

28,-

Danmarks mest solgte edb-publikation

Brug din PC optimalt

CONFIG.SYS og AUTOEXEC.BAT

DOS 5, 6.0, 6.2 og 6.22
Windows 3.1 og 3.11

- også relevant for Windows95

Hukommelse nok til spil
Optimér din harddisk
DoubleSpace

Hold styr på dine filer med
Norton Commander 3 og 4
Brug hukommelsen optimalt
og mange andre tips

Michael Maardt

www.KnowWare.dk

4. udgave

Acrobat Reader: tips ...

F5/F6 åbner/lukker **Bogmærker**

I Menu **AVis** **sindstiller du, hvordan filen vises på skærmen**

CTRL+0 = Hele siden **CTRL+1** = Originalstørrelse **CTRL+2** = Vinduesbredde

I samme menu kan du osse sætte: **Enkelt side**, **Fortløbende** eller **Fortløbende - Dobbelsider** .. Prøv, saa ser du forskellen.

Navigation

Pil til højre/venstre: fremad/tilbage en side

Alt+Pil Højre/Venstre: som i Browser: fremad/tilbage

Ctrl+ + forstørrer og **Ctrl+ -** formindsker

<http://www.knowware.dk>

Forord til dette 6. oplag af 4. udgave

Dette hæfte startede en lille revolution inden for edb-litteratur i Danmark. Tilgiv mig et vist nostalgisk forhold til dette hæfte. I denne tekst lagde jeg en del af min sjæl i efteråret 1992 og foråret 1993 i håbet om genklang hos andre PC-brugere. Tak for alle tilbagemeldinger, feedback og først og fremmest støtten til projektet, som nu står over for sin internationale entré. Dette lille hæfte har nu alene i Danmark solgt over 170.000 - og det sælger stadig, hvilket ikke alene glæder mit hjerte, men også viser, at mange stadig bruger det glimrende Windows 3.1 eller 3.11 eller at flere ting stadig er relevante under Windows 95.

Eneste rettelse i dette 6. oplag er dette forord, resten har jeg ikke rettet, selv om flere ting er ret uaktuelle for mange, men jeg synes, at hæftets indhold har en vis historisk værdi, og måske du også synes, at det er lidt sjovt at se, hvad jeg skrev for nogle år siden.

Hæftets indholdsmæssige kerne og årsagen til det høje salgstal er afsnittene om memoryformer og de to startfiler, config.sys og autoexec.bat. Disse afsnit var og er måske stadig unikke set i international sammenhæng. (oktober 1997)

Er hæftet noget for dig?

Dette er ikke et begynderhæfte, men 'begyndere' kan sikkert hente lidt hist og her. Hvis du er begynder, så læs først *Start med DOS*.

Afsnittene er bevidst forskellige i sværhedsgrad. Du får mest ud af teksten ved at have minimum DOS 5, mens DOS 6 beskrives separat.

Formålet med det hele

Et af formålene med hele dette projekt er at hjælpe så mange som muligt i brugen af PC. Hvis du vil støtte projektet, så fortæl andre om hæfterne.

Ændringer i denne 4. udgave

I forhold til 2. udgave:

Der har været mange forskellige oplag af 2. udgave, men generelt i forhold til 2. udgave: afsnittene DoubleSpace og Bootdisketter er kraftigt revideret. Afsnit om Faxmodem og pakkeprogrammer er slettet, mens Windows 3.11 er nyt.

I forhold til 3. udgave:

overlap mellem Start med DOS og dette hæfte har betydet, at jeg har slettet noget, men har henvist til Start med DOS. Har du 3. udgave, skal du kun købe dette hæfte, hvis du samler på alle de forskellige oplag.

OBS

Jeg har forsøgt at ajourføre hæftet så godt som muligt. Husk på, at jeg skrev dette hæfte i begyndelsen af 1993,

så selv om udviklingen tilsyneladende går hurtigt i computerbranchen, så bruger de fleste af os versioner af programmer i længere tid end producenterne ønsker, så det går trods alt ikke så hurtigt, som mange ynder at fortælle. Vi bestemmer, hvor hurtigt det skal gå, thi det er os forbrugere, der betaler gildet. Husk det!

Ideen fra KnowWare

er at videregive relevant og letforståelig viden til en rimelig pris.

April 1993 kom første udgave af *Brug din PC optimalt* på 48 sider og jeg opfordrede læserne til at komme med forslag til forbedringer. Jeg fik heldigvis flere hundrede breve og telefonopringninger fra mennesker, som syntes, at et hæfte til 28 kroner var en fantastisk ide.

Vigtigste motiv var at give min PC-viden til så mange mennesker som muligt så billigt som muligt. Edb-bøger er eller rettere *var* for dyre. Vi er interesseret i information og viden, som kan lette arbejdet. Om jeg får denne viden fra en fin bog med 4 farvet omslag eller fra et hæfte, er mig ligeegyldigt, men prisen er ikke ligeegyldig.

Hæfterne trykkes i *rotation*, som også bruges til avistryk. På lange ruller papir trykkes mange sider ad gangen på begge sider. Et hæfte kan trykkes i 10.000 eksemplarer på et par timer, hvorefter bogbinderen beskærer, sætter hæfteklammer i og pakker i bundter. Jeg kan godt lide, at hæftet kan knækkes i ryggen hele vejen rundt *uden* at springe op. Det kan foldes, ligge ved siden af PC'en eller puttes i inderlommen.

I starten tænkte jeg: Hvis der sælges mange rimelig hurtigt, så kan det sikkert løbe rundt økonomisk. Jeg tog en chance. Jeg håbede, at nogle aviser ville skrive om den nye ide, hvilket heldigvis skete. Tak til journalisterne, som hjalp.

Nu har vi danskere glæde af et enormt udvalg af relativt billig computerlitteratur - desværre er ikke alle publikationer af samme kvalitet, men det udjævner sig nok med tiden. Dette store billige udvalg eksisterer ikke i noget andet land i verden!

Ide og baggrund

I efteråret 1992 fik jeg tid og ide til denne tekst. Den nødvendige viden om det 'svære' havde jeg i marts 1993 efter at have læst min første større PC bog: John M. Goodman: *Memory management for all of us*.

Lidt om mig selv

Jeg er født 1952, har gået i gymnasiet, har spillet en masse fodbold og 1970-80 har jeg læst sociologi og psykologi i Danmark samt nogle år i Tyskland. Siden 1986 har jeg arbejdet med PC - dog med en pause 1988-91, hvor jeg arbejdede med en anden computer.

Tak

til alle, som har støttet mig og lært mig forskellige ting, og tak til alle, der hjælper med at forbedre og sprede denne tekst. Tak for alle de positive henvendelser og forslag. Taknemmelighed og hjertelig tak til mine andre lærere og vejledere.

God fornøjelse

med læsningen. Jeg håber, du får større glæde i dit arbejde med din PC, og at denne tekst vil bringe dig en større forståelse og indsigt i PC'ens hemmeligheder.

Vigtigt

Alt, hvad du foretager dig, efter ideer eller forslag fra denne tekst, gør du på *eget ansvar*. Jeg har primært erfaring med *kloner*, som typisk er produceret i Taiwan.

Hvis du er utålmodig og straks vil rette din CONFIG.SYS, så husk for himlens skyld at have en boot-diskette, der *virker!* Med en boot-diskette kan du starte din PC. Sæt en tom *formateret* diskette i din PC og skriv

```
C: \>SYS A:
```

Se også afsnittet *Bootdisketter* s. 49. Mit råd er at læse det hele i den rækkefølge, det er skrevet - incl. de henvisninger, der tvinger dig til at springe lidt. Dette gælder specielt indholdet i CONFIG.SYS under DOS 5, hvor du *kan* komme til at skrive noget, der bevirker, at du ikke kan starte din PC fra harddisken, dvs. at den låser under 'boot'. Nu er du advaret. Ved DOS 6 findes dette problem ikke.

Hvis du er vant til at rette *startfilerne* CONFIG.SYS og AUTOEXEC.BAT og har en boot-diskette, der *virker*, kan du starte med *CPU og Memory* s. 8.

Hvis du ikke forstår så meget af disse afsnit, skal du for det første ikke være ked af det. For det andet kan du nøjes med at ændre startfilerne - eller få en anden til det! - med mine forslag. Jeg skriver undervejs, hvad der er teknisk, avanceret el. lign. Jeg forudsætter, at du har installeret DOS i directory C:\DOS. Kun Windows version 3.1x beskrives.

Engelske udtryk

Jeg bruger med vilje hovedsageligt de engelske ord directory, path, upper memory, high memory, extended memory osv., men de bliver forklaret. Jeg bryder mig ikke om de besynderlige oversættelser, specielt af directory, som i mange år har været kaldt katalog, bibliotek, indeks osv. (*directory* bliver beskrevet senere). Du møder alligevel mange engelske udtryk, så du kan lige så godt lære dem.

Jargon

Edb- og selvfølgelig også PC-verdenen er fyldt med jargon, og i Danmark er skribenter desværre ikke blevet enige om at oversætte ens. Jeg vil prøve undervejs at nævne nogle af disse udtryk, som jeg og andre bruger. DOS kommandoer skrives

med courier - sådan

Hvis jeg skriver 'skriv i DOS', betyder det altså, at du står i DOS prompten (ser sådan ud C: \>) eller 'er i DOS'. På dette *niveau* kan du skrive DOS - kommandoer. Nogle programmer kan midlertidigt hoppe ud til dette niveau og vende tilbage med EXIT.

At stå eller være i et directory - ex: du skal stå i C:\EXTRA - betyder, at DOS kommandoer uden nærmere angivelse (engelsk *default*) vil blive udført på filer i dette directory.

Default er et godt begreb eller ord, som vi desværre ikke har på dansk. Det nærmeste er *forhåndenværende* eller *aktuel*, men mere nøjagtigt betyder det 'i mangel af nærmere angivelse eller specifikation'. Hvis du står i et directory og skriver DIR ville DOS sige: 'Hvis du ikke skriver i hvilket directory, jeg skal vise filer, så vælger jeg default for dig' - og default er så det aktuelle directory. Når du er i DOS, befinder du dig *altid* i et *directory* på et *drev*.

I programmer siges ofte noget med default, fx 'using default settings'. Når du vil hente eller gemme et dokument i tekstbehandling, bruges et default directory til dette. Dette kan ofte ændres. Når du starter et tekstbehandlingsprogram og starter med et tomt dokument, er mange ting som fx venstre og højre margin som standard (default) sat til nogle bestemte værdier, og som regel kan de fleste eller alle af disse default-værdier ændres af brugeren. Forhåbentlig har du nu en ide om begrebet default.

Filnavne skrives hovedsageligt sådan - HIMEM.SYS. *Drev- og directorynavne* skrives altid med stort - C:\DOS (A) betegner efter min vurdering afsnit med *avanceret* tekst - markeret med denne skrifttype.

Henvisninger til *manualen* betyder Microsofts DOS 5 manual. DOS 6 behandles særskilt.

Læs lige det her

Dette afsnit indeholder intet teknisk. Inden jeg startede med at skrive denne tekst hjalp jeg venner og så, hvor lang tid der bruges på at lære at bruge PC og tilhørende programmer. Hvis man støder ind i et vanskeligt problem, kan der gå rigtig lang tid - ikke alle har råd eller lyst til at betale sig fra det. Jeg håber at kunne bidrage til at reducere denne tid for dig, men det kræver, at du i første omgang investerer lidt tid. Du får den flerfoldigt igen.

Hvis du er begynder, skal du ikke være nervøs. Millioner før dig har lært det, men ting tager tid. Du får i starten måske kun glæde af få afsnit, men kan senere vende tilbage til de andre. For nogle år siden forstod jeg ikke en brik af meget af det, jeg i dag skriver om. Som bekendt er den bedste læreproces at forklare det til andre.

Jo senere du er begyndt i denne PC-jungle, jo hurtigere kommer du til at forstå og bruge din PC. Programmerne og deres on-line (direkte på skærmen) hjælp bliver stadig bedre, og litteratur kommer der også stadigt mere af. De danske PC-tidsskrifter, som har en meget høj standard, indeholder trods alt også lidt begynderstof og kan lånes på næsten alle biblioteker.

Der er visse grundlæggende ting, som skal være i orden, før din PC fungerer optimalt, og på længere sigt bliver du nok nødt til at lære en vis orden og disciplin, hvis du skal have rigtig godt udbytte af den.

Det sørgelige paradoks i denne computertid er, at der findes en masse aktuel viden i bøger, specielt i tidsskrifter og hos mennesker, men ingen ved, hvor eller hvordan man hurtigt finder den. Ingen ved alt i denne branche, fordi alt sker så hurtigt, markedet er meget stort og nye produkter kommer i en stadig stigende strøm.

En af årsagerne til denne tekst er den dårlige manual fra Microsoft til DOS 5. Ikke et eneste sted i denne 600 sideres manual står der et eksempel på de berømte *startfiler* AUTOEXEC.BAT og CONFIG.SYS.

Jo mere jeg skrev om det tekniske, jo mere kom teksten til at handle om optimering. Dele af teksten er blevet mere avanceret end jeg oprindeligt havde tænkt mig, men de, der forstår at bruge det, bliver så forhåbentlig glade, og hjælper så vennerne, ik'?

Startfilerne er ikke det mest ophidsende ved en PC, men er for os brugere noget af det vigtigste, hvilket 'overses' af de fleste i branchen. En af mine ideer, da jeg i sept. 1992 startede med at skrive, var delvis at skrive *min egen ønskebog*: den bog, jeg selv gerne havde haft, mens jeg havde alle problemerne.

Pga. tekstens karakter med de forskellige sværhedsgrader har det været lidt svært at placere de forskellige afsnit i en sådan rækkefølge, at en enkel gennemlæsning er tilstrækkelig, og du vil måske savne en rød

tråd. Hvis du er erfaren bruger, må du finde dig i en del begynderstof.

Nogle læsere bliver nok nødt til at læse teksten hurtigt igennem for at orientere sig for derefter at starte forfra. Et begreb bliver så vidt muligt defineret, første gang, det optræder.

Jeg har tidligere skrevet om andre emner, men jeg må sige, at det har været svært at strukturere denne tekst, hvor emnerne til tider er svære. Det er ikke altid let at finde den rette balance mellem teori og praksis.

Jeg forudsætter, at du er parat til at bruge knolden, dvs. du har *lyst til at tænke selv!* Der findes tusindvis af specifikke problemer med en PC. Jeg har forsøgt at give en *teoretisk viden* med *sammenhørende konkrete eksempler*, som forhåbentligt sætter dig i stand til at løse *andre* problemer end de i denne tekst beskrevne.

Batfiler

En BAT-fil er en samling DOS kommandoer, som er gemt i en fil. Hvis du ikke er vant til at rette i tekstfiler eller lave bat(ch)-filer, er her et minikursus samt advarsel om først at begynde at rette startfilerne, når du har styr på det. Editoren EDIT, som følger med DOS 5 og 6, er beskrevet i *Start med DOS* s. 35.

En hurtig måde at oprette en tekstfil er følgende. Du laver for at se princippet lige en fil, som hedder EASY.BAT, bare for at prøve det.

Copy con betyder 'kopiér fra consollen'. Consollen er en betegnelse for skærm+tastatur.

```
C:\BAT>copy con easy.bat
```

cursoren blinker, du er 'inde' i en fil, skriv

```
c:
cd\
dir
cd \dos
dir/w
cd \bat
```

Tryk **F6** og **Enter**. DOS melder

```
1 File(s) copied.
```

Filen EASY.BAT er skabt. Nu kan du 'køre' denne batfil ved blot at skrive EASY. Prøv. Du kan skiftevis starte/standse processen med **Ctrl+S**, hvis du vil følge med undervejs. **Ctrl+S** betyder hold **Ctrl** nede, mens du trykker S.

Nu laver du en batfil PRI.BAT, som automatisk starter EDIT og henter EASY.BAT ind i editoren.

```
C:\BAT>copy con pri.bat
edit c:\bat\easy.bat
```

Tryk **F6** og **Enter**.

Hvis du nu skriver PRI, starter EDIT og indlæser (loader) filen C:\BAT\EASY.BAT.

Skab nu følgende to bat-filer. EC.BAT starter editering af CONFIG.SYS og EA.BAT starter editering af AUTOEXEC.BAT. De to directories BAT og DOS skal være i din PATH.

```
C:\BAT>copy con ec.bat
edit c:\config.sys
```

Tryk **F6** og **Enter**.

```
C:\BAT>copy con ea.bat
edit c:\autoexec.bat
```

Tryk **F6** og **Enter**.

Det specielle ved EDIT er, at den ikke som normale tekstbehandlere laver 'word-wrap' - skærmen starter på en ny linie, selvom selve teksten egentlig ikke gør det.

Editorer fungerer forskelligt, men de fleste fungerer sådan, at et afsnit (tekst mellem **to Return/Enter**) kun vises i én linie, dvs. fortsætter 'udover skærmens højrekant' (max 256 tegn i én linie). Det skal man lige vænne sig til, men i EDIT fungerer fx **Ctrl**+pile som ved mange andre programmer. At editoren ikke arbejder med word-wrap er imidlertid smart, når man arbejder med filer som fx Windows' SETUP.INF. Hvert program har sine fordele.

EDIT er en glimrende, hurtig editor. Jeg brugte den ofte til noter, kunne hurtigt starte denne fil med en batfil, og søgning og editering er enkelt.

Læs mere om batfiler i *Start med DOS* s. 48.

ASCII og talsystemer

Det følgende gælder kun under DOS, ikke Windows. ASCII (American Standard Code for Information Interchange) var noget af det første, jeg for mange år siden var glad for at forstå. Det udtales aski og er en aftale eller definition om forholdet mellem et tegn og et tal.

Når en tast trykkes og skærmen lyser med et bogstav, så forledes man til at tro, at bogstavet så at sige ligger nede i tasten og bare venter på at blive aktiveret, men så let er det ikke.

Jeg forbigår lige tasternes *scankode*, men det er reelt en sådan, der bliver sendt afsted, når du trykker en tast. Denne kode sendes til en oversætter, som finder ud af, hvilken ASCII værdi, der er knyttet til scankoden, som så igen oversættes til et tegn på skærmen.

I edb findes kun to tilstande: tændt eller slukket, 0 eller 1 og intet andet, og dette fylder det, der kaldes 1 *bit*. En computer kan ikke andet end sende et 0 eller 1 tal afsted! 8 af slagsen kaldes 1 *byte*, og bogstaver og andre tegn er egentlig lagret som en serie af nuller og et-taller i en kasse med 8 rum. Et rum kan kun indeholde enten 0 eller 1 (2 tal-systemet).

Med 8 rum har man 2 gange med sig selv 8 gange = 256 forskellige muligheder for at skabe en talværdi. Nul er også en OK værdi, så man hører tit om 0-255 værdier.

Man blev heldigvis på et tidspunkt enige om, hvilke værdier de forskellige tegn og bogstaver skulle have i dette system, ASCII, som altså er begrænset af disse 256 muligheder. Lad os tage det store A som eksempel. Prøv i DOS eller i et program at holde venstre **Alt** nede, mens du på det numeriske tastatur taster 65 og slipper **Alt**.

Og for at gøre det hele lidt sjovere (men lettere for programmører) har Microsoft ved introduktionen af Windows benyttet en anden overenskomst kaldet ANSI, som har andre talværdier.

Heldigvis er de normale bogstaver uforandrede, men en tekstfil lagret efter ASCII systemet og en tekstfil lagret efter ANSI systemet er to forskellige ting. Tekstfiler lagret af DOS programmer lagres altså efter ASCII, og tekstfiler under Windows lagres (default i hvert fald) under ANSI. Write og andre tekstbehandlere under Windows har mulighed for at lagre og konvertere i det ene eller andet format.

Dette er nogle af grundpillerne i *datakommunikation*. Længere er vi mennesker stort set ikke nået i forhold til hjernens og sindets udvikling, men det går da fremad. Den eneste grund til computerens udbredelse er dens hastighed. Lad være med at tro, at en computer kan være intelligent, selv om nogle forsøgte at bilde os det ind - kunstig intelligens, neurale netværk osv.

Se også *Start med DOS* s. 45.

16 talsystemet

(A) I edb-verdenen er 16 talsystemet meget benyttet. Vort normale er titalsystemet, som vi har fået banket ind i knolden.

0123456789 og så skifter vi (fordi vi ikke har flere symboler (fingre/tæer)) for at nulstille førstepladsen, bagfra altså, og starter forfra på andenpladsen (stadig bagfra) med 1, og får så 10.

Hexadecimalt, dvs. i 16 talsystemet ser det sådan ud 0123456789ABCDEF og først derefter (ved vores normale 16) bliver det 10.

I 10 talsystemet siger vi: første ciffer bagfra er enere, næste er 10'ere, næste er 100'ere osv. I 16 talsystemet siger man (fordi 'man' tænker i 10 talsystemet) bagfra: første er enere op til F(=15), næste er 16'ere, næste er 256'ere osv. Det er ikke så svært, vel!

10-talsystemet kan slet ikke bruges i forbindelse med computere. Det er yt, et levn fra mennesket, et kreatur, der tænker med 10 fingre.

2 tal	10 tal	16 tal
00010000	16	10
00100010	34	22
11111111	255	FF
	256	100
	65536	10000
	1048576	100000

Der arbejdes med store tal i edb-sammenhæng, så hexadecimalt er smart og tallene angives sommetider med et h i enden, fx A0000h. Som vi senere skal se, 'snyder' man ofte for det sidste nul, hvorefter ovenstående bliver til A000h, som i 10 talsystemet er de berømte 640KB = 640 x 1024 bytes = 655360 bytes.

Hvis du har Windows, kan du med regnemaskinen (vælg: **view, scientific**) lege med disse tal og få dem konverteret.

CPU og memory

En del af det følgende er ret teknisk og kan springes over. Jeg bruger ordene memory og hukommelse i flæng. Hukommelse er et sted, hvor information kan opbevares. I en PC består hukommelsen af to forskellige former, RAM og ROM.

Read Only Memory omtales ikke meget i denne tekst, men enkelt fortalt har man valgt at lægge en del af styresystemet fast i chips. Andet ROM kommer fra andre fysiske dele som fx grafikkort og harddiskcontroller.

Random Access Memory er den flygtige hukommelse, som er tilgængelig, mens PC'en er tændt. RAM er vigtig. CPU'en dirigerer data rundt i RAM, og her kan indholdet løbende ændres.

Først lidt om opdelingen af RAM i en PC, som er indviklet, så længe styresystemet hedder DOS. Computere arbejder efter to-tal-systemet. Du møder ofte tal, som er potenser af tallet to. 2, 4, 8, 16, 32, 64, 128, 256, 512 bytes, og når man kommer til 1.024 bytes kaldes det 1 KiloByte (**KB** eller **K**), og derefter samme række 2, 4.. indtil 1.024 KB, som er lig 1 MegaByte (**MB**) osv. til 1.024 MB=1 GigaByte.

Da IBM lavede den første PC, troede de ikke, at den ville blive så stor en succes, og dengang regnede de med, at 1 MB (1024 KB) RAM med 640 KB afsat til programmer ville være rigeligt.

Lad os se på den første MB RAM, altså de første 1.024 KB af RAM. Forestil dig en kæmpe reol med alle disse pladser til ens bøger. Hver bogplads er plads til en byte. Denne plads har en såkaldt adresse. Det følgende handler meget om, hvordan disse egentlig logiske adresseområder defineres og behandles.

CPU'ens arbejde består hovedsageligt i at flytte rundt på alle disse bytes mellem harddisk, sig selv, RAM, skærm osv. - og det går stærkt. Stort set al kommunikation i en computer, dvs. transport af data mellem de forskellige enheder (devices), skal rundt om CPU'en. Skærm, tastatur, disk osv. er fysiske devices, men også andre mere logiske eller strukturelle dele i en PC kan være et device.

Som vi senere skal se, skal ethvert *device* styres, håndteres af et specielt program, som kaldes en *device driver* eller forkortet: en *driver*. Device er et underligt begreb, som jeg var flere år om at forstå.

DOS kan kun 'se' den første MB. Når processoren kører i sin mest primitive 'mode', nemlig *real*, kan den kun 'se' dette adresseområde i RAM. DOS er et såkaldt *real mode* program. Når du booter din PC, 'vågner' processoren i denne real mode. Man kan sige, at hjørnestenen i en PC er en processor, der vågner i real mode sammen med DOS. Sådan har det været lige siden DOS så dagens lys. Hvis man vil mere med sin PC end real mode og DOS, så skal der bygges mere oven på grundstenene.

For ikke at udelukke 8086 og 80286 og tidligere DOS versioner er DOS 5 og 6 stadig *real mode* programmer og såkaldt *bagud kompatibel*, hvilket betyder, at alle programmer skrevet under tidligere versioner fungerer under nyere.

For at udnytte RAM udover 1 MB skal man have et program, som kan få processoren til at køre i såkaldt *protected mode*, hvilket bevirker at 2 programmer, som befinder sig i RAM på samme tid, ikke bruger samme område i hukommelsen. 80286 var den første processor, som udover real mode også kunne køre i protected mode.

Processoren kan kun være i én af tilstandene på et givent tidspunkt, og det tager tid at skifte mellem real og protected mode.

En CPU arbejder ved en såkaldt clockfrekvens, dvs. svingninger pr. sekund (Hertz), fx 20, 25, 33, 40 Mhz eller højere - måles i Mhz (mio. Hertz - det er rigtig nok).

Memoryformer

Afsnittet er teknisk, men kan hjælpe til forståelsen inden afsnittet om startfilerne.

	extended
1024-	high = første 64K
640-1024	upper
0-640	conventional

Conventional memory

Området i memory fra 0-640 KB kaldes *conventional* eller *lower memory*. Her foregår det! DOS er memory manager for (håndterer og styrer) conventional memory.

Upper memory area

Den følgende beskrivelse er medtaget, fordi det er vigtigt fra og med DOS 5, hvor muligheden for at indlæse visse programmer i dette adresseområde gjorde, at der blev frigivet plads i conventional memory til afvikling af DOS programmer.

En stor del af denne tekst kredser om dette emne, og du skal ikke være ked af, hvis du ikke forstår en brik af det. Jeg har selv ikke fattet det hele endnu, men for hver gang jeg skriver, fatter jeg lidt til - håber jeg da, og jeg har altså beskæftiget mig med dette ret intensivt.

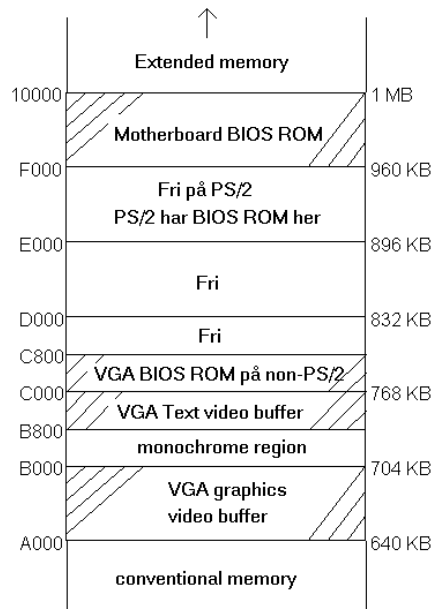
Nogle gange tænker jeg, hvorfor f... visse personer har konstrueret en PC med tilhørende styresystem, memory osv. så besværligt, men det har de altså, og en del af miseren skyldes, at IBM for mange år siden ikke kunne forudse fremtiden.

Adresseområdet fra 640-1024, altså 384 KB, reserverede IBM klogt nok til systemtekniske og fremtidige formål. Dette reserverede adresseområde i memory fra 640-1024 KB kaldes af de fleste for *upper memory*. IBM og Hewlett-Packard kalder det sommetider *reserved memory*. Da begreberne high og upper memory blev indført, var der i branchen lidt begrebsforvirring

Upper memory area er et navn for et *adresseområde*. Der er regler for, hvor i upper memory, de forskellige *devices* skal placere deres *data*, så CPU'en kan have direkte adgang til dem for at kommunikere med dem - skrueret på figuren.

Hele området er imidlertid ikke fyldt op, og den ledige plads kan så benyttes til såkaldte drivere og residente programmer. (*Resident* vil sige, at programmet er permanent i memory og skal derfor *ikke* kaldes fra harddisk, hver gang det skal køres. Det ligger og lurder hele tiden. Et resident program kaldes også TSR (Terminate and Stay Resident)).

Det interessante er, at området ligger inden for DOS' og real modes rækkevidde, DOS kan 'nå' dem og kontrollere/kommunikere med dem, og *derfor* kan DOS drivere og DOS residente programmer køre i dette *adresseområde*.



Man har fundet ud af at 'kontakte' dette adresseområde, hvilket for DOS vedkommende sker via EMM386.EXE, som kan skabe denne adgang - denne tekst behandler ikke programmer som fx QEMM, 386MAX og Net-Room.

Begrebet upper memory area er *ikke* det samme som de såkaldte blokke, kaldet Upper Memory Blocks (UMB), selv om disse blokke - hvis de skabes (af fx EMM386.EXE) - memorymæssigt befinder sig i upper memorys adresseområde. Upper memory er et navn for et *adresseområde*. UMB er *reel anvendelig RAM* i dette område.

Jeg troede længe, at upper memory var et helt nyt tilgængeligt område, som ikke havde eksisteret før, og som reelt øgede den samlede tilgængelige mængde RAM.

Memory manager for upper memory kan både være HIMEM.SYS eller DOS. I første omgang under boot er det HIMEM.SYS. Det normale er imidlertid, at via DOS=UMB i CONFIG.SYS bliver DOS memory manager.

Figuren viser, hvordan upper memory kan se ud på en PC, og kan måske hjælpe dig, hvis du begynder at finde ud af, hvor i dette område de forskellige programmer lægger sig.

IBM PS/2 har BIOS ROM i de to 'øverste' områder, dvs. i E og F-page. VGA BIOS ROM er placeret fra E000-E7FF. Til gengæld er C000-E000 ledigt til UMBs.

Expanded og extended memory

I det følgende er det vigtigt at skelne mellem den rent fysiske RAM eller memory og de logiske betegnelser (upper, extended osv.) for de måder, hvorpå den fysiske RAM anvendes. Har du fx 4 MB fysisk RAM, kan disse anvendes på mange forskellige måder, og det er denne anvendelse, det følgende handler om. Man taler om at konfigurere memory.

Brugen af memory over 1 MB startede i gamle dage primært med regnearket Lotus 1-2-3, hvor brugerne hurtigt løb mod loftet, de 640 KB, som var grænsen for DOS programmer. De tre firmaer Lotus-Intel-Microsoft udviklede regler for, hvordan såkaldt *expanded memory* skulle benyttes. Denne standard blev kaldt Expanded Memory Specification (EMS) - evt. LIM EMS.

Dengang produceredes ekstra kort med RAM, som kunne sættes i PC'en. Primært ældre programmer kan udnytte denne form for memory, men også mange spil kræver EMS memory (se *Flere Boots* s. 39). En memory manager for expanded memory kaldes Expanded Memory Manager (EMM). DOS har en sådan, nemlig EMM386.EXE.

Senere blev en anden standard introduceret for brug af memory over 1 MB, nemlig eXtended Memory Specification (XMS) - meget uheldigt med de næsten ens navne. Den er mere relevant for de fleste programmer i dag, da nyere programmer primært skrives til at udnytte extended memory, som ligeledes kræver en eXtended Memory Manager (XMM). DOS har en sådan, nemlig HIMEM.SYS.

Disse navne er altså betegnelser for, hvordan den fysiske RAM kan benyttes. Man kan kalde det logiske betegnelser eller standardiserede regler for, hvordan den fysiske memory kan bruges. RAM udover 1 MB kan altså benyttes som en kombination af forskellige former for memory, men først efter, at den relevante memory manager er aktiv og evt. bestemmer, hvor meget der skal bruges til hvad. Expanded memory kaldes også EMS-memory, når den er tilgængelig, konfigureret ifølge nævnte standard.

High Memory Area

forkortet HMA, er defineret som de første 64 KB af extended memory, dvs. fra 1.024-1.088 KB. Ved hjælp af et lille trick er dette område blevet tilgængeligt, som om det tilhørte den første MB. DOS kan benytte dette område. De fleste bruger dette område til at placere en del af DOS vha. DOS=HIGH. Memory manager for dette område er meget naturligt HIMEM.SYS, da det håndterer al memory over 1 MB. HIMEM.SYS styrer altså både high memory og extended memory.

MEM/A er mig bekendt den eneste DOS 6.x kommando, der giver oplysning om udnyttelsen af dette område. Switchen er udokumenteret.

(A) HMA er tilsyneladende 'en del' af extended memory - både ja og nej: HMA er tilgængelig for processoren i *real mode*, mens extended memory er tilgængelig for processoren i *protected mode*.

Forholdet upper-extended

(A) Upper memory 'låner' den reelle RAM af extended memory.

I jargon siger man, at extended memory bliver 'mapped' til upper memorys adresseområde. At *mappe* vil sige, at CPU'en snydes til at tro, at bestemte adresser i memory ligger et bestemt sted, hvad de egentlig ikke gør set i forhold til, hvor den fysiske RAM befinder sig.

Konkret: vil du have adgang til fx 92 KB upper memory, lånes de af extended memory, som dermed får 92 KB mindre. Prøv at skrive *REM* forrest i linien i CONFIG.SYS med EMM386, boot og se resultatet af MEM. Slet *REM* igen, boot og se MEM. Summen af upper memory + XMS er konstant!

Device

Enhver af disse former for memory undtagen conventional memory er et device på samme måde som skærm/tastatur, printer osv, og derfor kræver disse memoryformer en devicedriver på samme måde som andre devices gør.

Der findes både rent fysiske og, hvad jeg vil kalde for logiske devices. Skærm/tastatur er i sig selv ikke et device, men det logiske begreb *con* eller *consol* er et device, som så henviser til to fysiske genstande, en inputdel og en outputdel.

Begrebet *consol* stammer fra tiden med de originale gamle mainframe computere. De havde ingen separat skærm eller tastatur, men fik input og udskrev resultatet til en slags telex maskine. Denne maskine blev kaldt en *consol* og blev altså brugt til både input og output.

Startfilerne

Startfilerne CONFIG.SYS og AUTOEXEC.BAT sætter PC'en op på en bestemt måde, afhængig af indholdet i disse filer. DOS 5 og 6 er grundlæggende ens, hvad angår disse filer. Jeg har valgt først at omtale DOS 5 med enkelte henvisninger til DOS 6, som så beskrives til sidst under ét. De følgende eksempler skal du kun betragte som forslag. Filerne skal befinde sig i rod-directory på det drev, hvorfra der bootes, typisk C:\

Når du installerer programmer, foreslår programmerne ofte at ændre noget i startfilerne, og så er det altså rart at vide lidt om, hvad de forskellige linier i startfilerne betyder.

CONFIG.SYS indeholder typisk *kald* af device-drivers (programmer, der håndterer devices).

AUTOEXEC.BAT indeholder DOS kommandoer, som automatisk bliver udført, hver gang computeren startes. Det *er* DOS kommandoer, blot lidt specielle for det meste, dvs. de kunne lige så godt have været skrevet i DOS.

For begge startfiler gælder: skrives `REM` eller `rem` efterfulgt af mellemrum i starten af en linie, fortolkes linien som en bemærkning (remark) og ikke som en kommando.

Der er ingen forskel på store og små bogstaver. Husk endelig inden du begynder at rette i CONFIG.SYS at have en boot diskette, som *virker*. Jeg har valgt eksempler med én codepage, men kommentarerne beskriver, hvis du ønsker to codepages.

Codepage er en pudsigt betegnelse, som beskriver en samling af 256 forskellige (et sæt af) tegn, som du ser på skærmen. Hver codepage har et nummer.

De vigtige filer

HIMEM, EMM386, SMARTDRV, RAMDRIVE og MEM er vigtige filer (her kun angivet ved deres fornavn) i DOS 5, Win 3.1, DOS 6 og Win 3.11. Rækkefølgen af disse versioner af DOS og Windows er bevidst angivet og i teksten behandlet i *kronologisk* rækkefølge efter deres udgivelse. SMARTDRV er speciel. Dette disk cache program er blevet ændret meget og har derfor også fået sit eget kapitel.

DOS 5

22-03-91

HIMEM.SYS, EMM386.EXE, SMARTDRV.SYS og MEM.EXE. Alle filer befinder/befandt sig typisk i C:\DOS. Det vigtige er, at SMARTDRV.SYS *kun* er en devicedriver, som *kun* kan benyttes i CONFIG.SYS. SMARTDRV.SYS blev kaldt version 3. MEM.EXE, som kan vise brugen af de forskellige former for memory, findes i denne version af DOS.

Windows 3.1

10-03-92

leverede en ny og klart forbedret version 4.0 af SMARTDRV, som nu var blevet til en EXE-fil - SMARTDRV.EXE. Den kunne nu indsættes i AUTO-EXEC.BAT. HIMEM.SYS og EMM386.EXE var ikke meget, hvis overhovedet, forbedret. Med Win 3.1 introduceredes et fint program MSD.EXE, som kan kigge i memorys første MB. Alle filer er typisk placeret i C:\WINDOWS.

DOS 6.0

10-03-93

leverede samme vigtige filer som Win 3.1. EMM386.EXE og SMARTDRV.EXE ver. 4.1 er forbedret. Alle filer er (igen!) placeret i C:\DOS. Pas derfor på, hvis du installerer Win 3.1 *efter* DOS 6 og samtidig beder Win 3.1's setup om automatisk at ændre startfilerne. På denne måde benyttes nemlig ikke de nyeste versioner fra DOS 6. Hvis du har gjort dette, ret startfilerne, så de omtalte filer læses fra C:\DOS og ikke fra C:\WINDOWS.

DOS 6.2

30-09-93

eneste ændring var SMARTDRV.EXE ver. 5.0

Windows 3.11

01-11-93

(Windows for Workgroups), leverede samme versioner af de vigtige filer som DOS 6.2.

Angiv i startfilerne de nyeste versioner af de vigtige filer. Når du har fundet ud af det, så slet de gamle, hvorved du ikke kun sparer plads på harddisken, men du undgår også misforståelser. I eksemplerne senere er angivet to prikker . . for det relevante directory og ikke det i manualer ofte benyttede [path]

Det anbefales at læse de relevante kommentarer til CONFIG.SYS igennem, *inden* du begynder at rette.

Læsevejledning

Disse startfiler er primært rettet mod såkaldte *standalone* PC'er, dvs. ikke tilsluttet et netværk. Jeg har ingen erfaring med netværk, hvor der kan være specielle drivere, der påvirker memory og evt. andre devices.

Af typografiske grunde er de lange linier fra startfilerne i de efterfølgende kommentarer forkortet. Linier med REM er muligheder, som du måske ikke har brug for, men for at vise min anbefalede placering er de medtaget.

Hvis du undrer dig over, at jeg *ikke* angiver C : foran de fleste directorynavne, skyldes det, at startfilerne derved lettere kan kopieres til boot-diskette med få rettelser (se *Boot disketter* s. 49).

Jeg har givet to eksempler

1. DOS 5 med SMARTDRV.SYS
2. DOS 6, Windows 3.1 eller 3.11

Se kommentarerne, hvor jeg også angiver mulige linier, som *ikke* optræder i eksemplerne. Hvis du *ikke* har brug for EMS memory er mine *eksempler* og *forslag* til startfiler sådan:

DOS 5 - SMARTDRV.SYS**CONFIG.SYS**

```
device=\DOS\himem.sys
dos=high
device=\DOS\emm386.exe noems
dos=umb
devicehigh=\DOS\smartdrv.sys 512 256
devicehigh=\DOS\display.sys con=(,,1)
rem devicehigh=\UTI\gmouse.sys
devicehigh=\DOS\ansi.sys
rem devicehigh=\DOS\setver.exe
country=045,865,\DOS\country.sys
files=30
rem fcbs=1
buffers=5
rem stacks=9,256
break=on
rem shell=\DOS\command.com \DOS /p /f /e:1024
```

AUTOEXEC.BAT

```
@echo off
path C:\BAT;C:\DOS;....
mode con cp prep=((865) \DOS\ega.cpi)
mode con cp sel=865
LH keyb.com dk,,\DOS\keyboard.sys
rem LH \DOS\doskey.com
LH \DOS\share.exe
rem LH \UTI\gmouse.com
set temp=c:\temp
set tmp=c:\temp
prompt $p$g
rem \DOS\emm386.exe auto
```

Windows 3.1x eller DOS 6 - SMARTDRV.EXE**CONFIG.SYS**

```

device=..\himem.sys
dos=high
device=..\emm386.exe noems
dos=umb
rem devicehigh=..\cdmke.sys /d:mscd01
rem devicehigh=..\ramdrive.sys 2048 /e
devicehigh=\DOS\display.sys con=(, ,1)
rem devicehigh=\UTI\gmouse.sys
devicehigh=\DOS\ansi.sys
rem devicehigh=\DOS\setver.exe
rem device=..\smartdrv.exe /double_buffer
country=045,865,\DOS\country.sys
files=40
rem fcbs=1
buffers=5
rem stacks=9,256
break=on
rem shell=\DOS\command.com \DOS /p /f /e:1024

```

AUTOEXEC.BAT

```

@echo off
path C:\BAT;C:\DOS;C:\WINDOWS;C:\UTI;....
rem LH MSCDEX /D:MSCD01 /M:20
..\smartdrv.exe
mode con cp prep=((865) \DOS\ega.cpi)
mode con cp sel=865
LH \DOS\keyb.com dk,,\DOS\keyboard.sys
rem LH \DOS\doskey.com
rem LH \UTI\gmouse.com
LH \DOS\share.exe
set temp=c:\temp
set tmp=c:\temp
prompt $p$g
rem win :

```


Bootprocessen

(A) Under boot sker bl.a. følgende: efter at en ROM chip checker hardware (memory, porte, grafikkort osv.) og indlæser oplysninger fra ROM BIOS starter et program. Først søges efter systemfiler, som hos MS-DOS hedder IO.SYS og eee MSDOS.SYS.

På dette tidspunkt kendes de forskellige devices ikke. Ethvert device kræver et program, altså software, som kan styre dette device. Når dette er sket, er det pågældende device så at sige indlemmet i familien af alle de dele i din PC, som kan kommunikere med hinanden. Hvis alle disse konfigureringer lykkes, er styresystemet DOS klart.

DOS 5: herefter kan PC'en under boot ikke låse, uanset indholdet i AUTOEXEC.BAT. Derfor er det vigtigt i det mindste at have en CONFIG.SYS, der ikke laver fatale fejl. Når CONFIG.SYS er læst færdig, er en boot vellykket.

DOS 6: her er det (næsten) muligt at springe CONFIG.SYS over ved at trykke F5 i starten af boot - se under DOS 6.

Ved start er conventional memory tom. Først indlæses systemtekniske ting fra ROM. Dernæst gennemløbes CONFIG.SYS to gange. Første gennemløb søger linien DOS=HIGH og tilsyneladende også DOS=UMB. Derfor er placeringen af disse to linier ligegyldig. Hvis andet gennemløb finder HIMEM.SYS, startes med at indlæse HIMEM.SYS i conventional memory, derefter en evt. EMM386, derefter DOS i high memory, derefter evt. DOS=UMB, og så går det ellers derudaf med at læse drivere og andre programmer i RAM.

CONFIG.SYS

Bortset fra linierne `device=` bliver linierne læst i følgende rækkefølge, *uanset* hvor de placeres. Først indlæses gruppen af `device (high)=` i den rækkefølge, de optræder. Derfor er denne rækkefølge vigtig mht. udnyttelsen af upper memory! Dernæst indlæses (også selv om linien ikke findes!) `FILES= FCBS= BUFFERS= LASTDRIVE=` og `STACKS=`.

Sidst indlæses programmet `COMMAND.COM`, der fortolker DOS kommandoer, evt. via en `SHELL=`

Drivere kan heldigvis fungere på mange forskellige måder, dvs. få et device til at opføre sig på forskellige måder. DU bestemmer, hvor meget af din RAM, der skal bruges til de forskellige ting. Man angiver i linien ved et kald af en device driver én eller flere såkaldte *parametre*, en slags variabel, samt evt. såkaldte *switches*. En *switch* kan være enten ON eller OFF. En *parameter* angiver som regel det objekt, der handles på, og i denne forbindelse et device, og er en værdi (ikke nødvendigvis et tal), som kun kan antage én af flere mulige værdier. Parameteren *ugedag* kan kun antage én af syv forskellige værdier.

Dette er både godt og ondt. Hvis man kan finde ud af alle de forskellige kombinationsmuligheder og deres indbyrdes samspil, så er det fint, men som mange ved - det er et rent h.. og derfor forsøges med denne tekst at hjælpe.

DEVICE=\. . \HIMEM.SYS

HIMEM.SYS skal placeres *før* EMM386.EXE. HIMEM.SYS er memory manager for både high memory og extended memory.

(A) High og extended memory *eksisterer* først nu i en form, der kan bruges (tilgængelig). Både high og extended memory har samtidig fået sig en memory manager, hvilket betyder mulig adgang til og brug af high og extended memory. Hermed konfigureres al memory over 1.088 KB til *available XMS memory*, såfremt `DOS=HIGH` benyttes.

DOS=HIGH

Betingelse er, at HIMEM.SYS er aktiv. `DOS=HIGH` betyder, at DOS lægger så meget som muligt af sig selv i high memory. Dette er en af de vigtigste ting ved DOS fra ver. 5, nemlig at der frigives mere plads i conventional memory til at køre programmer. High memory *kan* benyttes til andet.

286

Hvis du har en 286 med min 1 MB RAM, kan du måske udnytte high memory og/eller upper memory. For at udnytte upper memory skal du for det første have speciel hardware (nogle 286 har) og for det andet skal du have en upper memory manager. Jeg har desværre ingen erfaring med en upper memory manager til 286. Du kan selv teste, om din PC kan udnytte high memory. Skriv disse to linier først i CONFIG.SYS

```
device=\. . \Himem.sys
DOS=HIGH
```

og se, hvad den siger, når den booter. Hvis du ikke kan nå at se det, skriv `MEM`, når den har bootet færdig. Sidste linie skal hedde

```
MS-DOS resident in high memory Area
```

Hvis ikke, kommer meddelelsen

```
unable to control A20 line
```

Prøv med `..Himem.sys /machine:11`

Det sidste er en såkaldt maskin-identifikation. Gyldige tal er 1-14 (se manual s. 611) eller prøv dig frem, start evt. med 11, 12 eller 13.

Jeg har fået det til at virke på nogle maskiner. Hvis det lykkes, har du i hvert fald fået installeret DOS i high memory, hvilket gør din PC hurtigere, fordi du får mere conventional memory fri. Brug `MEM` eller `MEM/C/P` til at kigge. Du har fået mere hukommelse til rådighed til programmer. Hvis du ikke kan bruge upper memory, skal du skrive `device=` i stedet for `devicehigh=` samt udelade `LH` i `AUTOEXEC.BAT`.

386

.. EMM386.EXE ..

EMM386.EXE kan *kun* benyttes på en PC, der har minimum en 386 processor, og er en større videnskab med et væld af muligheder, og ikke alle omtales, men den intensive behandling skyldes, at det er en vigtig fil mht. udnyttelse af memory.

Programmet har to basale funktioner. Det kan være memory manager for evt. installeret EMS memory og kan *skabe* adgang til upper memory.

EMM386.EXE kan både anvendes som *driver* og som *DOS program*. Det har vi ikke været vant til, øv! Det er usædvanligt med en devicedriver, som samtidig kan fungere som et program, der kan aktiveres fra DOS prompten. Og manualen .. ak.

Inden denne linie under boot er al RAM over 1.088 KB konfigureret som extended memory. HIMEM.SYS er i første omgang memory manager for upper memory, men hvis DOS=UMB er angivet, afgiver HIMEM.SYS kontrollen over upper memory til DOS med EMM386's hjælp.

Først *efter* indlæsning af HIMEM.SYS og EMM386.EXE kan der indlæses programmer i upper memory via devicehigh og loadhigh.

LH er en forkortelse for LoadHigh, hvilket betyder: load (indlæs) i upper memory. LoadHigh burde have heddet LoadUpper og devicehigh på samme måde - begrebsforvirringen er fuldkommen. Forsøger du at indlæse et program/driver i upper memory via en devicehigh eller LH og er der *ikke* plads i upper memory, sker der blot det, at der indlæses i conventional memory. Du får ingen fejlmeddelelse.

device= .. EMM386.EXE noems

Vi starter med den mulighed, som er relevant for de fleste, nemlig parameteren NOEMS, hvilket betyder, at en del af extended memory *ikke* skal konverteres til EMS memory. NOEMS betyder *også samtidig*: 'giv adgang til upper memory'. Du bruger altså dette, hvis du ikke har brug for EMS memory, og samtidig ønsker at bruge upper memory samt ønsker al ledig RAM over 1.088 KB brugt som XMS memory.

Teknisk: under DOS 5 meldte min PC 92 KB total available upper memory og 92 KB som largest block, startende ved address C800, hvis jeg blot angav NOEMS.

... I=E000-EFFF

(A). Irrelevant for PS/2. Denne linie er en *parameter*, som er uafhængig af andre parametre og kan stå i linien med EMM386.EXE. Den inkluderer et adresseområde i upper memory og kræver, at området ikke er optaget af andet systemteknisk som fx motherboard BIOS.

Et område i memory defineres med en start- og en slutadresse. Som regel bruges 16 talsystemet og i denne sammenhæng 4 cifre ad gangen (det sidste 0 er strøget). Eksempelvis betegnes de 64 KB i området fra 896-960 KB E000-EFFF. EFFF er adressen lige før F000. Der tales om A-page, B-page osv. for hvert område a 64 KB. 640 KB=A000 osv.

EMM386 fra DOS 5 inkluderer default *ikke* dette område, fordi PS/2 har placeret BIOS ROM her, mens EMM386 fra DOS 6 gør.

Hvis du vil *ekskudere* et område i upper memory - for at være sikker på, at intet program bruger det, anføres dette på lignende måde, blot med et X i stedet for I - ex

... X=B000-B7FF

som ekskluderer det område, der kaldes *monochrome region*, altså dér, hvor CPU'en kommunikerer med et monokromt (sort/hvidt) videokort. Se også MEMMAKER s. 38. Det er dog mere relevant at *inkludere* dette område end at *ekskudere* det.

Kræver EMS memory

Visse programmer kræver EMS memory.

Det vigtige for forståelsen af EMS og upper memory er, at de fleste EMS krævende programmer fordrer, at der bliver brugt 64 KB *sammenhængende* UMB til en såkaldt *page frame*.

Hvad angår EMS memory omtales specielt to parametre, nemlig RAM og AUTO, fordi manualen kun meget kort nævner dem og endda efter min mening forkert med AUTO-switchen - s. 482 og s. 605-609. HELP i DOS 6 er også forkert. Se under AUTOEXEC.BAT, hvor AUTO hører til!

Lad os sige, du har i alt 4 MB RAM. Du har så 3 MB extended memory efter installation af HIMEM.SYS - jeg snyder lidt med de 64 KB i high memory, men det er lidt lettere. EMM386.EXE kan konvertere extended memory til EMS memory. Hvis du vil bruge fx 1 MB=1024 KB af dine 3 MB XMS til at være EMS memory (og beholde de resterende 2 MB som XMS memory) samt have adgang til upper memory skrives.

... RAM 1024

RAM betyder 'giv adgang til upper memory'. 1024 betyder 'brug max 1024 KB til EMS memory'. Hvis du blot angiver et tal, får du *ikke* adgang til upper memory.

... I=E000-EFFF FRAME=E000

(A). Se tidl. om I=E000-EFFF.

Irrelevant for PS/2. EMS memory skal ifølge LIM EMS reglerne version 3.2 eller 4.0 arbejde med en såkaldt *page frame*, som her ikke beskrives uddybende. Men de fleste programmer, der kræver EMS, kræver en page frame - et 'vindue' i upper memory ud mod en del af EMS memory, og det er så vigtigt at disse 64 KB er ledige i upper memory, ellers snappes de fra conventional, og det er en pæn portion at give afkald på.

EMM386 i DOS 5 vælger uheldigvis default startadresse for en page frame ved D000 - pga. PS/2, som ved denne adresse starter sin motherboard BIOS ROM. Hvis du installerer EMS memory med fx parametrene RAM 1024, vil du sandsynligvis under boot se startadressen ved *starting at address*

Parameteren FRAME=E000 bestemmer startadresse for page frame, dvs. overlader det *ikke* til EMM386 at finde startadressen. Ovennævnte ex. udnytter upper memory bedre.

EMM386 i DOS 6 er forbedret, thi her testes tilsyneladende i upper memory af MEMMAKER, om E000-EFFF (den såkaldte E-page) er ledig, og hvis ja, vælges E000 som startadresse. Fint. Sådan er det i hvert fald på min PC.

... FRAME=NONE

(A) Måske har dit program, der kan bruge EMS memory, ikke brug for en page frame (dette gælder de færreste programmer, der kræver EMS memory). Men hvis dette er tilfældet, og du hellere vil udnytte de 64 KB i upper memory til andre programmer, kan du angive denne parameter. Bruger du DOS extended programmer - programmet udvider selv memory over DOS' normale grænser - som fx AutoCAD386 eller Lotus 1-2-3 ver. 3.x, må du checke med programmets manual, om du med fordel kan angive dette.

FRAME=NONE er udokumenteret i DOS 5, mens DOS 6 nævner den med bemærkningen, at dette kan forårsage, at visse programmer, der kræver EMS memory, ikke fungerer ordentligt.

I rammen er angivet nogle eksempler på linien med EMM386.

DOS 6: se EMM386 s. 38.

Slut på EMS memory. Slut på EMM386.EXE

DOS=UMB

betyder, at DOS overtager håndteringen af upper memory. DOS beder HIMEM.SYS om at få kontrol over upper memory. Man *kan* skrive linierne

```
DOS=HIGH og DOS=UMB som DOS=HIGH, UMB
```

men da det handler om to forskellige kommandoer, har jeg i eksemplet skrevet dem som to separate linier.

REM DE..IGH=\..\CDMKE.SYS /D:MSCD01

Her er en driver fra et Panasonic CD-ROM drev, som jeg kun aktiverer, når jeg har brug for det - det fylder 11 K. Navnet efter /D: er det navn, som CD-ROM-drevet kan kendes på af MSCDEX, som kaldes i AUTOEXEC.BAT. Se

under MSCDEX s. 21.

..SMARTDRV.SYS 512 256

Dette er kun relevant, hvis du *kun* har det ikke så gode disk cache program SMARTDRV.SYS fra DOS 5. Hvis du ikke ved, hvad en disk cache er, så læst først den *generelle* beskrivelse s. 23, og vend tilbage hertil.

Tallene angiver antal KB for begyndelsesstørrelse og minimumstørrelse af cachen. Du er frit stillet, om du vil angive første og evt. andet. De optimale tal afhænger af din available XMS memory. Første tal har default 256.

Sidste tal er relevant for Windows. Visse programmer kan nemlig finde på at ændre minimumstørrelsen og evt. nulstille den for selv at bruge dette område i memory.

Hvis du vil undgå dette, skal du angive minimumstørrelsen ved det andet tal. Jeg forudsætter, at du *ikke* har installeret expanded memory. Ovenstående er et eksempel med 2 MB total RAM, NOEMS ved EMM386.EXE, dvs. 960 KB available XMS memory. Når du har brugt 512 til SMARTDRV, har du altså 448 tilbage, som kan bruges til programmer, der kan udnytte extended memory. Hvis du ved, at dine programmer ikke kan dette, kan du sætte første tal højere og se, om det går hurtigere.

Hvis du har 4 MB total RAM skriv 1024 512. Du *behøver* ikke at bruge disk cache, men det tilrådes.

(Først med version 4.0 fra Win 3.1 blev SMARTDRV acceptabel, og med SMARTDRV 4.1 fra DOS 6 god. Har du SMARTDRV.EXE, er du bedre kørende og så skal den ikke placeres i CONFIG.SYS, men i AUTOEXEC.BAT. Du kan godt bruge SMARTDRV.EXE 4.0, selv om du kun har DOS 5 - se s. 23.)

..RAMDRIVE.SYS 2048 /E

Relevant, hvis du har min. 6 MB RAM og bruger programmer, der arbejder intenst med temporære, midlertidige filer, hvilket mange Windows-programmer gør.

Danner en såkaldt RAMdisk, egentlig RAMdrev, deraf navnet - IBM plejer at kalde det virtuel disk. Under boot vil du på skærmen efter *virtual disk* se drevnavnet, som vælges - næste ledige drev.

Den ved tallet angivne del af RAM opfører sig som et drev. I eksemplet afsættes 2048 KB af *extended* memory - angivet ved /e.

Ønsker du at bruge *expanded* memory skrives /a

Hvis computeren låser eller strømmen går, risikerer du at miste ramdiskens data. Fordelen ved en ramdisk

```
device=\..\emm386.exe ram 1024
device=\..\emm386.exe i=e000-ffff frame=e000 ram 2048
device=\..\emm386.exe i=e000-ffff x=b000-b7ff noems
device=\..\emm386.exe i=e000-ffff noems
```

kan ved intens brug af temporære filer være hurtigere adgang til data i RAM - der skal i så fald ikke kommunikeres med harddisken. Jeg har med mine programmer ingen gavn af ramdisk, men du kan jo prøve.

..DISPLAY.SYS CON=(, , 1)

er driver til consollen. Den *første* parameter, som angiver typen af grafikkort/skærm, behøver ikke at angives, da DISPLAY.SYS checker grafikkortet.

Den *anden* angiver nummeret på den hardware-understøttede codepage for consollen. Typisk er grafikkort default konstrueret til USA, altså codepage 437. Så vidt mine erfaringer behøver den ikke angives her, hvis man senere i AUTOEXEC.BAT bruger `mode con cp prep` og `mode con cp select`, hvor denne codepage netop vælges (og det gør de fleste jo i Danmark - det pæne ø og Ø). Derfor har jeg udeladt den her. Hvis din hardware, dvs. grafikkort, ikke understøtter 865 eller 850, skal du angive den. For at se dette, skriv i DOS

```
c:\>mode con
```

Den *tredie* parameter angiver antal codepages, der skal reserveres plads til i memory udover den i anden parameter valgte, men stadig kun for CON. Default er 1. Hvis du kun vil bruge én codepage anderledes end 437, skriver du 1 eller ingenting til sidst.

Det hele er forudsætning for `MODE CON CP prep` og `select` i AUTOEXEC.BAT.

To codepages: Skriver du 2 til sidst, har du mulighed for at kunne skifte mellem fx 865 og 850. Betingelse er at du forbereder to codepages (se `mode con ..`) og at NLSFUNC indlæses i AUTOEXEC.BAT. Du kan så skifte codepage med kommandoen `CHCP 865` eller `CHCP 850`. DISPLAY.SYS fylder ca. 8 K.

Se kommentar under buffers s. 19, hvor mange buffers der kan indlæses i high memory.

..SETVER.EXE

Nogle, typisk ældre, programmer kommunikerer med DOS versionsnummeret og kræver, at dette nummer er tilgængeligt i memory. SETVER klarer dette ved at lyve om versionen for programmet. Skriv SETVER for at se. Fylder 432 bytes.

..GMOUSE.SYS

indlæser en driver for mus til DOS programmer. Windows har sin egen driver, men kun til Windows programmer. Dette er eksempel med en Genius mus. Din

musedriver hedder måske noget andet og ligger måske i et andet directory.

..ANSI.SYS

er driver til tegn på skærmen, markørbevægelse, samt definition af tasterne. Nødvendig, hvis du i DOS vil anvende andet end default 25 linier med 80 tegn i hver linie, ændre farver på DOS skærmen m.v. Fylder ca. 4 K.
rem ..smartdrv.exe /double_buffer
Se *doublebuffer* s. 25

COUNTRY=045,865,..

Jeg har i eksemplet valgt en bruger, som ønsker codepage 865 som *active* (generel) codepage.

Det første tal, i dette tilfælde 045, angiver danske formater for tid, dato, valutasymbol, sorteringsorden i filer, tegn brugt i DIR- og filnavne. Anden parameter sætter active codepage, den generelle active codepage for alle devices, der understøttes af dette.

Hvis du lader anden parameter blank, vælger du på dette tidspunkt under boot active codepage 850 (fordi 850 for Danmark er default, mens den alternative er 865) og ønsker du 865, bliver du nødt til senere i AUTOEXEC.BAT at indlæse `MODE CON CP prep 865`, `NLSFUNC` samt vælge codepage med `CHCP 865`.

Se evt. *Codepage diskussion* s. 23.

FILES=40

Default (hvis linien ikke findes) er 8. Gyldig: 8-255. Angiver, hvor mange filer, der kan være åbne samtidigt. En fil er 'åben', når den bruges eller læses. Mange nyere programmer har mange filer åbne samtidigt. Bruger du Windows og kører flere programmer samtidig, kræves et højere tal.

Der bliver brugt lidt plads (64 bytes for hver fil) i hukommelsen til dette. Prøv med 30, 40 eller 50 og se, hvordan det går. Større kartoteker med såkaldte referenc kartoteker har ofte mange filer åbne. Jeg undrer mig over, at Microsoft ikke leverer et program, som fortæller antal åbne filer.

REM FCBS=X

File Control Blocks er en gammel måde at få adgang til filer. FILES= er den moderne måde. Visse ældre programmer som fx SideKick bruger denne gamle metode. x angiver, hvor mange af disse blokke, DOS kan have åbne på samme tid. Default er 4.

Hvis du ikke ved, om du har et program, der kræver dette, kan du prøve at angive tallet 1, og får du ingen fejlmeddelelser, har du sparet lidt memory.

BUFFERS=5

er et mini disk cache program (se evt. s. 23 om disk cache). Hvis du bruger SMARTDRV, anbefales at sætte antallet af buffers lavt, fx 6. Hver buffer bruger ca. 532 bytes. Bruger du ikke et disk cache program, kan du eksperimentere lidt. Prøv værdier mellem 6 og 20, og find ud, hvornår din PC kører bedst. Du får ingen fejl, blot lidt hurtigere eller langsommere PC.

Størsteparten af buffers skulle ifølge Microsoft indlæses i high memory, hvis der er plads her, men jeg kunne ikke via MEM/C/P eller MEM/D/P se, at dette skete. Senere fandt jeg årsagen: hvis DISPLAY.SYS indlæses (og det gør alle ikke-engelsk talende for at kunne bruge de for landet specielle bogstaver) sker der et eller andet i high memory. Jeg tror, at de codepages, der via DISPLAY.SYS reserveres plads til i memory, optager en del af high memory.

Min test viste: indlæses DISPLAY.SYS og sættes sidste parameter (antal codepages) til 1, er resultatet: op til og med 27 buffers går fint i high memory og kun 512 bytes bruges i conventional. Ved 28 buffers ryger alle buffers i conventional memory. En anden pudsighed: ved buffers=24-27 fylder COMMAND.COM 5K, ellers kun 3K. Det optimale ved én codepage er buffers=23.

Sættes sidste parameter i DISPLAY.SYS til 2, placeres op til og med 8 buffers i high memory. Ved 9 buffers går det galt. Ved buffers=6-8 fylder COMMAND.COM 5K, ellers kun 3K. Det optimale ved to codepages er buffers=5. Heldigvis har de færreste brug for mange buffere, da de sikkert benytter en disk cache som fx SMARTDRV, men *det er en fejl, at manual eller HELP ikke oplyser dette*. Amerikanere opdager det sikkert ikke, fordi de ikke har brug for DISPLAY.SYS. (IBM's PC DOS 6.1 og 6.3 har samme problem)

Har du lyst til at teste: efter hver boot bruger du MEM/D/P og kigger efter BUFFERS. Du kan også bruge den udokumenterede MEM/A, som giver info om udnyttelse af high memory.

REM STACKS=9,256

Tallene er et eksempel. Default (dvs. linien findes ikke) for en original IBM PC er 0,0 og for andre 9,128. Angiver hvor meget hukommelse, der afsættes til at håndtere såkaldte *hardware interrupts*.

Interrupt betyder *afbrydelse*. En interrupt er, 'når der er telefon til DOS' midt i en igangværende opgave. Hvis du fx trykker en tast, mens DOS (og dermed processoren)

er midt i noget, bliver det opfattet som en afbrydelse, der ikke kan udskydes. DOS vil derfor øjeblikkeligt lægge det igangværende program på hylden i en såkaldt stack (en lille buffer), mens der bruges tid på afbrydelsen. Når afbrydelsen er færdig, vender DOS tilbage til den tidligere proces.

Eksempler på interrupts er klik eller flytning med musen, afslutningen af en læsning fra harddisken eller en indgående fax. Mange samtidige interrupts kan opbruge antallet af de stacks, som er defineret her, og løsningen er at forøge antallet.

Det første tal angiver antallet af stakke, og gyldige værdier er 0 eller fra 8 til og med 64. Det andet tal angiver, hvor mange bytes hver stak er på. Gyldige værdier er 0, 32, 64, 128, 256 eller 512. Hvis du har extra kort til scanner, faxmodem, modem el. lign. og får fejlmeddelelsen *stack overflow*, kan du prøve at sætte værdierne op og se, om det hjælper. Prøv 9,256 - 9,512 - 10,128 osv.

Hvis du ikke har en original IBM og samtidig ikke har denne linie, bruges memory. Hvis du går efter det optimale, kan du prøve 0,0 - det kan jo være, at ingen af dine programmer kræver disse stakke. Får du problemer skriver du 9,128 - hvilket altså er det samme som at untlade linien, men hvis du har brug for højere tal, har du skrevet den og kan så rette.

Bruger conventional memory. Vi taler her om petitesser i memory, men skal det være ..

BREAK=ON

Forøger det antal gange, som DOS checker, om brugeren har trykket **Ctrl+C** eller **Ctrl+Break** (pause-tasten) for at afbryde et program. Visse programmer er skrevet i forsøg på, at disse tastetryk ikke virker. BREAK=ON bevirker så, at man alligevel kan afbryde med **Ctrl+C** eller **Ctrl+Break**. Har ingen indflydelse på forbrug af RAM, men *kan* nedsætte hastigheden, fordi der oftere skal checkes efter interrupts.


```
shell=.. /p /f /e:1024
```

```
shell=..\dos\.. /p /f /e:1024
```

Nogle har placeret COMMAND.COM i roden og har ingen linie med SHELL, hvilket er OK. Andre har COMMAND.COM i C:\DOS sammen med ovenstående linie, hvor det angives, hvor den skal findes.

Det første er det letteste, da bootprocessen default leder her efter COMMAND.COM. Placering i C:\DOS er fornuftigt, fordi man sjældent arbejder i sit DOS directory, og kommer derfor ikke til at slette COMMAND.COM. Slettes den, har du nemlig ingen kommandofortolker. Redning: boot fra diskette og kopier COMMAND.COM fra diskette til harddisk. Nu kan du igen boote fra harddisk.

/p betyder, at du har specificeret den permanente kommandofortolker.

/f betyder, at Fail automatisk vælges blandt de 3 muligheder ved den ellers opståede fejlmeddelelse Abort, Retry or Fail - typisk, når et program forsøger at læse en diskette i et diskettedrev uden diskette.

/e:1024 - et andet vigtigt formål med denne kommando er at udvide det såkaldte *DOS environment* (en slags opslagstavle i memory, hvor DOS og programmer lagrer og henter visse interne informationer, variable) til det antal bytes specificeret sidst i linien.

Nogle programmer kræver, at en variabel er defineret i dette environment og under installation af et sådant program indsættes eller foreslås en linie i AUTOEXEC.BAT - fx SET LIB=C:\QB. Hvis du i DOS skriver SET og HELP SET, får du en ide, hvad det handler om.

Hvis linien med SHELL= ikke findes eller hvis /e ikke angives, afsættes for DOS 5: 160 bytes og for DOS 6: 256 bytes.

Hvis du får fejlmeddelelsen *Out of environment space*, kan du prøve at forøge antallet i denne linie. Jeg har aldrig haft behov for denne linie til at forøge dette antal bytes, men Microsoft anbefaler mindst 1024 til Win 3.1. Det drejer sig om få bytes, så du bruger lidt i conventional memory.

Har du en lang path op mod 127 tegn, bruges altså 127 bytes til path, og så kan det være relevant at sætte dette tal, og for de fleste skulle 512 eller 1024 være rigeligt.

DOS 6: MEMMAKER indsætter denne linie, hvis den ikke findes, dog uden /e:1024

Andet i CONFIG.SYS

Denne tekst har sine grænser.

REM LASTDRIVE=X

angiver det maximale antal drev, du kan få adgang til. Default er næste ledige drev, dvs. for de fleste D. Du spilder memory ved at skrive Z.

Netværk: visse netværksprogrammer ønsker at skabe drevbetegnelse udover max - i dette tilfælde duer Z ikke.

FASTOPEN

Hvis der i din CONFIG.SYS står noget med FASTOPEN, så anbefales du at slette linien. Ideen med hurtigere adgang til filerne er fin, men der er rapporteret ødelagte filer fra brugere. DOS 6 indeholder også denne fil!

AUTOEXEC.BAT

er en fil, som indeholder DOS kommandoer. Det specielle ved denne fil er, at den aktiveres, når computeren bootes, deraf navnet AUTOMATIC EXECUTION. Betingelsen er, at filen ligger i rod directory på det drev, hvorfra der bootes, typisk C:\. Filen *behøver* ikke at eksistere.

@ECHO OFF

Det første tegn @ kaldet snabel A bevirker at linien eller bogstaverne `echo off` ikke vises på skærmen, men *selve kommandoen* `echo off` udføres.

Kommandoen `echo off` bevirker, at *efterfølgende* kommandoer (til forskel fra *resultatet* af kommandoen) i AUTOEXEC.BAT ikke vises på skærmen under udførsel. Du kan undlade @, som evt. kan skrives med **Alt+64** på nummertastaturet. Hvis denne forklaring lyder kryptisk, kan du prøve at skrive REM i starten af linien, aktivere AUTOEXEC.BAT en gang til, alternativt boote og se forskellen.

PATH C:\BAT;C:\DOS; . . .

Max antal tegn 127! Skriv PATH for at se de inkluderede directories. Semikolon adskiller de enkelte directories. Mange programmer ønsker under installation et directory tilføjet i AUTOEXEC.BAT. Hvis du siger ja til dette, ser du måske en ny linie med %, hvilket betyder 'tilføj dette directorynavn til PATH'. Ret det selv. Slet derefter linien med %.

Hvis C: udelades, ville path'en kun virke, hvis man står på drev C, hvilket ikke er så smart. Du kunne fx ikke udføre en `xcopy` kommando, hvis du stod på A:

rem LH MSCDEX /D:MSCD01 /M:20

Programmet MSCDEX giver adgang til CD-ROM-drevet. Navnet efter /D: skal være det samme, som angivet i driveren i CONFIG.SYS. Det er meget vigtigt, at denne linie kommer *før* linien med SMARTDRV, således at SMARTDRV kan udføre *read-cache* på CD-ROM-drevet. Se nærmere under SMARTDRV s. 23. Når SMARTDRV starter, checker den, om MSCDEX er tilstede, og hvis ja, sætter SMARTDRV pr. default *read-cache* på CD-ROM-drevet.

Hvis du vil se, hvordan memory benyttes, kan du skrive /V i linien, hvorefter dette vises ved boot. /M:20 angiver antal sector buffers.

\. . \SMARTDRV . EXE

I dette eksempel uden parametre for begyndelses- og minimumstørrelsen checker SMARTDRV selv *available XMS memory* og vælger selv disse størrelser, hvilket

virker fint for langt de fleste brugere. For at sikre SMARTDRV plads i upper memory, er det en god ide at placere den tidligt i AUTOEXEC.BAT, da den fylder 26 KB. Se s. 23 *Disk cache Smartdrv*, hvis du vil vide mere.

MODE CON CP PREP=(865) . .

MODE CON CP SEL=865

MODE CON er en *måde* (tilstand at befinde sig i) for consollen. CP er en forkortelse af CodePage. Prep(are) betyder forbered. Filen EGA.CPI indeholder disse grafiske tegn, som vises på skærmen.

Jeg har valgt eksempel med forberedelse for 1 codepage. Den første linie forbereder brug af 1 codepage for consollen. Den anden linie vælger den codepage, som consollen skal bruge, og det vil i praksis sige: hvordan skal ASCII værdier vises på skærmen.

C:\>MODE CON giver info om

- 1 den evt. valgte hardwareunderstøttede codepage for CON - anden parameter i `display.sys` skal være angivet.
- 2 forberedt(e) codepage(s) - styres af `MODE CON CP prep=..`
- 3 den valgte codepage - styres af `MODE CON CP sel=..`

Hvis du i linien med `DISPLAY.SYS` kun har reserveret plads til én codepage, forbereder og vælger du kun den ene codepage, 865 eller 850. Har du valgt 2, skrives i første linie

```
.. prep=((865 850) ..
```

Se litt. for forskellen mellem 865 og 850 - typisk copyright tegnet © og »sildebene« samt grafiske tegn til at tegne rammer.

.KEYB.COM . .KEYBOARD.SYS

Skal komme efter `mode con cp select`. Installerer eller konfigurerer tastaturet til dansk layout. Da anden parameter ikke angives, vælges den codepage, som er valgt med `mode con cp select=`

Hvis anden parameter angives specifikt, skal den være identisk med den af `select` valgte.

REM LH . .NLSFUNC.EXE

indlæser et program, der muliggør skift mellem codepages vha. `chcp` kommandoen, fx `chcp 865` eller `chcp 850`. Kun relevant, hvis du vil skifte mellem to codepages, hvilket kræver at de to codepages er indlæst - se *display.sys* i CONFIG.SYS og *mode con ..* i AUTOEXEC.BAT.

rem LH . .DOSKEY.COM

DOSKEY er et smart lille program, som husker de sidste DOS kommandoer, når du er i DOS. Tryk pil op og ned for at bladre i dem. Fylder 4 K.

LH \DOS\SHARE

SHARE.EXE holder styr på, hvilke filer, der er åbne, således at to programmer ikke åbner og evt. retter og lagrer forskellige rettelser i samme fil. To programmer kan dermed ikke *dele* (share) en fil samtidigt. Jeg har prøvet det med forskellige programmer, men ikke alle advarer. Så man kan altså ikke altid stole på, at alle programmer kan finde ud af det.

Det anbefales at bruge SHARE, specielt hvis du bruger Windows. Det fylder kun 6 K. Word for Windows (Winword) 2.0 fx ser gerne dette. Microsoft anbefaler til Winword 6

LH \DOS\SHARE /L:500 /F:5100

hvor /L:500 angiver antallet af 'locks', dvs. hvor mange låste filer den kan holde styr på samtidigt. /F:5100 angiver hvor meget memory (i antal bytes), der skal bruges til at kontrollere, at to programmer ikke bruger den samme fil samtidigt.

Hvis du anvender Win 3.11 og *altid* kører DOS programmer fra Windows, behøver du ikke SHARE.

set temp=C:\temp**set tmp=C:\temp**

tvinger visse programmer, fx Windows, til at bruge det angivne directory til midlertidig (TEMPorary) lagring af filer. Visse programmer bruger variabelen *temp*, andre *tmp*, derfor de to linier.

Det sker, at computeren 'låser', dvs. ikke kan komme videre, og man bliver nødt til at boote for at komme ud af problemet. I så fald efterlades midlertidige filer, som er lette at finde, specielt hvis de befinder sig i dette directory, og filnavnet begynder normalt med en tilde (bølge), og normalt hedder de TMP til efternavn. Slet disse filer regelmæssigt fra din harddisk, men kun når du befinder dig i DOS uden programmer kørende.

rem SET TEMP=D:**rem SET TMP=D:**

Hvis du i CONFIG.SYS har installeret en RAMdisk og samtidig vil bruge denne som det drev (disk), hvor midlertidige filer lagres, kan du angive ramdisken (i dette tilfælde drev D) som dette drev. Hvis dit næste ledige drev er et andet bogstav, skal dette selvfølgelig angives. Under boot vil RAMDRIVE.SYS melde det valgte drev. Husk begge linier.

prompt \$p\$g

DOS 6 indsætter automatisk denne prompt. Du behøver altså ikke skrive den i AUTOEXEC.BAT. Næsten alle bruger denne prompt, som bevirker, at man får at vide, i hvilket drev og directory, man befinder sig i. Dette C:\> kaldes en prompt, fordi computeren viser, at den er klar og *prompter*, dvs. beder dig om at skrive noget.

Kommandoen PROMPT kan også bruges til at redefinere taster eller lave andre farver på skærmen. Den følgende prompt kommando kræver, at du i CONFIG.SYS har installeret ANSISYS som tidligere beskrevet. Hvis du har det, kan du prøve at skrive

C:\>prompt \$e[0;1;37;44m\$p\$g

Den kantede parentes kan skrives med **Alt+91**, hvis du ikke har eller kan finde den på tastaturet. Skriv så CLS. Det giver blå skærm med hvide bogstaver. Hvis du bedre kan lide dette, kan du skrive ovenstående i AUTOEXEC.BAT i stedet for prompt \$p\$g. Hele skærmen bliver blå, når du arbejder i DOS. Hvis du ikke kan lide det, boot computeren eller skriv PROMPT \$e[m\$p\$g - hvilket bringer dig til den 'normale' prompt \$p\$g

rem ..gmouse.com

Jeg havde en Genius mus, og driveren C:\UTI\GMOUSE.COM. Hvis jeg fjernede `rem` indlæste jeg GMOUSE.COM i upper memory. Generelt bruger jeg ikke mus i DOS programmer, da jeg synes at tastaturet er hurtigere.

Musedrivere til *DOS programmer* leveres ofte både med én fil, der hedder SYS til efternavn og én, der hedder COM til efternavn. SYS-filen kan indlæses i CONFIG.SYS. COM-filen kan indlæses i AUTOEXEC.BAT.

Windows har sine egne musedrivere til de forskellige mus, som så virker i *Windows*-programmer. Hvis du vil bruge mus i DOS programmer under *Windows*, skal musedriveren installeres, inden du starter *Windows*. Hvis musen også skal virke i et såkaldt vindue (ikke fuld skærm), skal driveren være minimum version 8.20 fra Microsoft eller kompatibel. Du skal måske også tilføje i filen SYSTEM.INI

```
[NonWindowsApp]
MouseInDosBox=1
```

rem ..Emm386.exe auto

Kun relevant under DOS 5, hvis du har installeret expanded memory. Denne parameter kan angives, når EMM386.EXE bruges som program, dvs. som DOS kommando. Den virker *ikke* i CONFIG.SYS. AUTO betyder, at det antal KB, som i CONFIG.SYS er afsat til EMS memory, kun bruges som sådan, hvis et program beder om det, ellers bruges det som extended memory. Men én gang brugt til EMS, ej tilbage til XMS, kan altså ikke skifte automatisk.

REM WIN :

Mange starter et program i sidste linie, typisk et menuprogram eller *Windows*. WIN mellemrum kolon - springer reklamebilledet over.

set winpmt=Skriv EXIT for at vende tilbage til Windows\$_\$_p\$g

er en god ide, og det skal stå i én lang linie. winpmt er forkortelse af *Windows* prompt, og betyder, at du i et DOS vindue bliver mindet om, at du kører DOS i et DOS vindue under *Windows*. Hvis man glemmer det, kan man komme til at boote, hvilket er uheldigt. \$_ svarer til tryk på [Enter](#), altså linieskift.

Codepage diskussion

Det følgende gælder kun codepage under DOS. *Windows* arbejder med sin egen codepage, som defineres under *Windows* Setup.

Jeg har brugt lang tid på test, læsning af manualen, skrive `chcp`, `keyb` og `mode con` flere steder i AUTOEXEC.BAT under boot.

Jeg har på forskellige PC'ere set mange forskellige varianter af de implicerede linier i startfilerne. De fleste får trods forskel i active cp (skriv CHCP) og cp for consollen (skriv MODE CON) sjældent problemer, men nogle får, fx under *Windows* Setup, hvis de *ikke* angiver anden parameter i country kommandoen

Jeg synes, det er for ringe, at Microsoft ikke har givet informationer til danskerne på dansk om codepage. For det første er DOS 5 manualen sammenholdt med skærmens meddelelser fra ovennævnte kommandoer, der har med cp at gøre, det mest sprogligt og definitions-mæssigt inkonsistente, jeg længe har set. DOS 6 manualen er bedre, men ikke på de kritiske punkter.

Microsoft har tilsyneladende valgt CHCP til at være 'kongen' af kommandoer, der i ét hug kan ændre alle cp definitioner for alle devices. Så vidt mine erfaringer er den anden parameter i country kommandoen lige så god.

Det er himmelråbende, at man skal skrive flere ret komplicerede linier i startfilerne og endda i en bestemt rækkefølge for at definere en codepage eller for at kunne skifte mellem dem. Jeg har forgæves forsøgt at få svar hos Microsoft omkring codepage.

Som nævnt tidligere eksisterer et potentielt problem, hvis man *ikke* angiver anden parameter i country kommandoen i CONFIG.SYS, derfor denne behandling, som er irrelevant for mange. Måske bliver det i fremtiden med større international datakommunikation vigtigere.

Disk Cache Smartdrv

Generelt

Der findes forskellige disk cache programmer, men her beskrives kun SMARTDRV, da den følger med som extra program i DOS. En cache - udtales populært som kontanter - er en buffer, et mellemled, et mellemlager mellem CPU og harddisken. Den primitive cache er `buffers=` i `CONFIG.SYS`.

SMARTDRV bruger default en del af extended memory til sin midlertidige lagerplads. Når CPU'en skal bruge data, kigger den først i cachen for at se, om de stadig skulle befinde sig her, og hvis ja (et hit), så tages de herfra, hvilket kan være mere end 100 gange hurtigere end, hvis data ikke findes i cache (en misser), og CPU'en bliver nødt til at hente data fra harddisken. Dette er at læse/*read* i cachen.

Hvis CPU'en får besked om at lagre fx et dokument, lagres det i 1. omgang i cache-lageret. Dette er at skrive/*write* i cachen.

Hvis du ikke ved om SMARTDRV er installeret på din PC, så skriv

```
C:\>SMARTDRV/S
```

det sker der ikke noget ved. Hvis programmet *ikke* er installeret, vises for det første SMARTDRV's versionsnummer samt de forskellige muligheder, parametre.

Hvis det *er* installeret, vises den første linie med `cache size`: begyndelsesstørrelsen og den anden linie med `cache size while running Windows`: minimumstørrelse, mens Windows kører. Dernæst, hvor mange hits og missere SMARTDRV har haft. Det gælder selvfølgelig om relativt at få så mange hits som muligt, så du kan skrive ovenstående for at se, hvordan det går - også mens du kører Windows. Derefter vises, hvilke drev, hvorpå der udføres `read` og `write cache`.

SMARTDRV.EXE

Version 4.0 og 4.1: Det følgende gælder version 4.0 (fra Windows 3.1) og 4.1 (fra DOS 6.0). SMARTDRV indlæses om muligt i upper memory. Uden angivelse af begyndelses- og Windows-minimumstørrelse finder SMARTDRV selv ud af, hvor meget af *XMS memory*, den vil benytte, hvilket fungerer fint for langt de fleste brugere. Default sættes `read-` og `write-cache` på drev C. Begyndelsesværdien eller størrelsen kaldes af nogle også for disk cachens størrelse, mens der køres DOS programmer.

Ved 4 MB RAM, som jeg havde i starten, snupper 1 MB under DOS og min. 512 KB, når Windows kører. Windows bruger selv extended og ledig upper memory til forskellige ting - Windows har sin egen memory manager! - og Windows og SMARTDRV samarbejder om at

udnytte memory optimalt, derfor muligheden for at angive minimumstørrelse under Windows.

SMARTDRV 1024 1024

Her et eksempel, hvor begyndelses- og Windows-minimumstørrelse specifikt er angivet (angives i KB). Du må eksperimentere for at finde ud af, hvilke tal, der er bedst for din brug. Jeg vil anbefale dig at køre med en indstilling et stykke tid, skifte til en anden indstilling, skifte tilbage igen og så se, om du kan *mærke* forskel. Reelt eller 'objektivt' er der måske forskel, men hvis den ikke *opleves*, kan forskellen ikke være stor - brug så de mindste værdier. Det gælder om at finde proportionerne.

Du bliver nødt til at boote for at sætte nye størrelser! De følgende kommandoer gælder for alle versioner:

```
C:\>SMARTDRV/?
```

oplyser de forskellige parametre.

```
C:\>SMARTDRV C+
```

slår både `read` og `write cache` *til*.

```
C:\>SMARTDRV C-
```

slår både `read` og `write cache` *fra* på drev C.

```
C:\>SMARTDRV C
```

slår `write cache` på C fra, men beholder `read cache`. Hvis du har flere drev på harddisken, ønsker du måske at slå `write cache` fra på et af drevene. Kan blot skrives i DOS, men anbefales at vente indtil al aktivitet på harddisken er standset for at være sikker, at der ikke 'hænger' data i cachen, hvilket naturligt fører til

```
C:\>SMARTDRV/C
```

tvinger SMARTDRV til at lagre alt fra cache, som endnu ikke er lagret på disk. Det er altid en god ting at udføre denne kommando (fx i slutningen af en BAT-fil, der starter et program), specielt hvis du skifter fra det ene program til det andet meget hurtigt.

I DOS 6.0 manualen advares for version 4.1: 'Skriv `Smartdrv/C` i DOS, inden PC'en slukkes, for at være sikker på, at SMARTDRV har lagret alle data på harddisken. Efter afslutning af al diskaktivitet kan du slukke din computer uden risiko.'

Første råd: afslut altid Windows, inden du slukker din PC. Andet råd: inden du slukker din PC, vender du tilbage til DOS prompten. Vent indtil evt. harddiskaktivitet er slut og derefter slukker du din PC.

SMARTDRV 4.2: mange brugere klagede over mistede data, fordi de i god tro slukkede deres PC for hurtigt. Derfor udsendte Microsoft i al stilhed version 4.2 før DOS 6.2, som default stadig som tidligere versioner har `write cache`, men skriver til disk, *inden* der vendes tilbage til DOS prompten efter afslutningen af et program.

Denne facilitet er ny! Hermed er man sikker på ikke at miste data, men betaler så en smule i hastighedsreduktion.

SMARTDRV 5.0: leveres med DOS 6.2 og Win 3.11, som ved en helt ny installation slår write cache fra, skriver altså direkte til disk. Hvis DOS 6.2 installationen finder, at SMARTDRV bruger write cache, beholdes den.

Ver 4.2 og 5.0 indeholder 2 nye switches:

/X slår write cache fra på alle drev, dvs. der skrives direkte til disken.

/N bevirker, at SMARTDRV venter (ligesom i version 4.0 og 4.1) med at skrive til disk, når CPU'en får en ledig stund.

Ver. 5.0 understøtter caching af CD-ROM drev.

Forudsat du kun har et drev på din fysiske harddisk er det sikreste og enkleste at angive følgende i AUTOEXEC.BAT: Version 4.0 og 4.1:

```
C:\..\Smartdrv C
```

(der skrives direkte til disk)

Version 4.2 og 5.0:

```
C:\..\Smartdrv
```

(der skrives til cachen, men til disk inden der returneres til prompten). Win 3.11 installationen sætter /X, altså ingen write cache overhovedet.

I alle tilfælde benyttes read cache (hvilket er det vigtigste til at få PC'en til at køre hurtigere) og i alle tilfælde skrives til disk, *inden* du ser prompten. Ved at skrive SMARTDRV/S i prompten kan du se, hvilken version af SMARTDRV du har.

DoubleBuffering

er, som begrebet antyder, en dobbelt buffer, som visse harddisk controllere har brug for. AT-bus har ikke brug for det, men SCSI og nogle ESDI og MCA controllere har. Linien i CONFIG.SYS

```
device=..\smartdrv.exe/double_buffer
```

gør disse controllere kompatible med den form for memory, som skabes af EMM386.EXE eller af Windows 386 enhanced mode memory. Driveren fylder 2 K i conventional og kan *ikke* indlæses i upper memory. På godt dansk: en PC med en sådan controller har brug for denne linie i CONFIG.SYS. Og hvordan du finder ud af det? Indsæt linien i CONFIG.SYS og SMARTDRV i AUTOEXEC.BAT, boot, og skriv SMARTDRV/S

Hvis der i søjlen med *buffering* står **Yes** i en af linierne, så har din PC brug for linien i CONFIG.SYS. Hvis der ikke står **Yes**, kan du slette linien igen i CONFIG.SYS.

Harddisken

Mange faktorer bestemmer, hvor hurtigt data kan hentes fra disk til CPU, og vi skal se lidt på nogle af dem.

Følgende beskrivelse er ikke helt korrekt, men pædagogisk.

Fileerne på harddisken ligger på en bestemt måde og i en bestemt rækkefølge. Hvis vi forestiller os harddisken som en cirkulær plade med harddiskens læsehoved placeret i midten, kan det sammenlignes med en grammofonplade. Fileerne starter i midten, den ene efter den anden i en lang spiralformet kæde, ligesom musikken ligger i én lang rille.

Forestil dig, at læsehovedet er grammofonens pickup - placeret i midten. Når læsehovedet får besked på at læse en fil, kigges først i en tabel, Fil Allokeringens Tabel, hvori filens fysiske placering på disken findes. Læsehovedet bevæger sig så et stykke udad og drejer disken, indtil begyndelsen af filen er fundet.

Nu er der den forskel på en grammofonplade og en harddisk, at fileerne på harddisken kommer og går. De bliver slettet en gang imellem og nye kommer til. Forestil dig, at du har lagt 100 filer på din harddisk, uden at have slettet nogen. De 100 filer ligger pænt efter hinanden. Nu sletter du 3 filer med disse størrelser: den første var 50 KB stor, den anden 100 KB og den tredje 80 KB, og disse tre filer lå ikke lige efter hinanden. Nu er der 3 'tomme' pladser på disken.

Nu lagres/kopieres en fil. Hvis filen er større end 230 KB, vil DOS placere de første 50 KB på den første tomme plads, de næste 100 KB på anden plads, de næste 80 KB på tredje plads og sidste del af filen bliver placeret efter alle de andre. Filen er blevet opdelt eller *fragmenteret*.

Når filen skal anvendes/læses næste gang, skal læsehovedet på en større rejse på harddisken, og det tager tid. Nu har jeg med vilje forenklet sagen i denne sammenhæng, for i virkeligheden snurrer harddisken konstant, hvilket er nødvendigt for at opnå den hastighed, hvormed der kan læses på disken. Diskettedrev roterer først, når de får besked på det.

Du har sikkert regnet ud, at filer, der ligger tættest på diskens centrum og samtidig er sammenhængende, læses hurtigst. Den vigtigste faktor er, at fileerne er sammenhængende. Det er af mindre betydning, hvor på disken de ligger (jo højere søgetid, desto større betydning). Senere beskrives et program, som kan flytte data rundt på harddisken, samle filers enkelte stumper og placere dem sammenhængende på disken.

Den tid, som læsehovedet er om at få adgang (access) til en vilkårlig fils begyndelse på disken, kaldes *den gennemsnitlige søgetid* eller *accesstime*. De mest solgte harddiske for tiden har en gennemsnitlig søgetid på 10-15

millisekunder. Annoncer angiver som regel kun denne egenskab ved en harddisk, som om det var den eneste, der siger noget om kvaliteten.

Den er vigtig, men andre faktorer såsom diskcache er mere væsentlige. PC Magazine taler i deres tests om 'throughput', et gennemsnit af forskellige faktorer. Windowsprogrammer arbejder med langt større data-mængder, end man gjorde for et par år siden.

Lad os se på en faktor, som *du* kan gøre noget ved for at få din PC til at køre hurtigere. Du kan gøre filerne sammenhængende og placere de filer, som du mest bruger, så tæt på harddiskens midte (logiske 'start') som muligt, og derfor ser vi på et program, der kan dette.

Diskoptimering

Der findes flere programmer, der kan optimere eller defragmentere harddisken. DOS 5 har ikke et sådant. DOS 6 leveres med en mini-version af Nortons SPEEDISK.EXE kaldet DEFRAG. Her beskrives klassikeren SPEEDISK.EXE ver. 6.0/7.0 fra Norton Utilities (en samling af forskellige nyttige programmer, som DOS traditionelt ikke har leveret).

Jeg skrev dette afsnit, mens jeg havde en harddisk med en søgetid på 18 ms, og senere er harddiskene blevet meget hurtigere med søgetider under 10 ms. Diskoptimering har *større* effekt, jo *langsommere* harddisken er.

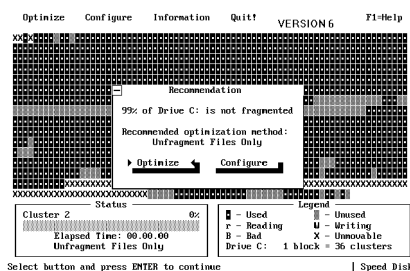
Uanset hvilket optimeringsprogram du bruger, så lav inden du kører programmet sikkerhedskopi af dine vigtige filer. Jeg har aldrig haft problemer med SPEEDISK, men det *kan* gå galt - strømsvigt fx. Det anbefales ikke at have en disk cache aktiv. Hvis du har en sådan i en af startfilerne, så sæt den midlertidigt ud af funktion ved at skrive REM først i linien og boot derefter. Microsoft anbefaler ikke dette for SMARTDRV. Jeg har selv nogle gange 'glem' at sætte SMARTDRV ud af funktion, og det er gået godt hidtil, men hellere være sikker.

1) Du skal helst være i 'ren DOS', dvs. umiddelbart efter en boot uden aktive programmer - kaldes også en *clean boot*. Har du DOS 6, kan du trykke F5 efter boot, når der står *Starting MS-DOS*, så er du sikker på, at ingen disk cache bliver aktiveret - se s. 40

2) Lav som det andet *vigtigere* punkt et check af drev C (eller det drev, du vil optimere) med CHKDSK/F, som reparerer evt. fejl (se s. 48). Hvis du vil checke på et komprimeret drev - DBLSPACE, STACKER eller SUPERSTOR - har disse programmer som regel deres eget checkprogram, som erstatter DOS' CHKDSK. Bruger du DOS 6.2, startes SCANDISK. Bruger du DBLSPACE, skal du benytte DEFRAG.

Det vigtigste ved optimeringen er, at filerne bliver gjort sammenhængende, dvs. at hver fil rent fysisk kommer til at ligge i én lang kæde. Har du aldrig før kørt

et sådant program, vil du sandsynligvis blive positivt overrasket over forbedringen.



Hvis du i 1. omgang ikke har lyst til at eksperimentere med at ændre rækkefølgen af directories og filer m.m., kan du nøjes med at lave en optimering, som kun laver filerne sammenhængende. Hos SPEEDISK hedder det 'Unfragment files only'.

Programmet har mange forskellige valgmuligheder. Du kan fx bestemme rækkefølgen af directories, og inden for hvert directory kan du vælge at sortere filerne efter ét af følgende kriterier: fornavn, efternavn, størrelse, tid og enten stigende eller faldende. Du kan som nævnt vælge 'blot' at optimere de filer, som er fragmenteret, og intet andet.

Det bedste er at planlægge, inden man starter, men det kan du nok lettere, når du får lidt erfaring med programmet. Det kræver nok også, at du har arbejdet med din PC et stykke tid, før du rigtigt får erfaringer med et diskoptimeringsprogram. Når den tid kommer, skal du overveje, hvilke programmer du bruger mest og hvilke directories disse programmers filer ligger i.

Dernæst skal du have en fornemmelse af, hvilke programmer, der læser meget på harddisken. Hvis du ikke har det, så lyt evt. til harddisken, når du arbejder med forskellige programmer. Windows og de fleste Windows-programmer arbejder meget på harddisken.

Min erfaring siger, at tekstbehandlingsprogrammer ikke arbejder særlig intensivt på harddisken, heller ikke tekstbehandling under Windows. Jeg har ikke arbejdet meget med regneark, men store regneark er givetvis aktive, databaser er måske i endnu større grad. Programmer, der arbejder meget med grafik, er krævende.

Du skal altså veje forskellige ting op mod hinanden: hvor tit bruger du programmet og hvor diskaktivt er programmet. Jeg har placeret de diskaktive programmers directories først, hvis jeg bruger dem regelmæssigt og ret ofte, og skifter i øvrigt alt afhængig af opgaverne.

Lad os se på SPEEDISK ver. 6. Selve optimeringen behøver du ikke starte. Vi kigger kun på mulighederne. Hvis du har en mus, så aktiver den. Du starter programmet ved at skrive SPEEDISK i ren DOS. Først spørges, hvilket drev, du vil optimere, typisk C. Programmet checker bl.a. directory-strukturen, men også hvor fragmenteret disken er. Hvis relativt få filer er fragmenteret,

foreslås blot en optimering af disse filer. Du skal *ikke* trykke **Enter** - hvis du kommer til det, tryk **Esc** og svar Cancel. Du kan standse en optimering med **Esc**.

Gå videre ved at vælge `configure`, dvs. højrepil og **Enter** eller klik med din mus. Programmet viser din disk, opdelt i mindre enheder. Du kan andetsteds på skærmen se, hvilke enheder, der er brugt og hvilke, der er ledige. Krydser er filer, som ikke må flyttes (unmovable files) som fx DOS systemfiler og evt. Windows *permanente swapfile* - beskrives under afsnittet *Windows*. SPEEDISK flytter *ikke* disse filer.

Jeg gennemgår kun de vigtigste muligheder. Det vigtigste valg er metoden, som SPEEDISK bruger. Vælg under **Optimize** i øverste menulinie punktet **Optim. Method**. Med tastatur flytter du med pile og af-/markerer med mellemrum.

Full optimerer alle filer, men tager ikke specielt hensyn til rækkefølgen af filer og directories. **Full with File reorder** bruger jeg selv. Den placerer filerne i den rækkefølge, som bestemmes under **Directory Order** under **Configure**, som beskrives om lidt. **Unfragment Files Only** er den hurtigste måde til at få sammenhængende filer.

Pil hen til **Configure** og vælg **Directory Order**. Den er lidt besværlig at betjene med tastatur. Til venstre har du din directory-struktur. Til højre har du den valgte rækkefølge, som SPEEDISK bruger, hvis du vælger en metode, som skal tage hensyn til rækkefølgen af directories. Hvis du som **Optim. method** vælger **Full with File reorder** flyttes de under **Directory Order** valgte directories filer 'forrest' på disken.

Fejl! Ugyldigt filnavn.

Tab springer mellem vinduerne, og pilene flytter også markøren. Bemærk, at du kan markere et directory til højre, og derefter flytte det til en anden prioritet. **Enter** fungerer forskelligt, alt afhængig af, hvad du laver. Det drillede mig i begyndelsen, så derfor er det lettere, hvis du bruger mus. Eksperimentér lidt, så finder du ud af, hvordan du får skrevet de directories, som du vil vælge.

Vælg **Save Options** for at gemme. Under **File Sort** kan du vælge, hvordan filerne i hvert directory skal sorteres. Hvis du har mange eller meget store filer i et eller flere directories er det relevant, ellers spiller det en mindre rolle i forhold til rækkefølgen af directories. Vigtigere er derimod mulighederne under **Other Options**. Det anbefales på det kraftigste at vælge **Read after Write**, hvilket betyder, at programmet checker, at de flyttede data er ens før og efter flytningen. Hvis du undlader dette, går optimeringen meget hurtigere, men du risikerer at få ødelagt enkelte filer, hvilket kan betyde, at du må installere programmer på ny. Gem dette med **Save Options**.

På en PC fra 1991 kan en fuld optimering af en 90 MB disk, som ikke er blevet optimeret i lang tid, godt tage et par timer. Når du har afsluttet optimeringen, kan det være interessant at se, hvor enkelte filer er placeret på disken. Dette kan gøres med **Walk Map** under **Info**. Her er det igen rart at bruge mus.

Hvis du gemmer nye filer på en fuldt optimeret disk, kommer de altså til at ligge sidst på disken, og det er ikke altid det, man vil, så her et lille tip. Princippet er at kopiere en fil til et directory, hvor du efter at have kørt SPEEDISK gerne vil have 'huller', dvs ledig plads, og denne plads skal være forrest på disken.

Lad os sige, at du lagrer dine datafiler i C:\DATA\DOC. Du skaber C:\EXTRA, hvis du ikke har det. Til dette directory kopierer du en stor fil. Filen omdøbes til fx A.A. Inden du kører en fuld optimering med SPEEDISK kopieres denne fil til C:\DATA\DOC. Nu køres SPEEDISK og derefter slettes C:\DATA\DOC\A.A. Du har nu ledig plads i C:\DATA\DOC, hvor dine dernæst lagrede filer bliver placeret, og denne plads er først på disken. Dette kan du selvfølgelig gøre med flere forskellige directories, hvor du gemmer dine datafiler. En bat-fil, som gør dette:

```
XCOPY C:\EXTRA\A.A C:\DATA\DOC
SPEEDISK
DEL C:\DATA\DOC\A.A
```

DEFRAG fra DOS 6 kan kun benytte conventional memory og har ikke følgende muligheder: **Directory Order, Full with file reorder, Other Options** og dermed heller ikke **Read after write** - Microsoft har åbenbart tillid til sikkerheden. Til avanceret brug er DEFRAG utilfredsstillende, men dog langt bedre end ingen optimering. DEFRAG har dog en udokumenteret switch

```
C:\>DEFRAG/Q
```

som betyder Quick, og den er virkelig hurtig. Denne metode samler alle filerne i en lang række (dog uden at optimere hver enkel fil) og laver dermed sammenhængende plads til fx Windows' swapfile, se s. 35. Switchen **/H** gør, at DEFRAG kan flytte skjulte (hidden) filer.

Norton Commander (NC)

Dette afsnit er anderledes end de andre afsnit på den måde, at jeg fører dig igennem de funktioner, som jeg selv bruger mest. Det minder altså om kursusmateriale.

Da mange har og bruger NC, behandles dette fremragende program, som oprindeligt blev skrevet af John Socha. Jeg beskriver primært den engelske udgave ver. 3.0 og håber, at du selv finder ud af oversættelsen, hvis du har en dansk version.

Efterhånden får man mange filer på harddisken, og filerne ligger i mange forskellige directories. Jeg har altid brugt Norton Commander til at håndtere directories og filer. Hvis du ikke har dette program, kan du desværre ikke få noget ud af dette afsnit - beklager, men du kan jo læse lidt hist og her, hvilket måske gør, at du bliver interesseret i at anskaffe det. Du kan med god grund spørge, hvorfor jeg dog i en sådan tekst har valgt at skrive så meget om et program, som ikke følger med DOS.

Der findes lignende programmer, bl.a. DOSSHELL, PCTools og Windows' FileManager. Jeg synes, at NC er det bedste, letteste, hurtigste og smarteste. Programmet blev lavet for mange år siden for at lette håndtering af filer og directories. NC kan betjenes med mus, men jeg beskriver kun brugen af tastatur.

NC startes ved at skrive NC og afsluttes med F10. Tab springer mellem de to vinduer. Esc bruges til at fortryde, så hvis du åbner et vindue, kommer til at vælge en kommando, som du ikke vil udføre: tryk Esc.

Skærmen er opdelt i to vinduer, svarende til at man

står og har overblik over to directories på evt. to forskellige drev. Denne ide er genial! Allerøverst i hvert vindue står de to directories, som venstre og højre vindue befinder sig i. Den vandrette markering kalder jeg bjælken. Den markerer et drev, et directory og evt en fil. Hvis det aktuelle directory er tomt, vil bjælken stå i toppen af skærmen.

Hvert af de to vinduer viser filer/directories på den form, som bestemmes via valg i rullemenuerne. Prøv at trykke F9 og Enter eller ↓.

Dette aktiverer vinduesmenuerne. Du kan bevæge dig i et vindue med ↓ og ↑. Du kan flytte til et nyt vindue med → eller ←. Tryk Esc 2 gange for at returnere.

Nederst på skærmen står, hvad funktionstasterne laver. Holdes Alt nede, så vises hvad Alt + funktionstasten laver. Hvert af de to vinduer viser filer/directories på den form, som bestemmes via valg i rullemenuerne.

Indstillinger

Lad os starte med opsætning eller indstillinger, så din skærm og program opfører sig på samme måde som min, mens du læser dette. Du kan altid lave det om senere. Tryk F9 og i vinduet options (indstillinger eller opsætning) vælger du configuration. I hver ramme vælger du at aktivere et punkt med mellemrum, bevæger dig med pile, tab eller Enter. Nogle af indstillingerne kan du senere ændre, men her er mine forslag.

Screen blank delay betyder, at skærmen bliver (næsten) blank efter et bestemt tidsrum, hvis du ikke har rørt en tast. Dette beskytter din skærm. Min står på 3 min.

C:\DAN				C:\			
Name	Size	Date	Time	Name	Size	Date	Time
fakt8 doc	3126	17-05-93	14.34	DOS	▶SUB-DIR◀	7-05-93	8.18
faktpriv doc	2851	27-05-93	14.02	DSISYS	▶SUB-DIR◀	7-05-93	9.29
forudbog doc	2682	14-06-93	14.02	ENG	▶SUB-DIR◀	7-05-93	11.28
forudtak doc	2678	14-06-93	12.01	HD	▶SUB-DIR◀	18-06-93	14.04
isbn1 doc	2688	17-06-93	10.24	MOUSE	▶SUB-DIR◀	16-06-93	19.02
jan1 doc	2678	16-06-93	13.16	MW	▶SUB-DIR◀	7-05-93	9.01
johntape doc	2921	25-06-93	22.13	NC	▶SUB-DIR◀	7-05-93	8.10
kommiss doc	4488	26-05-93	17.42	NU	▶SUB-DIR◀	18-05-93	22.05
optimal doc	900253	26-06-93	13.20	PROTOS	▶SUB-DIR◀	19-05-93	17.49
optimal1 doc	17896	24-05-93	19.52	SYMANTEC	▶SUB-DIR◀	18-05-93	22.13
pris cdr	10294	24-05-93	6.54	TEMP	▶SUB-DIR◀	25-05-93	6.18
spørgesk doc	7870	9-05-93	11.02	UTI	▶SUB-DIR◀	7-05-93	9.35
storlabe doc	4016	20-04-93	20.38	WINDOWS	▶SUB-DIR◀	7-05-93	8.30
taklæser doc	2714	17-06-93	10.30	WINFAX	▶SUB-DIR◀	22-05-93	13.15
titel cdr	23974	25-05-93	5.54	WINWORD	▶SUB-DIR◀	7-05-93	9.06
titel3 cdr	24276	24-05-93	7.47	autoexec bat		543	26-06-93
tseng doc	9216	16-03-93	17.50	config sys		741	26-06-93
udg2 doc	2700	25-06-93	12.58	treeinfo ncd		923	26-06-93
tseng.doc	9216	16-03-93	17.50	treeinfo.ncd		923	26-06-93

C:\>

1Left 2Right 3View.. 4Edit.. 5 6 7Find 8History 9EGA Ln 10Tree

Configuration	
Screen colors () Black & White (*) Color () Laptop	File panel options [] Show hidden files [x] Ins moves down
Screen blank delay () 40 minutes () 20 minutes () 5 minutes (*) 3 minutes () 1 minute () Off	Tree panel options [x] Auto change directory
Other options [] Menu bar always visible [x] Auto save setup [] Left-handed mouse [x] Fast mouse reset	
Press Space to change an option, ↑ and ↓ to move between options	
[Ok] [Cancel]	

Show hidden files (vis skjulte filer) vælges blank, dvs. skjulte filer vises ikke. Du bør først vælge at se skjulte filer, når du er fortrolig med NC. *Ins moves down* (*Ins* flytter ned) betyder, at ved markering med *Ins* flytter bjælken sig et trin ned. *Auto change directory* betyder, at har du directorytree i fx venstre vindue og dette directory's filer i højre, skifter filerne, hver gang du flytter dig i venstre vindue. *Menu bar always visible* betyder, at den øverste bjælke for rullevinduerne altid er synlig. *Auto save setup* - x, betyder: afsluttes NC, lagres denne configuration. Afslut med **OK** - du bliver ved med **Enter**, indtil du når til **OK**.

Igen under *Options*, dvs. **F9** og enten ↓ eller **Enter**. Punkterne skifter mellem til og fra, hver gang de vælges. Du skal have følgende aktive, dvs. et lille hak ud for dem, hvilket gøres med **Enter** eller det fremhævede bogstav. Du kan desværre kun ændre én ad gangen.

Path prompt - DOS katalogsti: DOS prompten nederst på skærmen viser aktuelle directory.

Key bar - funktionstasterne bliver vist nederst på skærmen

F9, Enter. Vinduesmenuen for venstre og højre vindue er underdelt i 3 rammer, og inden for hver ramme kan du vælge én af mulighederne.

Brief	ØVERSTE RAMME - hvad vil du se i vinduet
√ Full	<i>Brief</i> oversigt: blot filnavne
Info	<i>Full</i> alle detaljer: filnavne, størrelse, tid
Tree	<i>Info</i> info om aktuelle drev, brugt og ledig diskplads
quick View	<i>Tree</i> katalogdiagram: directory træ i dette vindue og filerne i det andet
link	
On/Off	
√ Name	
eXtension	
tiMe	
SiZe	
Unsorted	MELLEMSTE RAMME - filernes sortering
Re-read	De forskellige muligheder siger sig selv.
fiLter...	Normalt har jeg <i>Name</i> slået til, men hvis man fx ændrer noget i et program, der
Drive...	består af flere filer og gerne vil se, i hvilken fil programmet lagrer bestemte ting, er <i>Time</i> smart. Filen vil så stå

øverst, hvis du vælger sortering efter tid. Skal du over-

skue mange filer, som hedder det samme til efternavn, vælger du *Extension* (type eller efternavn).

NEDERSTE RAMME - bruger jeg aldrig.

Du kan vælge enten ved at pile og trykke **Enter**, eller ved at skrive det markerede bogstav direkte. Prøv at vælge *Brief*. Vinduet viser det aktuelle directory med filerne, blot med deres navne. Gør det samme med det andet vindue, dvs. **Tab, F9, ↓**, vælg *Brief*. Hvis du trykker **F9** og vil fortryde, tryk **Esc**. Bevæg bjælken op eller ned med pilene. Læg mærke til, at nederst i den lille ramme får du den fulde information om den fil eller directory, som bjælken står på. Directories er skrevet med store bogstaver, filnavne med små. Vælg nu for begge vinduer **F9, Enter, Full**, hvilket viser filerne med navn, størrelse, dato og tidspunkt. Jeg kan bedst lide denne, men det er en smagssag.

Nu laver vi noget!

Du skal stå i rod-directory i begge vinduer, dvs. allerøverst i hvert vindue skal der stå C:\ Hvis der ikke står det, står du i et eller begge vinduer i et under-directory eller SUB-Directory. Tryk Home, bjælken står på to punktummer (UP-DIR eller TILBAGE). Tryk **Enter** og du kommer et trin nærmere rod-directory. Bliv ved indtil du i begge vinduer er i rod-directory. **Tab** skifter mellem vinduerne. Bevæg dig lidt rundt i directories, under-directories, så du bliver fortrolig med det. Prøv Home, End, PgDn, PgUp og pilene samt **Tab**.

Stå med bjælke i højre vindue. Vi laver et nyt directory i roden. Tryk **F7** og kald det EXTRA - forudsætter, at du ikke har et directory med dette navn. Bemærk at bjælken automatisk springer hen til det nye directory EXTRA. Tryk **Enter**. Hermed vælger du dette directory.

I toppen af vinduet står C:\EXTRA. Directoryet er tomt, indeholder ingen filer eller sub-directories. De to punktummer markerer det sted, hvor du med et tryk med **Enter** kommer et trin nærmere roden. DOS prompten helt i bunden af skærmen viser det aktuelle drev og directory for bjælken. Det smarte ved NC er, at man samtidigt kan manøvrere i NC og skrive normale DOS kommandoer i den normale DOS prompt. Brug **Tab** til at skifte mellem de to vinduer og kig på DOS prompten, den skifter. Placer bjælken i højre vindue, så du står i sub-directory EXTRA.

Tag en tilfældig diskette med filer. På diskettens bagside, øverste venstre hjørne sidder en skydeknop. Tryk den med negl eller kuglepen ud mod kanten, så der er hul igennem. Nu kan filerne på disketten ikke slettes, og der kan heller ikke lagres (skrives) på disketten, som nu er skrivebeskyttet. Man kan dog stadig læse filerne. Man taler om at *læse* (read), dvs. hente fra en disk og

skrive (write), dvs. lagre/gemme til en disk. Sæt disketten i diskettedrevet.

Nu vil vi i venstre vindue gerne se indholdet af disketten, dvs. A. Tryk **Alt+F1**, tryk **A** på tastaturet direkte. Skærmen blanker et øjeblik, læser på drev A og viser derefter indholdet. **Alt+F2** vælger på lignende måde højre vindue.

Sæt bjælken i højre vindue. Vi kopierer nu alle filer fra diskettens rod directory til directory EXTRA.

Kopiering med XCOPY

Skriv ved denne prompt

```
C:\EXTRA>XCOPY A:
```

Filerne kopieres og venstre vindue bliver opdateret. Med XCOPY fortæller man, *hvad* man vil kopiere *hvorfra* og *hvertil*. Her udnyttes, at kommandoen selv indsætter standardindstillinger for det, man *ikke* skriver. I sin fulde længde burde vi have skrevet.

```
C:\EXTRA>XCOPY A: . C:
```

Punktum angiver alle filer. Da prompten står i directory C:\EXTRA indsættes dette automatisk for *hvorhen*.

Default er alle filer - vi undlod punktum efter **A:**

Hvis vi ville kopiere den anden vej, altså fra EXTRA til A, havde vi skrevet

```
C:\EXTRA>XCOPY . A:
```

kopiér alle filer i aktuelle directory til A:. Man kopierer hovedsageligt mellem to forskellige drev, typisk mellem harddisk og diskette.

Kommandoen XCOPY kunne have været udført, uden at vi først med **Alt+F1** havde kigget, hvad der lå på disketten, men jeg synes det er en god ide først at kigge. Man behøver heller ikke være inde i NC for at lave XCOPY. Fordelen ved NC er at man let kan skifte directory, og endda to forskellige steder samtidigt.

Men pas på, at du skriver det rigtige, dvs. at du angiver *fra* og *til* rigtigt. Hvis du fx har alle dine tekstbehandlingsfiler i et directory på harddisken, som du regelmæssigt laver sikkerhedskopi af, så er det vigtigt at du får kopieret fra C til A - og ikke omvendt, for XCOPY advarer dig IKKE om, at filerne allerede eksisterer dér, hvor du vil kopiere dem til. Det gør COPY i øvrigt heller ikke.

Hvis du er lidt usikker med XCOPY, så brug hellere i starten NC uden DOS kommandoer i prompten. Se senere, hvordan filer markeres eller udvælges.

XCOPY kan desværre ikke kopiere skjulte filer.

Skjulte filer

Skjulte filer er netop skjulte, så brugeren ikke piller ved dem. De er skjulte for DIR og DELEte, hvilket betyder, at man ikke kan slette dem i DOS med DELETE, og man får dem heller ikke at se ved DIR. Yderst fornuftigt, fordi man sjældent har brug for at slette en skjult fil.

Hvis du skal slette, kopiere, omdøbe eller flytte en skjult fil, skal du via F9, options, configuration, sætte x i *show hidden files* (vis skjulte filer), og derefter kan du håndtere disse filer.

Udvalgte filer

Man selekterer - udvælger - ofte nogle bestemte filer for at gøre et eller andet ved dem: kopiere, slette eller flytte. Sæt bjælken i højre vindue. Vi vil slette nogle af de filer, som vi før kopierede til C:\EXTRA. Du skal være sikker på, at du står i C:\EXTRA (øverst i vinduet skal der stå C:\EXTRA).

Stil bjælken på en fil og tryk Ins (Insert hedder den på nogle tastaturer). Du kan enten bruge den lille grå Ins eller den store hvide Ins, men med sidstnævnte må der ikke være lys i NUMLOCK. Hele linien for filen skifter farve, sandsynligvis gul. Filen er udvalgt, selekteret eller markeret.

Bemærk at bjælken springer automatisk videre til næste fil. Dette er bestemt af konfigurationen (**F9+Options+Configuration**) *Ins moves down*. Man kan i ver. 3.0 *ikke* markere et directory, kun filer (ver. 4.0 kan imidlertid). Markér nogle af filerne. ↑ igen på de gule filer og tryk **Ins**. De fravælges. På den måde kan du udvælge enkelte filer. Vælg nogle stykker, men ikke alle. Tryk **F8** for at slette dem. Først advares du om, at du har udvalgt nogle filer til sletning. Tryk **Enter**. Du advares endnu en gang og kan endnu nå at fortryde - med **Esc** eller markere Cancel. Tryk **Enter**. Filerne slettes.

Nederst på skærmen angives funktionstasterne. Hold **Alt** nede og kombinationen med **Alt** vises. **F5**, **F6** og **F8** behandler primært udvalgte filer, sekundært den fil, som bjælken står på. Det vil sige: har du udvalgt filer, udføres handlingen kun på de udvalgte filer - det er ligegyldigt, hvor i directoryet bjælken står, men den skal stå i det vindue (directory), hvor du vil udføre handlingen. Er ingen filer markeret, udføres handlingen på den fil, hvorpå bjælken står.

Vi vil nu flytte udvalgte filer fra et directory til et andet. Højre vindue er stadig C:\EXTRA, og du har nogle filer tilbage, håber jeg. **Tab** til venstre vindue. Hvis du ikke er på C, så tryk **Alt+F1**, tast C direkte. Gå til roden: **Home** og **Enter**, indtil du i øverste linie på skærmen har C:\

Tryk **F7**, kald det nye directory for SLET. Gå ind i directoryet. Venstre vindue står nu i C:\SLET, højre i

C:\EXTRA. **Tab** til højre. Udvælg nogle filer med **Ins**. Tryk **F6**. Programmet melder

```
Rename or move x files to
C:\SLET
```

F6 og **F5** vil gå ud fra, at du vil flytte, omdøbe eller kopiere til det directory, som er i det andet vindue, hvilket er meget smart. Tryk **Enter**. De udvalgte filer flyttes til det andet directory. Filerne optræder nu ikke længere i directory EXTRA, men i SLET.

Hvis du undrer dig over, at en fil kan flyttes så hurtigt fra ét directory til et andet, har du god grund. Man kalder det ganske vist at *flytte*, men filen beholder sin fysiske plads på disken. Som andetsteds omtalt indeholder FAT informationer om, i hvilket directory en fil befinder sig samt dens fysiske placering på disken. Flyttes en fil, er det altså kun directorynavnet i FAT, der skal ændres, hvilket ikke tager så lang tid.

Nu kopierer vi alle filerne i SLET til EXTRA. **Tab**, så bjælken står i venstre vindue. Tryk på den store grå plustast helt til højre på tastaturet. Et vindue på skærmen viser

```
Select the files
*.*
```

. foreslås, men du kan skrive noget andet, hvis du vil. Tastes **Enter** markeres alle filerne i det aktuelle directory. Bemærk nederste linie i vinduet, der fortæller antal og størrelse af alle filerne. Den grå tast med minustegnet virker omvendt. Den fravælger (UN-selector)

Tryk **F5** og **Enter**. Filerne *kopieres* nu til EXTRA. Jeg bruger hovedsageligt plustasten til at se, hvor mange filer, der findes i et directory, og hvor meget de fylder. Hvis jeg skal lave noget med næsten alle filer i et directory, markerer jeg først alle og **FRA**vælger dernæst med **Ins** de filer, som ikke skal med i markeringen.

En anden nyttig funktion med plustasten er: Du har et directory med filer, som fylder flere MB og vil kopiere dem til disketter. Dit andet vindue er valgt med A. Vælg alle med plustasten, tryk **F5** og tryk **Enter**. Kopieringen starter, indtil disketten ikke kan rumme mere. Det smarte er nu, at NC har afmarkeret de allerede kopierede filer. Du sætter en ny diskette i og fortsætter, indtil alle filer er blevet kopieret.

Tilbage til eksemplet. Højre vindue er stadig EXTRA og venstre SLET. Stil dig i højre. Marker alle filer igen, tryk **F5** og **Enter**.

NC finder ud af, at filen, som NC skal til at kopiere, allerede eksisterer dér, hvor du vil kopiere den hen og spørger: **Overwrite**, **All** or **Skip** (Overskriv denne, Overskriv alle eller spring over). I den engelske version kan du bruge bogstaverne **O**, **A** eller **S** (**Esc** laver det samme som **S**). **O** for at overskrive den aktuelle fil, **A** for

at overskrive alle udvalgte filer eller **Esc** for ikke at overskrive den aktuelle fil, men gå videre til den næste. I den danske udgave må du flytte med pil. Prøv forskellige muligheder, indtil du forstår systemet. Bemærk at ét tryk på **Esc** kun svarer for én fil, mens et **A** svarer for alle filer. DOS kommandoerne **COPY** og **XCOPY** giver *ikke* disse advarsler før DOS 6.2.

Omdøbe

F6 kan én af tre ting: 1) flytte en fil til et andet directory, 2) give en fil et nyt navn inden for samme directory eller 3) give et directory nyt navn (et directory er egentlig en fil, blot en speciel fil).

'Flyttes' en fil fra et directory til et andet ser det ud som om, filen flyttes, men reelt får den blot et nyt navn (drev + directory + fornavn + efternavn).

Stil dig på en tilfældig fil i C:\EXTRA eller C:\SLET. Vi vil kalde filen MARIA.DOC. **F6** og skriv MARIA.DOC og **Enter** - og bemærk at **NC** *blanker* indtastningsfeltet, når du skriver første bogstav. Filen har fået nyt navn. Nu vil vi kalde filen KRISTINA.DOC. Tryk **F6**, og nu behøver du kun at skrive KRISTINA.*

Stjernen til sidst betyder, at vi vil beholde efternavnet. Filen hedder nu KRISTINA.DOC. Hvis du vil kalde filen KRISTINA.BRE skriver du blot på lignende måde *.*.BRE. Tip: Hvis du vil undgå, at **F5** eller **F6** foreslår det andet vindue, bruger du **Ctrl+F1** eller **Ctrl+F2** for midlertidigt at slukke det andet vindue. Samme kombination tænder igen.

Nu ved NC, at du ikke vil flytte filen til et andet directory. Dette er specielt smart, hvis du ved omdøbning vil rette lidt i navnet eller vil lave en kopi af en fil med et næsten identisk navn. Filnavnet fremkommer som forslag, brug først piletast til højre, derefter kan du rette i navnet - starter du med et bogstav, *blankes* hele navnet.

Nogle gange vil man gerne omdøbe flere filer, som hedder det samme til efternavn, til et andet efternavn. Lad os sige, at du har kaldt alle dine dokumenter i tekstbehandlingen for DOC til efternavn - eller programmet gør det automatisk. Du vil nu gerne have, at de i stedet hedder fx DOK til efternavn. Vær blot sikker på, at ét af navnene ikke findes i forvejen med det nye efternavn. Du kan så markere dem enkeltvis med **Ins** eller - lettere, hvis der er mange - markere dem alle med den grå plustast, og så angive *.*.DOC. Nu har du markeret alle filer, som hedder DOC til efternavn. **F6** og skriv *.*.DOK og **Enter**. Alle filer hedder nu DOK til efternavn.

Hvis du vil lave kopier af alle filer med efternavn DOC, bruger du **F5** i stedet for **F6**, og skriver på samme måde som før *.*.DOK. Så har du filerne med to forskellige efternavne i samme directory. Slet nu alle filerne i de to directories C:\EXTRA og C:\SLET. Og slet de to directories.

Se eller rette i tekstfiler

F3 og **F4** åbner en tekstfil lige som en tekstbehandler. **F3** kan kun kigge. **F4** kan rette i en fil på max 26464 bytes. NC har en indbygget editor. Hvis du vil bruge en anden editor ved **F4**, så: **F9**, **Options**, **Editor**, marker **Extern**, og skriv det program, der skal kaldes.

Jeg bruger NC's indbyggede editor til at lave mindre rettelser i tekstfiler, typisk AUTOEXEC.BAT, CONFIG.SYS, INI-filer og BAT-filer. Det er nemt og hurtigt. **F7** søger.

Ctrl+Y sletter en hel linie, og **Home**, **End**, **PgDn** og **PgUp** virker, som de plejer i andre programmer. Har du lavet en rettelser og vil afslutte, trykker du **Esc**, hvorefter du bliver spurgt, om du vil lagre eller gå ud af filen uden at lagre.

Søge efter fil

Sommetider ved man, at man har en fil, men kan ikke huske, hvor på harddisken den ligger. Tryk **Alt+F7**, skriv navnet fx MUSIK.DOC og tryk **Enter**. Eller blot en del af navnet, hvis man ikke kan huske hele navnet. MU* finder alle filer, som begynder med MU. Programmet søger på hele disken, lister alle fundne filer. Pil til den ønskede fil, **Enter**, og du står på filen.

Skabe en ny fil

Shift+F4. Skriv navnet og tryk **Enter** to gange.

Tidligere dos-kommandoer

Ctrl+E bruges til at bladre baglæns i udførte DOS kommandoer. **Ctrl+X** bladrer den anden vej. På samme måde som **↑** og **↓** i DOSKEY.

Tænd/sluk et vindue

Ctrl+O tænder og slukker for NC på skærmen. Prøv. Bruges typisk, hvis man skriver en DOS kommando, og bagefter dækker NC for resultatet. Hvis du ved, at de næste kommandoer er DOS kommandoer, og du gerne vil følge med på hele skærmen, kan du slå NC's skærm fra midlertidigt med **Ctrl+O**. Nu er du i DOS og kan bruge **↑** og **↓** til at bladre i kommandoerne. Når du igen taster **Ctrl+O**, bruges **Ctrl+E** for at bladre baglæns.

Byt vinduerne

Ctrl+U bytter de to vinduer om. Prøv. Bruges typisk, når man har valgt directories i de to vinduer og pludselig finder ud af, at man egentlig hellere ville have det omvendt. Det er en god ide fra starten at beslutte sig til, hvor man helst vil have A og C, når man arbejder med disketter. Der er en vis logik i at følge alfabetet og vanen med at vi læser fra venstre mod højre. A til venstre og C til højre. Har man to diskdrev A og B på samme måde. På

denne måde er der mindre sandsynlighed for at lave fejl ved kopiering.

Skift directory.

Alt+F10 bruges til hurtigt at skifte directory. Prøv. Når du taster et bogstav finder NC det første directory, som begynder med bogstavet. Skriv tilstrækkelig med bogstaver, indtil du lander det rigtige sted.

Hvis to directories har samme begyndelsesbogstaver, brug **Ctrl+Enter** til at hoppe til næste. Du kan også bevæge dig rundt med pilene. Tryk derefter **Enter**. Det vindue, bjælken stod i, sættes nu i det valgte directory. Filen C:\TREEINFO.NCD indeholder dette træ, og hvis den slettes, skaber NC træstrukturen og dermed TREE-INFO.NCD, næste gang du trykker **Alt+F10**.

Hvis du står godt inde i træstrukturen og hurtigt vil ud i roden: **Ctrl+<**

Bladre i directories

F9, venstre eller højre vindue, **Enter**, **Tree (Katalog)**. Pil op og ned, og det andet vindue viser automatisk filerne i det valgte directory. God til hurtigt at få overblik: hvor mange filer ligger i de forskellige directories. Eller hvis man overhovedet ikke kender navnet på en fil, man leder efter, kan man være heldig at støde på den, når man ser navnet.

Sammenlignbare directories

F9, **Commands**, **Compare** (sammenlign) er praktisk, hvis du laver sikkerhedskopiering fra et directory på harddisken til et tilsvarende directory på en diskette. Du får med det samme at vide, om der er forskel på de to directories. Filer, som er anderledes det ene sted, markeres, og du kan gennemgå de markerede filer for at se, om du måske kan slette nogle filer eller om du burde lave en sikkerhedskopiering af nogle filer fra harddisk til diskette.

Nu når du har læst dette, øvet dig lidt, så er du mester i at håndtere dine filer, både på disketter og på harddisken. Så her mine råd om at opdele harddisken i directories. Når du installerer programmer, så følg forslagene om directories, som programmet foreslår. De directories, du selv laver: lav så få underdirectories som muligt. Lav hellere flere ude i roden.

Efterhånden vil du arbejde mere og mere med NC. Du vil få flere filer at holde styr på. Placer dine datafiler i et separat directory, fx C:\DATA og herunder subdirectories for hvert program, du bruger. Du bruger sikkert kun ét tekstbehandlingsprogram (data placeres i fx C:\DATA\DOC), én database (data placeres i fx C:\DATA\DB), ét regneark (data placeres i fx C:\DATA\REGN) osv.

Med NC kan du hurtigt overskue dine dokumenter, du kan let kopiere alle eller enkelte til diskette.

Menu

Filen C:\NC\NC.MNU indeholder den menu, som fremkommer ved **F2**. Man kan lave mange smarte ting med denne tekstfil.

Lad os tage et eksempel på nogle linier, som kan stå i NC.MNU. **Mellemrum** eller **Tab** kan benyttes til indryk.

```
D:  Skift til DOC
'  her er en kommentar
   cd c:\data\doc
```

```
Ret CONFIG.SYS
   C:\BAT\ec.bat
```

Den første linie i hver gruppe, som skal starte helt ude til venstre, vises i menuen på skærmen. **D**: i starten betyder, at du blot ved tryk på **D** aktiverer punktet, hvilket er smart, hvis du har mange punkter.

Næste linie består af en eller flere DOS kommandoer. Du kan lave flere NC.MNU i andre directories, som bliver den aktive menu, hvis du skifter til disse directories. På den måde kan du hurtigt køre rundt mellem mange forskellige directories og dermed forskellige menuer.

Version 4.0

er fremragende forbedret. Her blot nogle af nyhederne: lige som i Windows' Filemanager kan du håndtere et directory med underliggende sub-directories. Under configuration (indstillinger) kan du vælge **Select Directories**, hvilket betyder, at den grå plus-tast også markerer directories. Du kan kopiere directories med under-directories, hvilket er godt ved backup.

Editoren (**F4**) kan en masse forskellige ting: blok-funktioner (start blok med **F3**, slut blok med **F3**, ophæv blok med **Shift+F3**), søg/erstat osv.

Quick View giver info om det valgte directory: antal sub-directories samt antal og størrelse af samtlige filer.

Der er et indbygget program, hvor du kan håndtere ZIP-filer og andre komprimerede filer (kræver at pakkeprogrammerne ligger i PATH). NC har sit eget pakkeprogram i ZIP formatet. Tast **Enter** på en ZIP-fil og så vises navnene på de filer, der er indeholdt i ZIP-filen, og derefter kan du udføre fil-funktioner såsom kopiere, slette osv., men ikke kigge. Marker en række filer og komprimer dem til ZIP format med **Alt+F5**. Hvis du har LHA og andre pakkeprogrammer liggende i din PATH, kan du med **Alt+F6** udpakke (de-komprimere) i disse programmets formater.

Ctrl+F9 sender en fil til printerens. **Ctrl+F3**, **F4**, **F5** og **F6** sorterer filer i henholdsvis fornavn, efternavn, tid eller størrelse.

En anden nyhed er et indbygget kommunikationsprogram. Det er glimrende. Jeg bruger det til at sende og modtage med venner og BBS.

Jeg savnede i version 3: efter en sammenligning af directories, markeres visse filer i et directory. Nu kan man med den grå stjerne *invertere*: de markerede filer bliver umarkerede og omvendt. Dette kan bruges, hvis du vil lave noget med de filer, som netop *ikke* er markeret.

Version 5.0

Denne nyeste version indeholder intet væsentligt nyt. Enkelte ting er endda ændret til det dårligere.

Version 4.0 er bedst.

Sikkerhedskopiering

Rigtige mænd tager backup, om ikke før, så siden. Rigtige kvinder tager forhåbentlig også backup, thi på et tidspunkt sker for nogle af os dét, der ikke må ske. Harddisken kan i sjældne tilfælde bryde sammen eller man kommer til at slette en fil og opdager det for sent.

Mit bedste råd er: køb en ekstra harddisk, som du bruger som backup. Sandsynligheden for, at begge fysiske harddiske går i stykker samtidig, er lille. Jeg har to ens harddiske, og det tager ikke mange minutter at tage backup af flere hundrede MB.

Jeg tager flere gange om dagen backup, når jeg har lavet noget, jeg nødt vil lave en gang til. Jeg bruger Norton Commander

Eller køb et ZIP-drev, hvor hver diskette kan rumme 100 MB eller mere. Hvis du vælger at bruge 1,44 MB disketter, så læs følgende:

Microsoft har i mange år leveret et program kaldet BACKUP, som er så upraktisk at bruge, at de færreste har brugt det. Der findes andre programmer til backup. DOS 6 indeholder en mindre udgave af Nortons Backup.

Min harddisk er primært fyldt med filer fra programmer og måske 5-20% er *datafiler*, dvs. filer, som jeg selv har skabt.

Programmerne har du på originaldisketter og kan installeres igen, hvis det går galt. Det er kun relevant at tage kopi af datafiler samt af filer, som har med initiering eller opsætning af programmer at gøre. Sidstnævnte filer har typisk efternavn INI.

Det er en god ide at holde programmer og data adskilt i forskellige directories. Lav et under-directory som fx C:\DATA\DOC til dine tekstbehandlingsdokumenter. Jeg har tit været ude for at skulle installere et program flere gange. Hvis man ved, hvilke directories et program skaber og placerer sine filer i, er det let at slette det hele og begynde forfra med en frisk installation. Windows er dog speciel.

Du kan lige så godt vænne dig til ideen om, at sikkerhedskopiering og vedligehold af din harddisk og disketter tager tid. I starten gik jeg ikke så højt op i det, håbede at jeg kunne huske filernes navne, håbede at harddisken aldrig lavede problemer, håbede at jeg kunne finde rundt i alle mine disketter, men jeg blev klogere..

Mens jeg arbejder, laver jeg sikkerhedskopi ca. hver time, og det gør jeg i øvrigt med alt, hvad jeg laver. Jeg har prøvet at miste flere timers arbejde.

Jeg har lavede directories på mine disketter, som svarede til directories på harddisken, så var jeg ikke i tvivl om, hvor filerne stammede fra. Som regel brugte jeg Norton Commander, da jeg typisk arbejdede på flere filer samtidig, men også pakkeprogrammet PKZIP. Hvis du ar-

bejder med store filer, er det en god ide at bruge et program, som kan pakke filerne, så de fylder mindre.

Sikkerhedskopiering hænger meget sammen med dit overblik over dine filer. Jeg vil kraftigt anbefale dig regelmæssigt at bruge tid til at gennemgå både harddisk og disketter. Har du programmer eller datafiler på harddisken, som du sjældent bruger? Vi installerer jo ofte et program for at se, hvad det er for noget, men glemmer det senere. Hvis du har installationsdisketter til programmet, så slet det fra harddisken eller pak relevante directories med et pakkeprogram. Jo færre filer på harddisken, jo hurtigere fungerer din PC.

Hvis dine filer ikke er større end, at du let kan organisere dem i forskellige directories, så hvert directory ikke fylder mere end 1.44 MB, kan du let lave en diskette til hvert directory. Evt. kan du måske ved hjælp af et pakkeprogram stadig have en diskette til hvert directory. Et pakkeprogram kan typisk reducere filernes størrelse 50%. Det er ikke let at overskue directories med mange filer.

Generelt anbefales, hvis du eksperimenterer med CONFIG.SYS, løbende at kopiere en fungerende C:\CONFIG.SYS til C:\DOS

```
C:\>copy config.sys c:\dos
```

så kan du altid kopiere tilbage til roden - evt. efter en boot fra boot-diskette, som beskrives s. 49.

Windows 3.1

Her behandles blot emner, som har forbindelse til resten af teksten. Se hæftet *Bag Om Windows 3.1*.

Jeg bruger Windows meget og kører som de fleste både DOS og Windows programmer samtidigt. Man skal ikke tro, at blot man installerer Windows, så flyver det hele derudaf med samme hastighed, som man har været vant til fra sine DOS programmer. Opdatering af en skærm i såkaldt *graphics mode* tager væsentligt længere tid end opdatering af en skærm i *text mode*.

Windows 3.1 er *ikke* et styresystem, men et *operating environment*, et miljø, omgivelser, en *overbygning* på DOS, et grafisk brugerinterface, en mere praktisk og smartere måde at håndtere programmer, filer, directories osv. end den traditionelle i DOS.

Windows er særlig relevant, hvis du vil arbejde meget med grafiske ting, fx layout, tegneprogrammer osv., eller hvis du løbende vil se, hvordan det *nogenlunde* kommer til at se ud på printeren. I alle Windows-programmer kan bruges mus, og du kan klikke dig igennem næsten det hele. En anden behagelighed er, at programmerne ligner hinanden mht. menuer ligesom på Macintosh. Yderligere behøver du ikke kende så meget til DOS kommandoer.

Du kan køre dine vante DOS programmer, både som du kender det på hele skærmen, men også i et vindue mindre end hele skærmen. Visse steder kan du flytte data fra ét program til et andet. Du kan køre flere programmer samtidigt, let skifte fra det ene til det andet.

Problemet med Windows, når man har været vant til hastigheden under DOS, er at Windows kræver hurtigere CPU og gerne masser af RAM. En 386DX/33 MHz med 4 MB RAM anser jeg som minimum for acceptable hastigheder.

Windows kræver XMS hukommelse og også gerne hurtig harddisk og grafikkort. Windows starter (forudsat tilstrækkelig RAM) default i *386 enhanced mode*. Windows såkaldte *standard mode* - C:\>**WIN/S** - er ca. 10-20% hurtigere. Brug dette, når du kun kører mindre memorykrævende Windowsprogrammer. I *386 enhanced mode*, kan du danne en såkaldt swapfile.

Swapfile

Når der ikke er tilstrækkelig hukommelse (fordi du åbner flere programmer end din RAM kan rumme) kan harddisken benyttes som forlænget memory, hvilket kaldes *virtual memory*. Når du med **Alt+Tab** skifter program hentes det, der evt. ikke kunne rummes i RAM, ind fra *swapfilen*. Dette er en væsentlig forbedring af Windows hastighed og du undgår 'out of memory', hvis du da ellers laver swapfilen stor nok.

Du kan lave enten en *temporær* eller en *permanent*. Den temporære skal skabes, hver gang Windows starter, hvilket tager tid, og dette kan kun anbefales, hvis du er i pladsnød på harddisken. Men du burde nok hellere rydde op på din harddisk, så du kan få plads til en permanent swapfile, hvilket er langt bedre.

Den permanente swapfile bliver liggende på harddisken. Inden du laver denne fil, kører du et diskoptimeringsprogram eller

```
C:\>DEFRAG/Q
```

Vælg [Control Panel](#), [386 enhanced](#), [Virtual Memory](#), [Change](#). Hvor stor skal den være? En generel regel er, at din ledige XMS memory efter boot plus swapfilen er lig med 12 MB. 4-8 MB er for de flestes vedkommende passende.

Du kan teste ved at kigge på lampen for harddisken, hver gang du skifter med **Alt+Tab**, om CPU'en skal ud i swapfilen for at hente data. Du kan som regel også høre flittig aktivitet på harddisken, hvis du har mange og ofte skift, hvordan der 'swappes' frem og tilbage. Hvis det er tilfældet, prøv at forøge swapfilens størrelse. Den kan max være en vis brøkdel af den ledige plads på harddisken. Du bør også aktivere 32 bit disk access.

32 bit disk access

som giver hurtigere kommunikation med harddisken uden om DOS og den langsomme BIOS, når der 'swappes'. DOS programmer kører også hurtigere i enhanced mode. Hvis Windows under test af harddiskcontrolleren finder den kompatibel (i overensstemmelse) med en bestemt standard (Western Digital 1003), vil du have mulighed for at aktivere 32 bit disk access - du skal sætte kryds. Dette er nemlig af Microsoft *ikke* default sat pga. visse tekniske forhold ved bærbare PC'ers måde at spare batteriet.

(4) Uheldigvis valgte Microsoft at kalde denne kommunikation noget med 32 bit. Et andet navn er FastDisk, som intet har at gøre med I/O bussen eller CPU'ens bredde for adressebussen. Det drejer sig om noget teknisk omkring 386'erenes adresseregister. Windows bruger en device driver, som i *protected mode* direkte kommunikerer med harddiskcontrolleren, hvilket forøger kommunikationen (throughput) med ca. 20% og forøger antallet af DOS programmer, der kan køres samtidigt.

Hvis Windows derefter ikke kan starte, start med

```
C : \>WIN/D:F
```

og fjern krydset i boksen for 32 bit disk access.

Win 3.1 kan køre i protected og enhanced mode. Når Windows kører i enhanced mode, får hvert DOS program memory, som om det kørte på en 8086 baseret PC. Har du 4 DOS programmer kørende, har du simuleret 4 ekstra PC'ere. Denne *mode* kaldes Virtuel 8086 eller forkortet V86 mode, og her simulerer 386'eren 8086 processoren, mens den kører i protected mode.

Fordelen er, at du kan køre real mode DOS programmer samtidig med fordelene fra protected mode - beskyttelse imod memorykonflikter. Yderligere kan du køre flere programmer tilsyneladende samtidigt - det ser sådan ud, selv om 386 processoren reelt blot skifter hurtigt mellem de forskellige programmer, som i et givet meget lille tidsrum får CPU kraft - også kaldet multitasking (task: opgave, proces). Hvert DOS program råder altså over den totalt tilgængelige conventional memory, og derfor er memory optimering vigtig.

En 486 processor indeholder grundlæggende i denne forbindelse det samme som en 386'er.

Tips til Windows

som jeg haft glæde af. Hvis du vil lagre Program Managers opsætning *uden* at afslutte Windows, holdes **Skift** nede, mens Windows 'afsluttes' med fx **Alt+F4**. Opsætningen gemmes, men Windows afsluttes *ikke*. Du kan så under *Options* frakoble *Save settings on Exit*.

Jeg havde svært ved at læse de grønne ord i Windows Help. I WIN.INI under [Windows Help] kan skrives

```
Jumpcolor=0 0 128
Popupcolor=128 0 0
```

tallene angiver hvor mange enheder rød, grøn og blå du giver. Du kan lege lidt og se, hvad du bedst synes om. Tak til Brian Livingston, som også har videregivet at skrive i SYSTEM.INI

```
[386enh]
MaxBPs=768
```

Det betyder *max antal breakpoints*. Et breakpoint er 10 byte, som Windows bruger til kontrol af DOS sessioner, dvs. DOS programmer, der kører som en *virtuel PC*, hvor DOS programmet anvender PC'en, som om den havde den for sig selv. Mere nøjagtigt bruges et breakpoint, hver gang Windows skal kommunikere i *real mode*, og kort fortalt handler det om, at der default ved Windows start kun er afsat et vist antal breakpoints, som programmørerne regnede for tilstrækkeligt, men mange når i én Windows session over dette antal, og så begynder de ubehagelige fejl at komme. Siden jeg indførte denne linie, har jeg haft færre ubehagelige stop i programmer.

Som vi alle ved, er Windows 3.1 eller 3.11 ikke perfekt og går ofte ned. Jeg har vænnet mig til ca. hver time at lukke Windows helt og starte forfra (og evt. boote) for at rense memory. Windows æder langsomt memory, hver gang man åbner og lukker programmer. Jeg vil hellere bruge dette minut i timen end at få ubehagelige overraskelser.

Punktum på nummertastaturet med lys i NumLock: Ret i WINDOWS\SYSTEM\KBDDA.DLL med en editor. Søg , , **--++ og ret til . . **--++ og du får punktum i stedet for komma. Lav forinden en kopi af filen for en sikkerheds skyld.

Andre måder at starte WIN, skriv WIN/?

DOS 6

I det følgende handler det om Microsofts udgave MS-DOS. Lad mig starte med at sige, at den blotte installation får ikke din PC til at køre *hurtigere* end med DOS 5.

Af DOS 6 beskrives det, som er relevant i forhold til den øvrige tekst. Programmer som Undelete, Anti-virus, Interlnk, Backup m.m. beskrives ikke, primært fordi jeg ikke selv bruger dem.

Installation

Det er muligt at installere, således at du - hvis du fortryder - kan vende tilbage til din tidligere version. Dette anser jeg for urealistisk og anbefaler derfor, at du skriver

```
A:\>setup/G
```

som ikke kræver såkaldte *Uninstalldisketter*. Hvis du vil se andre muligheder ved Setup, kan du skrive

```
A:\>setup/?
```

Forbered dig på, at installationen tager et stykke tid. Visse residente disk-cache (SMARTDRV undtaget), delete-protecting og anti-virus programmer kan ikke køre sammen med Setup-programmet. Det anbefales derfor, at sætte disse ud af funktion under installationen ved at skrive REM foran sådanne linier i dine startfiler og derefter boote.

↑ og vælg for hvert punkt (tryk **Enter**), om du vil installere 1) kun DOS, 2) kun Windows eller 3) både DOS og Windows for de tre angivne programmer, som både leveres i en DOS og en Windows version.

Hvis du ikke tager det hele, kan du altid senere installere det, du i første omgang ikke installerede. Du kører blot setup igen.

Der skabes et directory C:\OLD_DOS.1, hvortil den tidligere DOS versions filer kopieres. Bemærk, at filer i C:\DOS, som ikke bliver erstattet af nye fra DOS 6 med samme navn, bliver liggende i C:\DOS. Derfor er det mit forslag *inden* installationen i C:\DOS *kun* at have DOS filer og intet andet for lettere at kunne finde filer efter installationen - jeg lavede faktisk en kopi af hele C:\DOS til C:\DOS5.

Jeg blev forvirret, da jeg sammenlignede C:\OLD_DOS.1 og det nu nye C:\DOS. Jeg troede faktisk, at setup kopierede hele C:\DOS til C:\OLD_DOS.1, men det gjorde den ikke. En fil i C:\DOS, som *ikke* stammer fra DOS 5, forbliver i C:\DOS. Setup laver det sådan for at kunne skifte tilbage til DOS 5, hvilket de færreste vel ønsker.

Hvis du vil beholde DOS 6, kan du slette C:\OLD_DOS.1, fx med C:\>**DELOLDOS**

Startfilerne beholdes i roden. Til sidst bootes med den nye DOS 6. Her er der sket ændringer. Skærmen viser i 2 sek.

```
Starting MS-DOS..
```

Mange brugere har enten ikke haft brug for eller blevet forvirret over de tekniske meddelelser fra specielt EMM386 og SMARTDRV, så de er nu fjernet ved defaultlinier i startfilerne. God ide. Hvis du som under DOS 5 vil se meddelelser på skærmen under boot, skriver du /V et sted i linien for henholdsvis EMM386 og SMARTDRV.

Jeg var spændt på at se, om der var sket noget med mine startfiler. Setup havde ikke fortalt noget undervejs. Noget overrasket konstaterede jeg, at den uden at spørge havde tilføjet

```
device=C:\DOS\setver.exe
shell=C:\DOS\command.com c:\dos\ /p
```

og hermed stjal nogle KB af min conventional RAM. Den kunne da i det mindste have indlæst SETVER i upper memory. Hvor jeg før havde 617.5 KB free i conventional, havde jeg nu kun 615.

Help

Her er der sket noget. Hele manualen for DOS kommandoer ligger i en fil. Skriv fx DIR/? for at få hurtig, kort help på skærmen om de forskellige parametre og switches. Skriv HELP DIR, og EDIT startes - brug **Tab** eller forbogstav. Det er virkelig behageligt (og økologisk), at man hurtigt kan slå op her, i stedet for at kigge i en stor bog.

Vi ventede længe på *Deltree*, som kan slette et directory med underliggende directories.

```
C:\>deltree c:\extra
```


EMM386

EMM386.EXE tager nu med parameteren RAM al extended memory under sine vinger og giver et program, hvad det har brug for, hvad enten det drejer sig om EMS eller XMS memory. Dette kræver altså brug af en page frame, som fylder 64 KB i upper memory.

Under DOS 5 kunne man kun konvertere en vis del af XMS memory til EMS memory, og så var denne del af memory 'låst' til udelukkende dette formål (EMS), og man måtte ændre CONFIG.SYS og boote for at ændre dette.

Hvis du i CONFIG.SYS skriver fx

```
..emm386.exe ram min=0 .. /v
```

vil du under boot af skærmeddelelsen se, at EMM386 kan simulere den mængde EMS eller XMS memory, som et program har behov for. Har du sommetider brug for EMS memory, anbefales ovenstående linie. `min=0` betyder, at der fra start ikke reserveres 256 K til EMS, hvilket ellers ville ske, hvis ikke `min=0` angives.

Du bruger 64 K i upper memory (som altså tages fra extended memory), men behøver ikke spekulere så meget på om dine programmer bruger EMS eller extended memory.

Se også s. 39 *Flere Boots*, hvor du får mere kontrol over opsætningen af memory.

MEMMAKER

MEMMAKER kan en hel del mht. optimering af startfilerne. Start med at skrive

```
C : \>MEM/?
```

for at se de nye muligheder ved dette program. Skærmens informationer fra MEM er i forhold til tidligere version (fra Win 3.1) lettere at læse og forstå. Du kan skrive

```
C : \>MEM/C/P
```

og notere informationerne om free memory for bedre at kunne følge med, når du senere kører MEMMAKER.

Du behøver ikke at være nervøs for at køre MEMMAKER. Du kan altid vende tilbage til den tilstand, dine startfiler var i, inden du kørte programmet - men kun ét skridt tilbage. Hvis du vil have mulighed for flere skridt baglæns, kan du efter hver kørsel `rename` startfilerne med efternavn UMB til efternavn 1, 2 osv. (husk, at de ligger i C:\DOS - lidt usmart).

Du starter ved at skrive

```
C : \>Memmaker
```

Du kan enten køre den enkle udgave, kaldet *express*, eller den avancerede, kaldet *custom*. MEMMAKER kopierer først de nuværende startfiler til C:\DOS med efternavnet UMB. Derefter rettes kun i dine startfiler, intet andet.

Hvis du vil tilbage til før sidste rettelse - altså kun ét skridt tilbage - skriver du

```
C : \>Memmaker/Undo
```

MEMMAKER indsætter selv visse linier, retter andre til med diverse parametre og switches - og der er kommet mange!

Jeg kørte først *express* for at se, hvad der skete. Der bliver bootet og testet nogle gange, og det tager lidt tid. Løbende gives engelske nogenlunde let forståelige skærmeddelelser, som fortæller, hvor langt MEMMAKER er kommet i processen. Derefter kan du se ændringer i dine startfiler.

Det gode ved MEMMAKER er, at man kan køre programmet gentagne gange, indtil man finder det optimale. Hold øje med meddelelser på skærmen om evt. problemer. Jeg har ikke oplevet nogen, men hvis du gør, må du kigge i manualen. Hvis der går kludder i det, så kø `Memmaker/Undo`.

Du skifter ved et valg med mellemrum, og pilene bruges op og ned mellem valgene.

Brug for expanded memory?

Det handler om linien med EMM386.EXE og dens parametre.

Ved både *express* og *custom* spørges, om et af dine programmer har brug for EMS memory. Hvis du ikke ved det, siger MEMMAKER, at du skal svare **No**. Svarer du **Yes**, skrives RAM i linien, og svares **No** skrives NOEMS.

Det er netop de to muligheder, der skal angives for at give adgang til upper memory. Ved RAM bruges som bekendt 64 KB til en page frame, så det er vigtigt at have disse 64 KB 'til overs' for at undgå brug af conventional til denne page frame.

Custom

Den avancerede mulighed kaldes både Custom og Advanced Options. Manualen er ret kortfattet om den avancerede og hjælpen er ikke meget bedre. Jeg giver i det følgende en mini oversættelse af hjælpen. Husk først at trykke **Enter**, når du har foretaget alle valg. MEMMAKER indsætter forskellige parametre i linien med EMM386.EXE.

Specify . .

Hvis du ikke har haft problemer, mens MEMMAKER arbejder, svares **No**. Hvis du har haft problemer, kan det skyldes et af de programmer, som indlæses i en af startfilerne. Svarer du Ja, promptes du for hver enkelt program for at svare **Yes/No** og dermed finde ud af hvilket, der er problemer med.

Når du har fundet programmet, der giver problemer, skriver du REM foran linien, kører MEMMAKER færdig, og forsøger så ved at slette REM, om programmet nu vil indlæses. Microsoft foreslår at skrive programmets navn

(med eller uden en stjerne foran) i filen C:\DOS\MEMMAKER.INF, hvilket har samme funktion som mit forslag.

Scan the upper ..

MEMMAKER fra DOS 6.0 forsøger default - via EMM386 - at få fat i al ledig plads, hvis du svarer **Yes**, hvilket du skal gøre, hvis du ikke oplever problemer hermed - i EMM386 linien skrives HIGHSCAN. Ellers svares **No**, hvilket bevirker, at et mere sikkert område i upper memory scannes. Prøv først med **Yes**, og hvis det ikke går, kører du igen med **No**. DOS 6.2 har default uden HIGHSCAN, altså mere forsigtig scanning af upper memory.

Move Extended BIOS ..

Svar **Yes** og frigør 1 KB conventional memory. Hvis der senere opstår problemer, prøv at køre MEMMAKER igen, svar **No** her for at se, om det løser problemet. Her handler det om petitesser.

monochrome region ..

Se figur for upper memory s. 9. De fleste har sikkert en såkaldt Super VGA skærm, dvs. i stand til at vise opløsningen 800 x 600. Hvis du har styr på at rette i System.Ini, svares **Yes**. MEMMAKER indsætter parameteren

```
I=B000-B7FF
```

i linien med EMM386.EXE. *Inden* du starter Windows, skrives i Windows' SYSTEM.INI under sektionen [386Enh]:

```
device=C:\DOS\monoumb.386
```

Har du en EGA eller VGA, dvs. max opløsning 640 x 480, kan du også forsøge med **Yes** og se, hvordan det går. Hvis det går godt, har du 'vundet' 32 KB i upper memory, som du kan bruge til at få endnu flere programmer i upper memory, hvis du ønsker dette. Testen består i at starte Windows!

Keep current EMM386 ..

Dette handler om parametrene I=.. og X=.., som enten inkluderer eller ekskluderer områder i upper memory. Hvis du er klar over funktionen af de parametre, der inkluderer og/eller ekskluderer, svares **Yes**, dvs. dine I=.. og X=.. beholdes, ellers svar **No**, dvs. du overlader jobbet til MEMMAKER, som måske kan gøre det bedre.

Optimize .. for Windows

Her handler det kun om de tidligere omtalte translation buffers, data i memory, som Windows behøver for at kunne køre DOS programmer. Jeg troede først, at det

handlede om noget mere, nemlig optimering af hele Windows.

Svares **Yes**, indsættes på min PC parametrene WIN=EA00-ECFF og WIN=ED00-EFFF. MEMMAKER tilføjer altså disse parametre i linien med EMM386.EXE for at reservere dette adresseområde i upper memory til brug for translation buffers, så andre ikke kommer og bruger disse UMBs.

Mit forslag er at svare **No** og med MEM kigger og sørger du for at have 8 K (eller 24 KB, hvis du er på netværk) free i upper memory, inden Windows startes.

Manualen og HELP væver frem og tilbage om DOS programmer under Windows eller ej, og min mening er stadig, at manualen er håbløs. Hvis du laver en CONFIG.SYS med menuer (beskrives under afsnittet *Flere Boots*) og derefter kører MEMMAKER, kan MEMMAKER ikke finde ud af det. Du bliver nødt til at opdele i separate startfiler og så køre MEMMAKER på hver af dem for til sidst at smelte de enkelte optimerede startfiler sammen til én stor. Ret besværligt.

Vigtigt: Memmaker kan heller *ikke* ændre rækkefølgen af devicehigh, hvilket er det vigtigste aspekt af det, der med et fint ord hedder memory management. Det må du selv gøre.

Min generelle vurdering: MEMMAKER er bedre end ingenting, men jeg er ikke imponeret.

Flere boots

F5

Prøv at boote og tryk F5, så snart skærmen viser Starting MS-DOS ..

Nu laves en såkaldt *clean boot*, dvs. startfilerne læses ikke. Hvis du skriver SET ser du, at det handler om en næsten clean boot. F5 er rar, hvis du eksperimenterer med CONFIG.SYS (du slipper for en bootdiskette, hvis problemet blot er fejl i CONFIG.SYS) eller inden du kører et diskoptimeringsprogram el. lign.

DOS 6.2: Hvis du har installeret DoubleSpace, kan du forbigå denne ved Ctrl+F5. Formålet med dette er oftest problemløsning i forbindelse med DBLSPACE.

F8 og ?

Boot igen og tryk nu F8. Nu får du mulighed for at svare Y eller N til hver enkel linie i CONFIG.SYS og derefter om du vil køre hele AUTOEXEC.BAT eller ej. I DOS 6.2 kan Yes/No til hver linie også udføres i AUTOEXEC.BAT.

Skrives et spørgsmålstejn umiddelbart efter en kommando i CONFIG.SYS (fx Numlock?=ON), spørges du, om du vil udføre kommandoen. DOS 6.2: Ctrl+F8 kan på samme vis som Ctrl+F5 forbigå DBLSPACE.

Boot med valg

Brugere med behov for flere forskellige konfigurationer eller en PC, som bruges af flere brugere, har fået det lettere. Du kan gøre opsætning og boot mere avanceret med menuer i CONFIG.SYS, hvor du vælger mellem forskellige muligheder.

Teksterne i linien med menuitem efter kommaet vises på skærmen under boot. Den første tekst er bloknavnet, som henviser til en senere forekommende blok. Navnene skal være ens!

Du kan skrive, hvad du vil, men kun et ord som bloknavn. Det anbefales i øvrigt at skrive en [COMMON] linie sidst i CONFIG.SYS i tilfælde af programmer, der under installation tilføjer en eller flere linier sidst i CONFIG.SYS. DoubleSpace gør dette.

En evt. tom blok med [COMMON] skader ikke. Der kan godt stå linier før [menu], som så er fælles (common). Jeg har prøvet at finde ud af, hvordan man får mest conventional memory til spil, der kræver dette. Nogle spil kræver kun conventional memory, mens andre kan bruge expanded memory.

Variablen %config% antager værdien afhængig af dit valg, og der springes til en såkaldt label, fx :SPIL1

Jeg får 625 KB fri i conventional, og 622, hvis jeg også skaber expanded memory. I AUTOEXEC.BAT har jeg i :SPIL1 angivet SMARTDRV. Du har skabt adgang til upper memory, og kan forsøge om spillet kører hurtigere ved brug af SMARTDRV.

Hvis du kun har DOS 5 og alligevel læser dette afsnit om DOS 6, behøver du ikke fortvivle. Du skaber 3 sæt af dine startfiler, som svarer til det ovenfor beskrevne.

Disse tre sæt startfiler placerer du i dit DOS directory, og giver begge startfiler efternavnene NOR, SP1 og SP2 samt skaber til sidst følgende bat-filer

```
C:\BAT>COPY CON NORMAL.BAT
COPY C:\DOS\CONFIG.NOR C:\*.SYS
COPY C:\DOS\AUTOEXEC.NOR C:\*.BAT
```

trykker F6, Enter og filen NORMAL.BAT er skabt. På samme måde med de to andre filer SPIL1.BAT og SPIL2.BAT. Når du vil boote normalt, skriver du

```
C:\>NORMAL
```

som kopierer dine 'normale' startfiler til C:\, og derefter booter du, og på lignende måde, når du vil spille, med eller uden expanded memory. Du skaber en BAT-fil til hvert spil, og disse bat-filer placeres i fx C:\BAT eller et andet directory, som er indholdt i din PATH. Jeg håber ikke, at beskrivelsen var for kort!

Numlock

NumLock=ON eller OFF i CONFIG.SYS bevirker .. gætt engang.

```
DEVICE=C:\DOS\HIMEM.SYS
DOS=HIGH

[menu]
menuitem=SPIL1, med expanded memory
menuitem=SPIL2, kun conventional
menuitem=NORMAL

[SPIL1]
DEVICE=C:\DOS\EMM386.EXE ram min=0
DOS=UMB
STACKS=0,0
FCBS=1
BUFFERS=10
SHELL=C:\DOS\COMMAND.COM C:\DOS\ /P

[SPIL2]
STACKS=0,0
FCBS=1
BUFFERS=30
SHELL=C:\DOS\COMMAND.COM C:\DOS\ /P

[NORMAL]
DEVICE=C:\DOS\EMM386.EXE NOEMS /V
.. og så videre
[COMMON]
```

Måske har du brug for FILES=15 eller et større tal. I SPIL2 bruger jeg 30 buffers. Du kan prøve forskellige værdier. Som nævnt andetsteds kan MEMMAKER ikke finde ud af at optimere en sådan CONFIG.SYS ligesom programmet heller ikke kan klare en [include] sætning.

```
PATH C:\DOS;C:\BAT;C:\;C:\WINDOWS
goto %config%
:NORMAL
@echo off
.. osv med din 'normale' autoexec.bat
goto slut
:SPIL1
LH smartdrv
goto slut
:SPIL2
:slut
```

MS-DOS 6.2

Den vigtigste grund til udgivelsen af DOS 6.2 er i følge Microsoft forbedring af DoubleSpace i forhold til ver. 6.0, og derfor behandles DoubleSpace i dette afsnit, men først kort om forbedringerne i version 6.2:

SCANDISK er et program, som kan undersøge og reparere fejl på en disk. Det er fra Microsofts side tænkt som erstatning for CHKDSK programmet. Programmet kan også teste og reparere fejl på et drev komprimeret af DBLSPACE.

VERIFY=ON i AUTOEXEC.BAT kan give problemer med visse harddiske, fx Conner. Slet derfor denne linie, hvis den findes.

DISKCOPY kommandoen, der laver en tro kopi af en diskette, kan nu endelig foretage denne operation ved først at sætte original og derefter kopidisketten i drevet og så er disketten kopieret.

COPY, XCOPY og MOVE advarer, hvis du er i gang med at overskrive en fil.

DIR, MEM, CHKDSK og FORMAT angiver nu komma som skilletegn ved 1.000, så det bliver lettere at læse fx 134,245,576 bytes.

DEFRAG kan nu bruge extended memory og dermed håndtere et større antal filer.

HIMEM.SYS understøtter nu 64 MB RAM, og checker default de fysiske RAM chips i extended memory under boot.

Doublespace

I de seneste år er kravet til plads på harddisken vokset voldsomt. Flere firmaer udviklede i 1993, da harddiske stadig var relativt dyre, programmer til at sammenpresse (komprimere) data på harddisken, så de fylder mindre end normalt. Derefter faldt priserne på harddiske! Hvis du har råd til at købe en ekstra harddisk, så gør det og glem alt om dette afsnit om DoubleSpace. Den langsomste harddisk kan du evt. bruge som backup-disk.

Microsoft leverede med DOS 6.0 og 6.2 komprimeringsprogrammet Doublespace (filnavn DBLSPACE).

Når et program lagrer data komprimeres det og når data skal bruges, pakkes de ud (de-komprimeres). Målet er, at brugeren ikke mærker det i form af nedsat hastighed.

Bruge DBLSPACE?

DBLSPACE var nok et af de mest omtalte programmer i 1993. Nogle mente, at versionen fra DOS 6.0 var fejlbehæftet, hvilket Microsoft måske *indirekte* indrømmede ved at lave DOS 6.2. Jeg har kun kort testet DBLSPACE fra DOS 6.2 for at kunne skrive om programmet, og mig bekendt er det uden fejl.

Bliver min PC langsommere? Microsoft svarer på denne måde i HELP (min oversættelse):

"Hvis du har en hurtig CPU og en hurtig harddisk, mærker du sikkert ingen forskel. Har du en hurtig CPU og en langsom harddisk, vil DBLSPACE måske øge ydeevnen. Hvis du har en langsom CPU, reduceres ydeevnen måske."

Spørgsmålene er: hvad er en hurtig CPU osv. Mine gæt er: en hurtig CPU var dengang en 386DX/40 Mhz. En langsom harddisk var en standardharddisk købt før 1992 (med en såkaldt *transfer rate* på 5-700 KB/s) og en langsom CPU var en 386SX. For at finde ud af det, må du prøve!

Hvis du mangler plads på harddisken, kan du jo i første omgang lave den af mig anbefalede installation og se, hvordan det virker. Husk, at du har brug for 38 ledige KB i upper memory, hvis du vil undgå at bruge noget af conventional memory! Erfaringer med DBLSPACE modtages gerne!

Hvis du vil bruge programmet, anbefales inden installation:

- 1 Opgrader til DOS 6.2, fordi DBLSPACE i denne version er forbedret og mere sikkert.
- 2 Tag backup af dine vigtige filer.
- 3 Lav diskoptimering. Vælg muligheden FULL for at skabe maximal ledig diskplads.
- 4 Hvis du er koblet på et netværk, så sørg for at være logget på, *inden* du kører DBLSPACE.

Jeg forudsætter i det følgende, at din harddisk kun består af ét drev, nemlig C, så når jeg taler om *drev*, svarer det til *disk*. Har du DOS 6.2, startes automatisk SCANDISK, inden selve komprimeringen foretages.

Hvis du vil vide lidt mere, kan du læse manualen eller skrive Help Dblspace for at hente informationer, inden du starter. DBLSPACE kan installeres på to måder.

C:\>DBLSPACE

Første skærbillede fortæller, at Setup indlæser filen DBLSPACE.BIN (en systemfil), som kræver ca. 40 KB i memory. Næste skærbillede fortæller de to mulige måder, *express* og *custom*, og at *express* er den letteste, mens *custom* er for den øvede bruger.

Det er rigtigt, men Microsoft fortæller *ikke*, at det er sværest at fortryde en *express* installation, dvs. komprimering af *hele* drev C: Du kan via

express komprimere et helt drev (hvis du kun har C: er det altså hele harddisken)

eller via *custom* oprette et nyt drev på et allerede eksisterende drev (den ledige diskplads bruges).

De fleste brugere vælger sikkert at komprimere hele harddisken ved hjælp af muligheden *express*, hvilket er det enkleste, men det er det sværeste at ændre, dvs. *fortryde*. Jeg vil anbefale dig (for at få nogle erfaringer

eller hvis du er lidt usikker i starten) at vælge *custom*, som er let at fortryde, hvilket beskrives senere.

Det nye drev, uanset om du vælger den ene eller anden metode, skal have et bogstav som drevbetegnelse. DBLSPACE springer over 4 bogstaver fra det sidste rent faktiske drev. Hvis dit sidste logiske drev er C, bliver næste ledige bogstav H, og på denne måde bliver der plads til andre programmer, der gerne vil eller måske i øjeblikket benytter drev udover de rent fysisk installerede, såsom ramdisk eller netværk.

Der er forskel på det nye drev H, afhængig af metode, men det vender jeg tilbage til.

Custom

Jeg vil anbefale, at du starter med at lave et lille DBLSPACE drev for at få lidt erfaring, specielt for at finde ud af, om din PC bliver langsommere. Hvis du så senere beslutter, at du vil droppe DBLSPACE, kan du let gøre det og vende tilbage til 'normal tilstand'. Hvis du i stedet vil komprimere *hele* harddisken, kan du rimelig let lave en *custom* installation om, og på det tidspunkt har du sikkert mere viden om DBLSPACE. Du vælger altså *custom*.

Forklaringen på næste skærmbillede er forvirrende. Programmet giver to valg

Komprimere et eksisterende drev

Skabe et nyt tomt komprimeret drev

(Hvis du kun har ét drev, er den første mulighed jo det samme som *express*, og vi har jo lige fortalt programmet, at vi ikke ville *express* installationen. Programmet burde teste, om PC'en kun har ét drev.)

Jeg anbefaler, at du med **Enter** vælger muligheden **Skabe et nyt...**, hvorefter næste skærmbillede giver et tal for ledig diskplads (current free space) og anslået størrelse på det nye drev (projected size of new drive). Skriv det første tal ned, du får brug for det senere.

Tast **Enter** og næste skærm fortæller, at det nye komprimerede drev H vil blive skabt ud af den *ledige* plads på C. Nu kommer det vigtige. Øverste linie i rammen siger, at der vil blive efterladt sølle 2 MB på drev C (af den ledige plads - der sker intet med dine filer på C). Hvis du har fx 50 MB fri plads, efterlades 2 MB i ukomprimeret form. De resterende 'normale' 48 MB vil DBLSPACE så komprimere til at kunne rumme ca. 96 MB.

Jeg anbefaler, at du ændrer de 2 MB til et større tal, så du kun laver et nyt komprimeret drev, der kan rumme 20 MB. Hvis du havde 50 MB fri, piler du op på de 2 MB, taster **Enter**, og i næste skærmbillede skriver du 40 og taster **Enter**, så har DBLSPACE fået $50-40=10$ MB af den frie plads til at skabe et nyt drev H.

Det er dårligt, at hvert skærmbillede ikke viser den ledige diskplads. DBLSPACE er det store trækplaster i DOS 6! Brugervenlighed?

Du kan 'snyde' lidt ved at skrive et alt for stort tal som fx 999 og programmet fortæller, hvor stort tallet max må være, nemlig den ledige diskplads. Programmørerne *har* altså testet, de gider bare ikke minde brugeren om det!

Sidste skærmbillede, før det går løs, fortæller at programmet nu er klart og du bedes trykke C for at fortsætte (continue). Da jeg testede, mente programmet, at det ville tage 53 minutter (det tog ca. 10). Nu starter SCANDISK, og findes fejl (*lost clusters* eller *crosslinked files*), må du kvitte programmet og først reparere disse (se s. 48 CHKDSK). Hvis du gør dette med SCANDISK, behøver du ikke også køre en *surface scan*, da DBLSPACE senere alligevel gør dette. Derefter må du starte forfra med DBLSPACE.

(A) Nu laves af SCANDISK en *surface scan*, og er den OK, bootes og filen C:\DBLSPACE.INI skabes. DBLSPACE.BIN kopieres fra C:\DOS til C:\ og begge filer sættes med attributterne: system, hidden og read-only.

Sidste skærm fortæller det nye komprimerede drevs størrelse og den resterende ledige diskplads på C. NB: drev C er stadig drev C, men er *vært* (eng.: host) for det nye komprimerede drev H.

Express

Hvis du vælger denne *lette* metode, sker der *umiddelbart* ikke andet end at din harddisk bliver større og måske langsommere. Du kan bruge din PC lige som du altid har gjort, men hvis der går kludder i noget eller du senere fortryder, kan det være relevant at vide, hvad der foregår.

(A) DBLSPACE opretter en nyt drev, typisk H, kaldet *Host Drive* (det er tilfældigt, at H er første bogstav i *Host*).

Dette drev H er ikke komprimeret og indeholder de rent faktiske data fra C: i én stor *komprimeret* fil kaldet **Compressed Volume File**. NB: Drev H er vært for drev C, som på drev H kaldes DBLSPACE.000, som 'swappes' (skiftes) til drev C under boot (før indlæsning af CONFIG.SYS), fordi DBLSPACE.BIN er en del af styresystemet i DOS 6, dvs. DBLSPACE.BIN er en systemfil på linie med de to andre systemfiler. Opretter du flere af DBLSPACE skabte drev, bliver efternavnet DBLSPACE.001 osv.

På drev H placeres også de andre systemfiler, dvs. bootfiler, andre skjulte filer fra andre programmer samt en evt. permanent swapfile fra Windows. Disse filer kan nemlig ikke eksistere i komprimeret form. Rør endelig ikke de filer, der tilhører DBLSPACE. Du risikerer at miste samtlige dine filer.

Windows permanente swapfil kan kun eksistere på et ukomprimeret drev. Hvis du derfor inden installering af DBLSPACE havde en permanent swapfil, skulle denne under installationen være flyttet til H.

Hvis du inden installation af DBLSPACE *ikke* havde en swapfile (måske har du slet ikke Windows) og senere vil oprette en sådan, bliver du måske (jeg har ikke selv prøvet) nødt til at skaffe mere ledig plads på drev H, hvilket du gør ved at starte DBLSPACE og under menu **Drive**, **Resize** øger du den ledige plads på drev H, så der er plads til swapfilen. Det samme skulle kunne aktiveres lettere med

```
C:\>DBLSPACE/SIZE/REServe=8
```

hvor **reserve=8** betyder: reservér 8 MB på det ukomprimerede værtdrev.

Custom og Express

For begge installationer gælder:

I CONFIG.SYS tilføjes som den sidste linie
devicehigh=c:\dos\dblspace.sys/move

Flyt linien, så den står som den første devicehigh linie, hvorved du er sikker på at DBLSPACE.BIN indlæses i upper memory

(A) DBLSPACE.SYS indlæser DBLSPACE.BIN, og hvis linien i CONFIG.SYS er buffers=8 eller mindre, er du ret sikker på, at der er plads i High Memory til en del af DBLSPACE.BIN. (se buffers= s. 19)

Generelt

Komprimering af data er enkelt forklaret: forestil dig et dokument med mange ens forekommende ord (som, der,

og osv.). I stedet for hele tiden at lagre ordene, kan man lave en meget kort kode for hvert ord, og så lagre kode plus en henvisning. Risikoen ved fejl er højere end ved 'normal' lagring, hvis der går kuk i den komprimerede datafil. Det svarer nogenlunde til at have samme kode til flere hundrede creditcards - og så glemme koden!

DBLSPACE.BIN er både 'disk space manager' og et program, der kan komprimere og de-komprimere. Da DBLSPACE er en integreret del af styresystemet, indlæses DBLSPACE.BIN *inden* CONFIG.SYS læses. DBLSPACE.BIN indlæses i første omgang i conventional memory, men læses senere (via den tilføjede linie i CONFIG.SYS) i upper memory.

SMARTDRV kan fange data 'bag' DBLSPACE, hvilket betyder at cachen kan indeholde relativt større datamængder.

Undlad at bruge Nortons Disk Doctor på et DBLSPACE drev - brug i stedet SCANDISK.

Fortryde

Det er let at fjerne et komprimeret drev, men det kræver, at du har plads nok andetsteds på din harddisk til de ukomprimerede versioner af de filer, som er komprimeret. Du bliver måske nødt til at flytte nogle filer til disketter eller tape.

Du starter DBLSPACE, vælger **TOOLS** fra menuen og herunder **uncompress**. Hvis der ikke findes flere komprimerede drev, tilbydes at afinstallerede DBLSPACE totalt, hvilket du siger **Yes** til. (Reelt slettes C:\DBLSPACE.INI, C:\DBLSPACE.BIN og det komprimerede drev DBLSPACE.000. I CONFIG.SYS slettes linien med DBLSPACE.SYS)

Windows f. Workgroups 3.11

Dette afsnit er placeret her, fordi programmet udkom efter DOS 6.2. Windows for Workgroups 3.11 er en mindre opdatering i forhold til Win 3.1 og kan anvendes på en PC, uanset den er koblet på netværk eller ej. Det handler primært om hurtigere kommunikation med harddisken via et 32 bit *fil-adgangs* system (en disk cache), men denne version skulle også være mere stabil. Win 3.11 indeholder bl.a. et mindre faxprogram og en ny hurtigere printmanager.

Installationen tilføjer i CONFIG.SYS

```
device=C:\WINDOWS\ifshlp.sys
```

som du bør ændre til `devicehigh`. Win 3.11 leverer de samme *versioner* af de 'vigtige' filer som DOS 6.0/6.2: HIMEM.SYS, EMM386.EXE og SMARTDRV.EXE, men ændrer i startfilerne det directory, hvorfra de kaldes, til Windows directory, typisk altså C:\WINDOWS. Det er OK, men pas på, hvis du derefter kører MEMMAKER, som ændrer det til C:\DOS. Hvorfor gør MEMMAKER det?

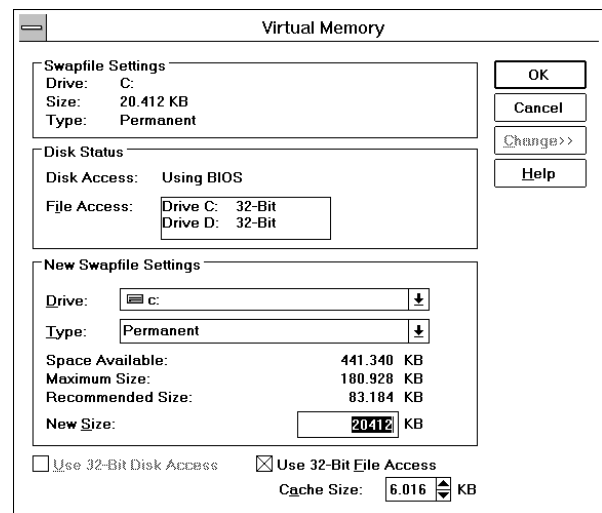
Mit gæt: Da DOS 6 blev introduceret, ville Microsoft 'hjælpe' Win 3.1-brugeren, der installerede DOS 6, til det 'rigtige' valg: nyere og bedre versioner af filerne fra C:\DOS. Hvis du *ikke* regner med at skulle ændre din DOS-installation, kan du *kopiere* filerne fra C:\WINDOWS til C:\DOS. I modsat fald må du ændre dine startfiler efter at have kørt MEMMAKER.

Det nye 32-bit filadgangssystem er en cache, som erstatter SMARTDRV, når Windows kører. I den dialogboks, hvor swapfile vælges (Control Panel, Enhanced, Virtual Memory, Change), vil du se de drev, hvor den nye cache benyttes, når du har sat kryds i 32 bit file access. Du vil også se, hvor stor cachen er. Programmet vælger en værdi i forhold til din ledige extended memory på samme måde, som også SMARTDRV gør. Installationen ændrer i AUTOEXEC.BAT den parameter (den anden) for SMARTDRV, der gælder SMARTDRV's cache størrelse under Windows, til 128. Med 8 MB RAM ændres til:

```
\..\smartdrv 2048 128 /X
```

En af forskellene på SMARTDRV og den nye med 32 bit filadgang (VCACHE.386): SMARTDRV læser nogle ekstra data fra de *næste sektorer* (i forhold til de sidst læste, som et program havde brug for) ind i sit lager i håbet om, at de næste data, som behøves, befinder sig i de næste (fysiske) sektorer på disken.

Den nye cache indlæser de næste data i forhold til den sidst læste *fil*, dvs. i håbet om, at de næste data, som behøves, er fra samme fil. Da dette er meget sandsynligt, fås altså flere 'hits' i forhold til 'missere'.



Denne nye cache har størst virkning, når man sjældent eller aldrig kører et diskoptimeringsprogram! Kører du regelmæssigt diskoptimering er denne fordel næsten nul.

(jeg har ikke kryds ved 32 bit disk access, fordi min disk controller kører 32 bit)

En anden forskel er, at den nye cache flytter nogle processer fra *real mode* til *protected mode*, hvilket betyder, at det hele går hurtigere.

Min erfaring: Win 3.11 kører lidt hurtigere med harddisken, men ikke meget. Derimod er andre funktioner som fx åbning og lukning af vinduer, rulning i dialogbokse osv. noget hurtigere.

Optimeringstips

Du har sikkert fundet ud af, at der findes ingen generelle svar på den optimale PC. Vi bruger alle vores PC på hver vores måde. Det bedste råd er: prøv dig frem. Du vil lære noget undervejs og bliver sandsynligvis tvunget til at tænke over nogle ting, som du måske ikke har tænkt på før - det skader ikke. Dette er min erfaring og resultatet er denne bog!

Generelt kan siges, at jo mindre RAM, du har, desto vigtigere er udnyttelsen af memory. Jeg er klar over, at jeg har skrevet både små og store tips, så her de vigtigste:

- 1) DOS=HIGH
- 2) Disk Cache
- 3) Optimering af harddisken
- 4) Permanent Swapfile

2 MB RAM

Har du en 286 og en upper memory manager til din PC eller en 386SX/SL (her kan EMM386 skabe UMB, og DOS kan håndtere dem), gælder det om at indlæse så meget som muligt i UMBs. Har du et fysisk såkaldt add-on EMS-kort med RAM, bør du kunne udnytte denne EMS RAM med den med kortet medfølgende EMS-driver. Se medfølgende manual for kortet og/eller software. Du indlæser sikkert DOS=HIGH og sikkert også DOS=UMB. Alt afhængig af, hvor meget XMS memory, du låner til at fungere som UMB, har du måske ca. 800 KB RAM tilbage. Så må du vurdere, hvordan du vil bruge dem. Hvis du kun kører DOS programmer, så brug en pæn del - måske 512 KB - til en disk cache, fx SMARTDRV.

Cachens størrelse - de 512 KB - tages af extended (eller expanded) og berører ikke conventional memory. Hvis du ikke bruger programmer - fx Windows - der kan bruge extended memory, kan du evt. bruge al ledig extended memory til din disk cache.

Bruger du Windows, må du finde en balance mellem størrelserne (min. og max. værdier) for SMARTDRV og den resterende ledige extended memory, som Windows så kan bruge. Jeg har ikke prøvet, men 400 KB til hver er nok ikke helt ved siden af, men prøv dig frem.

4 MB Du har nok min. en 386DX. Teksten skulle gerne hjælpe. SMARTDRV tager som skrevet default 1024 KB til cache under DOS og 512 KB under Windows, hvilket nok er fint for de fleste. Hvis du bruger Windows, anbefaler jeg at eksperimentere med større cache under Windows, dvs. i AUTOEXEC.BAT skriver du fx

```
smartdrv 1024 768          eller
smartdrv 1024 1024
```

Hvis du typisk kun bruger et eller to ikke alt for memory krævende programmer, kan det godt være, at du får disse programmer til at køre hurtigere ved en større cache end 512 KB.

6-8 MB Det er ikke let at give generelle råd om størrelsen af disk cache og evt. ramdisk. Det afhænger af, hvor mange samtidige programmer, du normalt kører under Windows, og det afhænger også af, *hvilke* programmer du bruger. Det letteste for mig er selvfølgelig at sige, at du må prøve dig frem, men jeg har 8 MB og kører ofte med Winword, en database og måske NC og/eller Winfax åben, og jeg får hurtigste resultat med SMARTDRV sat til 2048 2048, hvilket er default ved 8 MB RAM, dvs. jeg bruger ingen parametre.

Jeg havde med 8 MB glæde af generelt hurtigere Windows og mulighed for at køre flere programmer samtidigt. Der skal ikke swappes så meget med den permanente swapfil.

Udnyt upper memory

(A) Hvis du vil have mange programmer i upper memory, gælder det om at skabe et stort område med sammenhængende upper memory blocks - UMB. Du skal ikke via en evt. I= . . . inkludere mere, end du har brug for.

Dernæst gælder det om først at indlæse de programmer, som fylder mest, og så mindre og mindre. Det første program fra CONFIG.SYS, der indlæses, placeres i den største ledige UMB, næste i største nu ledige UMB osv., hvilket betyder 'huller'.

Du kan med MEM/C/P nogenlunde se, hvor meget hvert program fylder. Brug evt. også MEM/D/P. Du bliver måske nødt til at ændre rækkefølgen af devicehigh og loadhigh.

Tricky: du kan tilsyneladende have plads i upper memory til et program, som du tidligere har fundet ud af, fylder et bestemt antal KB, men programmet vil alligevel ikke op i upper memory. Årsag: der indlæses ikke direkte i upper memory. Først indlæses i conventional, derefter checkes for plads i upper, og hvis ja, så indlæses i upper, men..

programmerne bruger under indlæsning til upper memory mere plads for at komme 'derop' end de til sidst fylder. Bagefter afgives så den plads i upper memory, som programmet nu ikke længere har brug for.

I MEMMAKER.STS angives ved *MaxSize* størrelsen på det antal bytes, som programmet kræver for at komme 'op'. Hvis du ikke kan få et program i upper, så undlad nogle andre midlertidigt, kør MEMMAKER og se derefter i MEMMAKER.STS.

Det vigtigste er altså rækkefølgen af devicehigh og dernæst af LH i forsøget på at få så mange programmer

som muligt i upper memory. MEMMAKER kan *ikke* hjælpe på denne rækkefølge. *Du* bestemmer.

Hvis du bruger Windows og ikke har installeret netværksdriver: indlæs ikke mere i upper memory, end at du har min. 8 K ledig i upper memory. Har du installeret netværksdriver, skal dette tal være 24.

Windows i enhanced mode har nemlig brug for såkaldte translation buffers, og de fylder enten de 8 eller 24 KB. Translation buffers bruges til at overføre data mellem real og protected mode - DOS kører i real mode og Windows kører i protected mode, og når et DOS program kører i et DOS vindue under enhanced mode sker dette skift mellem real og protected mode, og her er der altså brug for disse buffers, midlertidige lagerpladser for visse informationer eller data.

Windows: Det gælder om i startfilerne kun at indlæse de programmer (drivere, residente programmer osv.), som *alle* dine programmer har brug for. Hvis du kun har brug for et resident program, mens du kører et *bestemt* DOS program under Windows, er det smart at lave en bat-fil til dette DOS program, hvor du indlæser det residente program, *inden* du starter DOS programmet. Når programmet afsluttes, forsvinder det residente program så fra memory.

Kig i memory - MSD.EXE

(A) Hvis du har programmet MSD.EXE (fra Win 3.1 eller DOS 6), kan du køre det i DOS og efter start trykke M og her se, hvordan upper memory udnyttes. Det grå er det reserverede systemtekniske. **F=Free**, **U=Used**. Da PS/2 har motherboard BIOS ROM i området E000-EFFF inkluderes dette område *ikke* default af EMM386.EXE i DOS 5.

På de fleste kloner er dette område imidlertid ikke optaget af systemtekniske ting og går 'til spilde' i udnyttelsen af *upper memory*, såfremt det ikke inkluderes. Dette gælder kun, hvis du har brug for en del af disse 64KB yderligere i upper memory. Parameteren **I=E000-EFFF** inkluderer dette område. Herved 'vindes' 64 KB i upper memory, men check først med MSD, om området er frit (vist med F).

Vælges **Utilities - Alt+U** - og **Memory Block Display** kan du se i den første MB - måske bortset fra E og F pagen, altså fra E000. Hvis du har startet Windows eller andre programmer og kan skifte til MSD eller andet program, der kan vise memory, kan du få info om, hvor i memory de forskellige programmer eller filer er placeret. Vær dog opmærksom på (som skærmeddelelsen også siger), at oplysningerne måske ikke er helt korrekte, fordi du samtidigt har Windows kørende. Men alligevel er det smart, når du vil checke, om en ændring virkede.

Du kan få mange informationer om din PC ved at vælge **Alt, File, Print, File** og **Enter**. Rapporten findes i filen REPORT.MSD. Næsten samme resultat opnås ved at skrive

```
C:\>MSD/P sysinfo.doc
```

hvor filnavnet selvfølgelig er mit forslag, men skriv (som ved alle andre DOS kommandoer) for at få info om mulighederne

```
C:\>MSD/?
```

DOS

Slette alle filer i et directory

En anden læser fortalte (tak) en smart måde til at slippe for at vente på en af de berømte *Are you sure* meddelelser - jeg ved ikke, hvad Microsoft og programmører tænker om os brugere og vores IQ. Lav en BAT-fil som fx hedder SLET.BAT med indholdet

```
echo y | del *.*
```

Har man installeret et stort program, som opretter mange underdirectories, og man vil have det fjernet fra disken, er der med DOS 5 kun den tunge vej. Start 'baglæns' med 'dybeste' directory og slet ét directory ad gangen. Windows File Manager, NC 4.0 og DOS 6 kommandoen DELTREE kan slette et directory med tilhørende underdirectories i ét hug.

```
C:\>DELTREE C:\EXTRA
```

og svar Y.

Du kommer sikkert ofte ud for at skulle slette samtlige filer på en diskette. Jeg plejer lige at checke, hvad jeg egentlig er i gang med at slette, dvs. jeg bruger NC og Alt+F1, A, kigger på filerne, skriver i prompten SLET for at køre min bat-fil. Hvis der er mange directories på disketten, kan du overveje at lave en *quick-format* af disketten, dvs.

```
C:\>FORMAT/Q A:
```

eller en UN-conditional formatering

```
C:\>FORMAT/Q/U A:
```

som går endnu hurtigere, men du kan ikke bruge UNFORMAT efter en sådan formatering.

CHKDSK/F

er omtalt flere steder i teksten. Før DOS 6.2, hvor dette program er blevet 'erstattet' af SCANDISK, kan dette benyttes til at reparere fejl på disken. Hvis du har DOS 6.2, bruger du SCANDISK i stedet for CHKDSK.

Typiske fejl (fra CHKDSK) på en disk er *lost clusters* og *crosslinked files*. Lost clusters er stumper af data, som ikke har noget navn knyttet til dem, mens crosslinked files betegner den tilstand, når to filer deler eller henviser til samme sted på disken.

Lost clusters: Hvis der findes fejl, svar Yes til at fixedem, undersøg filerne (FILE0000.CHK osv. og ligger i roden), om du vil beholde dem (omdøb filen eller filerne til et sigende navn), ellers slet dem.

Crosslinked files: Hvis CHKDSK siger, at der er crosslinked files, så notér navnene på filerne, kopiér dem med nyt navn til et andet sted på disken og slet originalerne. Dermed fjernes krydshenvisningen til samme sted på disken.

Fra , til . i DOS programmer

Hvis du indtaster mange tal fra det numeriske tastatur og gerne vil have Del-tasten til at skrive punktum og ikke komma, som den gør ved en dansk opsætning af tastaturet, kan du med Ctrl+Alt+F1 og Ctrl+Alt+F2 skifte mellem den standard, din PC kommer med, og den, du har sat den til via dine startfiler. Det virker også, mens du er i et DOS vindue under Windows. Du mister muligheden for æ, ø og å, mens du har punktum, men du kan bare skifte frem og tilbage efter behov.

FDISK

Dette program, som bruges til at partitionere en harddisk, inden den formateres (beskrives i et kommende hæfte *Opgrader din PC*) har en udokumenteret switch, der er ganske harmløs, men i nogle tilfælde kan fjerne en virus i master-boot-record.

```
C:\>FDISK/MBR
```

Boot disketter

Efter at du nu ved noget om startfilerne, giver jeg nogle forslag til nogle disketter, som du kan få brug for, hvis du får problemer. En *boot-diskette* kaldes også en system-diskette. En setup-diskette er noget andet, nemlig den diskette, der installerer (kopierer) et program (i dette tilfælde DOS) fra diskette til harddisk. Du har måske en setup-diskette, men ikke nødvendigvis en boot-diskette.

Erfaringen viser, at det er en god ide at forsikre sig - især hvis du under *DOS 5* begynder at eksperimentere med indholdet i CONFIG.SYS, skal du lave en sådan diskette. Jeg har flere gange ikke været i stand til at boote fra harddisken, mens jeg eksperimenterede. Indholdet i filen CONFIG.SYS kan få din PC til at låse, dvs. standse boot-processen totalt. DOS 6 har heldigvis løst dette problem.

En manglende eller ødelagt COMMAND.COM kan også få den til at låse. Jeg har engang blot 'åbnet' COMMAND.COM med editoren i Norton Commander (man skal jo lige kigge) og gået ud igen uden at pille. Min PC strejkede og jeg kunne ikke boote fra harddisk. Rør *aldrig* COMMAND.COM.

En PC kan desværre ikke blot tændes og så virker den - det ville ellers være rart. Styresystemet skal være indlæst i PC'ens hukommelse.

Selv om du har DOS, og én af setup-disketterne kan få din PC til at starte, er en boot-diskette ikke helt det samme. Du anbefales derfor at lave de følgende disketter. En PC kan kun boote fra drev C eller A. Mange disketter er allerede formateret ved køb, men hvis du har en *uformateret* diskette, formateres den med

```
C:\>FORMAT A:
```

Diskette 1 - clean boot

Du sætter en *tom, formateret* diskette i diskettedrevet og skriver

```
C:\>SYS A:
```

Der meldes `system transferred` hvilket betyder, at to skjulte system- eller bootfiler og COMMAND.COM er blevet kopieret til disketten (hvis du har DOS 6, kopieres også DBLSPACE.BIN). Disse filer er nødvendige for at DOS kommer i gang. Bruger du DoubleSpace, er DBLSPACE.BIN nødvendig, hvis du vil have adgang til et drev komprimeret af DoubleSpace.

Kopier filen SYS.COM til disketten:

```
C:\DOS>COPY SYS.COM A:
```

Skriv på disketten *Clean Boot*. Nu har du en bootdiskette, hvormed du altid kan starte PC'en, hvis den ikke kan starte fra harddisken. Denne boot disk indeholder kun de 3 eller evt. 4 vigtige filer fra styresystemet, og booter du

med denne diskette, laves en såkaldt *clean boot*. NB: visse tegn fx kolon (Æ) findes nu på andre taster, fordi den danske keyboarddriver ikke er indlæst.

Hvis du vil se filerne på disketten kan du skrive denne udokumenterede DIR med komma, som også viser skjulte filer

```
A:\>DIR,
```

Visse spil kræver en clean boot fra diskette, og så kan denne diskette benyttes. Disketten kan løse to problemer:

Problem 1

Du kan ikke boote fra harddisken og får fejlen Non-System disk. Hvis problemet blot er, at der er noget galt med én af systemfilerne eller COMMAND.COM, kan du efter boot med disketten skrive

```
A:\>SYS C:
```

som kopierer de 3 eller 4 filer til C:\ Hvis meldingen er `system transferred`, skulle du igen kunne boote fra harddisken.

Problem 2

Hvis du får fejlmeddelelsen `missing or bad Command Interpreter`, er det sket fejl på COMMAND.COM. Sker dette, booter du med denne diskette og skriver

```
A:\>copy command.com C:\
```

```
A:\>copy command.com C:\DOS
```

som kopierer en velfungerende COMMAND.COM fra disketten til de to directories på harddisken, hvor den er relevant. Måske har du kun brug for COMMAND.COM det ene sted. Fjern disketten fra drevet og boot igen fra harddisken.

Diskette 2

Tag endnu en formateret diskette. Gør det samme som ved den første diskette.

```
C:\>SYS A:
```

Denne diskette skal lave samme boot som en boot fra harddisken, blot fra disketten. Alle nødvendige filer skal derfor findes på disketten, og her kommer ideen fra startfilerne uden angivelse af C: foran kommandoerne. Startfilerne er ens bortset fra PATH.

De følgende mange kopieringer er selvfølgelig lettere med et program som NC, men for at alle kan være med, kommer her DOS kommandoerne. Hvis du lige nu er i stand til at boote fra harddisken, så kopier startfilerne til disketten

```
C:\>copy config.sys a:
```

```
C:\>copy autoexec.bat a:
```

```
Ret i A:AUTOEXEC.BAT til path=\; \DOS
```

(De følgende kommandoer er måske uvante for nogle, bl.a. fordi jeg taler om *at sætte et directory som default* med CD kommandoen, men det er rent faktisk det, som CD gør. Det for mange måske pudsige er, at jeg opretter/skifter directory på A: uden 'at være der eller stå på A:')

Opret directory A:\DOS.

```
C:\>MD A:DOS
```

Sæt dette directory som default på A:

```
C:\>CD A:DOS
```

Sæt C:\DOS som default på C:

```
C:\>CD DOS
```

Kopier til A:\DOS de filer, som optræder i dine startfiler. Brug evt. F3 efter hver kommando og ret lidt i den forrige kommando. De første kommandoer kan se sådan ud (A:\DOS er default!):

```
C:\DOS>copy himem.sys a:
```

```
C:\DOS>copy emm386.exe a:
```

```
C:\DOS>copy display.sys a:
```

osv. indtil alle filer, som optræder i dine startfiler er kopieret.

Denne diskette kan boote din PC med samme opsætning som en boot fra harddisken, men uden brug af filer fra harddisken. Kopiér på lignende måde også disse filer fra C:\DOS til A:\DOS: UNFORMAT.COM, FORMAT.COM, CHKDSK.EXE (SCANDISK.EXE), UNDELETE.EXE, FDISK.EXE og SYS.COM. Kopier evt. også andre hjælpe-programmer til disketten. Check, at den virker. Skrivebeskyt den og gem den godt.

Disketten kan bruges i det tilfælde, hvor din harddisk går så meget i stykker, at du ikke kan få adgang til filerne på harddisken, dvs. ikke kan *læse* filer på harddisken (cannot read drive C: eller error reading drive C:).

Hvis du installerer fx CD-ROM eller lydkort eller andet hardware, der samtidig tilføjer linier til dine startfiler, skal du huske at kopiere de nye startfiler til denne boot-diskette.

Iøvrigt anbefales månedligt at teste dine bootdisketter! Husk det nu. Det er surt show, når du har brug for den efter et års tid og disketten er blevet "dårlig". Det sker nemlig!

Altså: ajourfør dine bootdisketter. Arbejdet er minimalt i forhold til det, du evt. sparer.

Hvis det store uheld kommer og eneste løsning er en evt. ny partitionering og/eller en evt. formatering eller en UNFORMAT af harddisken, kan denne diskette benyttes. Det er heldigvis de færreste, der oplever dette, men den, det rammer, er ligeglædet med, at det kun rammer de få. Hvis du kan få hjælp, så få det i dette specielle tilfælde.

Din ekspert vil langt bedre kunne hjælpe dig i denne kritiske situation. Hvis du selv er eksperten, der hjælper andre, anbefales en sådan diskette. Sæt disketten i og boot for at se, om din PC booter uden problemer.

Manglende kontakt med harddisken kan også skyldes noget, som beskrives i følgende afsnit.

CMOS og Setup

(A) En sikkerhed, som manualer sjældent fortæller, er at nedskrive informationer fra den såkaldte CMOS (udtales *semos*), som er RAM indeholdende informationer om PC'ens konfiguration, bl.a. harddisken. Informationerne bliver holdt i live af et batteri, som regel genopladeligt. Hvis det uheldige skulle ske, at din harddisk af en eller anden grund ikke virker, og årsagen måske er, at værdierne i CMOS er blevet ændret eller nulstillet, så skal du kende værdierne fra CMOS for at få den til at virke igen.

Det sker heldigvis sjældent, og risikoen er mindre, jo nyere din harddisk er. Min harddisk blev engang smadret, da jeg blev nødt til at trykke på Reset-knappen, mens jeg var i Windows. Jeg kunne ikke gøre andet, og klik sagde det på harddisken. Ingen af mine nødhjælpsprogrammer kunne klare den. FAT, Boot Record osv. var forsvundet. Jeg kunne starte forfra med at formatere harddisk, installere programmer og lægge mine data ind. Jeg havde næsten det hele på disketter! Jeg lærte at tage sikkerhedskopier.

Du burde med din PC have fået et lille blad fra harddisk-fabrikken. Kender du type-betegnelsen på din harddisk, står værdierne heri. Hvis du ikke har dette blad, så spørg din forhandler, hvad værdierne er.

Du kan også kigge i din hardware SETUP, men gør kun følgende, hvis du føler dig tryk ved det. Boot med Reset-knappen. På skærmen plejer at stå, at du kan trykke på en eller flere taster for at komme ind i SETUP, hvor CMOS befinder sig. Esc, Del eller evt Ctrl+Alt+Esc er de mest benyttede tastetryk. Hvis det ikke står under opstart, står det i en manual for din PC.

Du skal finde CMOS setup eller noget med Fixed disk, skrive værdierne ned for din harddisk: type (oftest *user defined* nr 46 eller 47), antal cylindre, heads og sectors. Skriv på en label og sæt den på kabinetet. En af mine gamle harddiske har følgende værdier: type 47, 1024 cylindre, 10 heads og 17 sectors. Typisk er at bruge F10, Esc eller lignende til at komme ud igen. Du bliver måske spurgt om du vil lagre - Do you want to save changes? Svar **No**. Hvis du er tvunget til at svare Yes, så gør det, men vær sikker på, at du ikke har ændret nogle af værdierne.

Hvis du kommer til at ændre værdier eller rode dig ud i noget, hvor du ikke ved, hvad der vil ske, så tryk på Resetknappen igen. Hvis man ved, at noget endnu ikke er lagret, er det altid den sikreste måde at komme ud igen til den 'gamle' tilstand: Reset maskinen!

Nogle setups har både standard setup og advanced setup. Du skal vide, hvad du gør. Du skal endelig ikke ændre værdier bare for sjov eller for at se, hvad der sker, og din forhandler bliver ikke glad for dig, hvis du har gjort noget, du som bruger helst ikke må. Jeg har prøvet! Men generelt: vær forsigtig med setup, specielt advanced setup, som jeg ikke vil behandle i denne *generelle* tekst.

Blindskrift

Det bedste råd, jeg måske kan give, har ikke direkte med edb at gøre. Jeg brugte 30 timer på en gammel rejseskribe-

maskine til at lære blindskrift. Jeg skulle slå hårdt med lillefingeren, men i dag er det en smal sag på tastatur. Der findes bøger og programmer, der tvinger dig til at træne fingermusklerne.

Jeg har sparet mange timer og har fået skrevet ting, som jeg ellers ikke ville have skrevet. Jeg skriver sjældent forkert. Jeg kender ikke et godt tastatur til rimelig penge (IBM's er stadig suverænt, men dyrt).

Din sundhed

Skærm

Jeg har min skærm på en lille kasse, så jeg kigger svagt nedad, dvs. næsten i øjenhøjde. Hvis du kan blindskrift, skal du sjældent kigge på tastaturet, hvilket sparer dine nakke- og skuldermuskler. Hvis du lærer blindskrift, kommer du sandsynligvis til at stille større krav til tastaturet.

Skru lige netop så højt op for lysstyrke og kontrast, at du ser tydeligt og tilstrækkeligt. Til Windows-brugere: prøv Control Panel, Color, LCD default/standard. Anbring skærmen så langt væk som muligt - min er 75 cm væk.

Håndklæde

Dit håndled bliver belastet, hvis du ikke beskytter det. Jeg har foldet et håndklæde et par gange og lagt det lige foran tastaturet - fin håndledsstøtte.

Støj

Ventilatoren og harddisken larmer, også med såkaldt Low Noise. Jeg er specielt sensitiv over for de højfrekvente lyde fra harddisken. Sæt selve computerkassen så langt væk som muligt, evt. ned under bordet - forlængereledninger til skærmens forbindelse til computeren (både strøm og data) kan købes. Prøv med tæpper, håndklæder eller andet for at dæmpe støjen uden selvfølgelig at hindre gennemstrømningen af luft. Der skal lidt køling til, men her i Norden har vi ikke brug for den samme køling, som de har i troperne. Prøv dig frem.

Stol

Anskaf en god kontorstol at sidde på, hvis du sidder længe ad gangen. Jeg har i pressede situationer siddet 12 timer i træk. Det skal du ikke prøve. Jeg blev syg til sidst. Bevæg dig en gang imellem, hold en pause, gå en tur, vask op, lav gennemtræk, knæ- og armbøjninger, lav noget helt andet.

Lyt til din krop.

Uanset hvor spændende det er, hvis din krop siger, at den skal have en pause, så giv kroppen det. Hvis du ikke gør det, skal den nok på et eller andet tidspunkt kræve sit. Hvis du kigger meget på skærm, så lav øjenøvelser en gang imellem.

Øjenøvelser

Kig helt til højre, helt til venstre, helt opad, helt nedad, sydøst, nordvest, sydvest, nordøst både med åbne og lukkede øjne - lige nok til at du kan mærke musklerne, og træk samtidigt vejret godt igennem.

Når du har trænet lidt, kan du lave cirkler både med lukkede og åbne øjne - børn elsker det sidste! Disse øvelser er meget enkle, men har stor virkning. To minutters øvelser om dagen gør underværker. Hvad koster din sundhed? Du skal ikke tro, at de fleste skærmproducenter primært tænker på din sundhed, selv om den 'grønne linie' også er ved at bryde igennem i denne branche.

- 286, 11; 13; 20; 54
- 32 bit disk access, 44
386 enhanced mode, 43
- acceleratorkort, 13
access time, 32
 ANSI, 10
 ANSI.SYS, 24
 AT-bus, 6; 31
- BIOS, 6
 bit, 10
 blå skærm, 28
boot-diskette, 57
BREAK=, 25
 byte, 10
- cache*, 8
 CD-ROM, 23; 27
 CHKDSK/F, 33; 56
clean boot, 32; 57
 clockfrekvens, 11
 CMOS, 59
 codepage, 16; 27
 COMMAND.COM, 26; 57
 consol, 16
coprocessor, 11
 COUNTRY=, 24
CPU cache, 12
 crosslinked files, 56
- datafiler*, 42
default, 4
 DEFRAG, 32; 34; 50
 device, 11; 14; 16; 20
 device driver, 11
 devicehigh, 21
 DISPLAY.SYS, 23
 DOS, 6
DOS environment, 26
 DOS=HIGH, 20
 DOS=UMB, 23
 DoubleBuffering, 31
- editor, 39
 EISA, 13
 EMM386.EXE, 21
 EMS, 22
Enhanced IDE, 13
 enhanced mode, 44
environment, 26
- FastDisk, 44
- FASTOPEN, 26
 FAT, 38
 FCBS=, 24
 FILES=, 24
fragmenteret, 32
- graphics mode*, 7
- HIMEM.SYS, 20; 50
 host drive, 52
- I/O porte, 6
 IDE, 6; 7
interlaced, 7
interrupt, 25
ISA, 13
- kloner*, 4
- LASTDRIVE**=, 26
 LH, 21
LoadHigh, 21
Local Bus, 13
Lost clusters, 56
- MCA, 13
 missing or bad Command Interpreter,
 57
 MODE CON, 27
 monitor, 7
monochrome region, 22
 multitasking, 44
- NOEMS, 21
non-interlaced, 7
 non-interlaced, 7
 Non-system disk, 57
- page frame*, 22; 46
parametre, 20
PATH, 27
 pixel, 8
 porte, 6
 protected mode, 11; 15; 44
PS/2, 15
- quick-format*, 56
- RAM, 6
 RAMdisk, 23; 28
 RAMDRIVE.SYS, 23
read-ahead-buffer, 8
real mode, 11; 15
 REM, 16
 resident, 55
- resident, 14
 ROM, 6
- SCANDISK, 50
scankode, 10
 SCSI, 7
 SETVER, 24
shell=, 26; 45
 SIMM, 6
slot, 6
 SMARTDRV.SYS, 23
 SPEEDISK, 32
 spil, 48
stack overflow, 25
STACKS=, 25
standalone, 17
standard mode, 43
 Super VGA, 7
 SVGA, 7
swapfile, 33; 43
 switch, 20
 system-diskette, 57
 systemfiler, 20
søgetid, 32
- tekstfiler, 10
text mode, 7
 translation buffers, 47; 55
 TSR, 14
- UMB, 14
- V86 mode, 44
 VESA, 13
 VGA, 7
 videokort, 7
 virtual disk, 23
- Windows, 8; 9; 10; 13; 23; 28; 42; 45;
 54
- øjenøvelser, 60

Ideen fra KnowWare	3	Flere boots.....	40
Batfiler.....	6	F5.....	40
ASCII og talsystemer	7	F8 og ?.....	40
16 talsystemet.....	7	Boot med valg	40
CPU og memory	8	MS-DOS 6.2.....	42
Memoryformer	9	Doublespace	42
Conventional memory.....	9	Brug DBLSPACE?	42
Upper memory area.....	9	Custom.....	43
Expanded og extended memory	10	Express.....	44
High Memory Area	10	Custom og Express	44
Forholdet upper-extended	10	Windows f. Workgroups 3.11.....	45
Device.....	11	Optimeringstips	46
Startfilerne	11	Udnyt upper memory.....	46
De vigtige filer.....	12	Kig i memory - MSD.EXE.....	47
Bootprocessen	15	DOS.....	48
CONFIG.SYS	15	Slette alle filer i et directory	48
286.....	15	CHKDSK/F	48
386.....	16	Fra , til . i DOS programmer.....	48
Kræver EMS memory.....	16	FDISK	48
AUTOEXEC.BAT	21	Boot disketter	49
Codepage diskussion	23	Diskette 1 - clean boot	49
Disk Cache Smartdrv.....	24	Diskette 2	50
Generelt	24	CMOS og Setup.....	51
SMARTDRV.EXE	24	Blindskrift	51
DoubleBuffering.....	25	Din sundhed	52
Harddisken.....	25		
Diskoptimering.....	26		
Norton Commander (NC).....	28		
Indstillinger	28		
Kopiering med XCOPY	30		
Skjulte filer	30		
Udvalgte filer.....	30		
Omdøbe	31		
Se eller rette i tekstfiler.....	32		
Søge efter fil	32		
Skabe en ny fil.....	32		
Tidligere dos-kommandoer	32		
Tænd/sluk et vindue	32		
Byt vinduerne	32		
Skift directory.....	32		
Bladre i directories	32		
Sammenligne directories	32		
Menu.....	33		
Version 4.0	33		
Sikkerhedskopiering	34		
Windows 3.1.....	35		
Swapfile.....	35		
32 bit disk access	36		
Tips til Windows	36		
DOS 6	37		
Installation.....	37		
Help.....	37		
EMM386	38		
MEMMAKER.....	38		
Custom.....	38		