

Sinclair

Magazine

Nummer - 4 - august 1991. Copyright N.A.S.A.

ADVERTISING

DIRECTOR and MANAGER

Vi håper at dette var en tankeløst for deg og at du gleder deg til neste nummer.

Sinclair Magazine august 1991
Følgende er en del av innholdet i dette nummeret:

Innhold i dette nummeret av SM:

side	3	Redaktøren
	4	Informasjon
	5	Diverse
	5	Gold Card
	7	Perfection
	9	Rutiner i SB - del 3
	11	Mer om systemvariabler
	13	Liste over varer fra N.A.S.A.
	15	Konkurranse
	15	Resetknapp på Spectrum - revidert
	15	Z80 maskinkode
	19	Softwaretest
	20	Spectrum super sommersalg
	21	QLs diskhemmeligheter
	27	Football Director/Manager
	28	Rabatt på programmer

Redaktøren (Pål Monstad)

Selv om vi har sommer og ferietid, kommer en ny utgave av SM. Dette nummeret er preget av disse ferietider, men vi valgte likevel å gi ut et nummer. Vi i redaksjonen har fått en del tips angående forslag til innhold fra dere medlemmer, i form av beskjeder på baksiden av betalingsblanketten som vi sende ut forrige gang. Vi skal prøve å ta med det dere ønsker, men det er også opp til dere. Send oss bidrag. Mange er helt sikkert interessert i å vite hva nettopp DU bruker maskinen din til, hvilke programmer og hardware du bruker osv.

Det er særlig Spectrum-avdelingen som sliter med lite stoff. Dette har mange av Spectrum-medlemmene klaget på, forståelig nok. Men det ser ikke ut til at noen gjør noe for å bedre situasjonen heller, f.eks. i form av å sende oss bidrag!

For at klubben skal bli bedre ønsker vi å gjøre følgende:

- at alle svarer på det spørreskjemaet som er vedlagt dette nummeret av SM.
- vi har en såkalt on-line-help, det vil si at medlemmene kan ringe oss om akutte problemer som oppstår (angående Sinclairmaskinen, hardware/programmer mm.). Bruk denne, og vi vil da hjelpe så godt vi kan. Ikke sitt inne med dine vanskeligheter, la oss hjelpe hverandre slik at man ikke mister interessen for maskinen.
- at alle medlemmene sender oss et bidrag, og ikke tenker at det gjør sikkert noen andre (da kommer det nemlig ingen ting).
- hvis du kjenner noen som ikke er medlem i klubben, og i tillegg har en Sinclairmaskin, må du være snill å overtale personen til å bli medlem. (Vi har vervepremier).

Det som kan skje uten medlemmenes hjelp er følgende:

- vi må legge ned klubben fordi vi slipper opp for ideer og stoff.
- du kan miste muligheten til å få tak i programmer, hardware deler reparasjoner og alt det andre vi i dag tilbyr. Du kan selvsagt bestille dette fra utlandet, men det blir i alle tilfeller mer arbeid for deg, og i mange tilfeller også dyrere.

Vi håper at dette var en tankevekker for deg, og at du gjør noe med situasjonen!

Informasjon (Pål Monstad)

Alle henvendelser angående QL, THOR, N.A.S.A. og Sinclair Magazine rettes til:

Norwegian All Sinclair Association
Nerheim
5580 ØLEN

Telefon : 04 - 76 84 63
Postgiro: 0824 0432375

Alle henvendelser angående Spectrum og SAM rettes til:

Frode Tennebø
Gt. 3 nr. 4
6700 MÅLØY

Telefon : 057 - 50 33 8
Postgiro: 0802 3829801
Bankgiro: 3632.15.15201

Vi samarbeider for tiden med følgende klubber i utlandet:

Sinclair Freakeren
v/ Leif Mortensen
Bryggervangen 29
DK - 7120 VEJLE ØST
DANMARK

Svenska QL Gruppen
Toftaåsgatan 73
S - 471 47 VESTRA FRÖLUNDA
SVERIGE

Sinclair QL User Club E.v.
Co/ Franz Herrmann
Talstrasse 21
D - 5460 OCKENFELS
GERMANY

Bidrag mottas i form av filer (helst rene ASCII-filer) på diskett eller cartridge. Vi tar også imot utskrifter. Alle bidrag sendes til N.A.S.A. i Ølen.

Medlemskap i N.A.S.A. koster for tiden NOK 55.00 pr. halvår, eller NOK 100.00 pr. helår. Medlemskapet er uten forpliktelser, selv om vi ser at medlemmene bidrar med stoff, tips og ideer.

Norwegian All Sinclair Association (N.A.S.A.), Frode Tennebø eller Sinclair Magazine (SM) er helt uten ansvar for feil som måtte oppstå på ditt eller andres datautstyr som direkte eller indirekte følge av våre artikler.

Husk at både Tennebø og meg selv er tilgjengelige for spørsmål over både telefon og brev, selv om vi (i alle fall undertegnende) til tider kan være vanskelig å nå pr. telefon. Ikke gi opp, før eller siden er jeg å treffe, om det så er klokken 23.00 en søndag kveld.

Diverse (Pål Monstad)

Rettelser:

På side 24 i forrige nummer skrev vi at vi hadde fått en ny medlem. Navnet skal være Didrik Seip. Beklager at vi rotet, Didrik.

På side 10 presenterte vi en løsning på konkurransen. Det snek seg dessverre inn en feil i programlistningen. Linje 120 skal være:

```
120 IF (sum - n) / 10000 = INT ((sum - n) / 10000)
```

Nye medlemmer:

Roar Hove, Lier, er meldt inn i klubben. Han bor i Lier og har QL. QL har også Knut G. Walle som bor på Landås. Velkommen som medlemmer begge to!

Vinner av konkurranse:

Vinneren av vår "innbetalingskonkurranse" ble Bjørn Gamst, Brattvåg. Gratulerer! (Premien er sendt i posten). Ny konkurranse på side 15.

Gold Card (Pål Monstad)

Gold Card er navnet på det nye mirakelet som Miracle Systems Ltd lanserer i disse dager. Gold Card er et nytt kort som skal erstatte alle tidligere ekspansjonskort og interface til QLen. Selv deres tidligere mesterverk, Trumpcard, blir småting i forhold til dette vidunderet.

I størrelse er kortet mindre enn alle tidligere kort. Det får plass inne i selve QLen, og bare en liten kjøleribbe stikker ut. Enkelt er det også å montere, det er bare å stikke det inn på venstre siden, akkurat på samme måte som med andre interface.

De tekniske data som Miracle annonserer med er som følger:

- * 16 Mhz 68000 prosessor
- * 2 Megabytes RAM
- * Batteriklokke
- * Kompatibelt med Trumpcard

- * Kontrollerer 3 diskettstasjoner med opp til 3.2 Mbytes lagringskapasitet
- * Sub-directories (også på vanlige disketter og ramdisken)
- * Får plass inni QLen, uten lodding el.
- * To års garanti

Alt dette høres for meg helt utrolig ut. Tenk å ha to MB RAM og en maskin som er like rask, eller til dels raskere enn QL-emulatoren på Atari! I tillegg batteriklokke, det vil si at QLens klokke går selv om maskinen slås av.

Når det gjelder diskettstasjonen som kan kobles til dette vidunderet kan de vanlige brukes. Det vil si at du kan bruke din gamle diskettstasjon slik du gjorde før. Det nye er at du i tillegg kan bruke diskettstasjoner med større lagringskapasitet, såkalte HD-disketter (1.44 MB) eller de helt nye ED-diskettene (3.2 MB).

Disse nye diskettstasjonene kan bruke de vanlige 720K diskettene. Så for deg som bruker, vil det ikke oppstå noen problemer med gamle disketter. Du vil heller ikke få problemer med noen av programmene, fordi alt er som før bortsett fra blant annet hastigheten.

I og med at Gold Card er utstyrt med såkalte CMOS-chiper, blir ikke maskinen så varm som den kunne bli før. Dermed forsvinner også hovedårsaken til at QLen til tider kan krasje.

En annen fordel med Gold Card er det store minnet på to MBytes. Av dette har brukeren 1920 Kb til rådighet for programmer ol. De resterende 80 Kb er brukt til å legge QDOS i. QDOS blir kopiert fra ROMen over til det mye mer håndterlige RAM. Dette vil i praksis si at brukeren kan "poke" rundt i QDOS (selv om det kan krasje maskinen). Du kan forandre på diverse meldinger ol. (mest nyttig for de som virkelig kan kalle seg "hackere").

Så det kanskje aller viktigste, nemlig prisen. Hvis du fra før av har et interface, for eksempel Trumpcard eller expanderam, får du Gold Card billigere.

Her er en oversikt over prisene (export):

Uten innbytte	£330
Trumpcard 2	£200
Trumpcard 768K	£241
Trumpcard 256K	£263

Expanderam	£284
Disk Card	£284
RAM+ Disk interface	£284
Andre ekspansjonskort	£305

Hvis det er flere som går rundt med tanker om å kjøpe Gold Card, skal jeg prøve å oppnå rabatt hos Miracle. Det er ellers mulig å kontakte meg for mer informasjon.

Perfection (Pål Monstad)

Det var med spenning at jeg satte meg ned for å teste det så mye omtalte tekst-behandlingsprogrammet fra Digital Precision. I følge den som vanlig beskjedne reklamen fra Digital, skulle dette programmet være svært raskt, det skulle ha mange kommandoer, men samtidig være like enkelt å bruke som Quill. Det ville enkelt og greit "blow your socks off".

Det er ikke rart at jeg da hadde ganske store forventninger til dette programmet. Jeg kan først som sist si at jeg nok ikke ble skuffet, selv om det tok litt tid før jeg oppdaget alle finessene og mulighetene dette programmet har.

Vel, så over til selve programmet. Programdisketten leveres sammen med en 104 sider manual i standard DP utførelse, det vil si en meget god manual. Manualen har i oppbygning mange likhetstrekk med Editor, også den fra DP.

Etter at man har laget noen sikkerhetskopier med "backup"-programmet, startes programmet ved hjelp av "boot"-filen. Her får en valget mellom å starte Perfection, printerinstalleringsprogrammet eller et konfigureringsprogram. Velges ikke et av de to siste alternativene, startes Perfection automatisk etter tre sekunder.

Perfection multitasker, så det er nødvendig å trykke CTRL/C for å aktivere markøren. Skjermen skiller seg egentlig ikke så veldig fra den som finnes i Quill, men en oppdager raskt at her finnes mange flere kommandoer. Kommandoene finnes i et vindu på toppen av skjermen, akkurat som i Quill. F3 aktiviserer nye kommandovinduer, og F2 tar bort helt kommandovinduet, også det slik som i Quill.

F1 gir deg hjelp, uten å lese fra disketten, et fremskritt fra Quill. I hjelpeteksten kan du bruke de vanlige kommandoene, slik som f. eks. "search".

Alle kommandomulighetene står beskrevet, sammen med forklaring på bruk av piltaster, editering osv.

For å eksperimentere med en allerede ferdig tekst, kan en begynne med "readme"-filen som ligger på disketten. Man kan godt si at denne teksten er et slags opplæringsprogram for de viktigste kommandoene. Teksten gir deg beskjed om å plassere markøren på forskjellige steder, deretter kan du trykke de oppgitte tastene, og du vil se at margene stilles m.m.

Hvis du heller vil eksperimentere med en fil fra Quill, er det bare å bruke "load"-kommandoen. Både Quill-, Perfection- og vanlige ASCII-filer loades uten problemer. Dine gamle Quill-filer kan med andre ord brukes i Perfection uten noen form for "fiksfakseri". Det samme kan printer-filene dine som du bruker på Quill. Du trenger ikke lage nye filer, bruk de gamle og vær sikker på at det fungerer på Perfection, det gjør de!

Når det gjelder de mange kommandoene kan de aktiveres på to måter. Den ene måten er å bruke kommandomenyene, slik som i Quill. Denne måten er enkel å grei, du trenger ikke huske hvilken tast som aktiviserer hvilken kommando. Den andre muligheten er å bruke CTRL, SHIFT og ALT sammen med en annen tast. Da aktiviseres kommandoen direkte, uten omveien gjennom menyene. Denne måten er mye raskere, men den krever at du husker de forskjellige kombinasjonene. Det er denne metoden som er mest praktisk, men den går ikke jeg over til før jeg er litt mer kjent med programmet og dets muligheter.

"Search"-kommandoen er utrolig rask, det samme er "replace" og de forskjellige "block"- og "goto"-kommandoene. Ellers finnes de vanlige kommandoene slik som "margins", "justify", "tabs", "print" (selvsagt), "merge" og "export".

"Header" og "footer" er mye bedre enn i Quill. I Perfection er det mulig å kombinere alle skrifttypene og bruke opp til en hel side med "header" eller "footer", selv om det egentlig er lite praktisk med fullt så mye.

Du har muligheten til å dele skjermen i to, slik at dokumentet kan editeres i to forskjellige deler. Meget praktisk spør du meg. Praktisk er også muligheten til å se skjulte kontrollkoder. Du kan for eksempel load inn maskinkodeprogrammer og editere dem. (Du kan dessverre ikke lagre filen tilbake, da blir filen kaputt). Jeg prøvde å ta inn Perfection (!?!), noe som fungerte helt fint. De skjulte kontrollkodene finnes også i et vanlig dokument (som dette jeg skriver nå). Du vil da se mange merkelige tegn som forteller deg hvilken skrifttype som brukes og all mulig annen informasjon som finnes spredt rundt i dokumentet. Det

finnes også muligheter for å bruke grafikk i dokumentene. Dette har jeg enda ikke prøvd, så derfor skal jeg heller ikke omtale det. Det sammen gjelder bruk av Perfection som database.

Nyttig er også spellingchecker som leveres med en ordliste på helt 225.000 ord. Dette er rekord på QL, nummer to har "bare" 50.000 ord. Jeg har ikke hatt muligheten til å teste spellingchecheren enda, men det skal være mulig å lese ferdige filer, samt hele dokomenter. Det mest interessante er kanskje at det er mulig å få ordlisten til å korrigere deg etterhvert som du skriver. Så vidt jeg vet skal det også være mulig å benytte ø, æ og å i ordlisten, noe som gjør det mulig å putte inn horske ord. Kanskje er det mulig å bruke norske ordlister som er tilgjengelige på andre maskiner, f.eks. WordPerfect. (Dette vil jeg komme tilbake til en annen gang).

Nå lurar du sikkert på hva det skal være godt for å kjøpe Perfection til £79.95 (£124.95 "/ordliste) når du allerede har Quill. Til det er det bare å si at Quill sammenlignet med Perfection er som å sammenligne en Skoda med en Porsche. Bare tenk deg "block"-kommandoene på Quill..., eller skrolling..., for ikke å snakke om "erase"-kommandoen eller "import".

Nei, folkens, Perfection er å anbefale hvis du benytter deg av tekstbehandlig. Jo større dokumenter du pleier å skrive på Quill, jo større er behovet for dette vidunderet...

NB! Du trenger diskettstasjon og minimum 256K for å bruke Perfection.

Rutiner i SB - del 3 (Pål Monstad)

Denne gangen skal vi ta for oss en prosedyre som midtstiller tekst på skjermen. Du har sikker en eller annen gang opplevd at du kunne hatt behov for en liten rutine som skriver tekst midt på skjermen, f. eks. ved overskrifter ol.

Rutinen er avhengig av av å vite bredden på det vinduet (scr_ eller con_) du skal skrive teksten i, det vil si hvor mange karakterer som kan skrives på en linje. Det er teknisk mulig å lage en rutine som selv sjekker bredden på vinduet, men men da hadde vi måttet bruke en toolkit-kommando, som dessverre ikke alle medlemmene har. Derfor denne litt mer tungvinte metoden:

```
100 DEFine PROCedure SENTRER (kanal, x, y, bredde, rekke, tekst$)
110 LOCAl lengde, verdi
```

```

120 CSize #kanal, x, y
130 lengde = LEN(tekst$)
140 SElect ON x
150   =0 : verdi = (bredde * 6) / 2
160       lengde = lengde * 6
170   =1 : verdi = (bredde * 8) / 2
180       lengde = lengde * 8
190   =2 : verdi = (bredde * 12) / 2
200       lengde = lengde * 12
210   =3 : verdi = (bredde * 16) / 2
220       lengde = lengde * 16
230 END SElect
240 CURSOR #kanal, verdi - (lengde / 2), rekke
250 PRINT #kanal, tekst$
260 END DEFine

```

Prosedyren krever 6 parametre ved bruk. Du må angi kanalnummeret på det vinduet du vil skrive teksten til. Videre må du angi bokstavstørrelsen, som følger det vanlige mønsteret for "CSize"-kommandoen (0,0 - 3,1). Om du bruker vanlig høyde på bokstavene eller dobbel høyde, så har ikke det noe å si.

For å finne ut bredden på et vindu kan du bruke denne metoden: Skriv PRINT #kanal, FILL\$("o",tall), hvor kanal angir det vinduet du vil bruke, og tall er et passende tall mellom 1 og 85. Det er bare å prøve seg frem med forskjellige tall, helt til du finner det tallet som gjør at hele vindusbredden fylles med en linje med o'er. Det er dette tallet som angir bredden på vinduet, dvs. at du bruker det som fjerde parameter i kommandoen.

Til slutt skal du angi hvilken rekke teksten skal skrives i, og hvilken tekst som skal midtstilles. Hvis du ikke skjønnte alt dette, kan du prøve noen av de eksemplene som følger. Først lager vi et vindu som dekker hele skjermen:

```
WINDOW 512, 256, 0, 0 : PAPER 2: CLS
```

Dette vinduet har kanal=1, fordi vi ikke har spesifisert noen annen kanal (default). Kanal-parameteren er da 1. Siden vi laget et vindu som strekker seg over hele skjermen blir bredden 85, fordi det er plass til 85 tegn tvers over skjermen. Du kan bare prøve med: PRINT FILL\$("o",85). Dette gjelder bare ved bokstavstørrelse 0,0, hadde du brukt f.eks. CSize 2,0, hadde det bare vært plass til 42 tegn.

Nå har vi altså funnet bredden og kanalen, bestemt oss for bokstavstørrelsen, og da er det bare igjen rekke-nummeret og tekst.

Hvis vi vil skrive "Dette er en test på midtstilling" noenlunde midt på skjermen (i høyde fra bunn til topp), kan det passe med rekke=110.

Da kan vi føye til en linje i programmet vårt:

```
300 SENTRER 1, 0, 0, 85, 110, "Dette er en test på midtstilling"
```

Denne kommandoen, som du kan bruke i dine egne programmer, er mye mer nøyaktig enn om vi hadde brukt "AT" i stedet for "CURSOR". "CURSOR" deler skjermen inn i mindre enheter enn "AT", noe som gjør at alt blir helt nøyaktig. Bare prøv med flere linjer på skjermen men varierende lengde. Du kan prøve dette eksemplet:

```
270 CLS
280 OVER 1
290 INK 7
300 FOR a = 1 TO 80
310   SENTRER 1, 0, 0, 85, a, FILL$("-", a)
320 END FOR a
330 INK 2
340 FOR a = 80 TO 1 STEP -1
350   SENTRER 1, 0, 0, 85, a, FILL$("-", a)
360 END FOR a
```

Vi er interessert i å få forslag til nye prosedyrer og funksjoner, eller kanskje har du laget en nyttig rutine selv? Send oss i så fall en kopi, slik at flere kan få nytte av den. Vi premierer alle publiserte bidrag/ideer.

Mer om systemvariabler (Pål Monstad)

Forrige gang listet vi systemvariablene til QL. Jeg lovte å publisere noen flere eksempler på bruken av disse. Siden vi nettopp fikk spørsmål om hvordan det er mulig å få en oversikt over hvor mye ledig hukommelse det er i maskinen, kan vi ta det her:

```
PRINT PEEK_L(163856) - PEEK_L(163852)
```

Denne peek'en returnerer ledig hukommelse i bytes. Hvis du heller vil ha ledig hukommelse returnert i KB, kan du dele svaret med 1024.

Det er også mulig å få vite hvor mange jobs som ligger i maskinen til en hver tid. Basic-interpreteren eksisterer alltid, og er job 0, dvs at du får returnert 1 hvis du ikke har noen andre jobs. Starter du et program av typer exec/exec_w, blir det en ny job. Har du Toolkit II kan du skrive jobs, og få en oversikt over alle jobs listen på skjermen. For de som ikke har dette gode toolkitet, kan denne funksjonen brukes:

```
100 DEFine FuNction JOB%
120   telle% = 0
130   FOR i = PEEK_L(163944) TO 4*PEEK_W(163938) + PEEK_L(163944) STEP 4
140     IF PEEK_L(i) > 0 : telle% = telle% + 1
150   END FOR i
160   RETurn telle%
170 END DEFine
```

Nå kan du enkelt og greit skrive PRINT JOB%, og antall jobs dukker opp på skjermen.

En annen peek returnerer et tall som sier hvor stor RAM maskinen din har. Hvis du har en vanlig standard QL får du returnert 128, har du en med 512K SuperQboard får du returnert 640 osv.

```
100 DEFine FuNction SYSTEM
110 RETurn (PEEK_L(163872) - 131072) / 1024
120 END DEFine
```

PRINT SYSTEM returnerer den aktuelle verdien.

Til slutt tar vi med en peek som sier om du har en eprom i ROM-porten på baksiden av QLen eller ikke.

```
100 DEFine FuNction EPROM$
110 IF (PEEK_W(49152) <> 19195) OR (PEEK_W(49154) <> 1)
120   RETurn "Nei"
130 ELSE
140   RETurn "Ja"
150 END IF
160 END DEFine
```

Nå kan du skrive PRINT EPROM\$, og svaret blir "Ja" hvis du har eprom, og "Nei" hvis ikke.

Varer fra N.A.S.A.:

HARDWARE:

QL 128K (JM) komplett m/kabler, programmer og manual:	830.00	<u>ROM:</u>	AH rom :	90.00
QL 512K (JM) komplett " " " :	1270.00		JM rom :	160.00
SuperQboard 512K disk/parallell interface og TK II :	770.00		JS rom :	230.00
Tandata Modem 1200/75 1200/1200 m/terminalprogram :	470.00		MGN rom :	300.00
Miracle QL Modem 1200/75 1200/1200 m/terminalprogram:	310.00		MGN rom	
Miracle 512K expanderam :	580.00		m/norske	
Eidersoft ICE m/mus :	360.00		taster :	370.00
Monitorer fra kr :	290.00		MGD rom :	260.00
Miracle Centronics printer interface :	390.00			

DELER:

LITTERATUR:

ULA 8301 :	149.00	Dansk QL User Guide	:	160.00
ULA 8302 :	80.00	Engelsk QL User Guide	:	210.00
MC68008 :	110.00	QL SuperBASIC-The definitive QL Handbook (Jan Jones):		160.00
MC1377P :	15.00	Using Graphics on the Sinclair QL (Garry Marshall) :		40.00
CPU 8049 :	80.00	Profiting from the Sinclair QL (Barry Miles) :		40.00
RAM :	23.00	Desk-top computing with the Sinclair QL (B. Miles) :		40.00
7805 5V :	15.00	Word Processing with the Sinclair QL (Mike O'Reilly):		40.00
Kjøleribbe:	20.00	Introducing the Sinclair QL (Garry Marshall) 175 s. :		40.00
Membran :	125.00	Introducing to SB on the Sinclair QL (D. Meadows) :		40.00
Matte :	45.00	Introducing to Simulation Techniques on the QL 125 s:		50.00
Seriekabel		QL Gamesmaster (Med svært mange eksempler. Spill) :		50.00
m/plugg :	40.00	QL SuperBASIC (A. A. Berk) 165 s. :		45.00
Nettverk-		QL Computing (Ian Sinclair) 175 s. :		50.00
kabel :	35.00	Professional and Business uses of the QL (C. Lewis) :		40.00
TV-kabel :	35.00	The real Thing. Microcomputer simulations on the QL :		70.00
Strømfors.:	140.00	QL Handbook (Tim Hartnell) 270 s. :		50.00
Taster :	12.00	QL Technical Guide (David Carlin og Tony Tebby) 195s:		170.00
Fester :	2.00	The Sinclair QL 125 s. :		30.00
Plastluker:	5.00			
EPROM-hus :	90.00	Tidligere nummer av QL World: Spør etter egen liste :		15.00
UHF (TV)		Tidligere nummer av Sinclair Magazine og QLUB fra kr:		5.00
modulator :	78.00	Tidligere nummer av Quanta :		20.00
Skruer :	0.80	Abonnement på QL WORLD (12 nummer, 1 år) :		470.00

LAGRINGSMEDIA:

OPPBEVARING:

LAPPER:

Disketter 3.5" 2DD:	7.50	Plastmappe for 4 cartridger :	15.00	Diskett:	0.75
Nye cartridger :	25.00	Cartridgebokser (plass til 20):	50.00	Cartr. :	0.50
Brukte cartridger :	15.00	Diskettbokser (80 stk. m/lås) :	118.00	Univers:	0.10

NYTTEPROGRAMMER:

Turbo m/Turbo Toolkit (SuperBASIC-kompilator, 256K)	: 550.00
Turbo Toolkit (Meget bra toolkit. 70 kommandoer)	: 150.00
Editor Special Edition (Meget bra editor, 256K)	: 300.00
Better Basic (Strukturerer dine SB-programmer)	: 130.00
Eye-Q (Tegneprogram. Antakelig det beste...)	: 200.00
Super Sprite Generator V4.00 (Lag dine egne sprites)	: 150.00
Solution Vanilla (MS-DOS emulator, 640K og diskett)	: 240.00
Qspell (Spellingchecker 25.000 ord)	: 160.00
GraphiQL (Avansert tegneprogram)	: 200.00
QIMP (Front-end for kopiering ol.)	: 130.00
Mice Art (Meget bra tegneprogram for ICE m/mus)	: 85.00
Choice (Multitasker Quill m.m sammen med ICE)	: 90.00
TechniQL (Avansert tegneprogram)	: 280.00
TechniKIT (Printerdrivere til TechniQL for plotter)	: 140.00
Frontpage I (Desktop Publisher)	: 90.00
Desktop Publisher fra Digital Precision	: 130.00
QL Cash Trader (Regnskapsprogram)	: 240.00
Pointer's & Writer's Toolkit (Toolkits)	: 90.00
Spellbound (Spellingchecker som retter etter hvert)	: 176.00
Filebound (Undersøker filer. Til Spellbound)	: 48.00
Cartridge Doctor (Redder ødelagte cartridger)	: 165.00
QL Assembler (Assembler, linker og editor)	: 155.00
QL Macro Assembler fra GST (Assembler/linker/editor)	: 330.00
4Matter og Locksmith (Kopierer kopibeskyttede spill)	: 170.00
Super Monitor (Monitor/disassembler)	: 100.00
QL Paint (Tegneprogram. Bra)	: 140.00
Qdraw (Tegneprogram)	: 100.00
QL Art (Tegnerprogrammer, kopiprogrammer m.m)	: 100.00
Super Astrologer (Astrologiprogram)	: 130.00
Pro Fortran (Programmeringsspråk)	: 600.00
Super Forth + Reversi (Programmeringsspråk + spill)	: 400.00
QL Pascal (Programmeringsspråk)	: 390.00
QMON (Monitor av Tony Tebby. Bra)	: 140.00

SPILL:

BJ Returns:	60.00
Mr Smith	: 75.00
Bridge P.2:	90.00
Othello	: 90.00
The King	: 95.00
Football	
Manager/	
Director	: 125.00
Chess	: 185.00
StripPoker:	136.00
Type 22	: 185.00
Quest	: 186.00
Matchpoint:	148.00
Fictionary:	95.00
QL Hopper	: 45.00
Flight Sim:	150.00
Mort Manor:	140.00
DragonHold:	140.00
Backgammon:	90.00
Aquanaut	: 50.00
Snooker	: 100.00
Ambition	: 120.00

Alle priser er ekskl. porto. Vi skaffer også alt annet av utstyr: Ring for info.

Konkurranse (Frode Tennebø)

Som kjent er Den kinesiske mur et imponerende byggverk (for å korrigere en vanlig misforståelse med en gang: Den kinesiske mur kan IKKE sees fra månen!), men du har lyst å imponere verdens befolkning enda mer. Derfor setter du i gang med å bygge en mur rundt hele jorden ved ekvator. Den skal være 15 meter høy og 20m bred langs bakken som gradvis reduseres til 5m på toppen. Du bruker en stein som har en egenvekt på 15 g/cm^3 . Jorden har en radius på 6378 km ved ekvator. Spørsmålene er som følger: Hvor mye større blir jordens omkrets rundt ekvator og hvor mye tyngre blir jorden når muren er ferdig?

Svarene må vi ha innen 22. august. Vi trekker ut en vinner som får tilsendt en valgfri bibliotekpakke til enten QL eller Spectrum. (Gi beskjed om hvilken pakke du vil ha).

Resetknapp på Spectrum - revidert(Frode Tennebø)

I en tidligere utgave var det en artikkel av Knut Arne Vinger om hvordan man skulle sette på en resetknapp på en 48K Spectrum. Artikkelen er fullstendig riktig, men dessverre kun for Spectrum issue 3. Alle andre modeller trenger en annen fremgangsmåte enn forklart i artikkelen.

En meget enkel fremgangsmåte som virker på alle modellene er ganske enkelt å kortslutte kondensator C27. Denne er blå og ca. 1 cm. lang og er merket C27. Den ligger normalt rundt kjøleplaten eller den største av de to keyboard kontaktene. Lodd på to isolerte ledninger på kondensatoren, en på hvert ben, og kobl på en trykkbryter. Vips har du en reset knapp!

Z80 maskinkode (Frode Tennebø)

Vi har sett hvordan vi ganger/deler et tall med to ved å shifte en bit til venstre/høyre. En mer generell multipliserings metode kan utføres som en serie addisjoner - på samme måte som du gjør det for hånd. Tenk deg følgende regnestykke:

 798
 =====

Den samme kalkulasjonen er enda enklere i binær. Bits'ene i multiplikatoren (38) er enten null eller en. Dermed legger vi til enten null eller multiplikanden (21) respektivt, mens vi husker å skifte hvert svar en bit til venstre for hvert trinn i utregningen. I binær vil dette se slik ut:

```

      (21)      (38)
00010101 * 00100110
-----
              00000000 (0)
             00010101 (1)
            00010101 (1)
           00000000 (0)
          00000000 (0)
         00010101 (1)
        00000000 (0)
       00000000 (0)
      00000000 (0)
-----
000001100011110 (798)
=====
```

Hvis du ser på tallene i parantesen til høyre, lest fra bunnen og ned, er disse lik multiplikatoren. Ved å bruke denne teknikken kan vi lage en enkel multiplikasjonsrutine. Denne vil multiplisere to 8 bits nummer til et 16 bits resultat. Anta at A registeret holder multiplikatoren og E registeret multiplikanden. Vi kan så bruke rutinen under til å returnere resultatet i HL register paret. Divisjonen blir gjort på en tilsvarende måte, som en serien subtraksjoner, på samme måte som du gjør det for hånd.

00010	LD A,54	; A = multiplikator
00020	LD E,41	; E = multiplikand
00030	LD D,0	; Sett DE = E
00040	LD HL,0	; Sett resultat til 0
00050	LD B,8	; Gjør 8 bits multiplikasjon
00060	NEXT ADD HL,HL	; Shifter HL en bit til venstre
00070	RLA	; Shifter multiplikator en bit til venstre
00080	JR NC,NOADD	; Ingen addisjon hvis ingen carry
00090	ADD HL,DE	; Legg til multiplikanden
00100	NOADD DJNZ NEXT	; Gjenta for alle bits
00110	RET	; Retur til BASIC

Du bør ungå å bruke multiplikasjon og divisjon så sant dette er mulig. Heldigvis er det sjelden man har bruk for slike "kompliserte" kalkulasjoner. Når alt kommer til alt, hvor ofte trenger du divisjon og multiplikasjon for å flytte

Space Invaders rundt om på skjermen i fantastisk fart?!

Hvis du trenger mer kompliserte kalkulasjoner, har Spectrum (og SAM Coupe) programmerere en ekstra fordel. Har du noen gang lurt på hvordan BASIC kan regne ut uttrykk som "INT (25 * COS (LN (PI/(SIN * 2^8))))"? Tydeligvis har Sinclair skrevet koden som utfører alle disse funksjonene (og noen til) inn i operativsystemet (ROM'en), og disse har blitt gjort tilgjengelig for den vanlige maskinkode programmerer. De blir alle kalt opp med RST 40 instruksjonen.

Vi vil undersøke denne og lignende instruksjoner nærmere i en senere artikkel, så ikke tenk på hvordan den fungerer akkurat nå. I mellomtiden viser vi deg hvordan du skal bruke den. RST 40 instruksjonen forventer at det ligger en liste med nummer etter den som slutter med nummeret 56. Disse nummerene forteller ROM'en hvilke operasjoner som skal utføres på verdiene i kalkulator stacken. Alle disse nummerene er vist i tabellen under.

Tall	Label	Description
0	Jump-True	Betinget hopp basert på verdien på toppen av stacken
1	Exchange	Bytter to topp verdier på stacken
2	Delete	Sletter verdien på toppen av stacken
3	Subtract	Sletter to topp verdier og stacker subtraksjons resultatet
4	Multiply	Sletter to topp verdier og stacker multiplikasjons resultatet
5	Division	Sletter to topp verdier og stacker divisjons resultatet
6	Power	Sletter to topp verdier og stacker det opphøyde resultatet
7	X-or-Y	Gir X hvis Y=0, 1 i motsatt fall
8	X-and-Y	Gir X hvis Y<>0, 1 i motsatt fall
9	X<=Y	Gir sant (1) hvis X<=Y, ellers falskt (0)
10	X>=Y	Gir sant hvis X>=Y, ellers falskt
11	X<>Y	Gir sant hvis X<>Y, ellers falskt
12	X>Y	Gir sant hvis X>Y, ellers falskt
13	X<Y	Gir sant hvis X<Y, ellers falskt
14	X=Y	Gir sant hvis X=Y, ellers falskt
15	Addition	Sletter to topp verdier og stacker addisjons resultatet
16	X\$ and Y	Gir X\$ hvis Y=0, "" i motsatt fall
17	X\$<=Y\$	Gir sant hvis X\$<=Y\$, ellers falskt
18	X\$>=Y\$	Gir sant hvis X\$>=Y\$, ellers falskt
19	X\$<>Y\$	Gir sant hvis X\$<>Y\$, ellers falskt
20	X\$>Y\$	Gir sant hvis X\$>Y\$, ellers falskt
21	X\$<Y\$	Gir sant hvis X\$<Y\$, ellers falskt
22	X\$=Y\$	Gir sant hvis X\$=Y\$, ellers falskt
23	X\$+Y\$	Kombinerer X\$ and Y\$

24	Value\$	Erstatter toppen av stacken med VAL\$ av verdien
25	Usr\$	Erstatter toppen av stacken med USR av streng verdien
26	Read-in	Read (INKEY\$) fra en kanal
27	Negate	Gjør verdien på toppen negativ
28	Code	Erstatter toppen av stacken med CODE av strengen
29	Value	Erstatter toppen av stacken med VAL av strengen
30	Length	Erstatter toppen av stacken med LEN av strengen
31	Sine	Erstatter toppen av stacken med SIN av verdien
32	Cosine	Erstatter toppen av stacken med COS av verdien
33	Tangent	Erstatter toppen av stacken med TAN av verdien
34	Arcsine	Erstatter toppen av stacken med ASN av verdien
35	Arccosine	Erstatter toppen av stacken med ACS av verdien
36	Arctangnt	Erstatter toppen av stacken med ATN av verdien
37	Logarithm	Erstatter toppen av stacken med LN av verdien
38	Exponent	Erstatter toppen av stacken med EXP av verdien
39	Integer	Erstatter toppen av stacken med INT av verdien
40	Sq-root	Erstatter toppen av stacken med SQR av verdien
41	Sign	Erstatter toppen av stacken med SGN av verdien
42	Absolute	Erstatter toppen av stacken med ABS av verdien
43	Peek	Erstatter toppen av stacken med PEEK av verdien
44	In-port	Erstatter toppen av stacken med IN verdien
45	Usr	Erstatter toppen av stacken med USR av verdien
46	String\$	Erstatter toppen av stacken med STR\$ av verdien
47	Char\$	Erstatter toppen av stacken med CHR\$ av verdien
48	Not	Gir en hvis toppen av stacken er null, ellers null
49	Duplicate	Kopierer toppen av stacken til toppen av stacken
50	X mod Y	Erstatter to topp verdier med INT (X/Y) og resten
51	Jump	Ubetinget hopp basert på verdien på toppen av stacken
52	Stk-data	Stack listen med data etter denne koden
53	Dec-jr-nz	Gjør en DJNZ På BREG system variabel
54	X<0	Gir sant hvis toppen av stacken < 0, ellers falskt
55	X>0	Gir sant hvis toppen av stacken > 0, ellers falskt
56	End-calc	Slutter RST 40 kalkulasjonen
57	Get-oper	Denne rutinen omgjør en funksjon til en verdi
58	Truncate	Erstatter toppen av stacken med avrundingen (mot null)
59	Sgle-Calc	Gjør en enkelt kalkulasjon med koden i B
60	E-convert	Omgjør et nummer på formen "numEm" til toppen av stacken
61	Restack	Stacker nummeret en gang til
62	Series	Serie generator for SIN, COS, LN, etc.
63	Stk-lit	Stack null, en, en halv, en hal PI eller ti
64	Store-mem	Lagrer øverste verdi på stacken til minnet

65 Get-mem Henter fra minnet en verdi og plasserer den på stacken

Hvis vi f.eks. ønsker å gange de to tallene øverst på kalkulator stacken og så avrunde kvadratroten nedover, kan vi bruke koden:

```
RST 40 ; Start kalkulasjon
DEFB 4 ; Gjør gjennom gange
DEFB 40 ; Ta kvadratroten
DEFB 39 ; Rund av nedover
DEFB 56 ; Avslutt kalkulasjon
```

Det eneste problemet er å legge tallene på kalkulator stacken og hente tilbake resultatet. Dette gjøres ved å kalle passende ROM-rutiner:

```
11560 - Overfører A til kalkulator stacken
11563 - Overfører BC til kalkulator stacken
10334 - Overfører AEDCB til kalkulator stacken
11733 - Overfører toppen av stacken til A
11682 - Overfører toppen av stacken til BC
11249 - Overfører toppen av stacken til AEDCB
```

Neste gang skal vi gå igjennom et program som forstørker og minsker skjermen der vi nettop bruker en del av det vi nå har gått igjennom.

Softwaretest (Frode Tennebø)

Metodia-design-Norway har gitt ut tre programmer som alle er skrevet i BASIC.

Det helt klart beste av de tre, Kwik-Kalk, er et program beregnet for dem som sysler med elektronikk og da spesielt amatør-radio. Programmet starter opp med en meny med 18 muligheter, bl.a. for utregning av Ohms lov, Resonans, Bølgelengder, Frekvens, Tidskonstant og Kapasitans samt instruksjoner. Velger man en av mulighetene, blir man bedt om å taste inn de kjente verdiene, evt. først et valg av hva som er ukjent. Svaret kommer ut øyeblikket etter. Input sekvensen er oversiktlig og lettfattlig. Et minus med programmet er at det delvis er på engelsk og delvis på norsk.

Det andre programmet, Puslespill, er et matematikkprogram der du først velger om du vil bli spurt i plus, minus, gange, dele eller en kombinasjon. Deretter blir du bedt om å taste inn navn, alder, etc. Det er her verdt å merke seg at

oppgavene blir vanskeligere desto eldre man er. Nå kommer selve oppgave-delen. Spørsmålene blir skrevet ut på en skriver og du tester inn svaret. Svarer du riktig blir en ny del av et skjerm bilde tegnet på. Det hele går selvfølgelig ut på å få ferdig hele bildet så raskt som mulig. Etter å ha prøvd det en stund (og brukt opp en hel del papir), ble det klart for meg at vanskelighetsgraden burde trimmes noe - selv for meg tar det litt tid å regne ut $11047 * 8734$ i hodet!

Kaleidoskop, det siste programmet, generer over 300 farger og variasjoner ved å kombinere de 7 spectrum fargene på en smart måte. Fiffig rutine, men er den nyttig til annet enn fornøyelse?

Ut i fra en skala på 10, vil jeg gi følgende karrakter til programmene: Kwik-Kalk: 8, Puslespill: 5, Kaleidoskop: 2. Prisen ble ikke oppgitt på disse tre programmene, men interesserte kan henvende seg til Metodica-design-Norway, Sarpemyrveien 14, 1560 Larkollen, tlf. 092 - 63572.

Spectrum super sommersalg:

Følgende spill til salgs:

Kr. 15,-: Planet of Death, Chambers of Horrors, Cosmic Cruiser, Backgammon, Wiggler, Inventions 1, Kong, Spawn of Evil, Freex, Zzoom, Gatecrasher, Handy Andy, Logo, Reversi, Club Record Controller, Hungry Horace, Galactic Patrol, Embassy Assault, Beamrider, The Valley, Chess, VU-File, Blue Thunder, Flight Simulation, Gobbleman, Traxx, Danger Mouse - In the Black Forest Chateau, The Arc of Yesod 128.

Kr. 25,-: Everyone's a Wally, Flip Flap, Bomb Jack II, Roller Coaster, Kokotoni Wilf, Frank Bruno's Boxing, Viking Raiders, Batty, Saboteur II, Deep Strike, ATV Simulator, 3DC, Spellbound, Grand National, Booty, The Way of the Exploding Fist, Cyberun, Firestorm, Mantronix, Paintbox, Screen Machine, Lode Runner 128.

Kr. 40,-: Forgotten Worlds, Out Run, The Vindicator, Indiana Jones and the Temple of Doom, Blasteroids, California Games, Killed until Dead, Road Blasters, Typhon, Fernandez Must Die, Total Eclipse, 1943, Match Day II, The Pepsi Challenge - Mad Mix Game, Track Suit Manager, Foxx Fights Back, Monopoly, Gift From The Gods, Ace of Aces, PSI-5 Trading Compant, Rasputin, Dracula, Daley Thompson's Olympic Challenge, They Sold a Million (Beach Head, Daley Thompson's Decathlon, Sabre Wuld, Jet Set Willy), Hotshots (Mindshadow, Fighting Warrior, Gyron, Shadowfire).

Kr. 60,- (dette er alle flere spill i samme boks): We are the Champions (Super Sprint, Renegade, Rampage, IK+, Barbarian), Gold Silver Bronze (Summer Games I & II, Winter Games).

I tillegg har vi Turbo Esprit, Scooby Doo og Supertrux som såkalte "review copies", dvs. de blir sendt ut "halvferdig" (uten cover, labels, men med utgivers egne instruksjoner) til blader for test. Disse blir regnet som samleobjekter og selges for kr. 50,- m/cover fra oss. Alle spillene over er i god stand og kan loades.

Vi har også 1 stk. Challenger joystick for kr. 40,-, en 25 pins serial printer ledning for QL til kr. 30,- og Disc-Mate (kopierings program for Plus D) på 3.5" disk for kr. 60,-.

Varene sendes fraktfritt ved bestillinger over kr. 50,-, beregn ellers kr. 10,- ekstra for porto. Skriv bestillingen din på en postgiro og send den til: Frode Tennebø, gt.3 nr.4, 6700 Måløy, konto nr. 0802 3829801.

QLs diskhemmeligheter (Frode Tennebø)

I et forsøk på å overføre filer fra QL til min SAM Coupe, måtte jeg (med fare for kropp og sjel) sette meg inn i oppbygningen til QDOS. Til de av dere som planlegger disk utilities el.l., vil jeg nå gjennomgå det jeg fant ut. Jeg vil benytte anledningen til å advare deg om at det kan ha alvorlige følger for din mentale tilstand om du klusser med disketter med verdifult innhold på. Dette gjelder spesielt for QL'en - makan til tungvint system har jeg sjelden sett! (Det må være en av de største "blunderene" til Sinclair noen sinne!) Benytt back-up! Merk: Alle tall er i desimal bortsett fra der jeg har brukt benevnelsen HEX eller \$tall.

Som de fleste av dere sikkert vet består en formatert diskett av tracks, sectors og sider. I tillegg vil jeg komme til å bruke begrepet "sylinder" som beskriver den samme track-posisjonen på begge sidene (eks. sylinder 0 = side 0, track 0 + side 1, track 0, sylinder 1 = side 0, track 1 + side 1, track 1). En diskett kan ha 80 eller 40 tracks, og en eller to sider. Det hele kommer an på diskettstasjonen og diskettypen du bruker. Jeg vil referere til et DSDD (Double Sided, Double Density - to sider, 80-tracks) system, men mye vil nok kunne overføres på andre systemer.

På QL'en formateres hver track til ni fysiske sectors, nummerert fra 1-9. Hver

side blir så formatert til 80 tracks, nummerert fra 0-79. Når begge sidene blir formatert slik, har vi i alt 1440 sectors tilgjengelig på en disk - 720K når hver sector kan holde 512 bytes. Se fig. 1.

Når du så saver en fil til disk, blir dataene i filen lagret fortløpende i fysiske sectors i grupper på tre. Dette betyr at selv om en fil kun er en bytes lang, bruker den likevel 3 hele sectors for å lagre den. "Sløseri", sier du? Enig, men vi må bare leve med det. Selv om filen ble lagret i 3 fysiske sectors, betyr dette IKKE at filen blir lagret i tre etterfølgende sectors på disken. Her må vi skille mellom fysiske og logiske sectors, mer om dette senere.

Merk at alle filer har en 64-bytes fil header som blir lagret aller først i en fil:

Byte offset:	Beskrivelse
0	32-bit integer fil lengde (inkludert header'en.)
4	Fil access key byte.
5	File type byte.
6	Fil informasjon (8 bytes).
14	16-bit integer fil navn lengde.
16	Fil navn (ASCII). Maks 36 bytes.
52	Dato (4 bytes).
56-	Ledig.

File access key byte er normalt null. For execute-program holder filen nummeret til den, og da vil de fire første bytes'ene i file information holder default størrelsen av data området for dette programmet (opp med hånden de som forstod dette!). (Dette er det samme som dataspace, Pål.)

Ettersom de 64 første bytene i en fil er reservert til headeren, er det kun 448 bytes tilgjengelig i den første sectoren i en fil, "sector a" block 0. De resterende sectors kan holde 512 bytes.

Men hvordan holder systemet orden på hvor de forskjellige filene ligger på disken? Jo, den første allocation block'en - block 0 - er reservert for et "kart" over disken. De 96 første bytes'en i dette "kartet" inneholder informasjon om disken:

Byte	Beskrivelse
offset:	
0	4 bytes - format ID (QL5A)
4	10 bytes - medium navn (chr\$ 32)
14	2 bytes - format random nummer
16	4 bytes - antall oppdateringer
20	2 bytes - ledige sectors
22	2 bytes - gode sectors
24	2 bytes - totalt antall sectors
26	2 bytes - sectors pr. track (normalt 9)
28	2 bytes - sectors pr. sylinder (normalt 18)
30	2 bytes - antall sectors (normalt 80)
32	2 bytes - sectors pr. block (normalt 3)
34	2 bytes - block nr. for dir EOF
36	2 bytes - byte nr. for dir EOF (0 til 511)
38	2 bytes - sector offset/track
40	18 bytes - logisk til fysisk sector tabell
58	18 bytes - fysisk til logisk sector tabell
76-	Ledig

De siste 1440 bytes er oppdelt i ialt 480 "bokser", hver på 24 bits - 3 bytes. De første 12 bits'ene i den første "boksen" reffererer til hvilket fil nummer filen som ligger i allocation block 0. De siste 12 bits'ene reffererer til hvilken block i filen allocation block 0 hører til. Tomme blocks merkes med verdien 16646143 (eller FFFFFFF hex). Høres dette innviklet ut? Ta det bare med ro. Jeg skal prøve å eksplisjere de prinsipper jeg går i gjennom mot slutten av artikkelen. I mellomtiden kan du se på fig. 2 der jeg har illustrert "kartet" (disk mapping) grafisk.

For at tilgangen til disk-data skal bli så rask som mulig, blir logiske sectors lagret, med mellomrom, i fysiske sectors. Som nevnt tidligere lagres en fil på disk i grupper på tre sectors, såkalte allocation blocks (norsklærere som leser dette og irriterer seg over utstrakt bryk av engelske termer får ha meg unskyldt). Når vi siden snakker om en allocation block er dette altså en gruppe av tre sectors. Det er 18 sectors pr. sylinder på en DSDD-disk, og disse blir vanligvis brukt i følgende rekkefølge:

Sylinder	Sectors	Side	Allocation block
----------	---------	------	---------------------

0	1, 4, 7	0	0
0	1, 4, 7	1	1
0	2, 5, 8	0	2
0	2, 5, 8	1	3
0	3, 6, 9	0	4
0	3, 6, 9	1	5

Denne tabellen finner du også i den 96 bytes lange disk descriptoren (bytes 40-57). Dette betyr at vår "kart"-block - block 0 - ligger i sectors 0, 3 og 6 på side 0, track 0. Når man beveger seg fra en track til en annen, må man trekke en ny faktor inn i bildet, sector offset pr. track i byte 38 i disk descriptoren. Hver gang man beveger seg fra en track til en annen, en logisk/fysisk sector tabell, som vist over, vil bli modifisert. Hvert element i tabellen blir så modifisert etter følgende formel:

opprinnelig + (track * offset) MOD 9

Dette offset er vanligvis 5. Dermed blir allocation block 6 sectors 6, 9 og 3, i den rekkefølge, på side 0, track 1. Det er også verdt å merke seg at i tabellen i file descriptoren blir siden indikert med et tillegg på 128 for side en for de aktuelle sectors.

Hver fil har et nummer som er en index til directory'en. Opprinnelig blir filene gitt fortløpende, stigende nummer, dvs. 1, 2, 3,.... Både directory'en (fil 0) og "kartet" (fil 248) blir av QDOS behandlet som filer. Hver gang du sver en fil blir directory'en først oppdatert med de samme 64 bytes som saves sammen med selve filen, evt. erstatter det en tidligere slette fil. I begge tilfeller blir filen gitt et nummer som tilsvarer dens plass i directory'en (dette nummeret multiplisert med 64 gir en peker på hvor i directory'en headeren skal starte). Deretter blir "kartet" lest, det nødvendige antall sectors funnet, filen savet og til slutt blir "kartet" savet med de tilhørende endringer for den nye filen. Vær oppmerksom på at directory'en vokser etter hvert som det er nødvendig. Størrelsen blir gitt i bytes 34/35 og 36/37 i fil descriptoren. Om dette har en øvre grense vites ikke, men det ser ut som om at i hvert fall hele sylinder 0 bortsett fra "kartet" (block 0) er reservert for directory'en - dette gir 119 filer som er nok for de fleste.

Når du sletter en fil blir hele directory innholdet for denne filen slettet -

navnet finnes altså ikke lenger i directory'en. I "kartet" blir den viktigste (mest signifikante) byten i de tilhørende "boksene" satt til 253 (FD hex). Hvis vi f.eks. sletter fil nr. 6 som har 2 tilhørende blocks, blir de to "kart boksene" endre slik (her er det faktisk enklest å benytte HEX):

\$006000 ==> \$FD6000

\$006001 ==> \$FD6001

Så har du slette etter en fil ved et uhell, bør du først skaffe deg en liste over de blocks som er merket \$FDx000, \$FDx001, \$FDx002,..... der x angir det siste HEX sifferet for filnummeret. I eksempletet over kunne det like gjerne ha vært fil 22 (16 HEX) som var slettet. Deretter leter du gjennom alle sectorene på disken etter det aktuelle filnavnet med et offset på 16. Når (hvis!) du finner dette navnet kan du regne ut hvilken allocation block sectoren hører til og sammenligne det med de blocks du fant tidligere med \$xxx000 (husk at sectoren med headeren alltid er den første). Mar du en match er du nesten i mål. I motsatt fall er det bare å lete viderer.

Fant du den første block'en har du bare igjen å finne resten slik at lengden i bytes 0-3 stemmer. Som regel vil disse ligge etterhverandre, men ta ikke dette for gitt! Har du funnet alle block'ene, retter du disse opp ved å forandre \$FDx til det aktuelle fil-nummeret og saver "kartet". Deretter korrigerer du directory'en med en kopi av headeren på posisjonen: file_nr. * 64. Du må korrigere bytes 20/21 og kanskje også bytes 34/35 og 36/37 i file descriptoren i "kartet".

Alt dette kan virke litt tåkete, men jeg skal nå med et eksempel prøve å få det litt klarere.

Anta at du har en formatert disk (m/directory) i driven som er fri for andre filer. Du saver så et BASIC-program som er 7680 bytes langt. Dette tilsvarer 15 sectors - 5 alocation blocks. Men i tillegg til selve programmet må også headeren lagres, og ettersom det ikke er ledig plass på "program sectorene" trenger vi en ekstra sector - og en ekstra allocation block. Hadde programmet vært 64 bytes kortere hadde vi akkurat greid oss med 5 blocks, nå må vi ha 6. La oss kalle filen for "BOMLAGADAFSJIPOING". Headeren blir da (bytes merket x er ikke av interesse for oss):

Byte offset	Innhold
0	7744 (0, 0, 30, 64)
4	x
5	x
6	x
14	18 (0, 18)
16	BOMLAGADAFSJPOING
52	x
56	x

Vi vil videre forutsette at filen blir lagret i de nærmeste tilgjengelige block'ene og at det er den første filen på disken - fil 1.

Directory'en blir så oppdatert med 64 bytes fra byte 1 * 64 = 64 (hadde vi savet enda en fil hadde denne fått sine 64 bytes plassert fra byte 2 * 64 = 128) og lagt tilbake på disken. De 64 første bytes'ene (0 - 63) er reservert for directory'en.

Nå blir "kartet" tatt inn og disk descriptoren blir korrigert slik (x=ignorerer, z=ingen endring):

Byte offset:	Innhold før:	Innhold etter:
0	QL5A	z
4		z
14	x	x
16	x	x
20	1434	1416
22	1440	z
24	1440	z
26	9 (0, 9)	z
28	18 (0, 18)	z
30	80 (0, 80)	z
32	3 (0, 3)	z
34	0	0
36	64 (0, 64)	128 (0, 128)
38	5 (0, 5)	z
40	x	x

58	x	x
76-	x	x

Selve "kartet" vil se slik ut:

"Boks":	File nr.:	Block nr.:	Innhold:
0	3968 (\$F80)	0 (\$000)	"Kart"
1	0 (\$000)	0 (\$000)	Directory
2	4063 (\$FDF)	4095 (\$FFF)	Reservert dir. (?)
3	4063 (\$FDF)	4095 (\$FFF)	Reservert dir. (?)
4	4063 (\$FDF)	4095 (\$FFF)	Reservert dir. (?)
5	4063 (\$FDF)	4095 (\$FFF)	Reservert dir. (?)
6	1 (\$001)	0 (\$000)	Header + bytes 1 - 1472 i fil 1
7	1 (\$001)	1 (\$001)	Bytes 1473 - 3008 i fil 1
8	1 (\$001)	2 (\$002)	Bytes 3009 - 4544 i fil 1
9	1 (\$001)	3 (\$003)	Bytes 4545 - 6080 i fil 1
10	1 (\$001)	4 (\$004)	Bytes 6081 - 7616 i fil 1
11	1 (\$001)	5 (\$005)	Bytes 7617 - 7680 i fil 1
12	4063 (\$FDF)	4095 (\$FFF)	Tom

Før du savet noe på disken ville "boksene" 6 - 11 (som representerer allocation blocks'ene med de samme numrene) også ha vært tomme slik som "boks" 12.

Bestemmer du deg nå for å slette filen vil først headeren for den filen i directory'en bli slettet, deretter vil disk descriptoren bli korrigert til slik den var før du savet filen og til sist blir "boks" 6 - 11 i "kartet" endret til \$FD1 for fil nummeret og \$FD0, \$FD1, \$FD2, \$FD3, \$FD4 og \$FD5 for block nr. Merk: Ingen av datasector'ene blir endret!

Vel, håper at dette har vært interessant lesning, og at den har nøstet opp noen løse tråder. Hvis det er noe dere lurar på ang. oppbygningen av QL disketter, er det bare å skrive.

Football Director/Manager (Pål Monstad)

Dette spillet til QL er nå utkommet i nok en ny utgave. Programmet er nå bedre enn noen gang, blant annet er svært mange "skjønnhetsfeil" rettet opp. Layouten er også blitt mye bedre. Det er nå oversiktlig og greit med rullegardinsmenyer og muligheter for å ta utskrifter til skriver. Uttak av laget er mye enklere. Du

slipper å huske hvilke spillere som er gode, nå har du oversikten hele tiden. Spillerne kan nå også bli skadet på trening, i tillegg til under kampene. I E-cupen kan det nå bli straffekonk, samtidig som det er lagt inn utenlandske navn (franske, tyske, sovjetiske, nederlandske norske osv) på de aktuelle spillerne.

Det finnes også egne kommandoer for økonomien, rekorder og tidligere oppgjør. Bookmakerne har tips for kampene, du må betale regning til politiet for vakthold og regning for adm. kostnader. Det finnes også en egen kommando for sponsor.

Vel, det er egentlig ikke lett å si med 20 linjer alt som er positivt med dette programmet. Det eneste negative jeg kan komme på i farten, er at programmet er avhengig av 256K og diskettstasjon, hvis en kan kalle det negativt...

Prisen for spillet som vil holde deg foran skjermen i timesvis i strekk er kun kr 125.00 inkl. utskrevet norsk eller engelsk manual, som nå er blitt mer unødvendig enn før (programmer er blitt enda enklere å bruke). Kontakt N.A.S.A. for mer informasjon hvis du skulle være interessert i det.

Rabatt på programmer (Pål Monstad)

Digital Precision:

Vi har fått til en ordning med Digital Precision som gjør at medlemmene kan kjøpe programmer med opptil 30% rabatt. Dette gjelder ved kjøp av 6 programmer eller mer. Hvis 6 stykk av medlemmene er interesert i Perfection, vil programmet koste ca 680,- kroner da inkl. porto og div. gebyrer. Hvis mange nok melder sin interesse, kan det med andre ord være mange penger å spare. Selv med bare to interesserte, oppnår vi 12.5% rabatt!

N.A.S.A.:

N.A.S.A. (QL avdelingen) har sommersalg, og tilbyr medlemmene følgende rabatter:

-12% på alle QL-modellene, -8% på ekspansjonskort og modem, -5% på all software og -3% på alt annet. Tilbudene gjelder ut august måned

Deadline for neste nummer er satt til 12. september.