400  130   4410  128   5000  165
5010  26    5020  165   5025  112
5030  90    5040  63    5050  38
6000  165   6010  18    6020  165
6025  112   6030  89    6040  43
6050  38    7000  165   7010  77
7020  165   7030  14    7040  63
7050  38    8000  165   8010  69
8020  165   8030  201   8040  43
8050  38
```

NY SKØNSKRIFTSPRINTER

TEKNISKE SPECIFIKATIONER:

● Skrivehastighed: 18 tegn pr. sekund ● Typehjul: Triumph-Adler kompatibelt, 100 tegn ● Farvebånd: IBM 82 kassette ● Horisontal opløsning: 1/120″ min. ● Linietæthed: 1/48″ (1/96″ mulig ved hjælp af ESC koder) ● Tegntæthed: 10, 12 eller 15 tegn pr. tomme samt proportionalskrift og grafik
● **INTERFACE: COMMODORE 64/VIC-20.**

Intermedium EDB-Teknik og Service A/S,
Hedeager 2, 2600 Glostrup
Tlf. (02) 45 82 33

Møllevej 5, 8680 Ry
Tlf. (06) 89 31 22

CANYONS OF ZELAZ FOR COMMODORE 64 MED JOYSTICK

```
5 PRINTCHR$(147)
10 PRINT"[CRSR/DOWN]**********************
   **********************"
12 PRINT"*
          *"
14 PRINT"*       CANYONS OF ZELAZ
          *"
16 PRINT"*
          *"
18 PRINT"*
          *"
20 PRINT"*
          *"
22 PRINT"*       WRITTEN BY GARY D MCCLEL
   LAN      *"
24 PRINT"*
          *"
26 PRINT"*
          *"
28 PRINT"*
          *"
30 PRINT"**********************************
   *********"
60 OF=54272:GOSUB500:GOSUB900:TI$="00000
   0":FF=16:Y=71:POKEV+1,Y:GOSUB400
97 REM**********************************
98 REM*    MAIN PROGRAM LOOP            *
99 REM**********************************
100 CS=0:CP=0:POKEV+31,0:J=PEEK(56320)
105 IF(JAND16)OR(FU<1)THENC1=-9:POKEAD,0
    :POKEWA,0:GOTO150
107 POKE54296,15:POKEAD,88:POKESR,240:PO
    KEWA,129:FU=FU-3.5:C1=5.5:POKE2040,1
    4
110 IFNOTJAND8THENX1=X1+.25:FU=FU-.5
112 IFNOTJAND4THENX1=X1-.25:FU=FU-.5
113 IFNOTJAND1THENFU=FU+1.75:C1=32
150 POKE55296,2:POKE55326,2:C2=64/C1:V2=
    V1-.4+.2*C2:A2=A2+.1*(V1+V2):V1=V2
155 CP=PEEK(V+31)AND1:IFCP=0THENCR=0
176 Y=(222-INT(A2/D)):X=X+X1:IFY<55THENX
    1=X1*2
177 IFX<10RX>254ORY<45THENGOSUB300
178 POKE53279,0:POKEV,X:POKEV+1,Y:IFPEEK
    (53279)AND1THENCR=CR+1:IFCR>1THENCS=
    1
180 PRINTH$P$P$P$CHR$(17)TAB(32)TI$:PRIN
    TP$TAB(33)STR$(INT(FU))+"  "
185 PRINTP$TAB(33)STR$(INT(A2))+"  ":PRIN
    TP$TAB(32)"V:"STR$(INT(V2))+"  "
186 HV$=STR$(X1*3)+"      ":PRINTTAB(32)"H
    :"LEFT$(HV$,5)
190 FF=PEEK(V+30)AND30:IFFFTHENGOSUB400
195 IF(Y>222ANDV2<-10)ORCSTHENGOSUB300
200 POKE55296,7:POKE55326,7:POKE2040,15:
    GOTO100
297 REM**********************************
298 REM*    MISSION REPORT ROUTINE       *
299 REM**********************************
300 POKENH,2:POKENL,179:POKE54296,15:POK
    EAD,88:POKESR,240:POKEWA,129:CK=1
304 POKE2040,11:POKEV+28,1
305 GOSUB380:POKEV+29,31:GOSUB380:POKEV+
    23,1:GOSUB380:GOSUB390
310 POKEAD,00:POKESR,00:POKEWA,00
315 GOSUB392:PRINTH$P$:PRINTTAB(32)"CRAS
    HED"
317 POKEV+29,30:POKEV+23,0:POKEV+21,30:P
    OKEV+28,0:GOTO370
350 GOSUB392:PRINTH$P$:PRINTTAB(32)"ABOR
    TED":GOTO370
360 GOSUB392:FC=FC-3:IFFC>4THENFC=4
362 PRINTH$P$:PRINTTAB(32)"          "
363 PRINTH$P$:PRINTTAB(32)MR$(FC);TAB(23
    )"FLIGHT":FORJ=1TO4000:NEXTJ
370 J=PEEK(56320):IFJAND16THEN370
371 IFNOTJAND2THENPOKEV+21,0:CLR:PRINTCH
    R$(147):END
375 GOSUB395:GOSUB900:GOSUB584:FF=PEEK(V
    +30):FF=0:TI$="000000":RETURN
380 FORJ=1TO25:Z=INT(60*RND(1))+1:POKE70
    5+Z,J:NEXT:RETURN
390 FORJ=704TO766:POKEJ,00:NEXT:RETURN
392 PRINTH$:PRINTTAB(32)R$"MISSION"TAB(7
    2)R$"REPORT:":RETURN
395 PRINTH$P$:PRINTTAB(32)"          "TAB(7
    2)"          ":RETURN
396 REM**********************************
397 REM* LANDER SPRITE / PAD SPRITE    *
398 REM* COLLISION HANDLER ROUTINE     *
399 REM**********************************
400 IFV2<-10ORABS(X1)>1.5THENGOTO300
403 IFFF=16ANDX<210THENGOTO300
405 PRINTH$P$:PRINTTAB(32)"LANDED ":POKE
    2040,15:POKEAD,0:POKEWA,0
410 IFFF=16THENGOSUB495:IFQ=1THEN360
415 IFFF=16THENFU=FU+200
435 Y=Y-1:POKEV+1,Y
440 IFCKTHENPRINTH$P$:PRINTTAB(32)"STAND
    BY":CK=0
450 FORJ=1TO4:IFFF=2↑JTHENIFPEEK(V+39+J)
    AND5THENFU=FU+300:POKEV+39+J,2:FC=FC
    +1
455 POKE(V+43),5
460 NEXTJ:FORI=1TO1500:NEXT:PRINTH$P$P$:
    PRINTTAB(34)FC
465 PRINTH$P$P$P$;CHR$(17)TAB(32)TI$P$:P
    RINTTAB(33)STR$(INT(FU))+"  "
470 PRINTP$TAB(33)STR$(INT(A2))+"  ":PRIN
    TP$TAB(32)"V:"STR$(INT(V2))+"  "
471 HV$=STR$(X1*3)+"      ":PRINTTAB(32)"H
    :"LEFT$(HV$,5)
475 IFNOTPEEK(56320)AND 2THEN350
480 IFPEEK(56320)AND16THEN465
485 V1=5:C1=2:X1=0:FF=PEEK(V+30):FF=0:A2
    =A2+4
490 PRINTH$P$:PRINTTAB(32)"ACTIVE!":RETU
    RN
495 IF(NOTPEEK(V+40)AND5)AND(NOTPEEK(V+4
    1)AND5)AND(NOTPEEK(V+42)AND5)THENQ=1
496 RETURN
497 REM**********************************
498 REM* SET UP SPRITE DATA AND SCREEN
499 REM**********************************
500 FORI=13TO15:FORJ=0TO62:READA:POKE(I*
    64)+J,A:NEXTJ:NEXTI
504 DIMMR$(4):MR$(1)="GREAT":MR$(2)="GOO
    D":MR$(3)="FAIR":MR$(4)="POOR"
506 PRINTCHR$(147):POKE53281,11:POKE5328
    0,12:FORJ=55296TO56295:POKEJ,1:NEXTJ
508 FORJ=1055TO2023STEP40:POKEJ,160:POKE
    J+OF,4:POKEJ+8,160:POKEJ+8+OF,4:NEXT
    J
510 FORJ=1055TO1062:POKEJ,160:POKEJ+OF,4
    :POKEJ+960,160:POKEJ+960+OF,4:NEXTJ
512 FORJ=1249TO1252:POKEJ,160:NEXT:FORJ=
    1289TO1292:POKEJ,160:NEXT:POKE1288,9
    5
515 FORJ=1064TO1984STEP40:POKEJ,160:POKE
    J+30,160:NEXT
516 POKE1105,223:POKE1145,105:POKE1225,2
    23:FORJ=1265TO1425STEP40:POKEJ,160:N
    EXT
517 POKE1024,81:POKE1024+OF,2:POKE1054,8
    1:POKE1054+OF,2
518 POKE1105,223:POKE1145,105:POKE1225,2
    23:FORJ=1265TO1425STEP40:POKEJ,160:N
    EXT
520 POKE1465,105:POKE1745,223:POKE1785,1
    60:POKE1825,105:POKE1905,223
522 POKE1945,160:POKE1946,223
524 FORJ=1984TO2013:POKEJ,160:NEXT
526 POKE1093,233:FORJ=1133TO1293STEP40:P
    OKEJ,160:NEXT:POKE1333,95:POKE1413,233
```

Masser af spændende spil, seriøse- og hjælpe programmer

NYHED!

PROGRAM SPECIAL

UDKOMMER TIRSDAG DEN 4. DECEMBER

RUN I KIOSKEN

—tirsdag den 4. december!

```
530 FORJ=1453TO1893STEP40:POKEJ,160:NEXT
    :POKE1933,95:POKE1973,233
532 POKE1172,233
535 POKE1352,223:POKE1353,233:POKE1391,2
    33:POKE1394,105
536 FORJ=1392TO1552STEP40:POKEJ,160:NEXT
    :FORJ=1393TO1953STEP40:POKEJ,160:NEX
    T
538 POKE1431,95:POKE1592,95:POKE1672,95
539 POKE1752,233:POKE1792,160:POKE1832,1
    60:POKE1872,95:POKE1791,233:POKE1831
    ,95
542 POKE1471,233:POKE1511,160:POKE1551,9
    5:POKE1510,233
545 POKE1434,223:POKE1474,160:POKE1514,1
    05:POKE1554,223:POKE1594,160
547 POKE1595,223:FORJ=1634TO1954STEP40:P
    OKEJ,160:POKEJ+1,160:NEXT
549 POKE1716,233:POKE1717,223:POKE1757,1
    05:POKE1797,223:POKE1837,105
550 POKE1876,105:POKE1916,223:POKE1956,1
    05:FORJ=1756TO1836STEP40:POKEJ,160:N
    EXT
554 POKE1442,223:POKE1443,223:POKE1444,2
    33:POKE1445,223:POKE1485,105
556 POKE1481,95:FORX=1482TO1602STEP40:PO
    KEX,160:NEXT
558 FORJ=1483TO1963STEP40:POKEJ,160:POKE
    J+1,160:NEXT
560 FORJ=1525TO1965STEP40:POKEJ,160:NEXT
    :FORJ=1526TO1606STEP40:POKEJ,160:NEX
    T
565 FORJ=1766TO1966STEP40:POKEJ,160:NEXT
    :POKE1646,105:POKE1726,223
567 POKE1486,233:POKE1567,105:POKE1807,2
    23
570 POKE1967,233:POKE1968,233:POKE1451,9
    5:POKE1452,160:POKE1651,233:POKE1691
    ,95
573 POKE1492,233:FORJ=1492TO1812STEP40:P
    OKEJ,160:NEXT:POKE1852,95
575 POKE1521,233:POKE1561,95:POKE1642,95
    :POKE1723,95:POKE1763,32:POKE1803,23
    3
577 POKE1842,233:POKE1882,95:POKE1922,23
    3
584 P$=CHR$(17)+CHR$(17)+CHR$(17):H$=CHR
    $(19):R$=CHR$(18)
585 PRINTH$;CHR$(5 ):PRINTTAB(32)R$"MISS
    ION"TAB(72)R$"STATUS:"
587 PRINTH$P$P$;CHR$(17):PRINTTAB(32)R$"
     TIME: ":PRINTP$TAB(32)R$" FUEL: "P$
591 PRINTTAB(32)R$" ALT:   "P$:PRINTTAB(3
    2)R$" VEL:   "
596 PRINTH$P$P$TAB(32)R$"REFUELS"
597 RETURN
896 REM**********************************
897 REM* DEFINE AND ENABLE SPRITES     *
898 REM* 1 THRU 5 AND GAME VARIABLES   *
899 REM**********************************
900 V=53248:X=219:Y=70:D=3:CR=0:POKEV+31
    ,0:FU=0:A2=456:V1=0:X1=0
910 POKEV+21,31:POKE2040,15:POKEV,X:POKE
    V+1,Y:POKEV+27,1:CK=1:Q=0:V2=0
916 POKE2041,13:POKE2042,13:POKE2043,13:
    POKE2044,13
918 POKEV+2,48:POKEV+3,242:POKEV+4,124:P
    OKEV+5,242:POKEV+6,215:POKEV+7,242
919 POKEV+8,210:POKEV+9,90
920 POKEV+39,15:POKEV+40,5:POKEV+41,5:PO
    KEV+42,5:POKEV+43,2
922 POKEV+29,30:POKEV+37,7:POKEV+38,2
930 FORJ=54272TO54296:POKEJ,0:NEXT:WA=54
    276:AD=54277:SR=54278:NL=54272:NH=54
    273
940 POKENL,216:POKENH,12:FC=0
950 RETURN
996 REM**********************************
997 REM* DATA FOR SPRITE DATA BLOCK    *
998 REM* 13, 14, AND 15                *
999 REM**********************************
1000 DATA255,255,255,255,255,255,255,255
     ,255,255,255,255,255,255,255,255,25
     5
1002 DATA255,255,255,255,255,255,255
1003 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
     0,0,0
1004 DATA0,0,0,0,0,0,32,254,24,35,255,16
     0,31,255,216,15,255,192,31,255,192
1006 DATA63,255,248,127,255,252,127,255,
     252,127,255,248,63,255,240,15,255,2
     40
1008 DATA19,255,152,38,56,68,120,124,62,
     64,254,2,64,210,2,64,169,2
1010 DATA65,85,2,194,51,131,0,0,0
1012 DATA0,0,0,32,254,24,35,255,160,31,2
     55,216,15,255,192,31,255,192
1014 DATA63,255,248,127,255,252,127,255,
     252,127,255,248,63,255,240,15,255,2
     40
1016 DATA19,255,152,38,56,68,120,124,62,
     64,254,2,64,0,2,64,0,2
1018 DATA64,0,2,192,0,3,0,0,0
```

KONTROLSUM FOR CANYONS OF ZELAZ

5	77	10	126	12	241
14	103	16	241	18	241
20	241	22	219	24	241
26	241	28	241	30	109
60	134	97	249	98	90
99	249	100	1	105	61
107	17	110	62	112	59
113	38	150	223	155	244
176	90	177	7	178	171
180	16	185	216	186	101
190	223	195	127	200	187
297	207	298	71	299	207
300	54	304	131	305	174
310	159	315	32	317	218
350	132	360	80	362	161
363	227	370	117	371	114
375	101	380	154	390	138
392	178	395	194	396	207
397	75	398	147	399	207
400	26	403	64	405	41
410	63	415	251	435	199
440	64	450	112	455	176
460	143	465	75	470	216
471	101	475	134	480	250
485	235	490	102	495	195
496	142	497	207	498	142
499	207	500	204	504	166
506	0	508	186	510	125
512	42	515	84	516	121
517	144	518	121	520	189
522	149	524	87	526	84
530	14	532	38	535	77
536	8	538	113	539	207
542	23	545	178	547	138
549	199	550	163	554	174
556	181	558	36	560	7
565	58	567	248	570	211
573	4	575	96	577	203
584	183	585	154	587	91
591	178	596	238	597	142
896	207	897	50	898	18
899	207	900	60	910	56
916	108	918	82	919	45
920	240	922	201	930	20
940	145	950	142	996	207
997	64	998	157	999	207
1000	159	1002	207	1003	71
1004	231	1006	183	1008	133
1010	22	1012	211	1014	183
1016	178	1018	62		

Micro·Verden har 18 søstre

og flere er på vej. CW Communications Inc. udgiver micro & pc-magasiner i mange lande.
Har De brug for kontakt, vil vi gerne formidle den.

Micro verden

COMPUTERWORLD DANMARK A/S • Gammel Strand 50 • 1202 København K • Tlf.: 01-12 34 11

PASCAL

Pascal er et af de mest brugervenlige af de »højere« programmeringssprog. Det bruges ikke kun i større computeranlæg. Brugere af mikrocomputere kan også udnytte de fordele, Pascal tilbyder.

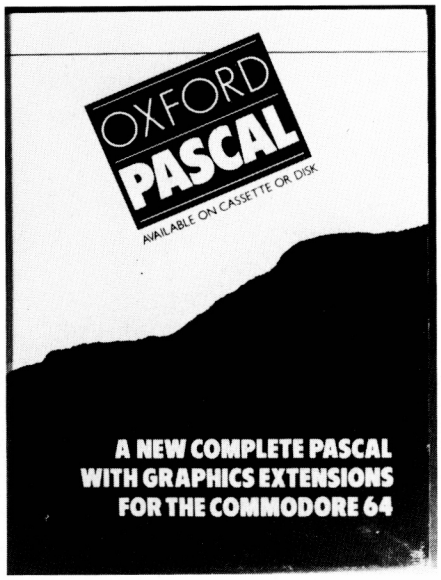
I maj i år udkom den nye version 3.0 af Pascal 64, udgivet af Data Becker. Den er stort set baseret på Standard Pascal, det sprog Nikolaus Wirth udviklede, for at lære sine studenter struktureret programmering. Han kaldte sproget »Pascal«, for at hædre mindet af den berømte franske matematiker fra det 17. århundrede.

Pascal gør det muligt at opbygge programmer med struktur, og uoverskuelige s.k. »spaghetti-programmer«, skulle dermed være noget, der hører fortiden til.

Pascal har en syntaks, der er helt anderledes end Basic'ens; programmerne er opbygget på en helt anden måde. Et Pascal program består af 4 dele. Øverst har vi programhovedet, der indeholder programmets navn, og hvis der er underprogrammer, også underprogrammernes navne og parametre. Derefter kommer deklarationsdelen, der definerer alle konstanter og variabler. Underprogrammets deklaration indeholder definitionen af de underprogrammer, og funktioner der bruges i hovedprogrammet og underprogrammerne. Uha! Så kommer endelig tilsidst sætningerne, det egentlige program. De begynder, og slutter med kommandoer, der i Pascal ikke hedder kommandoer, men »strukturerings-primitiver« (SP'er).

Programmet begynder med SP'en BEGIN, og slutter med END med et punktum efter.

Dette punktum har en vigtig funktion, det fortæller computeren, at hele programmet af afsluttet. Underprogrammer afsluttes ganske vist også med SP'en END, men derefter kommer semikolon (;), som fortæller computeren, at det var et underprogram (sub-rutine), der netop blev afsluttet. I et Pascal program, har semikolon den funktion, at skille de enkelte instruktioner i en programsætning fra hinanden.



I Basic bruges kolon (:) for at gøre det samme. I Pascal bruges kolon til definitioner. Definitionen af variabler er en af Pascal's meget store fordele. Hovedprogram og underprogram har hver sine variabler. Underprogrammerne kan have deres egne sub-rutiner. For programmøren har det den fordel, at han ikke behøver at spekulere på variablernes navne.

I Pascal er det ikke ligegyldigt, om man sætter mellemrum eller ej. Disse har nemlig funktion af styretegn, og det er svært at vænne sig til, når man har BASIC siddende i rygmarven.

Efter en styringsprimitiv, skal man altså trykke mellemrumstasten ind. Hele sætningen afsluttes med semikolon. Komma har en hel anden funktion, det bruges som skilletegn ved listningen. Med hensyn til mellemrumstegnet, er der i Pascal 64 ingen pardon.

Til strukturerede programmer, skal man bruge strukturfremmende sætninger. Er man rutineret i Basic-programmering, må man lære, at vænne sig af med den ubetingede springkommando GO-TO. GO-TO kommandoen er imod hele konceptet bag Pascal. Ved at undlade denne kommando kan man opbygge programmer, der let kan ændres, og har den fordel, at de er overskuelige.

Blokkenes organisation

Programmer og sætninger kan gennemfattes i blokke. Betingede forgreninger og løkker gør Pascal meget brugervenligt. REPEAT-UNTIL løkken gentager en blok lige så mange gange, indtil betingelsen der specificeres af UNTIL er opfyldt. Derefter afvikles resten af programmet. WHILE-DO virker nærmest omvendt. Her tester programmet, om den stillede betingelse er »true«, dvs. om den holder stik, og blokken udføres så længe det er tilfældet. FOR-TO-DO fungere anderledes end Basic'ens FOR-TO løkke. DO er en instruktion om, at udføre en hel blok. Løkken kan også forsynes med en ned-tælling. Instruktionen lyder så FOR-DOWN-TO.

Men Pascals bedste bud er den betingede forgrening IF-THEN-ELSE. Den fortæller computeren: udfør den pågældende programblok når en bestemt betingelse er opfyldt. Er den ikke opfyldt, bearbejder du en anden blok i stedet.

Foruden den udvidede Standard Pascal, tilbyder Pascal 64 noget ekstra: disketten har en »Linker«, der kan kæde to eller flere Pascalprogrammer sammen. Compiler'en er dog en forholdsvis langsom fortolker.

Sådan bruges funktionstasterne

Af Axel Bang

Commodore 64 og VIC-20 er forsynet med udmærkede tastaturer. 66 taster udgør dit absolut vigtigste bindeled til computeren, og heraf er de 62 taster temmelig indlysende. Men hvad med funktionstasterne - disse fire, grå ubestemmeligheder, som findes yderst til højre på tastaturet?

Før jeg kan give et svar på dette vel nok mest stillede spørgsmål (i hvert fald i forbindelse med Commodore computere), må jeg nok kort forklare begrebet ASCII-koder.

Hvad er ASCII-koder?

Alt det, du taster på tastaturet, lagres i form af talkoder. Dem kan du i øvrigt slå op bag i brugsvejledningen for din computer. Til »A« svarer 65, til »B« 66 og så videre. Alle computerens tal, tegn og bogstaver kan omsættes til talkoder, og det samme gælder mange af computerens særlige taster, som »DEL«, »CLEAR«, »HOME«, cursorstyringstasterne og så videre.

Disse talkoder betegnes »ASCII-koden« (selvom ASCII i virkeligheden er en verdensomfattende standard-Commodores koder adskiller sig en smule fra denne egentlige standard). Du bør lære denne kode, så du ved, hvilke tal der repræsenterer de hyppigst benyttede funktioner i computeren. 13 svarer således til RETURN-tasten, og 20 til slettetasten. Du kan generelt skifte mellem de enkelte tegn og deres ASCII-koder i flæng, og du kan blande de forskellige ting ganske som det passer dig.

Prøv f.eks. at taste denne lille programstump ind direkte som den står, og slut af med at trykke på RETURN:

```
PRINT»C«CHR$(66)»M«
```

På skærmen skulle der nu gerne stå CBM.

Men omvendt kan du bruge ASCII-koderne til at undersøge, om der er tastet det ene eller det andet tegn på tastaturet. Prøv dette lille eksempel:

```
10 GET A$: IF A$ = "" THEN 10
20 IF A$ = CHR$(13) THEN PRINT
»DU TRYKKEDE PÅ RETURN-TAS-
TEN«
30 IF A$ = CHR$(20) THEN PRINT
»DU TRYKKEDE PÅ DELETE-TAS-
TEN«

40 IF A$ = CHR$(133) THEN PRINT
   »DU TRYKKEDE PÅ F1-TASTEN«
50 GOTO 10
```

F-tasterne i programmer

Allerede nu begynder du at kunne skimte betydningen af funktionstasterne, og det bliver klart for dig, at der skal et egentligt program til, for at du kan styre funktionstasterne. På den anden side giver jo netop et program store muligheder for at knytte såvel omfattende som komplicerede funktioner til hver af funktionstasterne.

Det er almindeligt, at lade egentlige brugs-programmer starte med en »menu«, hvorfra brugeren kan vælge og vrage blandt programmets funktioner. Det er ganske almindeligt at anvende GET-kommandoen til at »hente« netop sådanne input. GET aflæser tastaturet, og lagrer netop ASVII-værdien af det tastede tegn i den variabel, der altid skal knyttes til GET-kommandone. At vi i linie 10 har et IF-udtryk, hænger sammen med at GET faktisk »undersøger« tastaturet i samme sekund, programmet når frem til GET-kommandoen – hvis der ikke er tastet noget som helst, accepterer GET-kommandoen dette som et »tomt« imput, hvilket betyder, at der slet ikke lægges nogen værdi ned i A$ – men det er GET-kommandoen bedøvende ligeglad med. IF-udtrykket er beregnet på at fange netop sådanne tomme input, og tvinger computeren til at udføre GET-kommandoen igen og igen, indtil der virkelig er tastet et eller andet.

Funktionstasterne kan anvendes alene, eller sammen med SHIFT-tasten. De kan også anvendes sammen med Commodore Logo-tasten, men der skal vi anvende en anden teknik, som jeg vil beskrive i en senere artikel.

Så vi vil i første omgang koncentrere os om at bruge tasterne alene, og sammen med SHIFT.

F1-tasten har ASCII-koden 133, F3 har koden 134, F5 135, F7 136. Og tilsvarende har F2 (= F1 + SHIFT) koden 137, F4 138, F6 139 og endelig F8 140. Ved at bruge den lille programstump herover som model, skulle du herefter let kunne konstruere et program, som faktisk reagerer på funktionstasterne – og herefter er det kun fantasien, som sætter grænserne. Et program kan f.eks. spille »Kong Kristian« når du trykker på F8, og »Højt fra træets grønne top« når du trykker på F1 – så længe dit program er indrettet til at spille begge melodier, og vælge imellem dem ved at GET'te en valgmulighed og reagere på den, ved at spille en af de to melodier. I øvrigt er det noget pjat at hævde, at funktionstasterne er noget særligt i den forbindelse, for dit program kunne jo ligeså godt have kigget efter ASCII-koden for f.eks. »Q« eller »Cursor-til-højre« for at udløse, at en af de to melodier bliver spillet. For GET tillader dig faktisk at »programmere« samtlige taster på tastaturet med undtagelse af SHIFT, SHIFT LOCK og Commodore Logo-tasten. Blot har funktionstasterne den fordel, at de sidder »afsides« og derfor ikke så let bliver trykket ved en fejl.

Uden programstyring

Når man bliver mere avanceret, kan man indrette et maskinsprogprogram, som i virkeligheden arbejder ligesom GET-rutinen herover. For nemheds skyld lægger man programmet tilrette, så computeren er tvunget til at gennemløbe det lille program meget ofte (nemlig 60 gange i sekundet) – det sker i praksis i forbindelse med en »interrupt«. 60 gange i sekundet afbryder computeren det, den er i gang med, og computerens operativsystem udfører i stedet et nøje fastlagt program, som er en del af computerens indbyggede »intelligens«. Under dette programforløb vedligeholdes skærmbilledet, og computeren foretager sig en hel del internt »rengøringsarbejde«, så den hele tiden kan holde styr på det, der er sket.

Man kan pille ved disse afbrydelser, så de starter med at gennemløbe et program, du selv har skrevet – men programmet skal være indrettet på en bestemt måde, og det skal være skrevet i maskinsprog. Endelig skal sådan et program afsluttes med en JMP (svarer til »GOTO« i Basic) til den operativsystems-rutine, der normalt skulle

gennemløbes under afbrydelsen – ellers er der en masse nødvendigt arbejde, som ikke bliver udført.

Det følgende program er stykket sammen af flere godbidder, og jeg skylder en række udenlandske programmører tak for detaljerne. Programmet er indrettet, så du kan have det liggende i computeren mens du f.eks. programmerer i Basic, og så vil du sikkert få glæde af funktionerne. Først har jeg aftrykt programmet i Assembler, et programmeringssprog, som ligger meget tæt ved rent maskinsprog, og som din computer kan fordøje hvis du har f.eks. en VICMON eller en maskinsprogsmonitor til Commodore 64. Jeg aftrykker kun programmet til Commodore 64, og den øvede maskinsprogsprogrammør, vil ikke have besvær med at omsætte programmet til VIC-20. Samtidig aftrykker jeg et Basic-program, som lægger maskinkodeprogrammet på plads ved hjælp af DATA-sæt som POKE's på plads. Til sidst gennemgår jeg programmets virkemåde i detaljer.

Indtast f.eks. Basic-programmet og gå det grundigt igennem for fejl. Det kontrollere selv, om dine DATA-indtastninger er korrekte, og herefter bør du SAVE programmet på tape eller diskette, inden du RUN'er det.

Når programmet er sat i gang, vil Basic-delen være slettet, og du kan frit bruge computeren til dine egne programmer. Men – nu kan du bruge F1 til at slette skærmen fra den linie som cursoren står i, og op til skærmens top. F3 sletter fra cursorens linie til skærmens bund, og endelig virker F5 som en slags »omvendt HOME«, idet cursoren hopper direkte til den nederste skærmlinie. Du skal være opmærksom på, at hvis du anvender funktionstasterne til f.eks. at holde skærmen »ren« mens du programmere, så virker slette-funktionerne kun på skærmen – selvom du på den måde »sletter« nogle programlinier, findes de stadig i computerens lager, og kan når som helst kaldes frem igen med LIST. Ikke desto mindre vil jeg tro, at programmet kan give en og anden et par gode ideer i forbindelse med f.eks. tekstbehandlingsprogram eller lignende.

God arbejdslyst.

F-TAST PROGRAM

```
200 FORI=0TO148
210 READJ:POKE49152+I,J:X=X+J:NEXTI
230 IFX<>17524THENPRINT"ERROR":STOP
240 SYS49152:NEW
300 DATA120,169,13,141,20,3,169,192,141,
    21,3,88,96,165,197,41
310 DATA127,201,4,208,27,169,0,133,25,16
    9,4,133,26,216,24,165
320 DATA209,105,40,133,27,165,210,133,28
    ,144,2,230,28,24,144,46
330 DATA165,197,41,127,201,5,208,19,165,
    209,133,25,165,210,133,26
340 DATA169,231,133,27,169,7,133,28,24,1
    44,19,201,6,208,67,169
350 DATA192,133,209,169,7,133,210,169,24
    ,133,214,24,144,44,216,56
360 DATA165,27,229,25,133,29,165,28,229,
    26,133,30,169,32,166,30
370 DATA240,12,160,0,145,25,200,208,251,
    230,26,202,208,246,166,29
380 DATA240,8,160,0,145,25,200,202,208,2
    50,169,0,133,211,169,32
390 DATA133,197,76,49,234
```

KONTROLSUM FOR F-TAST PROGRAM

```
200   237   210   165   230   34
240   127   300   194   310   183
320   15    330   130   340   254
350   133   360   44    370   107
380   4     390   222
```

Super sprite builder

Dette program vil lette dig væsentligt, når du skal lave dine egne sprites.

Programmet vil vise dig felter, du kan bruge til at opbygge din sprite. Du kan flytte rundt mellem felterne ved hjælp af tasterne I, J, K og M.

Du kan også erstatte linierne i hovedprogrammet med de linier, der er vist i program 2. I så tilfælde vil du kunne flytte rundt ved hjælp af joysticket.

Følgende taster skal anvendes til at styre programmet:

1 Vender spriten på hovedet.
2 Spejlvender din sprite.
3 Drejer din sprite 90 grader.
4 Ændrer farve (Normale farvetaster)
5 Tænder og slukker for multicolor mode.
6 og 7 flytter din sprite til højre eller venstre.
8 og 9 flytter din sprite op eller ned.
E Sætter TRACE-mode til og fra.
Q Slutter programmet med mulighed for DATA linier.
F1 Gemmer sprite data i en sekventiel file på diskette.
F3 Indlæser en tidligere file.
F5 Blander en tidligere file med den eksisterende sprite.
SHFT/CLR Sletter den eksisterende sprite.

```
SPRITE BUILDER FOR COMMODORE 64


3 REM SUPER SPRITE BUILDER
5 REM**********************************
6 REM*    M. J. CLIFFORD               *
7 REM*    2323 W. BANCROFT ST.         *
8 REM*    TOLEDO, OH 43607             *
9 REM**********************************
20 GOSUB9000
100 SL=SC+40*Y+X:CL=SL+54272
110 BY=Y*3+INT(X/8):BI=7-(X-INT(X/8)*8)
120 POKESL,102+128*CC
130 GETA$:IFA$=""THENCC=1-CC:GOTO120
140 POKESL,160
145 IFA$="[CLR/HOME]"THENFORQ=832TO895:P
    OKEQ,0:NEXT:GOSUB5000:GOTO100
150 IFASC(A$)>132ANDASC(A$)<136THEN320
160 K=(ASC(A$)>127):IFKTHENA$=CHR$(ASC(A
    $)-128)
170 IFVAL(A$)<>0THEN340
180 IFA$="E"THENE=16-E:A$="":GOTO360
190 IFA$="Q"THEN900
200 IFA$="I"THENY=Y-1
210 IFA$="M"THENY=Y+1
220 IFA$="J"THENX=X-1
230 IFA$="K"THENX=X+1
240 IFKGOTO270
250 IFETHENPOKECL,11:Q=PEEK(832+BY):POKE
    832+BY,QAND(255-2↑BI):GOTO270
260 POKECL,1:Q=PEEK(832+BY):POKE832+BY,Q
    OR2↑BI
270 IFX>23THENX=0:Y=Y+1
280 IFX<0THENX=23:Y=Y-1
290 IFY>20THENY=0
300 IFY<0THENY=20
310 GOTO100
320 ON(ASC(A$)-132)GOSUB2000,3000,4000
330 GOTO100
340 ONVAL(A$)GOSUB1000,1100,1200,500,400
    ,1300,1400,1500,1600
350 GOTO100
360 IFETHENPOKE1912,6:POKE1913,6:GOTO120
370 POKE1912,14:POKE1913,32:GOTO120
397 REM**********************************
398 REM*   MONO / MULTI COLOR           *
399 REM**********************************
400 MC=15-MC:POKEV+28,MC
410 POKE1677,14-11*(MC>0)
420 RETURN
497 REM**********************************
498 REM*   COLOR CHANGE                 *
499 REM**********************************
500 PRINTCL$
510 PRINT"[CRSR/UP]BACKGROUND ?":GOSUB58
    0:BC=I
515 IFMC=0THENPRINT"[CRSR/UP]SPRITE COLO
    R?  [CRSR/LEFT*3]":GOSUB580:C1=I:GO
    TO540
520 PRINT"[CRSR/UP]UNIQUE C (10)? [CRSR/
    LEFT]":GOSUB580:C1=I
525 PRINT"[CRSR/UP]COLOR 1 (01)? [CRSR/L
    EFT]":GOSUB580:C2=I
530 PRINT"[CRSR/UP]COLOR 2 (11)?  [CRSR/
    LEFT*2]":GOSUB580:C3=I
540 GOSUB600
550 PRINTCD$:E=0
555 POKEV+37,C2:POKEV+38,C3
560 FORQ=39TO42:POKEV+Q,C1:NEXT
570 RETURN
580 GETB$:IFB$=""THEN580
585 I=ASC(B$)-25:IFI>15THENI=I-24
590 IFI>0ANDI<16THENRETURN
595 GOTO580
597 REM**********************************
598 REM*   SPRITE BOXES                 *
599 REM**********************************
600 PRINTMID$(BG$,BC+1,1);
610 PRINT"[HOME][CRSR/DOWN][CRSR/RIGHT][
    RVS/ON]    [CRSR/RIGHT]      [RVS/
    OFF]"
620 PRINT"[RVS/ON][CRSR/RIGHT*2]
        [CRSR/DOWN][CRSR/RIGHT*14]
          "
630 PRINT
640 PRINT"[CRSR/UP][CRSR/RIGHT][RVS/ON]
        [CRSR/RIGHT]      [CRSR/DOWN][C
    RSR/RIGHT*14]          "
```

```
650 PRINT"[CRSR/UP][CRSR/RIGHT][RVS/ON]
      [CRSR/RIGHT]       [CRSR/DOWN][C
    RSR/RIGHT*14]"
660 PRINT"[CRSR/UP][CRSR/RIGHT][RVS/ON]
      [CRSR/RIGHT]       [CRSR/DOWN][C
    RSR/RIGHT*14]"
670 PRINTPI$:RETURN
900 POKEV+21,0
910 PRINT"[CLR/HOME]"
920 INPUT"DO YOU WANT THE DATA";Q$:IFLEF
    T$(Q$,1)<>"Y"THENEND
925 IFF$=""THENINPUT"NAME FOR SPRITE";F$
930 INPUT"BEGINNING LINE NUMBER";LN
935 PRINT"[CLR/HOME]":PRINTLN;"REM *** S
    PRITE DATA FOR ";F$;" ***":LN=LN+1
940 PRINTLN;"REM *** COLOR:"C1;:IFMC=0TH
    ENPRINT" ***":GOTO950
945 PRINT","C2","C3"***"
950 LN=LN+1:PRINTLN;"DATA";C1;:IFMC=15TH
    ENPRINT",";C2;",";C3;
955 PRINT:LN=LN+1
960 FORL=0TO3:PRINTLN;"DATA";
965 FORD=0TO15:PRINTMID$(STR$(PEEK(832+1
    6*L+D)),2);",";:NEXTD:PRINT"[CRSR/LE
    FT]"
970 LN=LN+1:NEXTL
980 END
1000 REM*******************************
1001 REM*   ROTATE ON X-AXIS          *
1002 REM*******************************
1010 FORZ=832TO859STEP3:ZZ=1724-Z
1020 FORA=0TO2:B=PEEK(Z+A):POKEZ+A,PEEK(
     ZZ+A):POKEZZ+A,B:NEXT:NEXT
1030 GOTO5000
1100 REM*******************************
1101 REM*   ROTATE ON Y-AXIS          *
1102 REM*******************************
1110 FORZ=832TO892STEP3
1120 B=PEEK(Z):POKEZ,PEEK(Z+2):POKEZ+2,B
     :NEXT
1130 SYS986
1140 IFMC=0THEN5000
1150 A=C1:C1=C2:C2=A:GOTO5000
1200 REM*******************************
1201 REM*   ROTATE ON Z-AXIS          *
1202 REM*******************************
1210 FORZ=832TO896:POKEZ,0:NEXTZ
1220 FORX=20TO0STEP-1:FORY=0TO20:BY=(20-
     X)*3+INT(Y/8):BI=7-(Y-INT(Y/8)*8)
1230 CL=SC+54272+40*Y+X:IF(PEEK(CL)AND15
     )<>1THEN1250
1240 A=PEEK(832+BY):POKE832+BY,AOR2↑BI
1250 NEXTY:NEXTX:X=0:Y=0:MC=0:GOTO5000
1300 REM*******************************
1301 REM*   MOVE LEFT                 *
1302 REM*******************************
1305 IFMCTHENGOSUB1310
1310 FORZ=832TO892STEP3
1320 POKEZ,(PEEK(Z)*2-1*(PEEK(Z+1)>127))
     AND255
1330 POKEZ+1,(PEEK(Z+1)*2-1*(PEEK(Z+2)>1
     27))AND255
1340 POKEZ+2,(PEEK(Z+2)*2)AND255
1350 NEXT:GOTO5000
1400 REM*******************************
1401 REM*   MOVE RIGHT                *
1402 REM*******************************
1405 IFMCTHENGOSUB1410
1410 FORZ=832TO892STEP3
1420 POKEZ+2,PEEK(Z+2)/2+128*(PEEK(Z+1)A
     ND1)
1430 POKEZ+1,PEEK(Z+1)/2+128*(PEEK(Z)AND
     1)
1440 POKEZ,PEEK(Z)/2
1450 NEXT:GOTO5000
1500 REM*******************************
1501 REM*   MOVE UP                   *
1502 REM*******************************
1510 FORZ=832TO891
1520 POKEZ,PEEK(Z+3):NEXT
1530 FORZ=892TO894:POKEZ,0:NEXT
1540 GOTO5000
1600 REM*******************************
1601 REM*   MOVE DOWN                 *
1602 REM*******************************
1610 FORZ=894TO835STEP-1
1620 POKEZ,PEEK(Z-3):NEXT
1630 FORZ=832TO834:POKEZ,0:NEXT
1640 GOTO5000
1997 REM*******************************
1998 REM*   SAVE SPRITE FILE          *
1999 REM*******************************
2000 PRINTCL$:PRINTC2$:F$=""
2010 PRINT"[CRSR/UP]SAVE IN FILE NAMED";
     :INPUTF$:IFF$=""THEN2080
2020 OPEN1,8,15
2030 FL$="0:"+F$+",SEQ,WRITE"
2040 OPEN2,8,14,FL$
2050 INPUT#1,A,B$,C,D:IFA<>0THEN2100
2060 PRINT#2,F$:PRINT#2,BC:PRINT#2,C1:PR
     INT#2,C2:PRINT#2,C3:PRINT#2,MC
2070 FORM=0TO63:PRINT#2,PEEK(832+M):NEXT
     M
2080 CLOSE2:CLOSE1
2090 PRINTCL$:PRINTCD$:E=0:RETURN
2097 REM*******************************
2098 REM*   DISK ERROR                *
2099 REM*******************************
2100 PRINT"[RVS/ON]DISK ERROR[RVS/OFF]";
     B$:GOTO2080
2997 REM*******************************
2998 REM*   LOAD SPRITE FILE          *
2999 REM*******************************
3000 PRINTCL$:PRINTC2$:F$=""
3010 PRINT"[CRSR/UP]LOAD FROM FILE NAMED
     ";:INPUTF$:IFF$=""THEN2090
3020 OPEN1,8,15
3030 FL$="0:"+F$+",SEQ,READ"
3040 OPEN2,8,14,FL$
3050 INPUT#1,A,B$,C,D:IFA<>0THEN2100
3060 INPUT#2,F$:INPUT#2,BC:INPUT#2,C1:IN
     PUT#2,C2:INPUT#2,C3:INPUT#2,MC
3070 FORM=0TO63:INPUT#2,Q:POKE832+M,Q:NE
     XTM
3080 CLOSE2:CLOSE1
3090 GOTO5000
3997 REM*******************************
3998 REM*   MERGE SPRITE FILE         *
3999 REM*******************************
4000 PRINTCL$:PRINTC2$:F$=""
4010 PRINT"[CRSR/UP]MERGE FROM FILE NAME
     D";:INPUTF$:IFF$=""THEN2090
4020 OPEN1,8,15
4030 FL$="0:"+F$+",SEQ,READ"
4040 OPEN2,8,14,FL$
4050 INPUT#1,A,B$,C,D:IFA<>0THEN2100
4060 INPUT#2,F$:INPUT#2,BC:INPUT#2,C1:IN
     PUT#2,C2:INPUT#2,C3:INPUT#2,MC
4070 FORM=0TO63:INPUT#2,Q:POKE832+M,QORP
     EEK(832+M):NEXT
4080 CLOSE2:CLOSE1
4090 GOTO5000
4997 REM*******************************
4998 REM*   DRAW SPRITE IN BIG BOX    *
4999 REM*******************************
5000 POKEV+28,MC:POKE1677,14-11*(MC>0):G
     OSUB5100
5010 SYS912:X=0:Y=0:GOTO540
5100 PRINT"[HOME]";:FORY=0TO20
5110 PRINT"[CRSR/RIGHT*16][RVS/ON]
                    [RVS/OFF]";
5120 NEXTY:Y=0
5130 RETURN
9000 POKE53269,0:POKE53281,15:POKE53280,
     7:PRINT"[LOGO/4]"
9010 PRINT"[CLR/HOME][CRSR/DOWN]
          SPRITE EDITOR[CRSR/DOWN]"
```

```
9020 PRINT"      IN TRACE MODE, THE CURSO
     R ([LOGO/T]) TURNS"
9030 PRINT"ON THE BIT WHEREVER IT MOVES.
       WHEN THE"
9040 PRINT"TRACE IS OFF, THE BIT UNDER T
     HE CURSOR"
9050 PRINT"IS TURNED OR LEFT OFF[CRSR/DO
     WN]"
9060 PRINT"    THE I,J,K AND M KEYS CONT
     ROL THE"
9070 PRINT"MOVEMENT OF THE CURSOR: I=UP,
      M=DOWN"
9080 PRINT"J=LEFT, AND K=RIGHT.  WITH TH
     E SHIFT"
9090 PRINT"KEY DOWN, THE CURSOR CAN BE M
     OVED"
9100 PRINT"WITHOUT CHANGING ANY BITS.  T
     HE CLR KEY"
9110 PRINT"CLEARS THE GRID.[CRSR/DOWN]"
9120 PRINT"  THE E KEY TURNS THE TRACE M
     ODE ON/OFF"
9130 PRINT"AND THE Q KEY ENDS THE PROGRA
     M WITH THE"
9140 PRINT"OPTION OF PREPARING DATA STAT
     EMENTS."
9150 PRINT"[CRSR/DOWN]    TO CHOOSE COLOR
     S, USE KEYS 1-8 AS"
9160 PRINT"MARKED AND SHIFT-1 TO SHIFT-8
      FOR THE"
9170 PRINT"[RVS/ON]C=[RVS/OFF] KEY COLOR
     S.  A MENU OF OTHER"
9180 PRINT"OPTIONS IS DISPLAYED AT ALL T
     IMES[CRSR/DOWN]"
9190 FORA=912TO912+101:READB:POKEA,B:NEX
     T
9200 V=53248:POKE2042,13
9210 POKE2040,13:POKE2041,13:POKE2043,13
9220 FORQ=832TO896:POKEQ,0:NEXT
9230 POKEV+29,10:POKEV+23,12
9240 POKEV,40:POKEV+1,60:POKEV+2,84:POKE
     V+3,60
9250 POKEV+4,40:POKEV+5,92:POKEV+6,84:PO
     KEV+7,92
9260 POKEV+21,15
9270 C1=2:C2=5:C3=7:BC=1
9280 FORQ=39TO42:POKEV+Q,C1:NEXT
9290 SC=1040:CS=55296:MC=0
9300 CD$="[HOME][CRSR/DOWN*22]TRACE=ON
       E=TRACE ON/OFF    Q=DONE "
9310 CL$="[HOME][CRSR/DOWN*22]
                                    "
9320 C2$="[HOME][CRSR/DOWN*23]
                                [CRSR/UP]
     "
9330 BG$="[BLK][WHT][RED][CYN][PUR][GRN]
     [BLU][YEL]":PT$="[LOGO/4]"
9340 PRINT"HIT A KEY TO START[CRSR/UP]"
9350 GETQ$:IFQ$=""THEN9350
9360 PRINT"[CLR/HOME]":GOSUB5100
9370 PRINT"[HOME][CRSR/DOWN*12][CRSR/RIG
     HT*2]1-TURN OVER[CRSR/DOWN][CRSR/LE
     FT*11]2-MIRROR IMAGE"
9380 PRINT"[CRSR/RIGHT*2]3-ROTATE 90[LOG
     O/V][CRSR/DOWN][CRSR/LEFT*12]4-COLO
     R CHG[CRSR/DOWN][CRSR/LEFT*11]5-MUL
TICLR"
9390 PRINT"[CRSR/UP][CRSR/RIGHT*2]6-MOVE
      LEFT[CRSR/DOWN][CRSR/LEFT*11]7-MOV
     E RIGHT[CRSR/DOWN][CRSR/LEFT*12]8-M
OVE UP"
9400 PRINT"[CRSR/RIGHT*2]9-MOVE DOWN"
9410 PRINTCD$:PRINT"[CRSR/DOWN][RVS/ON]D
     ISK:[RVS/OFF]F1=SAVE   F3=LOAD   F5
     =MERGE[CRSR/UP]"
9420 POKE55949,2:POKEV+28,0:POKE1677,14:
     X=0:Y=0:GOTO600
9500 DATA169,64,133,248,169,3,133,249,16
     9,16,133,250,169,216,133,251,162,3
9510 DATA160,0,177,248,10,144,6,72,169,1
     ,145,250,104,200,192,8,208,242,230,
     248
9520 DATA202,240,13,165,250,24,105,8,133
     ,250,144,224,230,251,208,220,165,24
     8
9530 DATA201,127,240,13,165,250,24,105,2
     4,133,250,144,203,230,251,208,199,9
     6
9540 DATA169,63,133,248,169,3,133,249,16
     0,63,162,8,177,248,74,38,250
9550 DATA202,208,250,165,250,145,248,136
     ,208,239,96
9999 :
10000 REM*******************************
10001 REM*   RUN 10000 TO CHECK DATA  *
10002 REM*******************************
10010 FORX=0TO101:READB:CK=CK+B:NEXTX
10020 IFCK=15921THENPRINT"OK":END
10030 PRINT"[RVS/ON]ERROR IN DATA":END
```

KONTROLSUM FOR SPRITE BUILDER

3	92	5	249	6	127
7	22	8	145	9	249
20	86	100	196	110	70
120	108	130	189	140	249
145	165	150	2	160	189
170	216	180	17	190	119
200	22	210	25	220	21
230	21	240	248	250	155
260	0	270	83	280	86
290	217	300	219	310	26
320	205	330	26	340	126
350	26	360	148	370	123
397	207	398	53	399	207
400	218	410	120	420	142
497	207	498	104	499	207
500	76	510	203	515	103
520	131	525	26	530	218
540	35	550	165	555	128
560	120	570	142	580	108
585	171	590	252	595	38
597	207	598	155	599	207
600	133	610	95	620	208
630	153	640	97	650	97
660	97	670	41	900	86
910	112	920	150	925	31
930	96	935	243	940	118
945	38	950	21	955	148
960	199	965	83	970	201
980	128	1000	165	1001	233
1002	165	1010	195	1020	74
1030	78	1100	165	1101	234
1102	165	1110	77	1120	50
1130	69	1140	105	1150	102
1200	165	1201	235	1202	165
1210	18	1220	34	1230	107
1240	35	1250	12	1300	165
1301	229	1302	165	1305	20
1310	77	1320	112	1330	39
1340	188	1350	10	1400	165
1401	24	1402	165	1405	21
1410	77	1420	139	1430	174
1440	105	1450	10	1500	165
1501	159	1502	165	1510	112
1520	35	1530	188	1540	78
1600	165	1601	242	1602	165
1610	251	1620	36	1630	176
1640	78	1997	165	1998	233
1999	165	2000	82	2010	19
2020	198	2030	228	2040	168
2050	151	2060	196	2070	157
2080	221	2090	243	2097	165
2098	24	2099	165	2100	132
2997	165	2998	218	2999	165
3000	82	3010	162	3020	198

```
3030  117     3040  168     3050  151
3060  76      3070  21      3080  221
3090  78      3997  165     3998  10
3999  165     4000  82      4010  242
4020  198     4030  117     4040  168
4050  151     4060  76      4070  31
4080  221     4090  78      4997  165
4998  154     4999  165     5000  252
5010  127     5100  39      5110  140
5120  80      5130  142     9000  70
9010  240     9020  230     9030  10
9040  204     9050  131     9060  146
9070  143     9080  29      9090  125
9100  35      9110  29      9120  196
9130  11      9140  195     9150  195
9160  78      9170  87      9180  229
9190  203     9200  55      9210  63
9220  166     9230  84      9240  66
9250  54      9260  140     9270  41
9280  120     9290  70      9300  255
9310  18      9320  58      9330  249
9340  46      9350  190     9360  253
9370  134     9380  106     9390  100
9400  12      9410  167     9420  21
9500  32      9510  198     9520  88
9530  96      9540  5       9550  172
9999  58      10000 123     10001 4
10002 123     10010 252     10020 165
10030 36
```

PROGRAM 2 - JOYSTICK RETTELSER

```
1 GOTO3
2 SAVE"S.B. V/JOYSTICK"
3 REM**********************************
4 REM*   CHANGES IN SPRITE BUILDER   *
5 REM*   FOR JOYSTICK OPERATION      *
6 REM**********************************
7 :
125 JV=PEEK(56320):FR=16-(JVAND16):JV=15
    -(JVAND15)
127 IFJVORFRTHENA$="Z":GOTO140
160 :
200 IFJVAND1THENY=Y-1
210 IFJVAND2THENY=Y+1
220 IFJVAND4THENX=X-1
230 IFJVAND8THENX=X+1
240 :
250 IFFRANDETHENPOKECL,11:Q=PEEK(832+BY)
    :POKE832+BY,QAND(255-2↑BI):GOTO270
260 IFFRTHENPOKECL,1:Q=PEEK(832+BY):POKE
    832+BY,QOR2↑BI
270 IFX>23THENX=0:Y=Y+1:IFY>20THENY=0
280 IFX<0THENX=23:Y=Y-1:IFY<0THENY=20
290 IFY>20THENY=0:X=X+1:IFX>23THENX=0
300 IFY<0THENY=20:X=X-1:IFX<0THENX=23
9020 PRINT"      THE JOYSTICK CONTROLS TH
     E MOVEMENT OF THE CURSOR[CRSR/DOWN]
     "
9030 PRINT"      WHEN TRACE IS ON, THE FI
     REBUTTON"
9040 PRINT"TURNS ON THE BIT UNDER THE CU
     RSOR."
9050 PRINT"WHEN TRACE IS OF, THE FIREBUT
     TON TURNS"
9060 PRINT"OFF THE BIT UNDER THE CURSOR.
     "
9070 :
9080 :
9090 :
9100 :
9110 PRINT"[CRSR/DOWN]      THE[RVS/ON]CL
     R[RVS/OFF] KEY CLEARS THE GRID.[CRS
     R/DOWN]"
```

KONTROLSUM FOR PROGRAM 2

```
3     249     4     113     5     75
6     249     7     58      125   165
127   39      160   58      200   242
210   242     220   243     230   246
240   58      250   226     260   202
270   102     280   107     290   100
300   105     9020  211     9030  6
9040  1       9050  26      9060  131
9070  58      9080  58      9090  58
9100  58      9110  93
```

SØSLAG

```
1000 REM**********************************
1010 REM SØSLAG FOR COMMODORE 64
1020 REM**********************************
1030 POKE55,255:POKE56,47:PRINT"[CLR/HOM
     E]":V=53248:CO=54272:POKE53281,0:PO
     KE53280,2:F=10
1040 PRINTTAB(14)"[CRSR/DOWN*8][RVS/ON]-
     -SØSLAG--"
1050 PRINTTAB(9)"[CRSR/DOWN*2]ET RUN PRO
     GRAM...VENT"
1060 FOR I=0TO2000:NEXT:PRINT"[CLR/HOME]
     "
1070 PRINTTAB(10)"[CRSR/DOWN*7]JOYSTICK
     I PORT 1":FORI=1TO2000:NEXT:PRINT"[
     CLR/HOME]"
1080 FORI=55816TO56296:POKEI,6:POKEI-CO,
     160:NEXT:G=0
1090 POKEV+21,63:POKE2040,192:POKE2041,1
     93:POKE2042,194:POKE2043,195:POKEV+
     44,7
1100 POKE2044,195:POKE2045,196:POKEV+39,
     1:POKEV+40,14:POKEV+41,0:POKEV+42,2
     1
1110 POKEV+43,7:POKEV+29,18:POKEV+23,16:
     FORI=0TO318:READQ:POKE12288+I,Q:NEX
     T:X=2
1120 Y=133:X1=29:Y1=160+INT(RND(1)*60):P
     OKEV,X:POKEV+1,Y:D=1:POKEV+2,X1:POK
     EV+3,
1130 FORJ=0TO24:FORI=55330TO55335:POKEI+
     J*40,2:POKEI+J*40-CO,160:NEXTI,J
1140 POKEV+5,160+INT(RND(1)*60):POKEV+4,
     70+INT(RND(1)*180)
1150 PRINT"[HOME][CRSR/DOWN*11][GRN][RVS
     /ON][LOGO/C][RVS/OFF][CRSR/DOWN][CR
     SR/LEFT][RVS/ON]     [RVS/OFF][SHIFT
     /£][RVS/ON]"
1151 POKEV+8,210:POKEV+9,55:GOSUB1400
1160 POKE54296,1:POKE54278,240:POKE54276
     ,129:TI$="000000"
1170 PRINT"[HOME][CRSR/DOWN*3][CRSR/LEFT
     *6][RED][RVS/ON]TIME"
1180 PRINT"[HOME][CRSR/DOWN*3][CRSR/LEFT
     *2][WHT]";RIGHT$(TI$,2):POKE55652+I
     NT(F)*40,4
1190 X1=X1+D*2:IFX1>240THEND=INT(RND(1)*
     3-3):Y1=160+INT(RND(1)*60)
1200 IFX1<28THEND=INT(RND(1)*3+1):Y1=160
     +INT(RND(1)*60)
1210 POKEV+2,X1:POKEV+3,Y1
1220 PE=PEEK(56321)
1230 IFPE=239ANDX>60ANDX2=0THENX2=X:Y2=Y
     +8:G1=0:F=F-.1:POKE53273,200
1240 IFX2>0THENGOSUB1420
1250 IFF<1THEN1320
1260 IFPE<>254ANDPE<>253ANDPE<>251ANDPE<
     >247THENPOKE54272,0:POKE54273,0:GOT
     O1350
1270 POKE54273,108:POKE54272,223
1280 IFPE=254ANDY>50THENY=Y-2
1290 IFPE=251ANDX>28THENX=X-4
```

```
1300 IFPE=247ANDX<230THENX=X+6
1310 F=F-.06:IFPE=253THENF=F-.1:G=G+2:GO
     TO1350
1320 IFY>140THEN1800
1330 IFF<1THENY=Y+G:GOTO1350
1340 G=0:GOTO1370
1350 Y=Y+G:G=G+.1:IFY>140THEN1800
1360 IFX>25ANDX<59ANDY>133ANDG>1.3THEN17
     40
1370 IFX>25ANDX<59ANDY>133THENG=0:F=10:Y
     =133:GOSUB1400
1380 POKEV,X:POKEV+1,Y:IFTI$="000100"THE
     N1670
1390 GOTO1180
1400 PRINT"[HOME][CRSR/DOWN*9][CRSR/LEFT
     *4][WHT]";
1401 PRINT"[RVS/ON][LOGO/@][CRSR/DOWN][C
     RSR/LEFT][LOGO/@][CRSR/DOWN][CRSR/L
     EFT][LOGO/@][CRSR/DOWN][CRSR/LEFT][
LOGO/@][CRSR/DOWN][CRSR/LEFT][LOGO/@][CR
SR/DOWN][CRSR/LEFT][LOGO/@]";
1402 PRINT"[CRSR/DOWN][CRSR/LEFT][LOGO/@
     ][CRSR/DOWN][CRSR/LEFT][LOGO/@][CRS
     R/DOWN][CRSR/LEFT][LOGO/@][CRSR/DOW
N][CRSR/LEFT][LOGO/@][CRSR/DOWN][CRSR/LE
FT][LOGO/@]"
1410 RETURN
1420 X2=X2+.5:IFY2>140THENY2=Y2+G1/2
1430 IFY2<141THENY2=Y2+G1
1440 IFG1=0THENPOKE54272,0:POKE54273,40
1450 G1=G1+.3
1460 POKEV+7,Y2
1470 POKEV+6,X2
1480 IFY2>230THENY2=0:X2=0:POKEV+7,0:G1=
     0
1490 IFPEEK(V+30)=10THEN1520
1500 IFPEEK(V+30)=12THEN1590
1510 RETURN
1520 POKEV+2,0:POKEV+3,0:POKEV+10,X1:POK
     EV+11,Y1:POKE54296,15
1530 FORI=70TO10STEP-.3:POKE54273,I:NEXT
     :POKE54273,0:POKE54272,0
1540 FORI=1TO11:POKEV+I,0:NEXT
1550 POKEV+10,0:POKEV+11,0:PRINT"[CLR/HO
     ME][CRSR/DOWN*2][WHT]DU HAR USKADEL
     IGGJORT U-BAADEN!!!!!"
1560 PRINT"[CRSR/DOWN*2][GRN]GLIMRENDE!!
     "
1570 PRINT"[CRSR/DOWN][YEL]DIN TID VAR "
     ;RIGHT$(TI$,2);" SEKUNDER"
1580 GOTO1850
1590 FORI=1TO11:POKEV+I,0:NEXT:POKEV+29,
     32:POKEV+23,32
1600 POKEV+2,0:POKEV+3,0:POKEV+10,X2:POK
     EV+11,Y2:POKE54296,15:POKE53281,8
1610 FORI=90TO0STEP-.2:POKE54273,I:POKE5
     3280,I/10:NEXT:POKE54273,0:POKE5427
     2,0
1620 POKE53280,0:POKE53281,0:POKEV+29,0:
     POKEV+23,0
1630 POKEV+10,0:POKEV+11,0:PRINT"[CLR/HO
     ME][CRSR/DOWN*2][WHT][RVS/ON]DU RAM
     TE UHELDIGVIS EN ATOMBOMBE!!!!!!!"
1640 PRINT"[CRSR/DOWN*2][GRN]IDIOT!!!!!"
1650 PRINT"[CRSR/DOWN][YEL]DU BRUGTE IAL
     T ";RIGHT$(TI$,2);" SEKUNDER."
1660 GOTO1850
1670 POKE54296,15:POKE54276,0:POKE54276,
     33
1680 FORI=255TO0STEP-2:POKE54273,I:NEXT:
     POKE54273,0:POKE54272,0
1690 FORI=0TO11:POKEV+I,0:NEXT:POKE53280
     ,0:POKE53281,0
1700 PRINT"[CLR/HOME][CRSR/DOWN*2][WHT]D
     UN TID ER UDLØBET!!!!"
1710 PRINT"[CLR/HOME][CRSR/DOWN*2][BLU]U
     -BAADEN VIL ANGRIBE DIT LAND!!!!!!!
     !"
1720 PRINT"[CRSR/DOWN*3][GRN]DIT STORE F
     JOLS!!!!!!"
1730 GOTO1850
1740 POKEV,0:POKEV+1,0:POKEV+10,X:POKEV+
     11,Y:POKE54296,15
1750 FORI=70TO20STEP-.2:POKE54273,I:NEXT
     :POKE54273,0:POKE54272,0
1760 FORI=1TO11:POKEV+I,0:NEXT
1770 POKEV+10,0:POKEV+11,0:PRINT"[CLR/HO
     ME][CRSR/DOWN*2][WHT]DU STØDTE IMOD
      DIT EGET SKIB!!!!!"
1780 PRINT"[CRSR/DOWN*2]BEDRE HELD EN AN
     DEN GANG!!"
1790 GOTO1850
1800 POKEV,0:POKEV+1,0:POKEV+10,X:POKEV+
     11,Y:POKE54296,15
1810 FORI=80TO30STEP-.2:POKE54273,I:NEXT
     :POKE54273,0:POKE54272,0
1820 FORI=1TO11:POKEV+I,0:NEXT
1830 POKEV+10,0:POKEV+11,0:PRINT"[CLR/HO
     ME][CRSR/DOWN*2][WHT]DU ER STYRTET
     I HAVET!!!!!"
1840 PRINT"[CRSR/DOWN*2][GRN]TRIST."
1850 PRINT"[HOME][CRSR/DOWN*14][CYN]VIL
     DU PRØVE IGEN? (J/N)."
1860 GETA$:IFA$="J"THENRUN
1870 IFA$<>"N"THEN1860
1880 PRINT"[CRSR/DOWN*2][PUR]TAK FOR NU!
     "
1890 END
1900 DATA0,0,0,0,0,0,192,0,0,192,0,0,240
     ,0,0,252,3,0,255,255,192,255,255,25
     2
1910 DATA63,255,255,63,255,240,31,240,0,
     63,252,0,63,0,0,0,0,0,0,0,0,0,0,0,0
     ,0
1920 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,99
1930 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
     ,56,0,0,56,0,0,124,0,3,255,192,127,
     255
1940 DATA252,255,255,255,255,255,252,15,
     255,224,0,0,0,0,0,0,0,0,0,0,0,0,0,0
     ,0,0
1950 DATA0,0,0,0,0,0,0,0,99
1960 DATA0,0,0,0,0,0,0,,0,0,0,0,4,8,16,2
     ,8,32,1,8,64,0,136,128,0,127,0,0,12
     7,0
1970 DATA7,255,240,0,127,0,0,127,0,0,136
     ,128,1,8,64,2,8,32,4,8
1980 DATA16,0,0,0,0,0,0,0,0,0,0,0,0,0,99
1990 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,28,0,0,62,0,0,127,0,0
2000 DATA127,0,0,127,0,0,62,0,0,28,0,0,0
     ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
     0,0
2010 DATA99,128,0,32,18,1,8,233,0,4,99,6
     3,1,48,84,32,19,8,1,89,109,22
2020 DATA8,12,38,0,52,149,128,243,211,99
     ,63,0,128,72,1,19,8,0,89,2,22
2030 DATA99,0,2,18,76,149,2,64,1,99,4,7,
     91,16,0,19,8,54,1,109,22
```

KONTROLSUM FOR SØSLAG

```
1000  165    1010  175    1020  165
1030  4      1040  6      1050  177
1060  152    1070  195    1080  234
1090  179    1100  76     1110  225
1120  147    1130  43     1140  88
1150  182    1151  186    1160  77
1170  46     1180  180    1190  206
1200  215    1210  56     1220  91
1230  63     1240  241    1250  34
1260  66     1270  249    1280  115
1290  116    1300  167    1310  244
1320  154    1330  61     1340  183
1350  172    1360  31     1370  62
1380  10     1390  83     1400  61
1401  199    1410  142    1420  217
1430  240    1440  233    1450  173
1460  133    1470  131    1480  14
1490  131    1500  140    1510  142
1520  194    1530  7      1540  229
1550  105    1560  246    1570  5
1580  87     1590  121    1600  252
1610  45     1620  79     1630  92
1640  59     1650  26     1660  87
1670  211    1680  220    1690  67
1700  202    1710  57     1720  32
1730  87     1740  130    1750  7
1760  229    1770  158    1780  86
1790  87     1800  130    1810  9
1820  229    1830  9      1840  225
1850  150    1860  161    1870  92
1880  102    1890  128    1900  38
1910  64     1920  161    1930  81
1940  167    1950  213    1960  93
1970  151    1980  216    1990  87
2000  29     2010  26     2020  15
2030  32
```

RUMREDNING FOR VIC-20

```
10 POKE52,28:POKE56,28:CLR
20 PRINTCHR$(147)CHR$(158):POKE36879,8:S
1=36875:S2=36876:S=210
30 FORJ=1TO5:READA$:N$(J)=A$:NEXT
40 SP$="[SHIFT/S*22]"
50 G$=CHR$(168)
60 FORJ=1TO5
70 T$=SP$+N$(J)+G$
80 FORI=1TO22
90 PRINTCHR$(19):FORK=1TOJ+5:PRINT:NEXT
100 PRINTMID$(T$,I,22)
110 NEXT:POKES1,S:POKES2,S:FORV=15TO0STE
P-1:POKE36878,V:NEXT:S=S+2:NEXT
120 FORV=15TO0STEP-.3:POKE36878,V:NEXT:P
OKES1,0:POKES2,0
130 FORI=7168TO7439:READA:POKEI,A:NEXT
140 FORI=7552TO7631:POKEI,PEEK(I+25600):
NEXT
150 POKE143,VAL(MID$(TI$,5,2))
160 POKE36879,25:PRINTCHR$(144)CHR$(147)
170 PRINT"DU HAR 3 MINUTTER TIL"
180 PRINT"AT REDDE 18 ASTRONAU-"
190 PRINT"TER, STRANDET I"
200 PRINT"ET RUM MINEFELDT. DU"
210 PRINT"SKAL SAMLE DEM OP,1"
220 PRINT"AF GANGEN,OG BRINGE"
230 PRINT"DEM TILBAGE TIL RUM-"
250 PRINT"SKIBET. RUMKAPSLERNE"
260 PRINT"HAR PHOTON STRAALER"
270 PRINT"TIL AT RYDDE MINERNE"
275 GOSUB400
280 PRINT"HVOR DYGTIG DU ER,"
290 PRINT"KOMMER AN PAA,HVOR"
300 PRINT"MANGE GANGE DU MAA"
310 PRINT"SKYDE.DU GENOPLADER"
320 PRINT"HVER GANG DU VENDER"
340 PRINT"TILBAGE TIL MODERSKI-"
350 PRINT"BET. PAS PAA IKKE AT"
360 PRINT"KOMME UDEN FOR RADAR"
370 PRINT"OMRAADET. BRUG JOY-"
380 PRINT"STIK TIL AT STYRE MED"
385 GOSUB400
390 POKE198,1:POKE631,131:END
400 PRINT:PRINT:PRINT"(BRUG EN AF TASTER
NE)"
410 GETA$:IFA$=""THEN410
420 PRINTCHR$(147):RETURN
430 DATARUM,REDNING,ET,RUN,PROGRAM
440 DATA24,24,126,24,24,36,36,36,0,0,36,
24,24,36,0,0
450 DATA24,60,24,24,60,126,255,36,36,255
,126,60,24,24,60,24
460 DATA64,96,242,127,127,242,96,64,2,6,
79,254,254,79,6,2
470 DATA0,0,3,4,15,16,42,64,24,231,0,24,
24,0,165,0
480 DATA255,120,32,16,15,0,0,0,255,0,36,
255,0,0,0,0
490 DATA0,0,192,32,240,8,84,2,255,30,4,8
,240,0,0,0
500 DATA0,0,247,132,244,20,247,0,0,0,189
,37,37,37,189,0,0,0,239,40,238,200,47,0
510 DATA153,126,126,255,255,126,126,153,
255,0,36,255,60,126,255,36
520 DATA255,0,36,255,0,0,0,24,60,24,24,6
0,126,255,36,0
530 DATA255,0,36,255,0,0,24,60,24,60,24,
24,60,126,255,36,0,0
540 DATA255,0,36,255,0,24,60,24,24,60,12
6,255,36,0,0,0
550 DATA255,0,36,255,24,60,24,24,60,126,
255,36,0,0,0,0
560 DATA255,0,36,255,60,24,24,60,126,255
,36,0,0,0,0,0
570 DATA255,0,36,255,24,24,60,126,255,36
,0,0,0,0,0,0
580 DATA255,0,36,255,24,60,126,255,36,0,
0,0,0,0,0,0
590 DATA255,255,255,255,255,255,255,255
600 DATA0,0,0,0,0,0,0,0,0,0,0,24,24,0,0,
0
```

DEL 2 INDLESES PAA BAANDET EFTER DEL 1.

```
100 CC=30720:SH=2:DR=0:S1=36877:VV=36878
    :AF=0
110 TS=3:DL=7756:SC=0:AC=0:M=6:S=15:R=36
    869
120 F$="                         FGJ"
130 K$="                         HIK"
140 POKE36879,238:PRINTCHR$(144)CHR$(147
    ):PRINT"HVOR DYGTIG ER DU? 1-9"
150 GETS$:IFS$=""ORS$<"1"ORS$>"9"THEN150
160 SK=(VAL(S$))*10:TI$="000000"
170 A$=TI$:PRINTCHR$(147):POKER,255:GOSU
    B910:AZ=3
180 RESTORE:FORH=1TO4:READA,B,C:FORT=ATO
    BSTEPC:POKET+CC,3:POKET,31:NEXT:NEXT
190 FORS=1TOSK:GOSUB580:POKET+CC,0:POKET
    ,1:NEXT:FORAS=1TO3:GOSUB580:POKET,0:
    NEXT
191 TI$=A$
200 GOSUB770
210 L=7778:GOSUB560
220 POKE37154,127:JY=(NOT((PEEK(37152)AN
    D128)/8+(PEEK(37151)AND60)/4))+32
221 POKE3754,255
230 IFTI$=>"0003000"THENPRINTCHR$(147):P
    RINT"TIME IS UP...":GOTO720
240 IFJY=1THENDR=-22:SH=2
250 IFJY=2THENDR=22:SH=3
260 IFJY=4THENDR=-1:SH=5
```

Fortsættes side 60.

Læserservice.

Benyt bestillingskupon på bagsiden

RUN's læserservice vil fremover bringe tilbud på bøger, software m.m. se annoncen i RUN. Har du ikke tid eller lyst til selv at indtaste de mange programmer i RUN, kan RUN's læserservice tilbyde et kassettebånd med samtlige programmer. Ved forudbetaling

kun kr. 98,00

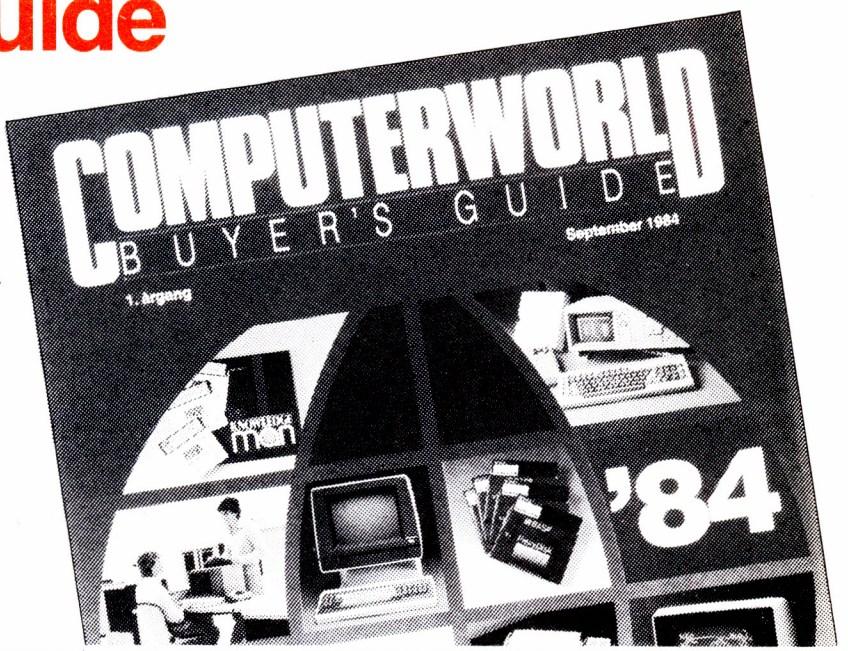
incl. moms, porto og ekspedition.

Buyer's Guide

Computerworld lancerer i september/oktober 1984 for første gang Buyers Guide. Buyers Guide er en indkøbs bog for professionelle edb-folk, brugere såvel som beslutningstagere. Buyers Guide vil være en uundværlig informationskilde i alle anskaffelsesfaser af edb-udstyr. I Buyers Guide vil man finde det samlede udbud af edb-maskiner, programmer og tilbehør på det danske marked.

Kr. 210;

+ moms kr. 46,20 i alt kr. 256,20
Benyt bestillingskupon på bagsiden.



Bøger fra Computerworld!

Bøger fra dataverden findes hurtigt og nemt i denne bogliste fra Computerworld litteraturservice. En lang række forlag er tilknyttet Computerworld litteraturservice. Det giver garanti for det største udvalg i edb-litteratur.
Nye edb-bøger annonceres i øvrigt i Computerworld som udkommer hver 14. dag.

Rekvirer bogliste. Benyt bestillingskupon på bagsiden.

JA!
Jeg (vi) bestiller

☐ Kassettebånd
☐ Bog nr.
☐ Bog nr.
☐
Navn:
Adresse:
Postnr. By:

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret
Computerworld betaler portoen

778

Reserveret postvæsenet

JA!
Jeg (vi) bestiller herved Buyers Guide

Firma:
Navn:
Adresse:
Postnr.: By:

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret
Computerworld betaler
portoen

778

Reserveret postvæsenet

JA!
Jeg (vi) bestiller herved boglist med
Computerworlds omfattende litteraturservice

Firma:
Navn:
Adresse:
Postnr.: By:

POSTKORT

Computerworld Danmark A/S
Gammel Strand 50
1045 København K

Postbesørges ufrankeret
Computerworld betaler
portoen

778

Reserveret postvæsenet

Computerworld

Computerworld Danmark A/S
Gammel Strand 50
1202 København K.

01·12 34 11

NYE BØGER

Norsk Programmeringskursus

– Vi synes, at bogen er virkelig god, og den fortjener at få en større udbredelse. Bogen udmærker sig blandt andet ved, at den både kan anvendes til selvstudium og være grundlag for studiekredse, siger Jan Nymann fra Commodore Data a/s om bogen »CBM 64 – Datakursus I«.

Bogen er den første i en serie på seks fra det norske firma RKD, og i Danmark er der to muligheder for at købe bogen. Enten gennem Commodore forhandlere, eller direkte fra importøren PLR-DATA CONSULT i Århus.

Datakursus I er begynderbogen i serien. Bind II, der er ved at være på markedet, når dette læses, beskæftiger sig med mere avanceret programmering, og går mere i dybden med enkeltemner. Blandt andet indeholder bogen en mere grundig gennemgang af programmet cursor-styring end bind et. Bind III omhandler styresystemer og hobby-elektronik, hvor Commodore 64 kan indgå. Bind IV handler om tekstbehandlings- og spreadsheetsystemer. Bind V behandler databaser og Bind VI regnskabssystemer.

Ifølge Steffen Høegh, PLR-DATA, vil hele serien blive taget hjem, efterhånden som de enkelte bind udkommer, hvis interessen for serien er tilstrækkelig stor.

Commodore Data er interesseret i, at markedsføre hele serien, hvis kvaliteten af de kommende bind holder sig på sammen niveau som Bind I.

– Vi er gået ind på at markedsføre bogen, fordi vi ønsker dokumentation omkring Commodore 64, og der findes meget lidt på det danske marked. Der er for så vidt ikke noget nyt i, at vi markedsfører andre firmaers produkter. Det gør vi også i andre sammenhæng, hvor vi mener, at der er et hul i Commodores eget udvalg, siger Jan Nymann.

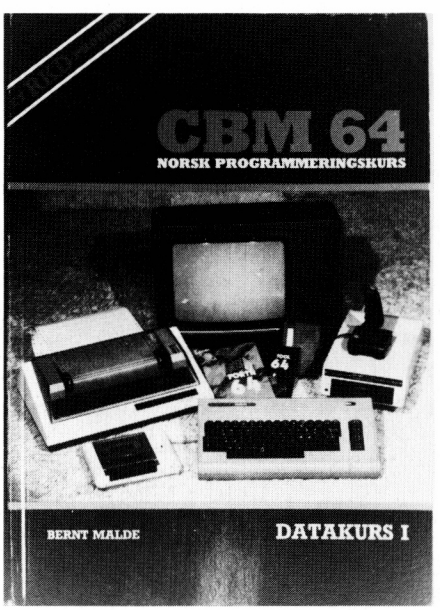
CBM 64 – Datakursus I er en begynderbog, der tager den nye Commodoreejer ved hånden helt fra starten. Men betegnelsen »begynderbog« skal i denne sammenhæng tages med et vist forbehold. Også den, der har arbejdet en del med sin Commodore 64 eller Vic 20, kan hente ideer og inspiration i bogen.

Forfatteren Bernt Malde er ansat ved Amtscentralerne i Norge, og bogen anvendes i EDB-undervisningen i mange norske skoler, hvor Commodore 64 er meget benyttet til undervisningsbrug. Men selv om bogen således er skrevet til lærerstyret undervisning, er den meget velegnet til selvstudium.

Bogen starter helt fra grunden med en gennemgang af, hvordan tastaturet er opbygget og eksempler på, hvad der sker på skærmen, når man taster noget ind. De første afsnit indeholder desuden et BASIC leksion, og en liste over fejlmeldinger og deres betydning. Man kan således lægge brugervejledningen helt væk, når man går frem efter bogen.

Selve indføringen i brugen af de enkelte kommandoer er klart og pædagogisk opbygget, med eksempler, der er godt forklaret, og opgaver til brugeren.

Opgaverne er så tilpas svære, at læseren løbende kan checke, at hun/han har fået det forventede udbytte af det gennemgåede stof. Opgaverne bliver løbende sværere i takt med inføringen i nyt stof, og de er hele tiden lagt på et niveau, hvor de tvinger læseren til at analysere problemet, før hun/han går i gang med løsningen af opgaven.



Dette giver i sig selv en meget dybtgående forståelse af stoffet, end lette opgaver ville have gjort. Forudsat selvfølgelig, at læseren ikke »snyder« og går om bag i bogen og ser det løsningsforslag forfatteren har angivet.

Som nævnt, kan også mere trænede brugere af Commodore-maskiner have glæde af bogen. Eksempelvis har bogen et program, der giver en pænere opstilling af udskrifter på skærm eller printer (formattering). Programmet er beregnet på at bruges som subrutine i et større program.

De sidste sider i bogen er helliget opbygningen af et program med diskettestation som database. Dette program kan anvendes som model for programdele, hvor man har burg for at arbejde med filer.

Hvis bogen skal have en kritisk bemærkning med på vejen, må det være for programlistningen i afsnittet om programmeret cursor-styring. Her har man anvendt de reserverede tegn for cursor-bevægelserne. Det kan på sin vis være i orden, fordi det er de tegn, man får frem på skærmen. Men det gør det sværere umiddelbart at forstå hvad der sker i programmet. Her ville det have været en fordel, hvis bogen havde anvendt samme princip for programlistningen som anvendes i RUN. Det forøger tilgængeligheden betydeligt, at man umiddelbart kan se hvad der sker i programmet, uden først at skulle tælle sig frem. Men det er en detalje i et ellers meget fint helhedsbillede af en bog, der rigeligt er sine 185,- kr. værd.

Alle bogens programeksempler leveres separat på to disekter til samme pris som bogen.

Hjælp til at komme godt fra start

Så er det store øjeblik kommet. Den nyerhvervede hjemmedatamat er pakket ud, ledningerne er forbundet, og skærmen melder, at computeren er klar til den første opgave. Den nybagte ejer slår op i brugervejledningen, og allerede nu opstår det første problem: Brugervejledningen er i bedste fald svær at forstå med eksempler, der kun siger meget lidt om de enkelte kommandoers brug. I værste fald er der tale om en uforståelig oversættelse fra en dårlig engelsk vejledning.

Mange bliver allerede på dette tidspunkt afskåret fra, at få den fulde glæde af deres datamat. De mange muligheder, der ligger gemt i den lille maskine, risikerer af forblive ubrugte, mens datamaten henslæber sit liv i familien som et avanceret »blip-spil«.

Men for de, der fortvivlet har lagt brugervejledningen væk, er der hjælp at hente i bogen »HJEMMEDATAMATEN – for dem der tror det er svært . . .« fra forlaget KOMMA.

Bogen tager fat helt fra bunden med små programmer. Langsomt bygges større programmer op, mens der inddrages flere og flere af de mest anvendte BASIC-kommandoer.

Hvert trin i de små programmer forklares løbende, så brugeren hele tiden er klar over, hvad der sker, når han/hun giver computeren den enkelte kommando.

Sideløbende med at nye kommandoer introduceres, giver bogen brugeren små opgaver, som er med til at øge forståelsen af detaljer i programmeringen – eksempelvis brugen af kommaer og semikolon i slutningen af en programlinie. Desuden henviser bogen løbende til et noteafsnit bagest i bogen, hvor læseren kan notere, hvordan man skal gøre, når man vil opnå et bestemt resultat, så de kommer til at virke som en art repetition.

Ind imellem forklaringerne af de forskellige kommandoer og opbygningen af småprogrammer, findes små programeksempler, der på en udmærket måde giver en ekstra illustration af, hvad det indlærte også kan bruges til.

I modsætning til brugervejledningen giver bogen på en letfattelig måde, selv den helt forudsætningsløse læser en god indgang til, at få glæde af hjemmedatamaten.

Det sidste afsnit i bogen handler om strengbehandling. Der er således en række muligheder tilbage i en commodore, som bogen ikke har behandlet. Men i kraft af, at bogen har givet den nye bruger en god indgang til selv at programmere, og dermed har afmystificeret datamaten, skulle det være muligt for læseren, at gå tilbage til brugervejledningen med så meget selvtillid, at det er muligt at finde ud af de mere avancerede muligheder.

Bogen fra KOMMA er den bedste begynderbog, jeg har set til dato, og den nye commodore-ejer vil have et godt udgangspunkt for at gå videre efter at have læst/brugt bogen, der primært er skrevet til commodore og sinclair-brugere. Det kan tilrådes enhver – gammel som ung – at ofre de knap 80 kroner bogen koster, samtidig med at man køber sin computer. *Khan.*

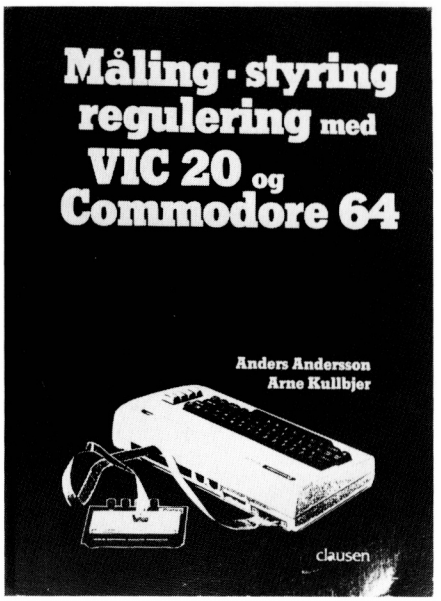
Den mystiske brugerport

Hvordan får man sin VIC-20 eller Commodore 64 til at styre husets energiforsyning eller hvordan får man den til at vande tomaterne i drivhuset, når fugtigheden bliver for lav?

De fleste af os har hørt noget om, at »sådan noget« kan en computer godt. Vi har også set, at der bag på computeren yderst til venstre sidder en såkaldt brugerport, som man aldrig bruger. Vi har nok en fornemmelse af, at de ting hænger lidt sammen.

Clausen bøger har netop udsendt en bog om disse ting. »Måling, styring og regulering med VIC-20 og Commodore 64« er titlen på bogen, der er oversat fra svensk.

Hvis man tror, at bogen kun er skrevet for eksperter, tager man fejl. Den



er, uden på nogen måde at være overfladisk, skrevet meget populært, så alle med en almindelig viden om programmering kan være med.

Bogen starter med at fortælle hvordan datamaten fungerer og fortsætter med at fortælle om brugerporten. Der er anvisninger på nogle ret enkle eksperimenter, som alle kan prøve.

Så fortsætter man med at fortælle om styring og måling, og her er det nok en fordel, at kende en lille smule til elektronik. Så går det videre med maskinkodning og kommunikation via RS-232C porten. Ganske kort fortælles om IEEE-bus for til sidst at runde af med nogle praktiske projekter.

Det er en fremragende bog, der tager et emne op, som de fleste viger uden om. Når bogen tilmed er skrevet, så enhver kan forstå det, er der ingen tvivl om, at bogen udfylder et hul og dækker et stort behov.

```
270 IFJY=16THENDR=1:SH=4
280 IFJY=8ANDDR<>0ANDPH<>0THENGOSUB600:I
    FGR=1THENGR=0:GOTO700
290 SL=PEEK(L+DR):IFL=7778ANDDR=-22THENG
    OSUB510:IFGR=1THENGR=0:GOTO800
300 IFGR=2THENGR=0:GOTO680
310 IFSL=>6ANDSL<=11ANDSL<>9THENGOSUB390
    :GOTO890
320 IFSL=1ANDAF=1THENAZ=AZ-1:AF=0:GOSUB3
    90:GOTO430
330 IFSL=1THENGOSUB390:GOTO430
340 IFSL=0ANDAF=1THENAZ=AZ-1
350 IFSL=0THENAF=1:POKE7690,0:N=220:GOSU
    B900
360 IFSL=31ANDAF=1THENPOKEL,32:AZ=AZ-1:A
    F=0:GOTO440
370 IFSL=31THENPOKEL,32:GOTO440
380 POKEL,32:POKEL+DR,SH:L=L+DR:POKEDL,9
    :GOTO220
390 POKEL,32:POKEL+DR+CC,2:POKEL+DR,15
400 A=132:FORV=15TO0STEP-.2:POKES1,A:POK
    EVV,V:A=A-.01:NEXT:POKES1,0:RETURN
410 POKEZ+DR+CC,2:POKEZ+DR,15:POKEZ+DR+C
    C,1:POKEZ+DR,32:POKES1,128
420 FORT=15TO0STEP-.5:POKEVV,T:NEXT:POKE
    S1,0:RETURN
430 POKEL+DR,32:POKEL+DR+CC,1
440 TS=TS-1:IFTS<>0ANDAZ<>0THEN880
450 IFTS<>0THEN800
460 PRINTCHR$(147):GOTO720
470 GOSUB760:POKER,240:PRINTCHR$(147):PR
    INT"NYT SPIL ?  J/N"
480 GETS$:IFS$=""THEN480
490 IFS$<>"N"THEN100
500 END
510 POKEDL,9:POKEDL+22,2:GOSUB860
520 DP=17:POKEVV,0:FORD=1TO7:POKEDL,DP:D
    P=DP+1:POKEDL+22,DP:GOSUB860:DP=DP+1
521 NEXT
530 POKEDL,16:POKEDL+22,32:GOSUB860:GOSU
    B860
540 POKE7690,32:N=190:GOSUB900:IFAZ=0THE
    NGR=1:RETURN
550 IFAF=1THENAF=0:AZ=AZ-1:AC=AC+1:SC=SC
    +(VAL(S$)):GR=2:RETURN
560 DP=30:FORD=1TO7:POKEDL+22,DP:DP=DP-1
    :POKEDL,DP:GOSUB860:DP=DP-1:NEXT:DR=
    0:
561 PH=SK/10
570 RETURN
580 T=INT(RND(1)*350)+7813:IFPEEK(T)<>32
    THEN580
590 RETURN
600 POKEVV,S:Z=L:PH=PH-1:FORB=1TO3:POKES
    1,230:G=PEEK(Z+DR):IFG=32THEN660
610 IFG=31THENB=3:GOTO670
620 IFG=0THENAZ-1:B=3
630 IFG=>6ANDG<=11THENGOSUB650:B=3:GR=1:
    RETURN
640 IFG=1THENGOSUB650:B=3:GOTO670
650 GOSUB410:GOTO670
660 POKEZ+DR+CC,0:POKEZ+DR,33:FORT=1TO5:
    NEXT:POKEZ+DR,32:POKEZ+DR+CC,1:Z=Z+D
    R
661 NEXT
670 POKES1,0:POKEVV,0:RETURN
680 GOSUB910:IFAZ=0ANDL=7778THEN800
690 GOSUB560:GOTO310
700 POKEZ+DR+CC,1:POKEZ=DR,36:FORT=1TO10
    00:NEXT:POKER,240
710 PRINTCHR$(147):PRINT"DU RAMTE MODERS
    KI-":PRINT:PRINT"BET!!!":SC=0:AC=0
720 POKER,240:PRINT:PRINT:PRINT"REDEDE A
    STRONAUTER:"AC:PRINT
721 PRINT"POINT OPNAAET:"SC
730 IFSC>HSTHENHS=SC
740 PRINT:PRINT"HIGH SCORE:"HS:GOTO470
750 PRINT:PRINT"DU HAR REDDET ":PRINT:PR
    INT"ASTRONAUTERNE!!":GOTO720
760 FORT=1TO4000:NEXT:RETURN
770 FORJ=23TO13STEP-1:GOSUB870
780 PRINTTAB(1)MID$(F$,J,20):PRINTTAB(1)
    MID$(K$,J,20):GOSUB860:NEXT
781 PRINTCHR$(144)
790 RETURN
800 FORJ=13TO1STEP-1:GOSUB870:GOSUB860:P
    RINTTAB(1)MID$(F$,J,20)
810 PRINTTAB(1)MID$(K$,J,20)
820 NEXT:M=M-1:IFAC=18THENPRINTCHR$(147)
    :GOTO750
830 IFM=0THENPRINTCHR$(147)CHR$(144):GOT
    O720
840 IFTS<>0THENPRINTCHR$(144):GOTO170
850 PRINTCHR$(147):GOTO720
860 FORT=1TO50:NEXT:RETURN
870 PRINTCHR$(5)CHR$(19):PRINT:RETURN
880 GOSUB910:AF=0:N=240:GOSUB900:GOTO210
890 POKEL+DR+CC,1:POKEL+DR,36:FORT=1TO10
    00:NEXT:GOTO710
900 POKEVV,S:POKES1-1,N:FORT=1TO25:NEXT:
    POKES1-1,0:POKEVV,0:RETURN
910 FORJ=7680TO7699:POKEJ,32:NEXT:PRINTC
    HR$(19)"LMN";SC:PRINTCHR$(19)TAB(17)
920 FORJ=1TOTS:PRINTCHR$(5)"B";:NEXT:PRI
    NTCHR$(144):RETURN
930 DATA7702,7723,1,7723,8185,22,8185,81
    64,-1,8164,7702,-22
```

Blok-delete til Commodore 64

Denne subrutine sætter dig i stand til at fjerne en »blok« at linienumre fra et program. Du lægger den her viste subroutine i slutningen af det program, hvorfra du vil have fjernet nogle linier. Du kan eventuelt bruge MERGE-routinen til at indlægge programmet.

Derefter skriver du RUN 60000 og du bliver nu bedt om at indtaste tre tal, adskilt af kommaer.

Første tal er første linienummer, du ønsker slettet. Andet tal er sidste linienummer og tredje tal angiver trin mellem hver slettet linienummer.

Et ettal til sidst vil således slette alle linier mellem de to første tal.

Hvis du derimod ved, at der f.eks. er et spring på 10 mellem dine programlinier, bør du skrive dette, da programmet ellers er 10 gange så lang tid om at køre igennem. Noget lynhurtigt program er det ikke.

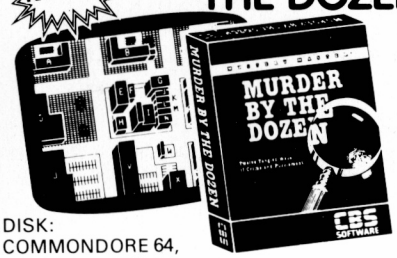
Program side 65.

EDB-avisen
Computerworld
Kontakt abonnementsafd.
01-12 34 11

Commodore-MARKED

NYHED

MURDER BY THE DOZEN



DISK:
COMMONDORE 64,
APPLE II og IBM PC.

Et nyt og anderledes spil.
Du er selv detektiv i 12 forskellige mordsager.
Du bestemmer selv, hvem du vil tage i forhør,
hvilke beviser der skal efterforskes o.s.v.
'MURDER BY THE DOZEN'
er intens og kræver planlægning,
god hukommelse og en smart hjerne.

CBS ELECTRONICS SOFTWARE

Fås i computer-, radio/TV-, fotoforretninger og boghandlere.

SOFT-WARE & TILBEHØR

LYS-PEN • JOYSTICK • TAPE
SPIL • BØGER • DISKETTER

Kun postordre, ring efter skema
for tilmelding til:

PROGRAMKLUB – køb/salg af dine programmer.

DATA-DEPOTET – bibliotek hvor du lejer programmer og tilbehør.

TAPEL APS

Rådmandsgade 8, 2. th. 2200 N
Tlf. 01-85 36 72. Giro 2 27 28 30

FOR KUN 890 KRONER KAN DU FÅ INTERFACE TIL VIC 20 OG COMMODORE 64 HOS CENTRONN.

Interface Parallel/Centronics og RS 232C. (Dobbelt handshake kontrol. Mulighed for printing af programmer). Afhentningspris 890 kroner incl. moms. Sendes over hele landet pr. efterkrav for en merpris (porto og gebyr) på 33 kroner. Ring 01-12 79 07 eller kom.

centronn
elektronik i centrum
Gl. Torv 6, 1457 København K.

Centronn er en afdeling af H. Christensen & Søn, der er datterselskab af De Forenede Papirfabrikker.

COMPUTER

COMMODORE JAMES SPECTRAVIDEO
DRAGON SHARP
SPECTRUM MEMOTECH
MICROBEE

Alt i computer og perifert udstyr, EDB-borde og tilbehør.

MIBOLA – stedet med den gode service.

MIBOLA MIKRODATA

Østerbrogade 25 • 2100 København Ø
Tlf. 01-42 19 66
Det rigtige datagrej til erhverv og fritid

TILBEHØR!

NYHED! **Bogen alle Commodore 64-brugere har ventet på... NU i Danmark. Machinecode games rutine for Commodore 64, KUN kr. .. 149:-**

QUICKSHOT JOYSTICK!
Det rigtige joystick, også til Commodore - med autofire og sugekopper. KUN kr.
188:-

NB! Også postordre!

JC JUMBO DATA *02-36 36 11
JERNBANEGADE 58
4000 ROSKILDE

MEICOM DATA 01-39 79 22
til Commodore 64

NYHEDER!!!!

QUICK SHOT 2	148,-
TEGNEPEN	395,-
BEACH HEAD	186,-
STRIP POKER	186,-
DECATHLON	186,-
MISSION 1	148,-
DISKBOX 40st	115,-

+20 andre nyheder til 64'eren.

Vi har også:

SPECTRUM QL	7995,-
MicD. bånd	58,-

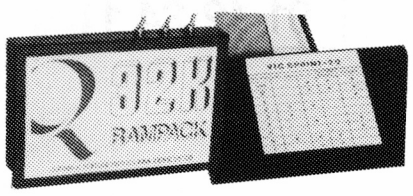
m.m.
Ring, skriv eller kig ind

MEICOM DATA

Fredensgade 5 . 2200 København N
Tlf. 01-39 79 22

Alt i interface og udstyr

til **VIC-20 + 64** og øvrige CBM varer. Også harddisk, der betjenes som en 8750, kan tilsluttes samtlige de af Commodore fremstillede computere.



U.I.B. Electronic & Data INC.
Lillevangsvej 3, 3520 Farum
02-95 51 70

NYHED

OPLEV EN NY VERDEN MED

PIXSTIK

- en fantastisk lyspen som tegner direkte på TV-skærmen.

KASSETTE:
COMMONDORE 64

Med Pixstik kan du selv tegne og farvelægge i op til 16 forskellige farver - fra frihånd til konstruktion.
Ved hjælp af hukommelsen kan du arbejde med flere billedvariationer.
Kombinations mulighederne er uendelige - det er din fantasi der bestemmer.

PIXSTIK leveres med tegneprogrammet PAINTBOX på kassette.
Vejledende udsalgspris: **395,-**

Ekstra programmer: MELODY - du komponerer dine egne melodier.
Vejledende udsalgspris kr. 99,-
TIC TAC TOW/FLAK - et dobbeltprogram, hvor du spiller 'kryds og bolle'.
Eller i Flak hvor du kæmper mod fjentlige fly og lyspennen er din aftrækker.
Vejledende udsalgspris kr 99,-

CBS

Fås i computer-, radio/TV-, fotoforretninger og boghandlere.

ELECTRONICS SOFTWARE

TRIGONOMETRI

```
100 REM ********************
110 REM ********************
120 REM ** TRIGONOMETRI **
130 REM **                **
140 REM **       BY       **
150 REM **                **
160 REM ** ARNE PETERS    **
170 REM **                **
180 REM **  22/9 1984     **
190 REM ********************
200 REM ********************
210 RA=180/[SHIFT/↑]
220 POKE53280,0:POKE53281,0
230 GOSUB570
240 PRINT"[HOME][CRSR/DOWN*5]"
250 PRINTTAB(14)"HVILKE VAERDIER ER KEND
    TE:"
260 PRINTTAB(60)"[RVS/ON] 1 [RVS/OFF]  B
    C & CA"
270 PRINTTAB(20)"[RVS/ON] 2 [RVS/OFF]  B
    C & BA"
280 PRINTTAB(20)"[RVS/ON] 3 [RVS/OFF]  C
    A & BA"
290 PRINTTAB(20)"[RVS/ON] 4 [RVS/OFF]  B
    C & A"
300 PRINTTAB(20)"[RVS/ON] 5 [RVS/OFF]  C
    A & A"
310 PRINTTAB(20)"[RVS/ON] 6 [RVS/OFF]  B
    A & A"
320 PRINTTAB(20)"[RVS/ON] 7 [RVS/OFF]  B
    C & B"
330 PRINTTAB(20)"[RVS/ON] 8 [RVS/OFF]  C
    A & B"
340 PRINTTAB(20)"[RVS/ON] 9 [RVS/OFF]  B
    A & B"
350 PRINTTAB(100)"[RVS/ON]VAELG[RVS/OFF]
     >>"
360 GETA$:IFA$=""THEN360
370 IFA$="1"THENL=1:GOTO470
380 IFA$="2"THENL=2:GOTO470
390 IFA$="3"THENL=3:GOTO470
400 IFA$="4"THENL=4:GOTO470
410 IFA$="5"THENL=5:GOTO470
420 IFA$="6"THENL=6:GOTO470
430 IFA$="7"THENL=7:GOTO470
440 IFA$="8"THENL=8:GOTO470
450 IFA$="9"THENL=9:GOTO470
460 GOTO360
470 GOSUB570
480 ONLGOSUB800,910,1030,1150,1270,1390,
    1510,1630,1750
490 GOSUB570
500 GOSUB1870
510 PRINTTAB(97)"ØNSKES FLERE"
520 PRINTTAB(17)"BEREGNINGER (J/N)"
530 GETA$:IFA$=""THEN530
540 IFA$="J"THEN230
550 IFA$="N"THENPRINT"[CLR/HOME]PROGRAMM
    ET ER AFSLUTTET !!!":END
560 GOTO530
570 PRINT"[CLR/HOME]"
580 PRINT"[CRSR/DOWN]TRIGONOMETRI:"
590 PRINT"[LOGO/T*12]"
600 PRINT"[CRSR/DOWN]"
610 PRINT"B"
620 PRINT" [LOGO/M][SHIFT/M]"
630 PRINT" [LOGO/M] [SHIFT/M]"
640 PRINT" [LOGO/M]  [SHIFT/M]"
650 PRINT" [LOGO/M]   [SHIFT/M]"
660 PRINT" [LOGO/M]    [SHIFT/M]"
670 PRINT" [LOGO/M]     [SHIFT/M]"
680 PRINT" [LOGO/M]      [SHIFT/M]"
690 PRINT" [LOGO/M]       [SHIFT/M]"
700 PRINT" [LOGO/M]        [SHIFT/M]"
710 PRINT" [LOGO/M]         [SHIFT/M]"
720 PRINT" [LOGO/M]          [SHIFT/M]"
730 PRINT" [LOGO/M]           [SHIFT/M]"
740 PRINT" [LOGO/M]            [SHIFT/M]
750 PRINT" [LOGO/M][SHIFT/P]
    [SHIFT/M]"
760 PRINT"  [LOGO/T*14]"
770 PRINT"C                  A"
780 PRINT"   VINKLER I GR."
790 RETURN
800 PRINT"[HOME][CRSR/DOWN*5]"
810 PRINTTAB(20)"BC = ";
820 INPUTBC
830 PRINTTAB(20)"CA = ";
840 INPUTCA
850 A=(ATN(BC/CA))*RA
860 B=(ATN(CA/BC))*RA
870 BA=SQR(BC↑2+CA↑2)
880 F=BC*CA/2
890 O=BC+CA+BA
900 RETURN
910 PRINT"[HOME][CRSR/DOWN*5]"
920 PRINTTAB(20)"BC = ";
930 INPUTBC
940 PRINTTAB(20)"BA = ";
950 INPUTBA
960 IFBC=>BATHENPRINTTAB(20)"UGYLDIGE VA
    ERDIER !!":GOTO910
970 A=(ATN((BC/BA)/SQR(-(BC/BA)*(BC/BA)+
    1)))*RA
980 B=(-ATN((BC/BA)/SQR(-(BC/BA)*(BC/BA)
    +1))+[SHIFT/↑]/2)*RA
990 CA=SQR(BA↑2-BC↑2)
1000 F=(BC/2)*(SQR((BA+BC)*(BA-BC)))
1010 O=BC+CA+BA
1020 RETURN
1030 PRINT"[HOME][CRSR/DOWN*5]"
1040 PRINTTAB(20)"CA =";
1050 INPUTCA
1060 PRINTTAB(20)"BA =";
1070 INPUTBA
1080 IFCA=>BATHENPRINTTAB(20)"UGYLDIGE V
     AERDIER !!":GOTO1030
1090 A=(-ATN((CA/BA)/SQR(-(CA/BA)*(CA/BA
     )+1))+[SHIFT/↑]/2)*RA
1100 B=90-A
1110 BC=SQR(BA↑2-CA↑2)
1120 F=BC*CA/2
1130 O=BC+CA+BA
1140 RETURN
1150 PRINT"[HOME][CRSR/DOWN*5]"
1160 PRINTTAB(20)"BC =";
1170 INPUTBC
1180 PRINTTAB(20)"A  =";
1190 INPUTA
1200 IFA=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL !!":GOTO1150
1210 CA=BC*(1/(TAN(A*[SHIFT/↑]/180)))
1220 BA=BC/(SIN(A*[SHIFT/↑]/180))
1230 B=90-A
1240 F=CA*BC/2
1250 O=BC+CA+BA
1260 RETURN
1270 PRINT"[HOME][CRSR/DOWN*5]"
1280 PRINTTAB(20)"CA =";
1290 INPUTCA
1300 PRINTTAB(20)"A  =";
```

```
1310 INPUTA
1320 IFA=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL !!":GOTO1270
1330 BC=CA*TAN(A*[SHIFT/↑]/180)
1340 BA=CA/COS(A*[SHIFT/↑]/180)
1350 B=(ATN(CA/BC))*RA
1360 F=CA*BC/2
1370 O=BC+CA+BA
1380 RETURN
1390 PRINT"[HOME][CRSR/DOWN*5]"
1400 PRINTTAB(20)"BA =";
1410 INPUTBA
1420 PRINTTAB(20)"A  =";
1430 INPUTA
1440 IFA=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL !!":GOTO1390
1450 BC=BA*SIN(A*[SHIFT/↑]/180)
1460 CA=BA*COS(A*[SHIFT/↑]/180)
1470 B=(ATN(CA/BC))*RA
1480 F=CA*BC/2
1490 O=BC+CA+BA
1500 RETURN
1510 PRINT"[HOME][CRSR/DOWN*5]"
1520 PRINTTAB(20)"BC =";
1530 INPUTBC
1540 PRINTTAB(20)"B  =";
1550 INPUTB
1560 IFB=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL[SHIFT/SPACE]!!":GOTO1510
1570 CA=BC*TAN(B*[SHIFT/↑]/180)
1580 BA=BC/COS(B*[SHIFT/↑]/180)
1590 A=(ATN(BC/CA))*RA
1600 F=BC*CA/2
1610 O=BC+CA+BA
1620 RETURN
1630 PRINT"[HOME][CRSR/DOWN*5]"
1640 PRINTTAB(20)"CA =";
1650 INPUTCA
1660 PRINTTAB(20)"B  =";
1670 INPUTB
1680 IFB=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL !!":GOTO1630
1690 BC=CA/TAN(B*[SHIFT/↑]/180)
1700 BA=SQR(BC↑2+CA↑2)
1710 A=90-B
1720 F=BC*CA/2
1730 O=BC+CA+BA
1740 RETURN
1750 PRINT"[HOME][CRSR/DOWN*5]"
1760 PRINTTAB(20)"BA =";
1770 INPUTBA
1780 PRINTTAB(20)"B  =";
1790 INPUTB
1800 IFB=>90THENPRINTTAB(20)"UGYLDIG VIN
     KEL !!":GOTO1750
1810 BC=COS(B*[SHIFT/↑]/180)*BA
1820 CA=SQR(BA↑2-BC↑2)
1830 A=90-B
1840 F=BC*CA/2
1850 O=BC+CA+BA
1860 RETURN
1870 PRINT"[HOME][CRSR/DOWN*3]"
1880 PRINTTAB(12)"VAERDIERNE ER:"
1890 PRINTTAB(52)"A     =";A
1900 PRINTTAB(12)"B     =";B
1920 PRINTTAB(12)"BC    =";BC
1930 PRINTTAB(12)"CA    =";CA
1940 PRINTTAB(12)"BA    =";BA
1950 PRINTTAB(12)"AREAL =";F
1960 PRINTTAB(12)"OMKR. =";O
1970 RETURN
```

KONTROLSUM FOR TRIGONOMETRI

```
100  163   110  163   120  58
130  23    140  114   150  23
160  208   170  23    180  185
190  163   200  163   210  138
220  37    230  41    240  69
250  15    260  211   270  207
280  207   290  143   300  143
310  143   320  147   330  147
340  147   350  233   360  102
370  75    380  77    390  79
400  81    410  83    420  85
430  87    440  89    450  91
460  34    470  41    480  124
490  41    500  93    510  91
520  113   530  101   540  108
550  25    560  33    570  112
580  203   590  129   600  238
610  31    620  113   630  145
640  177   650  209   660  241
670  17    680  49    690  81
700  113   710  145   720  177
730  209   740  241   750  193
760  7     770  97    780  168
790  142   800  69    810  72
820  10    830  71    840  9
850  75    860  76    870  179
880  140   890  225   900  142
910  69    920  72    930  10
940  70    950  8     960  51
970  145   980  197   990  180
1000 104   1010 225   1020 142
1030 69    1040 39    1050 9
1060 38    1070 8     1080 92
1090 193   1100 73    1110 180
1120 140   1130 225   1140 142
1150 69    1160 40    1170 10
1180 4     1190 198   1200 52
1210 42    1220 250   1230 73
1240 140   1250 225   1260 142
1270 69    1280 39    1290 9
1300 4     1310 198   1320 55
1330 170   1340 167   1350 76
1360 140   1370 225   1380 142
1390 69    1400 38    1410 8
1420 4     1430 198   1440 58
1450 168   1460 166   1470 76
1480 140   1490 225   1500 142
1510 69    1520 40    1530 10
1540 5     1550 199   1560 181
1570 171   1580 169   1590 75
1600 140   1610 225   1620 142
1630 69    1640 39    1650 9
1660 5     1670 199   1680 56
1690 172   1700 179   1710 73
1720 140   1730 225   1740 142
1750 69    1760 38    1770 8
1780 5     1790 199   1800 59
1810 168   1820 180   1830 73
1840 140   1850 225   1860 142
1870 35    1880 231   1890 175
1900 173   1920 19    1930 17
1940 15    1950 84    1960 95
1970 142
```

RUN Amok

Under denne overskrift vil vi nu og fremover komme med rettelser og korrektioner til ting, der tidligere har været bragt i RUN. Vi håber naturligvis, at denne spalte vil være af beskedent omfang, men det er svært at garantere 100%.

I RUN nr. 1 har vores printprogram som tidligere nævnt drillet lidt. Det har givet anledning til nogle programfejl, og vi vil her vise rettelser til linienumrene i de gamle programmer.

I BOWLING – side 18 og 19 – skal følgende linier rettes:

```
1010 C$(2)=C$(1)+"[CRSR/RIGHT*18]":C$(3)
     =C$(2)+"[CRSR/RIGHT*13]":C$=C$(1)
1360 PRINTLEFT$(C$,12)+"[CRSR/DOWN*7]"+"
                                      "
1460 PRINTLEFT$(C$,12)+"[CRSR/DOWN*7]"+"
                                      "
```

I VIC-BOWLING – på side 20 – skal der rettes disse linier:

```
1430 PRINTLEFT$(C$,11)+"[CRSR/DOWN*7]"+"
                                    "
1440 PRINT"                         "
1710 PRINT"[CRSR/RIGHT*5]"A$(1);T(1):IFA
     >1THENPRINT"[CRSR/RIGHT][CRSR/DOWN]
     [CRSR/RIGHT*4]"A$(2);T(2)
```

Desværre må PRIVATREGNSKAB del 2 af tekniske årsager udskydes til næste nummer. Til gengæld kan vi love, at dette afsnit vil blive kraftigt udvidet.

I PRIVATREGNSKAB – side 30 og 31 – skal følgende rettes:

```
11260 IFG1<>G2 AND G2<>0 THEN GOSUB 2040
      0: GOTO11210
13520 FORI=0TO12:FORII=0TO100
```

I MUSICBASE/C64 – på side 34 – er der en enkelt linie, der skal rettes.

```
60545 PRINT"[CRSR/DOWN]PRINT#3" :PRINT"[
      CRSR/DOWN*2]CLOSE 3" :PRINT"[CRSR/
      DOWN*2]@$" "[CRSR/UP*7]
```

SIMONS BASIC lider af samme skavank og her er der en fejl i linie nr. 970 på side 42.

```
970 PRINT"[CRSR/DOWN]VIL DU LAVE EN NY S
    PRITE? (J/N)!:POKE198,0:[SHIFT/D][SH
    IFT/F]1
```

I INVASION FRA RUMMET – på side 46 – skal følgende

```
10500 FOR I=15 TO 0 STEP-.1:POKE54296,I:
      NEXT:POKE54287,0:POKE54286,0
10501 POKESC+Y*40+X,0
10550 INPUT"NYT SPIL? J/N?";Q$
10560 IF Q$<>"J"ANDQ$<>"N"THEN10550
10610 FOR Y1=0 TO 4:FOR X1=0 TO5:POKE762
      0+Y1*12+X1*2,Y1*2+1
10611 POKE7621+Y1*12+X1*2,X1*3
10670 FORY1=1TO4:PRINT"[RED][HOME][CRSR/
      DOWN*18]"SPC(Y1*10-7)"EEE[CRSR/DOW
      N][CRSR/LEFT*3]EEE[CRSR/DOWN][CRSR
/LEFT*3]EEE";
```

SOFTWARE TIL DEM

- OG DERES
COMMODORE COMPUTER

FAKTURERING etc.
- 400 til 1400 kunder/debitorer
- 25 til 3000 varenumre
- debitor- og lagerstyring
- med mere...

FINANS
- 200 til 1000 konti
- 2800 til 4200 posteringer
- balancer etc., når De synes
- med mere...

- TIL AT FINDE UD AF!

Leveres til Commodore 64 med
1541 eller 8050 floppydisk.
Også til SX 64.

Kontakt Deres forhandler
eller

MICRO-SYS SERIØS SOFTWARE

BØGESKOV HØVEJ 7C 8260 VIBY J TLF 06-282299

MERGE PROGRAM TIL C64

```
100 S=49152:FORJ=STOS+78:READV
110 C=C+V:POKEJ,V:NEXT
130 DATA 169,0,133,10,32,212,225,165
140 DATA 43,72,165,44,72,56,165,45
150 DATA 233,2,133,43,165,46,233,0
160 DATA 133,44,169,0,133,185,166,43
170 DATA 164,44,169,0,32,213,255,176
180 DATA 14,134,45,132,46,32,51,165
190 DATA 104,133,44,104,133,43,96,170
200 DATA 201,4,144,244,240,10,104,133
210 DATA 44,104,133,43,24,108,0,3
220 DATA 164,186,136,240,209,208,239
230 NEW
```

MERGE PROGRAM TIL VIC-20

```
90 POKE55,176:POKE56,29
100 S=7601:FORJ=STOS+78:READV
110 C=C+V:POKEJ,V:NEXT
130 DATA 169,0,133,10,32,212,225,165
140 DATA 43,72,165,44,72,56,165,45
150 DATA 233,2,133,43,165,46,233,0
160 DATA 133,44,169,0,133,185,166,43
170 DATA 164,44,169,0,32,213,255,176
180 DATA 14,134,45,132,46,32,51,165
190 DATA 104,133,44,104,133,43,96,170
200 DATA 201,4,144,244,240,10,104,133
210 DATA 44,104,133,43,24,108,0,3
220 DATA 164,186,136,240,209,208,239
230 NEW
```

DELETE ROUTINE TIL C64

```
60000 REM BLOK DELETE (SUBROUTINE)
60010 INPUT"LINIE: FRA,TIL,TRIN";F,T,TR:
      PRINTCHR$(147)
60020 PRINTCHR$(19)F:F=F+TR:PRINT"60040
      F="F":T="T":TR="TR:PRINT"GOTO60040
      "
60030 POKE631,19:POKE632,13:POKE633,13:P
      OKE634,13:POKE198,4:END
60040 F=45:T=40:TR=5
60050 IFF>TTHENPRINTCHR$(147):END
60060 GOTO60020
```

KONTROLSUM

```
62000 REM KONTROLSUM PROGRAM
62010 GOSUB 62050
62020 GOTO 62200
62030 IF FL>=0 THEN 62020
62040 END
62050 DEFFN DEEK(X) = PEEK(X)+256*PEEK(X
      +1)
62060 DATA ***
62070 DATA 165,252,166,253,133,020,134,0
      21,032,019
62080 DATA 166,216,160,001,177,095,133,2
      54,240,013
62090 DATA 200,177,095,133,252,200,177,0
      95,133,253
62100 DATA 200,169,000,133,251,177,095,2
      40,006,024
62110 DATA 101,251,200,208,244,096
62120 DATA -1
62130 AD = 52992
62140 RESTORE
62150 READ T$: IF T$<>"***" THEN 62150
62160 READ T : IF T>=0 THEN POKE AD,T :
      AD = AD+1 : GOTO 62160
62170 PRINT"[CLR/HOME]HVOR ØNSKES KONTRO
      LSUM PRINTET:"
62180 INPUT"TV/MONITOR=0 PRINTER=4 PLOTT
      ER=6"; DEV
62190 RETURN
62200 REM DO INITALISATION
62210 FL = 0 : INPUT "FØRSTE LINE "; FL
      : IF FL<0 THEN RETURN
62220 LL = 65536 : INPUT "SIDSTE LINE ";
       LL:PRINT"[CLR/HOME]"
62230 IF DEV>0 THEN OPEN 1,DEV
62240 LN = FL : C = 0 : C1 = 0
62250 POKE 252,LN-INT(LN/256)*256 : POKE
       253,LN/256
62260 SYS 52992:CS=PEEK(251):LN=FNDEEK(2
      52)+1
62270 T$ = LEFT$(STR$(LN-1)+"       ",6)+
      LEFT$(STR$(CS)+"       ",7)
62280 IF DEV=0 THEN PRINT T$;
62290 IF DEV>0 THEN PRINT#1,T$;
62300 IF DEV=0 THEN C = C+1 : IF C>=3 TH
      EN PRINT : C = 0 : C1 = C1 + 1
62310 IF DEV>0 THEN C = C+1 : IF C>=3 TH
      EN PRINT#1 : C = 0 : C1 = C1 + 1
62320 IF LN<=LL AND PEEK(254) THEN 62250
62330 IF DEV>0 THEN PRINT#1:CLOSE1
62340 END
```

KONTROLSUM AF KONTROLSUM

```
62000 5      62010 170    62020 163
62030 177    62040 128    62050 179
62060 33     62070 30     62080 38
62090 46     62100 31     62110 14
62120 1      62130 130    62140 140
62150 167    62160 29     62170 65
62180 227    62190 142    62200 58
62210 47     62220 199    62230 77
62240 13     62250 207    62260 255
62270 189    62280 191    62290 250
62300 114    62310 161    62320 197
62330 38     62340 128
```

Læserservice

Programlistninger

Hjælpeprogrammer til Simons Basic 13
Diskdata 28
Total Music 34
Tipsprogram 42
Canyons of Zelaz 45
F-taster 51
Super Sprite 52
Søslag 55
Rumredning 57
Trigonometri 62
MERGE 65
Kontrolsum 65

Har du ikke tid eller lyst til selv at indtaste ovenstående programmer, kan du bestille dem på kassettebånd ved indsendelse af kuponen her i bladet.

Prisen er kr. 98,- incl. moms, porto og ekspeditionsgebyr. Af praktiske årsager bedes du venligst sende pengene samtidig med bestillingen – enten pr. check eller på postgirokonto 1 48 31 61. Husk at anføre på girotalonen, at bestillingen gælder et kassettebånd.

Ønsker du programmer leveret på discette, er prisen kr. 118,-.

Indtastnings-vejledning

Alle programmer, der bringes i RUN er forsynet med en såkaldt kontrolsum, så du bagefter kan kontrollere, om du har tastet rigtigt ind og evt. hurtigt finde indtastningsfejl.

Du starter med at indtaste programmet på foregående side og gemmer det til fremtidig brug.

Ved indtastning af de øvrige programmer er fremgangsmåde følgende:

1. Først indlæser du programmet »Kontrolsum«.
2. Så indtaster du dit program.
3. Tag altid en kopi af dit program inden du tester det.
4. Skriv GOTO 62000 og besvar de spørgsmål som programmet stiller.
5. Sammenlign de kontrolsumme med det, der er trykt i bladet og find de linier, hvor du evt. har tastet fejl.

Promramlistningen i RUN

Der er ved listningen af programmer i RUN anvendt et særligt printprogram, der skulle gøre indtastningen af programmer væsentlig lettere end ved en almindelig listning, hvor de grafiske tegn ikke er til at læse. I mange tilfælde skal du trykke på to taster for at indtaste et tegn. I programmet er de to taster anført mellem to kantede parenteser og delt med en skråstreg f.eks.

(SHIFT/A) eller (LOGO/A)

hvor du skal anvende SHIFT-tasten eller LOGO-tasten sammen med A.

Du vil også komme ud for

(BLK) eller (RVS/ON)

i de tilfælde skal du bruge CTRL-tasten sammen med henholdsvis 1 og 9-tasten.

Står der indenfor de kantede parenteser f.eks.

(SHIFT/A* 8)

skal du taste det pågældende tegn 8 gange.

Lige en ting, du til slut skal være opmærksom på. Kontrolsummen kommer ikke til at passe, hvis du sletter eller indsætter et mellemrum.

God fornøjelse.

Indsendelse af materiale til RUN

RUN vil som tidligere nævnt gerne have en bred læserkontakt. Har du derfor et program, som du mener har interesse for andre, er du meget velkommen til at sende det på bånd eller diskette til RUN, sammen med nogle bemærkninger om programmet. Ønsker du det returneret, bedes du venligst vedlægge svarkuvert.

Har du programmeringstips eller ideer til emner, du ønsker belyst, modtager vi også gerne din henvendelse. Vi vil ligeledes gerne støtte nuværende og fremtidige brugerklubber, ved at trykke adresserne her i bladet. Har du spørgsmål eller har du noget at sælge? Som privatperson er det gratis at bruge RUN.

Vi glæder os til at høre fra dig.

RUN EXTRA
udkommer omkring 1. december

med masser af programmer og programanmeldelser

RUN nr. 1 - 1985 udkommer i slutningen af januar.

Gennemgang af de sidste nyheder

Gennemgang af tekstbehandlingssystemer

Privatregnskab

Afslutning af Simons Basic

Grundig orientering om COMAL 80

. . . masser af programmer

Calc Result og Commodore 64
Den bedste hjælp man kan tænke sig.
Slut med kedelige kalkuler med papir og blyant.

Dette er Mogens Skovbon, direktør og indehaver af Skovbon Fashion A/S.

Han er konfektionsfabrikant, og anvender Calc Result Advanced til at foretage kalkulationer og budgetlægninger i sin virksomhed.

Calc Result er lige noget for mig

Jeg købte min Commodore 64 sammen med Calc Result. Jeg ville anvende Calc Result til at foretage produktkalkuler på den konfektion vi fremstiller. Min Commodore 64 med Calc Result står på skrivebordet, let tilgængelig, når jeg skal træffe en hurtig beslutning.

Af og til skal jeg hurtigt beslutte om om det kan betale sig, at fremstille en bestemt kollektion. Skal hele produktionen ske hos os selv, hos underleverandør eller i udlandet? Hvilken effekt vil det få, hvis vi anvender et andet stykke klæde? Det går lynhurtigt med Calc Result. Man indsætter et nyt tal, foretager en hurtig omkalkulering, og det er færdigt. Perfekt!

Jeg anvender også Calc Result til at foretage efterkalkulationer, så jeg kan kontrollere om mine beregninger stemte, eller om kalkulationsmodellen skal ændres. Jag får helt enkelt bedre kontrol med produktionsomkostningerne. Calc Result er et uundværderligt supplement til vort større dataanlæg.

På møder plejer jeg at medbringe udskrifter af mine kalkuler. Og har jeg behov for at anskueliggøre dette, er Calc Results søjlediagram perfekt.

Sparer tid med Calc Result

Tidligere foretog jeg alle produktionskalkuler i hånden, med papir og blyant, kedeligt og besværligt. Med Calc Result når jeg måske at foretage 10 produktionskalkuler på samme tid, som det før tog at lave en. Jeg kan ganske simpelt udnytte min tid meget bedre, takket være Calc Result.

Tjener penge på Calc Result

Den effektivitet og tidgevinst jeg har opnået ved at anvende Calc Result, har ganske givet resulteret i en forøget indtjening til firmaet på 10.000-tals kroner. Nu kan vi udnytte vore produktionsresurser mere effektivt og økonomisk end før.

De kan sikkert nu forstå, hvorfor jeg vil anbefale Calc Result og Commodore 64; det er den bedste hjælp man kan man kan tænke sig, ligelydigt hvilken form for tal og kalkuler man arbejder med.

skovbon fashion a/s
grimhøjvej 1 . dk 8220 brabrand

Kontakt din COMMODORE – forhandler idag!

Commodore
Bjerrevej 67
8700 Horsens
Tel 05-64 11 55

Han kan også vise dig andre kvalitetsprodukter fra Handic Software.

Alt, lige fra VIC-Switch flerbrugersystem til undervisningsprogrammer, programmeringsværktøj, kommunikation, videotex, databaser og spil.

handic software ab

HVEM HJÆLPER DIG ET SKRIDT VIDERE?

MATRIX PRINTER MPS 801

MPS 801 er til dig, der vil have en virkelig professionel printer. 50 karakterer pr. sek. og 80 karakterer pr. linie. Ideel til udskrivning af fakturaer, checks, breve o. lign.

Der er fuldt alfanumerisk tegnsæt samt Commodoregrafik. Den kan også skrive med negativ skrift og lave forstørrede karakterer. Desuden kan du designe tegn og bomærker med Commodore MPS 801.

MATRIX PRINTER MPS 802

Ideel til tekstbehandling. Fordi dens tegnopløsning er så høj. Velegnet til administrative systemer. Fordi den er beundringsværdig hurtig med 80 karakterer pr. sek. Fordi den ubesværet laver 20, 40 eller 80 tegn pr. linie. Bidirektional printning. Naturligvis med den linieafstand du programmerer den til. Printer MPS 802 arbejder med papirformater op til A4.

Commodore Computer er den største leverandør af hjemmecomputere i Danmark - og det forpligter.

Vore perifere enheder lever fuldt og helt op til vore computeres standard.

Med vore printere, plottere, diskettestationer og datasette udvider du dine muligheder såvel på det administrative område som på området for udvikling af egne programmer.

Commodore

Fordi fremtiden forlængst er begyndt.

JWT 2847 B

This was brought to you

from the archives of

http://retro-commodore.eu