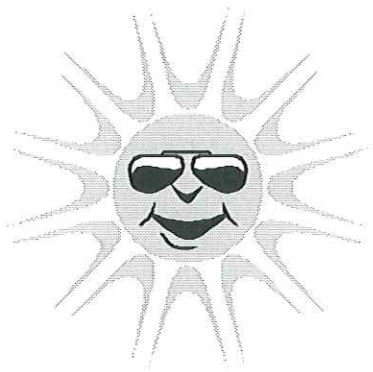


relik#0006

TIDNINGEN FÖR 8-BITARS DATORER OCH DERAS ANVÄNDARE

Relik
Juli 1997
Nummer 6
20 SEK



WWW RECENSIONER 4*av Jens Örtenholm*

Vi tar en gemensam titt på tio stycken åtta-bitars webbsiter i varje nummer med början här. Vi tittar bland annat på CaBooM!, Censor Design, CCCUG, Alter och MJK's Homepage.

LÄR DIG C64 ASSEMBLER DEL 6 6*av Jens Örtenholm*

I denna sjätte del av Relik's artikelserie om C64 assembler så händer det mycket. Först och främst presenteras lösningen på förra månads övning, en interrupt-driven scrollrutin, varefter vi tar en titt på hur en rasterrutin samt en FLD-rutin (Flexible Line Distance) fungerar.

LÖDPROJEKT: X1541-KABEL 11*av Jens Örtenholm*

Har du en PC hemma och vill kunna överföra filer mellan den och din C64 utan att använda nollmodem? En X1541-kabel är lösningen! Med denne kan du ansluta din 1541 eller 1571 direkt till PC:n via parallellporten, och kopiera filer med t.ex. StarCommander. Så här bygger du kablenn!

RELIK TESTAR: ACE v16 15*av Jens Örtenholm*

ACE är ett operativsystem för C64 och C128 som är starkt inspirerat från UNIX. I denna artikel tar vi en titt på ACE och de kommandon och program som följer med systemet, samt tittar lite på vilken hårdvara systemet stöder idag.

INTRODUKTION TILL GEOS 17*av Jens Örtenholm*

Många av Relik's läsare har uttryckt önskemål om en artikelserie kring GEOS. Nu är den här - denna första artikel är en lätt introduktion som förklarar för nybörjaren vad GEOS är och vad man kan använda det till, samt går igenom vad ni kan vänta er i denna artikelserie i framtiden.

NOTEMAKERS 19*av Jens Örtenholm*

Du har säkert sett en notemaker någon gång, ett enkelt program som man kan använda för att skriva små brev man sparar på diskett i form av körbara filer. I denna artikel tar vi en titt på Facenoter samt hur man kan använda den för att skapa s.k. notes.

LEDARE 3**INTERNET-LÄNKAR 12****PD-DISKETTER 13****BBS-LISTA 13****NYHETER FRÅN CMD 14**

Relik är en oberoende idéell tidning om åtta-bitars datorer och händelserna kring dem. Allt överskott från tidningen går till förbättringar av densamme.

Redaktör och ansvarig utgivare:
Jens Örtenholm

Adresser:

Relik
c/o Örtenholm
Råbystigen 9
197 31 Bro

Telefon:
08-58249088

Mobiltelefon:
070-5927747

email:
jens.ortenholm@pc-programs.se

www:
<http://home3.swipnet.se/~w-38277/>

Postgiro:
8356653-9

Betalningsmottagare:
"Jens Örtenholm 5365-0396823"

Prenumerera på Relik! 6 nummer för 100 kronor eller 12 nummer för 175 kronor, inklusive porto inom Sverige.
Lösnummer kostar 20 SEK per styck inklusive porto inom Sverige.

Betala in pengarna på postgirot ovan och ange tydligt namn, adress, telefonnummer och vad du vill beställa. I händelse av prenumeration, ange från vilket nummer du vill att den ska börja.

**Jens
Örtenholm**

Hej alla gamla och nya läsare!
Först och främst vill jag påminna er alla om att Relik flyttade för andra gången ganska nyligen. Än så länge är det i princip ingen som har använt sig utav den nya adressen, och jag tänkte att det därför kunde vara på sin plats med en liten repetition. Dessutom har jag nu egen telefon dit ni kan ringa om ni har problem med någonting rörande Relik, kontakta oss vid behov på följande adresser:

Relik
c/o Örtenholm
Råbystigen 9
197 31 Bro

email: jens.ortenholm@pc-programs.se
<http://home3.swipnet.se/~w-38277/>

Telefon: 08-58249088
Mobiltelefon: 070-5927747

Som du säkert redan märkt så har Relik genomgått betydande förändringar sen senast. Precis som jag lovade i förra ledaren så är det här numret 22 sidor istället för de gamla 16 sidorna, och så kommer det att förbli om det inte visar sig att ekonomin går kraftigt bakåt på grund av detta. Layouten är helt ny, jag bestämde mig för att göra om den helt från scratch, och jag hoppas att tidningen numera ser bättre ut än tidigare. Jag har också arbetat mycket på att bredda innehållet i tidningen, tidigare nummer har kanske varit lite väl inriktade på programmering (vilket i och för sig inte är så konstigt eftersom det är just programmering jag tillbringar större delen av min tid med). I detta nummer finns alltså nya artikelserier och typer av artiklar, bland annat startar en artikelserie om GEOS (i och för sig bara med en enkel introduktion, men ändå), operativsystemet ACE v16 testas, ett lödprojekt finns med för hur man bygger en X1541-kabel och jag recenserar ett par web-siter som är inriktade på åtta-bitars datorer. Samtidigt fortsätter den gamla artikelserien "Lär dig

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

C64 Assembler" med ytterligare ett avsnitt, och de som följer serien ser snabbt att den blir mer och mer inriktad på demo- samt spelprogrammering. Faktum är att vi kommer titta på en del nyttoprogrammering i framtiden, men det är en bit till dess.

Just det här numret av tidningen har varit extra jobbigt att framställa. Så här i sommartider har inte särskilt många skribenter haft någon tid över för Relik, vilket jag hyser full respekt för, men det resulterade i att jag blev tvungen att skriva ihop hela tidningen själv, från början till slut.

Relik söker med andra ord fler skribenter inom alla områden, gärna för andra åtta-bitars datorer också (speciellt artiklar om Spectrum är intressant). Skicka in prov på era alster eller färdiga artiklar till oss så hör vi av oss så fort vi kan! Ju mer du kan tänka dig att skriva, desto bättre

Så vart är Relik på väg nu? Oavsett om jag får hjälp från andra skribenter eller ej så tänker jag hålla Relik vid liv. Åtta-bitars datorerna är värda det tycker jag, dessutom lär jag mig en hel del under arbetet med tidningen, samt håller min maskinskrivning i trim. Jag tänker gradvis förbättra tidningen på alla områden från det här numret och framåt, allt från saker som layout till tidningens bredd, artiklarnas innehåll samt utseendet på hemsidan – you name it. Så länge det finns flera läsare kvar så kommer jag hålla kvar Relik.

Det vore givetvis hemskt roligt om ni läsare ville höra av er med era synpunkter på tidningen. Skriv brev eller email, ring, vad ni vill! Utan er skulle det inte finnas någon tidning, så det vore kul att veta vad ni vill läsa om.

Tills nästa gång, ha det så bra och sköt om er!

Jens Örtenholm
Redaktör

Jens
Örtenholm

I varje nummer av Relik har vi nu haft med en lista med Internet-länkar, där man kan hitta information om åtta-bitars datorer på Internet. Då det har blivit dags att validera alla dessa länkar, alltså gå igenom dem och se att de fortfarande fungerar som de skall och inte har flyttat till någon annan server eller liknande, så ska jag samtidigt passa på att utvärdera dem en smula och skriva ett par ord om vad du kan hitta på varje site. Med början från det här numret så tar vi 10 åtta-bitars siter per nummer, och jag kommer också betygsätta dem med ett procentuellt värde från 0% (inte bra) till 100% (mycket bra). Dessa resultat kommer du att hitta i såväl länklistan som på Reliks hemsida.

The C64 Fanatics Webpage

<http://huizen.dds.nl/~skynetw/frames.htm>

Här finns det mycket, och det handlar om C64. Allt från fler länkar till filer att hämta hem, olika typer av artiklar samt massor av annan information. Vad det stupar på är just att det är så mycket, och det är inte så väl organiserat i de fyra frames som finns på sidan. Dessutom laddas olika typer av grafik in om och om igen, vilket gör att sidan aldrig blir "färdigladdad", utan överraskar en med fartsänkningar sådär var tjugonde sekund. Grafiken på sidan är helt okej, inte det vackraste jag sett men heller inte direkt ful. Texten kunde dock varit snyggare presenterad, speciellt tabellerna. Innehållet är det däremot mycket intressant, och det är ju ändå viktigast.

60%

(Bra innehåll men rörigt!)

Alter

<http://www.geocities.com/SiliconValley/6645/>

Alter är en demogrupp på C64an där bland annat jag varit medlem och Thom som var redaktör för Åtta Bitar är medlem. I och med att jag bevisligen känner gänget så kan jag ju misstänkas för att inte vara riktigt objektiv, och det är därför jag informerar er om detta – ni är nu medvetna om att mitt omdöme kan vara en smula vinklat. Nå, nu till själva sidan. Denna sida har egentligen inget större värde för den genomsnittlige C64-användaren, den är

mer för dig som är intresserad av demos och då speciellt gruppen Alter. Du hittar här information om medlemmarna, de demos som gruppen tillverkat och lite bilder från olika partys samt gruppens historia. Innehållet är med andra ord lite tunt. Grafiken är inte heller världens bästa, men också här är den helt okej, och den ger helt klart C64-känsla. Texten är det däremot bättre med på denna sida, Nomolas och Wiggen som har gjort sidan har skapat en enhetlig design med rubriker och lättläst text. Sidan bygger på frames och är dessutom väldigt lätt att navigera, vilket också ger pluspoäng.

50%

(Snygg, lätt att hitta i men väldigt specialiserad information)

Byterapers Inc.

<http://www.iwn.fi/byterapers/>

Även Byterapers Inc. är en demogrupp, en av de större dessutom. Det första som väcker mitt gillande är att sidan är mer anpassad för just C64-användare, den är med andra ord inte byggd på frames och är lätt att hantera i textbaserade web-läsare. Designen är enkel och snygg utan att vara "flashig", inget avancerat här inte. Du kan läsa om medlemmarna och deras mörka hemligheter, om Byterapers historia (fast egentligen inte), cyparties och annat smått och gott. Självfallet finns även ett index över deras produktioner där du kan hitta länkar till filerna på Internet och ta hem dem samt prova dem själv.

60%

(Bra för C64-surfare, ser bra ut, mycket specialiserad information med lite allmänt)

MJK's Commodore 64 and LCD Page

<http://www.student.informatik.thdarmstadt.de/~mjk/c64.html>

Det här är en mycket användbar sida! Både

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

grafik och uppbyggnad är väldigt enkel, så det är ingen visuell fröjd av större mått, men innehållet håller mycket hög kvalitet! Sidan är uppdelad i tre delar varav två är värda guld. Den första delen är nämligen ett slags telefonbok där du kan hitta bland annat email-adresser och hemsidor till många av de människorna som varit aktiva på demoscenen. Den andra delen är ett slags översikt över Commodore-hårdvara för den som tänkt skaffa sig en C64, inte särskilt märkvärdigt men säkert nyttigt för nybörjaren. Tredje delen är vad alla som haft ett hårdvaruproblem alltid drömt om, nämligen en informationsdel om hur du håller din C64-utrustning i trim, inklusive kretsscheman.

80%

(Mycket bra innehåll, kunde dock ha sett bättre ut)

CaBooM!

<http://www.jbrain.com/caboom/>

CaBooM! är egentligen ingenting man recenserar, nästan alla vet vad det är och nästan alla avgudar den.

CaBooM! är en del av Jim Brain's ansträngningar för ett bättre Internet för åttabitars-användare, nämligen en sökmotor där du kan söka bland åtta-bitars länkar i massor! Letar du efter någonting som har med åtta-bitars datorer att göra på Internet, börja här.

90%

(Inte världens vackraste men outhärlig!)

Censor Design

<http://www.censor.net/>

Säg den demointresserade som inte känner till Censor Design, säkert en av Sveriges mest kända demogrupper genom tiderna. Vem kommer inte ihåg Wonderland-serien till exempel? Här hittar du demos att tanka hem såväl som bilder att titta på och Censor's diskmag Shock. Censor har även gjort en del till spelkonsoler, vilket du hittar här (ja, faktiskt demos). Lite fotografier, länkar och en gästbok finns också att beskåda. Som du säkert förstått så handlar det mesta på den här sidan om just Censor Design, och som alla demogrupper hemsidor så har den ett begränsat intresseområde. Det mesta ser ganska bra ut, om än lite tråkig bakgrund. Den största nackdelen är att det gick slött att titta på sidorna.

65%

(Intressant, mycket filer av intresse)

CCCUG Homepage

<http://www.slonet.org/~rtrissel/>

CCCUG står för Central Coast Commodore Users

Group, och det är givetvis en förening som specialiserat sig på Commodore. Deras hemsida är väldigt Commodore-vänlig – och väldigt tråkig. Dessutom innehåller den inte särskilt mycket överhuvudtaget, bara två sidor med länkar samt en samling med newsletters som de ger ut, som faktiskt inte är särskilt intressanta (det handlar mest om föreningen, vilket kan vara förståeligt).

5%

(Ett riktigt bottennapp!)

CMD Homepage

<http://www.cmdweb.com/index.phtml>

Creative Micro Designs är nog det enda företag som fortfarande producerar massor av tillbehör till C64, däribland den välkända SuperCPU:n. Detta är deras hemsida, väl uppbyggd med hyfsad grafik och en hel del intressant om man är C64- eller C128-ägare och har pengar på fickan. Här hittar du även hemsidor för Commodore World, en C64-tidning som CMD producerar.

65%

(Intressanta produkter, annars är det mest Commodore World som är någonting att hänga i gran)

Chris' Commodore Page

<http://www.cybermail.net/~cgi/>

Detta är en privat, allmän sida om Commodore-datorer. Sidan har en hyfsad uppbyggnad, inte alltför rörig utan faktiskt riktigt småmysig. Den är uppbyggd med frames, men en textversion finns tillgänglig. Runt hälften av länkarna fungerade inte dock, så den verkar lite misskött vilket ger en stor bock i protokollet. Informationen som fanns var mest länkar, möjligheten att söka med sökmotor, information om en Commodore WWW-ring och ett par enkla program att tanka hem.

15%

(Tunt med innehåll)

Commodore 8-bit Server

<http://www.hut.fi/Misc/cbm/>

När jag först laddade in den här sidan i min web-läsare så blev jag förtjust – det gick ju riktigt fort! Sidan är mestadels textbaserad, men ser riktigt bra ut med välplacerade Commodore-logos. Det stuper tyvärr på innehållet, för även om sidan innehåller lite eget material så är det mesta länkar till andra sidor.

20%

(En del bra, men mycket länkar. Laddar snabbt!)

Jens
Örtenholm

Hej och välkommen till sjätte delen av artikelserien "Lär dig C64 Assembler". Idag ska vi titta på mycket nytt, bland annat hur man kan åstadkomma en rasterrutin (ni vet, en sån där randig rullgardin på skärmen) samt en FLD-rutin (FLD = Flexible Line Distance). Först av allt ska vi ta en titt på hur förra numrets övning skulle kunna ha lösts, nämligen en kombination mellan rasterinterruptet från nummer 5 och scrollrutinen från nummer 4. Koden finns här nedan.

```

*      = $1000

lda   #$00
sta   $d020
sta   $d021
sei
lda   #$7f
sta   $dc0d
lda   #$01
sta   $d01a
sta   $d019
lda   #$32
sta   $d012
lda   #$1b
sta   $d011
lda   #<rutin
sta   $0314
lda   #>rutin
sta   $0315
cli
jmp   slut

slut   jmp   slut

rutin
lda   #$01
sta   $d019

varde
lda   #$07
sta   $d016
sec
lda   varde+1
sbc   #$01
bcc   flytt
sta   varde+1
lda   #$3a
stopp cmp   $d012
bne   stopp
lda   #$08
sta   $d016

```

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

```

jmp   $ea7e

flytt  and   #$07
      sta   varde+1
      lda   #$3a
stopp2 cmp   $d012
      bne   stopp2
      lda   #$08
      sta   $d016
      ldx   #$00
flyttkn lda   $0401,x
      sta   $0400,x
      inc
      cpx   #$28
      bne   flyttkn
textpek lda   text
      cmp   #$00
      bne   ejnoll
      lda   #<text
      sta   textpek+1
      lda   #>text
      sta   textpek+2
      lda   text
ejnoll sta   $0427
      inc   textpek+1
      lda   textpek+1
      bne   hoppover
      inc   textpek+2
hoppover
      jmp   $ea7e

text .text "detta är min"
     .text " scrolltext "
     .byte $00

```

Egentligen tror jag inte jag behöver säga så mycket om rutinen, om du jämför scrollrutinen samt rasterinterruptet från de tidigare numren med denna så ser du lätt hur den är sammanbyggd. Dock kanske jag ska nämna att jag glömde en sak i förra numrets rasterinterrupt, nämligen instruktionerna:

```
lda   #$1b
sta   $d011
```

Utan dessa så fungerar inte rasterinterruptet alla gånger. Varför? Jo, du kommer säkert ihåg att vissa rasterrader upprepar sig i register \$d012. För att se någon skillnad

LÄR DIG C64 ASSEMBLER DEL 6

mellan dem så används sista biten i register \$d011, vilket innebär att om den biten skulle råka vara ställd för de extra raderna precis när man sätter upp interruptet, så kommer det att inträffa på de extra raderna, och inte de vanliga. Alltså måste vi ställa in även denna bit, vilket även görs i den nya koden.

RASTER

Rasterrutin är det enda namn jag har hört på den här sortens grafisk effekt, så i brist på bättre så får den heta det här med. Vad som åsyftas är givetvis de "trasmattor" med färgade ränder som går från vänstra sidan av skärmen (i ramen) till den högra (också där i ramen). Dessa var mer populära i demos på 1980-talet, men att programmera dessa är en ganska nyttig övning i timing, så jag tänkte att vi skulle lära oss hur det går till.

I och med att rutinen har ordet raster i namnet så har du säkert redan listat ut att rasterstrålen är inblandad på något sätt, och mycket riktigt så är dess position avgörande för att denna effekt ska fungera. Den går helt enkelt ut på att varje rasterrad ändra ramens och bakgrundens färger på adresserna \$D020 och \$D021 så att man har en individuell färg på varje rad. I teorin är detta enkelt, men för att det ska bli snyggt så måste man till att börja med ändra färgen exakt vid rätt tillfälle, alltså inte bara på rätt rad utan också på rätt position på raden. Sedan måste man vänta en exakt tidsperiod innan man ändrar färgen igen. Låt oss ta det hela kodrad för kodrad.

Vår första uppgift är givetvis att sätta upp ett rasterinterrupt på den rad vi vill att vår effekt ska börja. Eftersom vi gjorde detta i förra avsnittet så förutsätter jag att du vet hur man gör, så jag hoppar över den biten (en fullständig listning finns med litet senare i den här artikeln). Det som är intressant är själva interruptet. Vår första uppgift (som inte har med interruptet att göra, värdet till \$D019 hoppar vi över) är att ta oss till exakt rätt position för att ändra färgen för första gången.

Men stopp ett tag. Hur fungerar egentligen det här med att tajma sig till rätt plats på en rasterrad? Hur vet man när man är framme? Varje instruktion som processorn utför tar en viss tid som man mäter i cykler. I "The C64 Programmers Reference Guide" (även känd som Bibeln bland beroende C64-programmerare) kan man slå upp varje instruktion för sig och läsa hur många cykler det tar för processorn att exekvera den. Under den tid som det tar för processorn att exekvera en cykel så hinner rasterstrålen förflytta sig en viss sträcka, och genom att räkna ut (eller mer verklighetstroget: gissa) hur många cykler man behöver vänta för att komma dit man vill så kan man lägga in instruktioner som fördröjer programmet tills rasterstrålen är på rätt plats. Ett

exempel på en instruktion man kan använda för det syftet är NOP (No Operation), en instruktion som inte gör någonting förutom att slösa bort 2 cykler. I detta program använder vi oss av små loopar som inte gör någonting för att slösa bort tid, rutinen börjar med:

```
ldy #$02 ; två varv
droj1    dey      ; tajmar in på
          ; första raden
bne droj1
```

Sedan ska vi gå in i en loop för att hämta varje färg i taget, ändra ramen och bakgrundsfärgen, fördröja och sedan fortsätta med det om och om igen. Första steget är att initiera loopen, hämta den första färgen och snabbt ändra ramen och bakgrundsfärgen (ramen först eftersom den kommer längst till vänster på skärmen). Koden för detta ser ut så här:

```
ldx #$00 ; rad nummer 0
nesta    lda color,x ; hämta färgen
          sta $d020 ; ändra
          ; ramfärgen
          sta $d021 ; och
          ; bakgrunds-
          ; färgen
```

Observera att vi använder X-registret för att hålla reda på vilken rad i ordningen vi använder oss av. På detta sätt så begränsar vi oss till 255 rader, men det får räcka den här gången (det är för övrigt en ganska stor rasterrutin). X innehåller alltså alltid numret på den rad vi håller på att behandla. Nå, låt oss fortsätta med fördröjningen.

```
ldy tid,x ; hämta
          ; fördröjning
          ; från
          ; tabellen
droj     dey      ; loopa tills
          ; vi har dröjt
          ; klart
bne droj
```

Men vänta nu.. Varför använder vi oss utav en tabell för fördröjningsvärdet (hur många varv vi ska göra i loopen)? Jo, problemet är lite mer komplicerat än jag riktigt har erkänt så här långt. Var åttonde rad så tar nämligen C-64an ett antal cykler för att rita upp den kommande teckenraden på skärmen, vilket gör att vi var åttonde gång ska vänta ett varv i loopen, alla andra gånger åtta varv i loopen. Tabellen kommer litet senare. Nu måste vi se till att loopen går runt som den skall.

```
inx
cpx #$18
```

LÄR DIG C64 ASSEMBLER DEL 6

```

        bne  nesta

```

Vi gör 24 varv i den här loopen, alltså 24 rasterrader. Det är bara att utvidga tabellerna och öka det här värdet om du vill ha fler rader, ända upp till 255 stycken. När vi är klara med detta så måste vi komma ihåg att vi har ändrat bakgrunden och ramfärgen, alltså bör vi ställa tillbaka dem till ursprunget.

```

        lda  #$00
        sta  $d020
        sta  $d021

```

Och så tabellerna:

```

        *   = $1500

```

```

color
        .byte  $01,$00,$03,$00,$05,$00
        .byte  $07,$00,$09,$00,$0b,$00
        .byte  $0d,$00,$0f,$00,$11,$00
        .byte  $13,$00,$15,$00,$17,$00

        *   = $1600

```

```

tid  .byte  $08,$08,$01,$08,$08,$08
      .byte  $08,$08,$08,$08,$01,$08
      .byte  $08,$08,$08,$08,$08,$08
      .byte  $01,$08,$08,$08,$08,$08

```

Men varför har vi satt tabellernas position i minnet vid \$1500 samt \$1600? Tja, förklaringen är ganska enkel – de är lättare att hålla reda på dem på det viset.

Om du försöker ändra programmet så att effekten uppkommer på en annan rasterrad och framåt, glöm inte att du måste ändra på tidtabellen!

Här följer den kompletta listningen:

```

        *   = $1000

        lda  #$00      ; skärmfärg
        sta  $d020     ; svart
        sta  $d021

        sei           ; sätt upp
        lda  #$7f      ; raster-
        sta  $dc0d     ; interrupt på
        lda  #$01      ; rad $38
        sta  $d01a
        sta  $d019
        lda  #$1b
        sta  $d011
        lda  #$38
        sta  $d012
        lda  #<irq
        sta  $0314

        lda  #>irq
        sta  $0315
        cli
        jmp  loop

loop
        irq
        lda  #$01
        sta  $d019
        ldy  #$02      ; tajma in
                        ; första raden
        droj1
        dey
        bne  droj1

        ldx  #$00      ; börja på rad
                        ; nummer 0
        lda  color,x   ; hämta färgen
        sta  $d020     ; ändra
                        ; ramfärg
        sta  $d021     ; och
                        ; bakgrundsfärg

        ldy  tid,x     ; hämta
                        ; fördröjnings-
                        ; värde
        droj
        dey            ; loopa tills
                        ; vi har dröjt
                        ; klart
        bne  droj

        inx           ; ta nästa rad
        cpx  #$18     ; är vi klara?
        bne  nesta    ; om inte, ta
                        ; ett varv
                        ; till

        lda  #$00      ; återställ
                        ; ram- och
                        ; bakgrunds-
                        ; färg
        sta  $d020
        sta  $d021

        jmp  $ea7e

        *   = $1500
        .
color
        .byte  $01,$00,$03,$00,$05,$00
        .byte  $07,$00,$09,$00,$0b,$00
        .byte  $0d,$00,$0f,$00,$11,$00
        .byte  $13,$00,$15,$00,$17,$00

        *   = $1600

```


LÄR DIG C64 ASSEMBLER DEL 6

```
tid .byte    $08,$08,$01,$08,$08,$08
    .byte    $08,$08,$08,$08,$01,$08
    .byte    $08,$08,$08,$08,$08,$08
    .byte    $01,$08,$08,$08,$08,$08
```

FLD

FLD står för Flexible Line Distance, och om du har spelat mycket spel på C64an eller alternativt tittat mycket på demos så har du nästan säkert sett denna effekt någon gång. Den används ofta till att rulla in en hel skärmbild på en gång i Y-led, att studsa med densamme eller som ett snyggt sätt att byta bild i en slideshow. Nog talat om den, du kommer säkert att känna igen resultatet när vi blir klara.

Tricket med den här effekten är faktiskt att lura VIC-chipet i C64an, som har hand om grafiken. Genom att lura VIC-chipet att vänta en viss tid med att börja rita ut innehållet på skärmen så kan man enkelt styra på vilken rad skärmbilden ska börja ritas beroende på hur lång tid man lurar VIC-chipet att låta bli. Hur går då detta till?

VIC jämför de 3 sista bitarna i \$D012 med \$D011, och varje gång alla bitarna är lika så ritas en ny teckenrad ut. Om bitarna däremot skulle vara olika så fortsätter VIC att rita ut föregående rad. Genom att lämna första raden på skärmen tom så behöver vi alltså bara se till att de 3 sista bitarna i \$D011 och \$D012 är olika – så länge vi gör detta så kommer VIC att rita ut just tomrummet på den första raden. När vi sedan tillåter dessa tre bitar att bli lika så kommer andra raden, där vår text eller grafik börjar, att ritas, och sedan fortsätter följande rader tills vi kommer till skärmens nedre kant.

Vårt första steg blir givetvis att sätta upp ett rasterinterrupt. Jag hoppar över detta just nu, men det finns givetvis med i den kompletta kodlistningen som kommer i slutet av den här artikeln. Vårt interrupt startar på rasterrad \$30, och då körs vår rutin. Den första uppgiften för vår rutin är att läsa av rasterstrålens position i \$D012. Det värdet adderar vi sedan ett till, vilket alltid förändrar bit 0:s värde. När det är gjort så isolerar vi de tre sista bitarna, och tänder sedan bit 6 för att lägga detta värde i \$D011. Anledningen till att vi tänder bit 6 är att sätta på teckenläge med ”utökade färger” (extended colors). Koden för allt detta blir som följer:

```
igen    lda    $d012    ; hämta
        ; raster-
        ; strålens
        ; position
        adc    #$01    ; lägg till
        ; ett
```

```
and    #$07    ; isolera de
        ; tre sista
        ; bitarna
ora    #$10    ; tänd bit 6
sta    $d011   ; lägg värdet
        ; i D011
```

Nästa steg är att upprepa denna rutin ifall vi inte har dröjt länge nog. Detta gör vi genom att jämföra värdet i \$D012 med hur långt vi vill nå. Vill vi alltså få skärmbilden att börja ritas på rasterrad #\$50 så väntar vi tills vi nått den raden. Eftersom vi vill använda den här rutinen på ett snyggt sätt så antar vi att det finns en tabell att hämta värden från, där man lämpligen lagrar en sinus mellan \$38 och \$F8. Du vet säkert hur en sinuskurva ser ut, och om du ser tabellens alla positioner som X-värden och värdet man lagrar där som Y-värden så ser du att som koordinater i Y-led så skulle de värdena representera en rörelse fram och tillbaka med mjuka inbromsningar. Om du inte har någon sinustabell för just den rörelse du vill ha så finns det mjukvara för att framställa sådana. Annars så duger det bra med den tabell som finns med här, om den passar dina syften förstås.

```
vart    lda    tabell
        cmp    $d012
        bne   igen
```

När detta är gjort så måste vi också se till att vi nästa gång tittar på ett annat ställe i tabellen, och det åstadkommer vi genom att öka lågbyten för adressen till tabell. Detta har vi gjort förut i vår scrollrutin, alltså vet du att det är minnesposition vart+1 som vi ska ändra på.

```
inc    vart+1
inc    vart+1
```

Vi ökar alltså värdet två gånger för att få lite fart på rutinen. Vill du ha en långsammare, litet mjukare rutin så kan du ta bort den ena ökningen, eventuellt göra den till en kommentar. Nå, detta var faktiskt allt. Ta en titt på den kompletta listningen nedan, som också innehåller den sinustabell du behöver för att pröva rutinen.

```
*    = $1000
lda    #$00
sta    $d020
sta    $d021
sei
lda    #$7f
sta    $dc0d
lda    #$01
sta    $d01a
sta    $d019
```

LÄR DIG C64 ASSEMBLER DEL 6

```

lda #$1b                .byte $45,$44,$43,$42,$41,$40
sta $d011              .byte $3f,$3e,$3d,$3d,$3c,$3b
lda #$30               .byte $39,$39,$39,$3a,$3a,$3a
sta $d012              .byte $3b,$3b,$3c,$3d,$3d,$3e
lda #<irq              .byte $3f,$40,$41,$42,$43,$44
sta $0314              .byte $45,$46,$48,$49,$4a,$4c
lda #>irq              .byte $4d,$4f,$50,$52,$53,$55
sta $0315              .byte $57,$58,$5a,$5c,$5e,$60
cli                    .byte $62,$64,$66,$68,$6a,$6c
loop                   .byte $6e,$70,$72,$74,$76,$79
jmp loop               .byte $7b,$7d,$7f,$82,$84,$86
                       .byte $89,$8b,$8d,$90,$92,$94
irq                    .byte $97,$99,$9b,$9e,$a0,$a3
                       .byte $a5,$a7,$aa,$ac,$ae,$b0
                       .byte $b3,$b5,$b7,$b9,$bc,$be
                       .byte $c0,$c2,$c4,$c6,$c8,$ca
igen                   .byte $cc,$ce,$d0,$d2,$d4,$d6
                       .byte $d8,$da,$db,$dd,$df,$e0
                       .byte $e2,$e3,$e5,$e6,$e7,$e9
                       .byte $ea,$eb,$ec,$ee,$ef,$f0
                       .byte $f1,$f2,$f2,$f3,$f4,$f5
                       .byte $f5,$f6,$f6,$f7,$f7,$f8
                       .byte $f8,$f8,$f8,$f8,$f8,$f8
                       .byte $f8,$f8,$f8,$f8,$f8,$f7
                       .byte $f7,$f6,$f6,$f5,$f5,$f4
                       .byte $f3,$f3,$f2,$f1,$f0,$ef
                       .byte $ee,$ed,$ec,$ea,$e9,$e8
                       .byte $e6,$e5,$e4,$e2,$e0,$df
                       .byte $dd,$dc,$da,$d8,$d6,$d5
                       .byte $d3,$d1,$cf,$cd,$cb,$c9
                       .byte $c7,$c5,$c3,$c0,$be,$bc
                       .byte $ba,$b8,$b5,$b3,$b1,$af
                       .byte $ac,$aa,$a8,$a5,$a3,$a1
                       .byte $9e,$9c,$9a,$99,$97

lda $d012              ; hämta
                       ; raster-
                       ; strålens
                       ; position
adc #$01               ; lägg till
                       ; ett
and #$07               ; isolera de
                       ; tre sista
                       ; bitarna
ora #$10               ; tänd bit 6
sta $d011              ; sätt värdet
                       ; till $d011

vart                   lda tabell ; hämta
                       ; sinusvärde
cmp $d012              ; har vi
                       ; väntat
                       ; tillräckligt?
bne igen              ; om inte, kör
                       ; hela rutinen
                       ; igen

inc vart+1             ; se till så
                       ; vi tar ett
                       ; annat
                       ; sinusvärde

inc vart+1

jmp $ea7e

* = $2000

tabell
.byte $97,$94,$91,$8f,$8d,$8a
.byte $88,$86,$83,$81,$7f,$7d
.byte $7a,$78,$76,$74,$72,$6f
.byte $6d,$6b,$69,$67,$65,$63
.byte $61,$5f,$5d,$5b,$5a,$58
.byte $56,$55,$53,$51,$50,$4e
.byte $4d,$4b,$4a,$49,$47,$46

```



Jens
Örtenholm

Internet är fullt av C64-program, program som oftast lagras i .D64-filer (diskimage-filer). Varje sådan fil innehåller en diskettsida, och för att använda dessa på din C64 så måste du föra över dem disketter med hjälp av speciella program. Enklaste sättet att göra detta är att använda sig utav StarCommander (det bästa överföringsprogrammet just nu), en X1541-kabel och en 1541 eller 1571 tillsammans med sin PC. Allt är frid och fröjd utom...vart får jag kabeln ifrån? Kan man bygga en själv kanske?

Det är just detta som den här artikeln handlar om, nämligen hur du bygger din egen X1541-kabel. Till detta behöver du: 1 st kabel, 1 st 6-pinnars DIN-hane, 1 st 25-pinnars PC-printer kontakt. Förmodligen kan det vara bra med lödkolv, lödtenn och eltejp också.

Bilderna här intill är översikt över kontaktorna. Kabeln är alltså så enkelt uppbyggd att det bara handlar om vilka pinnar i ena änden som ska vara anslutna till vilka pinnar i den andra änden. Anslut dem enligt följande tabell från vänster till höger, rad för rad.

Commodore Drive		PC Printer Port	
2	Gnd	18-25	Ground
3	Atn	1	Strobe
4	Clk	14	AutoFeed
5	Data	17	SelectIn
6	Reset	16	Init

Se bara till att anslutningarna på varje kontakt är skilda från varandra. Ett vanligt problem, speciellt för nybörjare, är när anslutningarna får kontakt med varandra och kortsluter.

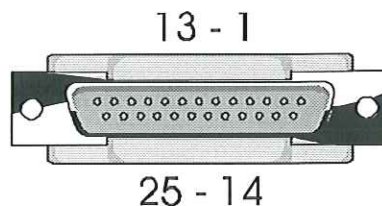
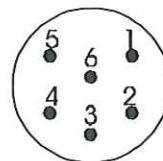
Om du vill ha en *riktig* X1541-kabel så ska du också kortsluta pinne 2 och 15 på den parallella kontakten. Detta används för automatisk detektion, men används inte av StarCommander (däremot bl.a. av vissa emulatorer etc.)

När allt detta är klart så kopplar du parallell-kontakten i skrivarporten på din PC och DIN-kontakten i seriell-porten på

din diskdrive. Sedan är du klar att köra! Om du får problem så är det dock ett par saker du kan undersöka innan du börjar pilla med kabeln. Det vanligaste felet är inställningen av någonting som kallas för interleave, vilket är något slags tajmningsvärde för att överföringen mellan diskdriven och PC:n ska gå i rätt hastighet och handskakningen ska fungera bra. På min Pentium 133 Mhz så har jag det värdet ställt på 20, har du en långsammare dator så ska du sätta ett mindre värde. Prova dig fram tills du får det att fungera.

Någonting annat du bör tänka på är att inte ha diskdriven påslagen när du drar igång PC:n, det leder till att massor av signaler skickas ut på printerporten nämligen. Vänta med att slå på diskdriven tills du har fått igång datorn och startar StarCommander.

Har du inte StarCommander men vill få tag på programmet? Sök på Internet, eller skicka en 3.5" HD-diskett till redaktionen samt svarsporto och kuvert så kopierar jag in programmet på disketten och skickar tillbaka den!



Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

INTERNET-LÄNKAR

64 The C64 Fanatics Web Page:

Alter:

Byterapers:

C64 Tuning and Repair Page:

CaBoom! - Your One-Stop Commodore Links Site:

Censor Design:

Central Coast Commodore Users Group:

Chris' Commodore Site:

CMD:

Commodore 8-bit Server:

Commodore Computer Cult Corner:

Commodore On-Line:

Commodore World Homepage:

Cosmic Style:

Dave's Commodore 64 Page:

DesTerm:

Fairlight:

Flash Inc:

Fridge:

In Medias Res:

Jim Brain's US Commodore 8-bit WWW Site:

Light:

Matthew Desmond's Homepage:

Megastyle Productions:

Motiv 8:

Noice:

Novaterm:

Omni / Revenge Homepage:

Padua:

Paramount:

Project 64:

Propaganda:

Reflex:

Relik:

RevolveR:s Homepage:

Science 451:

SID Homepage:

Talent:

The BlueNine C64 module collection:

The C64 Games Homepage:

The C64 Web:

The Electric Mayhem:

The Great C64 Cheats Compendium:

The Official I.N.C. Web Page:

The Ruling Company + Success:

Tips and Cheats to C64 games:

Triad:

Triangle 3532:

Xakk:

Åtta Bitar:

Craig Bruce's FTP Site:

DHP Demo Site:

Dyteca:

Erlangen FTP Site:

Funet:

Motiv8 FTP:

TPau:

TRC+Success:

ftp.funet.fi/pub

Newsgroups:

12

WWW:

<http://huizen.dds.nl/~skynetw/frames.htm>

<http://www.geocities.com/SiliconValley/6645/>

<http://www.iwn.fi/byterapers>

<http://www.student.informatik.th-darmstadt.de/~mjk/c64.html>

<http://www.jbrain.com/caboom/>

<http://www.censor.net/>

<http://www.slonet.org/~rtrissel/>

<http://www.cybermail.net/~cgi>

<http://www.cmdweb.com/cmd/>

<http://www.hut.fi/~msmakela/cbm/>

<http://www.ts.umu.se/~yak/cccc/>

<http://www.channel1.com/users/amazon/cbm/comm.htm>

<http://www.cmdweb.com/cwhome.html>

<http://www.tu-chemnitz.de/~dsc/c0smic/>

<http://www.csun.edu/~hbbuse08/c64.html>

<http://www.ionline.net/~mdesmond/desterm.html>

<http://www.ludd.luth.se/~watchman/fairlight/>

<http://www.abc.se/%7Em9656/flashinc/>

<http://stratus.esam.nwu.edu/~judd/fridge/>

<http://www.kuai.se/~zike/index.html>

<http://www.msen.com/~brain/cbmhome.html>

<http://www.hh.se/stud/d94fa/light.html>

[http://www.ionline.net/~mdesmond/ \(mannen bakom DesTerm\)](http://www.ionline.net/~mdesmond/)

<http://home.eunet.no/~scroll/msp/>

<http://www.Informatik.Uni-Oldenburg.DE/~grfrog/>

<http://www.ts.umu.se/~gnilk/>

<http://www.eskimo.com/~voyager/novaterm.html>

<http://flash.lakeheadu.ca/~jgvotour/index.html>

<http://www.padua.org/>

<http://ourworld.compuserve.com/homepages/midfit>

<http://ourworld.compuserve.com/homepages/pcgeek/proj64.htm>

<http://www.algonet.se/~motley/propa.htm>

<http://www.inf.tu-dresden.de/~jg10/reflex.htm>

<http://home3.swipnet.se/w-38277/>

<http://www.dra.nl/~vierstra/>

<http://www-und.ida.liu.se/~c89matlo/c64/science451.html>

<http://stud1.tuwien.ac.at/~e9426444/>

<http://www.talent.demon.co.uk/>

<http://inet.uni-c.dk/~bluenine/c64mod.html>

<http://www.nada.kth.se/~d93-alo/c64/>

<http://www.marathon.org/wilhelm/c64/index.htm>

<http://ltd.simplenet.com/c64/index.html>

<http://www.jyu.fi/%7EEnp/c64cheats.html>

<http://www.dhp.com/~shark/inc64/>

<http://www.geocities.com/SiliconValley/Pines/1016/>

http://www.student.nada.kth.se/~d93-alo/c64/tips_n_cheats.html

<http://www.df.lth.se/~triad/triad/>

<http://www.pip.dknet.dk/%7EPIP781/triangle/>

<http://www.engelholm.se/~fogelberg/xakk/>

<http://www.mds.mdh.se/%7EEdat95pkn/8bitar>

FTP:

<ftp.cnga.uwaterloo.ca/pub/cbm/>

<ftp.ftp.dhp.com/pub/c64/>

<ftp.ftp.cs.tu-berlin.de/pub/c64/>

<ftp.opheia.phil.uni-erlangen.de/pub/c64/>

<ftp.funet.fi/pub/cbm/>

<ftp.bbs.cc.uniud.it/pub/c64/>

<ftp.ftp.giga.or.at/pub/c64/>

<ftp.basement.replay.com>

<comp.sys.cbm>

PD-DISKETTER OCH BBS-LISTA

PD-disketterna kostar 30 kronor styck inklusive porto inom Sverige. Beställer du flera disketter vid samma tillfälle så kostar den första disketten 30 kronor och följande disketter 10 kronor styck. Betala in pengarna på postgirokonto 8356653-9 med betalningsmottagare Jens Örtenholm 5365-0396823 (siffrorna måste finnas med i fältet för betalningsmottagare, annars kommer inte pengarna fram!). Ange också tydligt vad du vill beställa samt namn, adress och telefonnummer (ifall någonting skulle vara oklart).

Också denna gång har jag satt ihop två PD-disketter, förhoppningsvis för sista gången då vi nu är ikapp igen. Den första disketten, Relik PD #5, innehåller först och främst ett demo från gruppen Reflex kallat "Drug Sixtyfour" som släpptes på partyt Assembly '96. Detta demo håller tyvärr inte Reflex' tidigare standard, och det är inte utan att man känner sig en smula besviken, men det är i alla fall värt att ta en titt på. Tyvärr har även Reflex hakat på marijuanatrenden på demoscenen, löven med den välkända formen finns med i detta demo, och det är därför inte lämpligt för känsliga och små barn.

Andra sidan på Relik PD #5 innehåller grafiken och musiken som deltog i tävlingarna på Assembly '96. Allt har inte toppklass, men det finns ett par godbitar man inte bör missa!

Den andra disketten, Relik PD #6, innehåller demot "Unsound Minds" vilket är tredje delen i serien "Follow the Sign" från gruppen Byterapers. Ett snyggt och bra demo som dock saknar fart och spänning samt är en smula för långt. Någon speciell design märker man inte

heller av i detta demo, men det innehåller många snygga effekter, som man tyvärr nästan får se lite för mycket av. Demot fyller båda sidorna på disketten, som därför inte innehåller någonting annat.

I kommande nummer av Relik kan det hända att PD-disketterna inte presenteras i en artikel som denna, utan i en artikel som samtidigt recenserar ett eller flera program på disketten. Håll därför ögonen öppna efter en diskettsymbol, vilken betyder att programmet som testas kan beställas på en av Relik's PD-disketter.

Har du skrivit ett eget program som du vill sprida? Skicka in det till oss! Så länge programmet verkligen är skrivet av dig och kan vara av intresse för andra så tar vi gärna med det på kommande PD-disketter.



LISTA PÅ BBS:ER MED INNEHÅLL FÖR ÅTTA-BITARS DATORER

<i>Namn:</i>	<i>Sysop:</i>	<i>Nummer:</i>	<i>Datorer:</i>
Antidote	Taper / Triad	042-76416	64/128
Fosie BBS	Natas	040-269767	64/128
The Studio	Jerry / Triad	0159-31991	64/128
<i>Utrustning: C64, 2400, 8 MB RamLink (10*1581)</i>			
Warez Aquarium	Sledge / FLT	08-371360	64/128
<i>Utrustning: C64, 2400, CMD HD 540 MB / JiffyDOS, 1581, C*Base 3.2</i>			

SuperCPU 128 Alfa-testas (7 juli 1997)

CMD har nu satt igång alfa-testningen av SuperCPU 128 prototypen. Resultaten från testet kommer att vara avgörande för huruvida designen av hårdvaran är tillräckligt bra för att projektet ska kunna fullföljas. Testningen av SuperCPU 128 i 64-läge är färdig, och testandet i 128-läget kommer snart att börja. Så snart 128-funktionerna verifierats så kommer layouten av kortet att uppdaterats, beta-varianter att produceras och beta-testningen av SuperCPU 128 att ta sin början.
Kanske är den snart här?

SuperRAM-kort (7 juli 1997)

CMD har börjat producera SuperRAM-kortet, ett RAM-expansionskort till SuperCPU-serien som kan öka minnet i din C64 med upp till 16 MB. Kortet är dock försenat medan en GEOS-drivrutin för kortet skrivs färdigt. Programutvecklare som vill ha tag på en version av kortet innan den släpps kan kontakta CMD:s tekniska support på telefonnummer (413) 525-0023.

CMD slutar med Boca-modem (7 juli 1997)

CMD kommer ej längre att sälja Boca-modem, och planerar att ersätta dessa modem i sitt sortiment med modem från Zoom Technology. Anledningen är att Boca ej längre stödjer Class 2-fax protokollet som krävs för att geoFAX ska fungera. Prisuppgifter och andra detaljer kring Zoom-modemen kommer inom kort.

CMD 1750 & 1750XL REU's (7 och 19 juli 1997)

Numera finns det två nya RAM-expansioner till C64:an, nämligen CMD 1750 och 1750XL REU's. Priserna på dessa enheter är just nu 99\$ för CMD 1750 och 139\$ för XL-varianten. Mängden minne på dem är 512Kb (CMD 1750) respektive 2 MB (CMD 1750XL).

SKRIBENTER SÖKES!

Relik har nu kommit ut i 6 nummer, och vi ger fortfarande inte upp. Det som orsakat alla våra förseningar är i princip bristen på skribenter - vi söker därför fler som vill arbeta med en tidning för 8-bitars datorer, och vi behöver artiklar inom alla områden!

Eftersom Relik är en helt idéell tidning så kan vi tyvärr inte betala ut någon ekonomisk ersättning, inkomsterna från försäljning och prenumeration täcker nätt och jämnt produktionskostnaderna i fråga om material och porto. Däremot så får varje skribent ett gratis-ex av det nummer hans eller hennes artikel är med i.

Skicka in prov på ditt skrivande nu, gärna en helt färdig artikel som är klar att tryckas i Relik nummer 7!
Kontakta oss på följande adress:

Relik
c/o Örtenholm
Råbystigen 9
197 31 Bro

Om det är någonting du undrar över, slå mig en signal.

Jens Örtenholm, 08-58249088

Jens
Örtenholm

När jag först började titta runt på Internet efter program till C64 och C128 så blev jag lite förvånad när jag stötte på denna kategori av mjukvara – operativsystem. Visst kände jag till GEOS, vilket jag dock såg mer som GUI (Graphical User Interface) än operativsystem, men jag kunde inte tänka mig att någon lagt ner tid på att utveckla ett operativsystem till C64/128. Givetvis hade jag fel.

ACE är just ett operativsystem utvecklat av Craig Bruce, starkt inspirerat av UNIX. Eftersom jag tyckte om ACE från första början (vilket dock kan ha med min vana vid UNIX att göra) så bestämde jag mig för att lägga det under luppen och skriva en artikel om det. Den här artikeln handlar om ACE version 16.

ACE är för både C64 och C128, och då talar jag inte om två olika versioner, programmet är detsamma förutom kärnan i operativsystemet där det finns en version för C64 och en för C128. Det är inte ens olika typer av applikationer beroende av vilken dator man använder, alla applikationer kan köras både på C64 och C128. ACE är dessutom multitaskande, det kan alltså köra flera processer samtidigt.

Systemet är lätt att starta. Man laddar helt enkelt in första filen på disketten, vid namn "ACE", och kör den. Programmet känner sedan av vilken dator man använder och laddar in lämplig kärna, drivrutiner och ett shell. Shell (betyder skal) är ett koncept från UNIX-världen vilket är miljön man använder för att kommunicera med operativsystemet, oftast via en enkel prompt. När skalet är inladdat så möts man av en prompt av typen [a:], som visar att skalet är redo att ta emot kommandon och att den diskettenhet som är aktiv är 8 (som döps om till a:, förmodligen inspirerat från MS-DOS). Man kan sedan köra något av de kommandon som är inbyggda i skalet eller ladda in ett program från någon enhet.

ACE stöder ett flertal olika enheter,

däribland ramdisk, 1541, 1571, 1581 och hårddisk. Dessutom kan man använda olika typer av minnesexpansioner, t.ex. REU, men stödet för hårdvara slutar inte där. Det finns även stöd för bland annat SuperCPU-64, Turbo232, Swiftlink och JiffyDOS från första början. I dokumentationen finns till och med ritningar på en kabel och anvisningar som man kan använda för att bygga ett mycket enkelt nätverk med två ACE-datorer!

Själva skalet stöder ett antal kommandon, varav några är bekanta från UNIX och några från DOS. Följande kommandon finns inbyggda:

echo	"Ekar" en text till skärmen.
clear	Rensar skärmen.
cls	Rensar skärmen.
dir	Rensar skärmen och visar innehållet på nuvarande enhet med en fil på varje rad.
d	Förkortning för dir.
ls	Det klassiska UNIX-kommandot för att visa vad som finns på nuvarande enhet. Filerna visas i kolumner.
cls	En kombination av cls och ls, rensar skärmen och kör sedan ls.
cd	Byt katalog eller enhet.
cat	Visar filers innehåll på skärmen (stdout).
exit	Avslutar skalet.
x	Förkortning för exit.
dos	Kör ett doskommando (alltså, ett kommando till diskettenheten, inte någon MS-DOS emulator inte).
@	Förkortning för dos.
path	Ställ in standard-sökvägen för de program du använder.
sh	Starta ytterligare ett skal.

När man använder sig utav kommandon eller försöker köra program i ACE så ska man vara uppmärksam på att ACE är likt UNIX i ytterligare ett avseende – det gör skillnad på stora och små bokstäver. Man

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

RELIK TESTAR: ACE v16

kan alltså inte köra ett kommando, exempelvis ls, genom att skriva det med stora bokstäver, det fungerar inte. Då CAPS-LOCK tangenten på min C128 råkade vara nedtryckt när jag startade ACE i C64-läge första gången så tog det en stund innan jag upptäckte detta (jo, som ni vet så används ju C128:s CAPS-LOCK för att tillåta svenska tecken i C64-läge, vilket gör att den oftast är nedtryckt).

Som du ser så finns kommandot cd, och det innebär mycket riktigt att man i ACE kan skapa en katalogstruktur på sina disketter. Den enda typen av enhet man inte kan göra detta med är på en RAM-disk, å andra sidan kan man skapa hur många RAM-diskar man vill (och som minnet räcker till för), vilket kompenserar detta.

För att kunna skapa en katalogstruktur så måste man ju ha kommandon för detta, vart finns de? Då kommer vi in på den andra kategorin av kommandon man kan använda, nämligen de som egentligen är vanliga program. Med ACE följer från början följande program med:

cp	Används för att kopiera filer.
rm	Remove, används för att radera filer.
mv	Move, används för att flytta eller byta namn på filer.
mkdir	Skapar kataloger.
rmdir	Tar bort kataloger.
wc	Räknar bl.a. antalet ord i en textfil.
grep	Ett mycket användbart program som från början kommer från UNIX-miljö. Programmet letar igenom en mängd filer som man anger med en filmask efter en söksträng som man anger. Man kan alltså t.ex. söka igenom filerna *.txt efter någon sträng, exempelvis "Relik".
tr	Används för att konvertera textfiler mellan olika teckenuppsättningar.
sort	Sorterar rader i en textfil.
wrap	Wordwrap.
date	Visar datum och tid.
forty	Byter till 40-kolumnsläge.
eighty	Byter till 80-kolumnsläge (i C64-läge finns emulation av detta med mjukvara).
read	Läser i en fil.
mem	Visar information om minnet, bl.a. hur mycket minne som är ledigt.
hello	Ett exempelprogram av typen "hello world".
window	Används för att ändra dimensionerna på fönstret.
acechr	Byter teckenuppsättning.
ketmat	Byter tangentbordsuppsättning.
head	Visar det antal linjer först i en fil som man anger.

jif	Visar Jiffy-tid.
mousetest	Testprogram för mus.
more	Visar textfiler i fullskärmsläge.
uencode	Kodar filer enligt UUCODE-standarden.
udecode	Avkodar uencode:ade filer.
bcode	Kodar filer, liknar uencode.
unbcode	Avkodar filer, liknar udecode.
crc32	Räknar ut CRC32 på en fil.
crc32a	Gör samma sak som crc32 fast enligt "gammal standard".
kar	Arkiveringsprogram, packar ihop filer.
unkar	Dearkiveringsprogram, packar upp ett arkiv.
vbm	Visar bitmapfiler.
vbmpr	Skriver ut bitmapfiler på skrivare.
as	Assembler för att utveckla ACE-program.
term	Terminalprogram.
fx	Filöverföringsprotokoll.
z	Enkel texteditor.

Efter att ha provat dessa kommandon så måste jag säga att de fungerar mycket bra. Tyvärr har jag inte haft möjlighet att testa terminalprogrammet samt filöverföringsprotokollet, men en god vän intygar att också dessa fungerar som de skall.

ACE är ett mycket intressant system. För den vanliga användaren med en C64 och en 1541 diskettstation är det kanske inte så mycket att hänga i granen, men med en maskin (C64 eller helst C128) utrustad med någon form av RAM-expansion och ett lagringsmedia med bättre kapacitet (exempelvis en hårddisk eller någon av CMD's diskettenheter) så blir det ett mycket användbart system. Den uppenbara nackdelen är att det inte finns någon uppsjö av mjukvara till detta system, så vill man använda det seriöst så måste man nästan kunna programmera och utveckla sina egna verktyg.

Om du är intresserad av att prova ACE så hittar du den på Jim Brain's FTP-site, adress <ftp.jbrain.com> i katalogen /pub/cbm/os/ace. Där finns ett antal arkiv av SFX-typ (som är kompatibelt med LHArc-formatet), samt en underkatalog som innehåller de filer du behöver för att göra en ACE-diskett utan tillgång till packningsprogrammet.

Om du inte har tillgång till FTP så kan jag trösta dig med att det är ganska troligt att ACE blir tillgängligt på någon kommande Relik PD-diskett.

Jens
Örtenholm

Länge har jag försökt få någon att skriva om detta ämne i Relik, men eftersom ingen verkar vilja göra det så bestämde jag mig för att göra det själv. Jag har ingen större erfarenhet av GEOS, men jag räknar med att jag kommer ha det efter ett par artiklar eller så.

Eftersom jag är nybörjare på GEOS och jag till att börja med också riktar mig till nybörjare så tänkte jag att vi först och främst skulle ta en titt på vad GEOS är och vad man kan använda det till, det är för övrigt detta som denna artikel handlar om. Därefter går vi igenom hela grunden i systemet och kikar på de möjligheter som finns, och när även detta ämne är behandlat så är det dags att gå in på det riktigt intressanta – recensioner av dels program till GEOS, dels hårdvara som kan göra ditt arbete i GEOS mer effektivt.

Men vad är då GEOS? Man skulle kunna säga att det är C64/128:ans motsvarighet till Windows på PC, fast mer ett operativsystem än t.ex. Windows 3.11. GEOS finns i två versioner – en för C64 och en för C128, varje version optimerad för att användas med respektive dator. Till skillnad från operativsystemet ACE som vi också tittar på i detta nummer av Relik så har GEOS dessutom olika kategorier av programvara, det är med andra ord olika program för C64 samt C128.

Det grafiska användargränssnittet (GUI:t) är det man först lägger märke till när man startar GEOS. Här är det inte kryptiska kommandon i textläge som gäller, utan fönster och ikoner att klicka på. Man använder en vanlig muspekare som man kan styra med mus eller joystick. Gränssnittet i GEOS är relativt välbyggt och lätt att förstå sig på, vilket är idén med ett grafiskt gränssnitt överhuvudtaget. En annan väsentlig fördel är att alla olika typer av mjukvara man använder på sitt system blir lättare att lära sig och använda med ett gemensamt gränssnitt – grundfunktioner i program som t.ex. ladda och spara finns på sina gamla välbekanta

platser och man känner sig som hemma redan första gången man startar ett nytt program. På så sätt kan man koncentrera sig på att lära sig använda programmet istället för att lära sig ett nytt gränssnitt varje gång. Den uppenbara nackdelen med ett grafiskt gränssnitt är just att det är grafiskt – grafiska system är oftast långsamma, speciellt gäller detta datorer som C64 och C128. Trots att GEOS inte är överdrivet slött så finns det ögonblick då man önskar att hastigheten hade varit en helt annan.

Med ett system som GEOS finns dessutom fler fördelar – man kan dela data och enheter på ett enkelt sätt. Istället för att varje program du använder ska ha egna programrutiner för att komma åt t.ex. minnet i en REU så sköter GEOS om den biten med sina drivrutiner. Programmen anropar bara GEOS och låter densamme tjäna som gränssnitt mellan programmet och hårdvaran. På samma sätt så underlättar det för all annan kringutrustning, t.ex. skrivare och liknande. Ett program skriver för GEOS behöver bara veta hur man skriver ut, sedan är det upp till GEOS att sköta resten, och förstås upp till användaren att skaffa en lämplig drivrutin för sin skrivare.

Datadelning är också mycket användbart. Att flytta någon form av data mellan två olika program är mycket enklare än att använda sig utav de många konverteringsprogram som ofta behövs för att flytta data mellan traditionella applikationer för C64 samt C128.

Förutom själva grunden, operativsystemet GEOS, så medföljer diverse programvara, däribland en ordbehandlare och ett ritprogram. Ordbehandlaren, geoWrite, är en äkta WYSIWYG-applikation (WYSIWYG = What You See Is What You Get), och man kan alltså enkelt göra en layout av sitt dokument på ett sätt som i alla fall jag inte är van vid från C64/128-miljö. Dessutom medföljer en stavningskontroll, geoSpell, som kan kolla stavningen i dina dokument. Ritprogrammet heter geoPaint

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

INTRODUKTION TILL GEOS

och är enkelt att använda. Från båda dessa program är det givetvis enkelt att t.ex. skriva ut dokumentet på en skrivare, med GEOS kan man till och med använda Postscript-skrivare, t.ex. HP LaserJet 4.

GEOS har funnits ett bra tag, nuvarande version är 2.0. Det finns en uppsjö av programvara, däribland programvara för kalkylark (geoCalc), för att skapa diagram (geoChart), för layout (geoPublish), för programmering (självfallet geoProgrammer samt geoBASIC för dig som inte programmerar assembler) – ja du kan till och med skaffa faxprogramvara för att sända och ta emot fax på din C64/128 (geoFAX). Om du är trött på det grafiska gränssnittet och längtar efter att få skriva lite kommandon igen så finns geoSHELL, ett CLI (Command Line Interface) som bör bota dina värsta mardrömmar om fönster och ikoner.

Detta är givetvis inte allt som finns, utan ett litet aplock. Från nästa nummer så ska vi börja titta på själva GEOS och hur man får ut det mesta av grunden i systemet. Därefter hoppas jag få möjlighet att göra tester på många utav de GEOS-program som finns, jag måste erkänna att jag är mest nyfiken på geoFAX, så det är nog ganska troligt att jag kommer försöka lägga vantarna på det programmet för ett test.

Till er som redan är erfarna GEOS-användare så har jag en liten tröst – jag kommer parallellt med dessa artiklar att recensera public domain och shareware-programvara för GEOS, så jag lämnar er inte helt ute i kulisserna. Troligen kommer jag dessutom börja sätta ihop någon form av GEOS PD-diskett också, men mer om allt detta i nästa nummer av Relik!

Du har väl inte glömt att beställa någon av Relik's PD-disketter? Följande disketter från tidigare nummer av tidningen är tillgängliga nu!

Relik PD #1

Mathematica / Reflex (demo)
Dawnfall / Oxyron (demo)
Turbo Assembler 7.0 / Alter (programmeringsverktyg)
TurboDOS Pack (snabbladdare för programmerare)
Mr Thomas / Vibrants (musiksamling)

Relik PD #2

Access Denied / Reflex (demo)
Radio Napalm / Reflex (demo)

Relik PD #3

Nyttoprogram:
Voodoo Noter
17xx-linker
80-columns
Alternative 1.1
Sinusmaker
Starion TextEditor
HandyFiler 64
Deditv7
AFLI-Convert
Dreamcopy v6
Gnylf 1.3
1541-1764 Copy
BAM-Zipcode

DOS1571or81
Blkcount
Scratcher
Starmenuload
DMC v4.0
Future Composer 4.0
Rob Hubbard Editor
Voicetracker
Rockmonitor
JCH Editor
Soft Tune Composer
Equaleditor
JCH Relocator

Relik PD #4

Mergezo Anyag 3 / Profik (demo)
Apache / Tobias Hultman (spel)
IFFL-Pack (laddningsrutiner för programmerare)
Zyron Music Collection 11 (musiksamling)

Varje diskett kostar 30 SEK inklusive porto inom Sverige. Beställer du flera disketter på en gång så kostar första disketten 30 SEK och följande disketter 10 SEK per styck. Betala in pengarna på postgirokonto 8356653-9 betalningsmottagare "Jens Örtenholm 5365-0396823" och ange tydligt namn, adress, telefonnummer (ifall någonting skulle vara oklart) samt vad du beställer.

Jens
Örtenholm

Vad är egentligen en notemaker? En notemaker är ett program som du kan skriva små brev i, och dessa brev sparar du som ett program på en diskett. Den person du ger disketten till kan sedan ladda in programmet och läsa ditt brev samtidigt som han eller hon lyssnar på en behaglig bit musik.

Jag är inte riktigt säker på notemakers ursprung, men jag kan mycket väl tänka mig att den härstammar från alla de swappers som skickat disketter med program (oftast crackade spel eller demos) fram och tillbaks mellan varandra. Om man då vill skicka med ett litet meddelande så är det jobbigt att använda penna och papper – varför göra det när man har en dator inom armlängds avstånd?

Denna artikel handlar om hur du som är nybörjare kan använda en notemaker för att skriva meddelanden till de du byter disketter med. Notemakern som använts kallas Facenoter, en av de bättre notemakers som finns. Själva teorin borde dock vara densamma, och oftast får man en liten hjälpskärm när man startar notemakern, så efter att ha läst denna artikel borde du klara av vilken vanlig notemaker som helst.

När man laddar in Facenoter och startar den så möts man av ett meddelande – tryck space för att läsa noten eller någon annan knapp för att redigera den. Om man är mottagaren av en note så trycker man space för att läsa brevet, om man å andra sidan vill skriva en egen note så trycker man på någon annan knapp. Om man gör detta så möts man av en enkel hjälpskärm där kommandona beskrivs.

Det första kommandot är E, Edit Pages, vilket betyder ungefär ”redigera sidor”. Med detta kommando skriver du själva texten. För att byta sida när en skärm är full trycker du Shift och +, för att gå tillbaka en sida trycker du Shift och -. Du kan givetvis rensa en skärm med Shift och

Clr/Home, och dessa enkla kommandon räcker för att skriva det du behöver. Tryck Run/Stop för att komma tillbaka till meny.

Det finns också lite mer avancerade funktioner i redigeringsläge. Bland annat så kan du låta delar av texten du skriver blinka rytmiskt, vilken typ av blink du vill använda (tre olika sorter samt inget blink) väljer du med knapparna F1, F3, F5 och F7. Du kan också välja att centrera eller vänster- samt högerjustera en rad eller sida. Detta gör du genom att kombinera följande tangenttryckningar:

C= (commodore-knappen) – Raden
Shift – Hela sidan

C – Centrera
L – Vänsterjustera
R – Högerjustera.

Exempel: genom att trycka C= och C så centrerar du alltså raden, och trycker du Shift R så högerjusterar du hela sidan.

I just den version av Facenoter som jag använde mig utav så fungerade inte vänsterjusteringen av en hel sida, istället centrerades den aktuella raden.

Du kan också radera eller skjuta in rader med följande kommandon:

C= och D – Radera rad.
C= och I – Skjut in en ny rad.

Musiken som spelas i noten väljer du från huvudmenyn genom att trycka M. Du kan här ladda in en bit musik från disk som dock måste ligga mellan \$1000 och \$2FFF i minnet. Det enda du behöver göra är att mata in filnamnet. Du kan eventuellt komma att behöva ändra parametrarna för musiken, vilket du kan göra med valet P i huvudmenyn. Här väljer du vilken adress notemakern ska hoppa till för att initiera musiken och vilken adress den ska hoppa till för att spela musiken.

Du kan också ladda in en egen

Jens Örtenholm är redaktör för Relik och kan nås via email på adress jens.ortenholm@pc-programs.se eller genom brev till redaktionen.

INTRODUKTION TILL GEOS

teckenuppsättning, från huvudmenyn använder du kommandot C. Den enda typ av teckenuppsättning du kan använda i denna notemaker är en vanlig 1x1-font, vilken du enkelt skapar med ett lämpligt program.

Med valet X, "Change FX" kan du ändra vilken typ av effekt du vill ska användas för att byta sida. Det finns fyra olika effekter att välja bland. Med effekter menar jag sådana saker som att texten t.ex. verkar försvinna punkt för punkt, vartefter nästa sida dyker upp på samma sätt.

D är ett menyval som jag saknat i en del notemakers, nämligen möjligheten att göra en directory på disketten, alltså se vad som finns på den. Det kan vara bra när man letar efter en bit musik, en teckenuppsättning eller vill veta om det finns plats på disketten för att spara en note.

Med kommandot R kan du ändra färgerna på de

rasterlinjer som finns på skärmen, inget avancerat här. Jag saknar dock möjligheten att se resultatet.

Det slutliga kommandot är S, Pack+Save note vilket använder en enkel packningsalgoritm för att noten inte ska ta så stor plats på disken, och sedan sparar den. Du får här skriva in ett filnamn, sedan flimrar skärmen lite grand medan noten packas. När den sedan är klar med packningen så får du en uppmaning om att sätta i en destinationsdiskett, när du har gjort detta så behöver du bara trycka på en knapp för att spara noten. Den tar från 50 blocks och uppåt, beroende på hur mycket text du skriver.

Hur får du då tag på en notemaker? Facenoter kommer att finnas tillgänglig på nästa Relik PD, som du kan beställa inom en månad. Annars finns det ett flertal att hitta på Internet, bl.a. på Funet's FTP. Ta en titt i listan för Internet-länkar och leta efter Facenoter.

INTRESSERAD AV PROGRAMMERING ELLER HTML?

En bok om HTML

av Jens Örtenholm

Denna bok går igenom grunderna i HTML och hur man bygger hemsidor för Internet eller ett Intranet. Det ingår dessutom en referensdel över HTML-standard 3.2, som listar alla märken och information om hur de används och vart man får använda dem. Du behöver inga förkunskaper för denna bok som börjar direkt från grunden. Ungefär 75 sidor i A4-format bundet med ringpärm så att den ska bli lätt att slå upp bredvid datorn.

80 SEK inklusive moms. Frakt tillkommer.

C++ från grunden

av Jens Örtenholm

Om du vill lära sig programmera på professionell nivå så är C++ det du bör lära dig idag. C++ är ett objektorienterat språk som bygger på C, och den här boken är skriven för dig som inte har någon tidigare programmeringsvana, men även du som har det kan lätt tillgodogöra dig innehållet. Boken går igenom saker som databehandling, funktioner, klasser och objekt, objektorientering, arv, operatoröverlagring, funktionsöverlagring och mycket annat med exempel. Ungefär 175 sidor i A4-format bundet med ringpärm så att den ska bli lätt att slå upp bredvid datorn.

120 SEK inklusive moms. Frakt tillkommer.

För beställning, kontakta PC-PROGRAMS på telefonnummer 08-58249122, eller skriv email till bjorn.ortenholm@pc-programs.se.

NÄSTA RELIK KOMMER I SLUTET AV AUGUSTI!



HAR DU ALLA RELIK?

Relik har nu kommit ut i sex nummer, och vi är här för att stanna! Om du vill komplettera din samling av Relik med exemplar av de nummer du inte hunnit med att införskaffa så går det bra att beställa i efterhand - vi har möjlighet att trycka upp dem exakt efter dina önskemål!

Dessa tidigare utgåvor är tillgängliga för beställning:

Relik #1

Internet och C64:an
1750 REU:n är död
Tio i topp fel på C64:an
Videotext på C64
NTSC Turbo
BBS Lista
Datorns matematik
Lär dig C64 Assembler del 1
Internet-länkar
Överför filer mellan C64 och PC

Relik #2

ROS-64 Del 1
Brotkasten CD
SuperCPU
B128
Internet-länkar
Lär dig C64 Assembler del 2
Internetuppkoppling
Listning: Division
BBS-Lista
Annonser

Relik #3

Programmera 17XX REU
Spelrecensioner
Färger med VIC-20
Internet-länkar
Lär dig C64 Assembler del 3
BBS-Lista

Relik #4

ROS-64 Del 2
BBS-Lista
Test: NovaTerm 9.6
Internet-länkar
Lär dig C64 Assembler del 4
Framtiden

Relik #5

Rädda dokumentationerna
ROS-64 Del 3
Lär dig C64 Assembler del 5
Internet-länkar
Pokes (spelfusk)

Beställ genom att sätta in pengarna, 20 SEK per nummer inklusive porto inom Sverige, på postgirokonto 8356653-9 Betalningsmottagare "Jens Örtenholm 5365-0396823". Ange även namn, adress, telefonnummer (ifall någonting skulle vara oklart) och vad du vill beställa, och tidningarna dimper ned i din brevlåda inom ungefär en vecka.