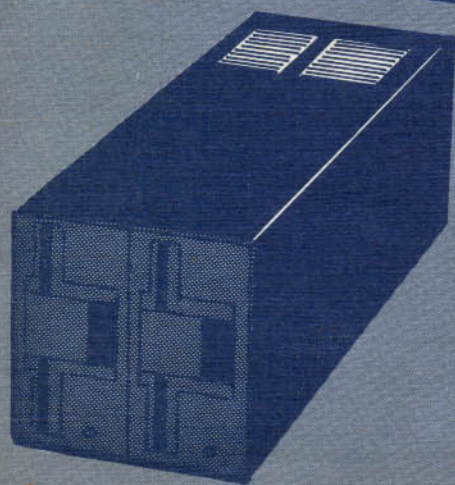
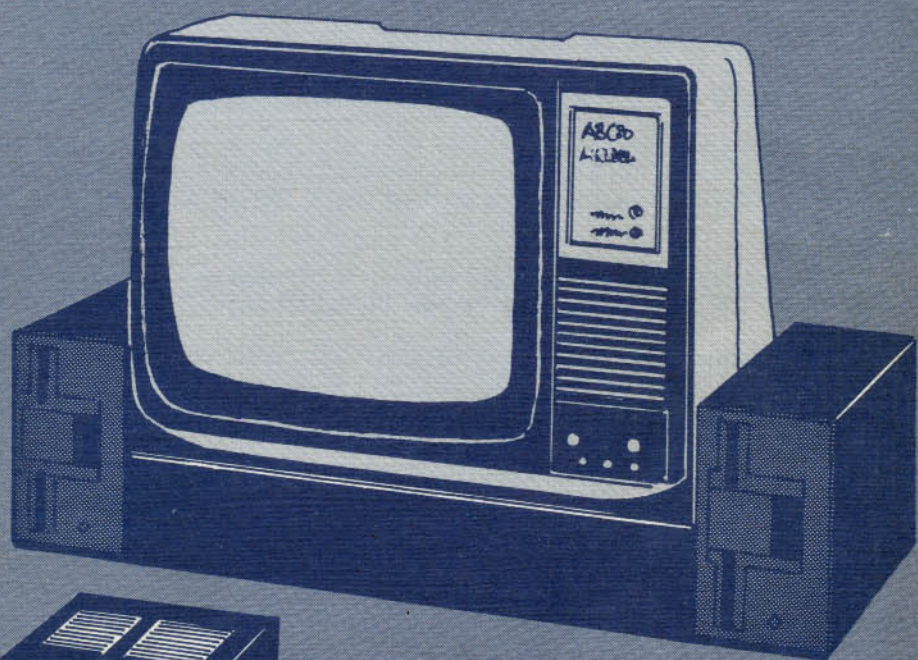


BRUKSANVISNING FD2 och FD2U

Flexskivenhet



ABC80

FÖRORD

Denna bruksanvisning är indelad i 4 kapitel, som vänder sig till användare med olika målsättningar och förkunskaper.

Kapitel 1 vänder sig till den, som egentligen inte vet någonting om datorer, utan bara vill veta precis vilka tangenter man skall trycka på för att få färdiga program och rutiner att fungera.

Kapitel 2 vänder sig till den, som vill veta mer om flexskivenhetens inneboende möjligheter. Kapitlet förutsätter kännedom om grundläggande BASIC-programmering på ABC 80, t ex från den blå boken "Bruksanvisning ABC 80".

Kapitel 3 behandlar ingående datafilhanteringen med flexskivenheten och förutsätter att läsaren är bekant med stränghantering på ABC 80.

Kapitel 4 slutligen behandlar diskoperativsystemet och de systemprogram, som finns tillgängliga till flexskivenheten.

ALLMÄNNA BESTÄMMELSER KRING PROGRAMMATERIAL TILL ABC 80	3
KAPITEL 1 PRESENTATION	
Beskrivning av flexskivenheten	4
Hantering av flexskiva	5
Inkoppling och start	5
LIB-programmet	6
Allmänna tips	7
Beskrivning av ett program för MEDLEMSREGISTRERING	8
Kommentarer till programmet MEDLEM	12
KAPITEL 2 PROGRAMKONTROLL	
Några enkla programexempel	13
Om filnamn och enheter	14
Kommandobeskrivning	14
Överföring från kassett till flexskiva	17
KAPITEL 3 DATAKONTROLL	
Begreppsdefinitioner	18
INPUT- och INPUTLINE-strängar	18
Några exempel med sekvensiella filer	19
Filer med direkt åtkomst	21
Beskrivning av ett lagerhållningsprogram	22
Kommentarer till programmet LAGER	27
KAPITEL 4 DISK OPERATIV SYSTEMET	
Kommandointerpretatorn	28
Kommandon på systemskivan	28
Space	
Memory	
Copy	
Copylib	
Dosgen, F	
Återgång till BASIC	30
APPENDIX	
A 1 Teknisk beskrivning	31
A 2 Teknisk specifikation av flexskivenhet till ABC 80	32
A 3 Blockschema över flexskivenheten	33
B 1 Flexskivor till ABC 80	34
FLÖDESSCHEMA OCH PROGRAMLISTNING "MEDLEM"	
C 1 Huvudprogram	35
C 2 Nya medlemmar	36
C 3 Medlemslistning	38
C 4 Medlemskorrigering	41
C 5 Nytt föreningsnamn	45
FLÖDESSCHEMAN "LAGER"	
D 1 Huvudprogram (lagerregister)	47
D 2 Uppläggning av nya varor	48
D 3 Listning av varor	49
D 4 Transaktion av varor	50
D 5 Borttagning av varor	51
D 6 Nytt lager	52

ALLMÄNNA BESTÄMMELSER KRING PROGRAMMATERIAL TILL ABC 80

Detta programmaterial är noggrant uttestat och antas vara felritt. Emellertid ges inga garantier och detta programmaterial ställs till köparens förfogande endast i befintligt skick med eventuella felaktigheter.

Copyright-innehavaren, säljare, förmedlare samt upphovsmannen till detta programmaterial skall under inga omständigheter vara ansvarig inför någon för eventuella fel i programmen eller för felaktigt handhavande eller för avbrott eller indirekt skada eller förlust för användaren.

Innehav av detta programmaterial berättigar till nyttjanderätt endast enligt följande villkor:

1. Nyttjanderätten till programaterialet tillkommer innehavaren, eller om innehavaren är ett företag, personalen vid nyttjande inom detta företag. Nyttjanderätten får ej överlätas utan skriftligt tillstånd.
2. Äganderätt, upphovsrätt eller annan immateriell rätt tillkommer copyright-innehavaren eller programmaterialets upphovsman. Detta programmaterial får ej försäljas, distribueras eller mångfaldigas i vare sig ursprungligt eller förändrat skick. Det åvilar innehavaren att vidtaga nödvändiga skyddsåtgärder för att förhindra obehörig spridning av programaterialet.
3. Om flexskivan inte går att använda i ABC 80 och om detta beror på felaktig kopia, äger innehavaren rätt att erhålla ny kopia kostnadsfritt inom en (1) månad efter leverans. Den felaktiga flexskivan skall, senast inom denna tid, lämnas till försäljaren eller order för utbyte kostnadsfritt. Efter utgången av den första månaden efter leverans, debiteras en självkostnadsavgift. Förevarande garanti gäller högst ett (1) år från leverans.
4. Programmaterial består av skrivna körinstruktioner och diskett. Program av "RUN ONLY-typ" kan ej listas. Efter skriftlig framställning kan innehavaren mot avgift erhålla programlista i vissa fall. Fri provningsrätt förbehålles.
5. Copyright-innehavaren förbehåller sig rätten att göra ändringar i programaterialet utan föregående avisering.

COPYRIGHT SCANDIA METRIC AB/LUXOR AB

Inga delar av denna bruksanvisning eller tillhörande program får mångfaldigas för kommersiellt syfte utan copyright-innehavarens skriftliga tillstånd.

Sålunda får ej flexskiva eller kassett kopieras, ej heller får information från flexskiva eller kassett överföras eller översättas - helt eller delvis - till likadant eller annat medium för försäljning.

KAPITEL 1

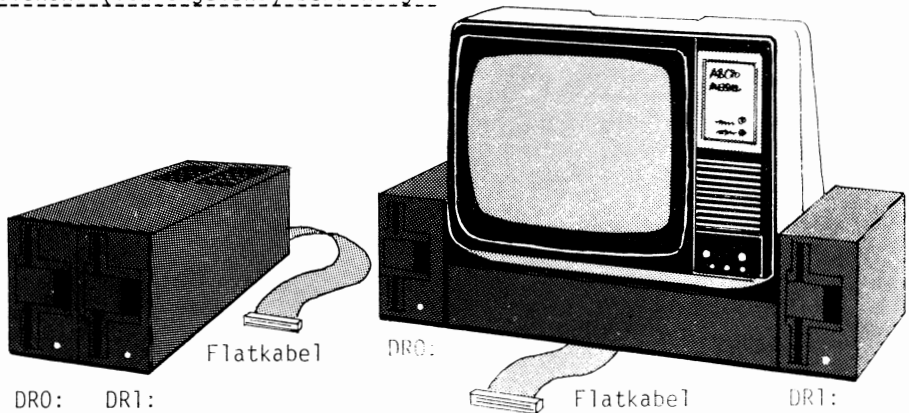
PRESENTATION

Beskrivning av flexskivenheten

Flexskivenheten FD till ABC 80 finns i 2 utföranden, som skiljer sig till utseende.

Det finns även en del elektriska skillnader (se "Teknisk specifikation" i Appendix A2). Denna bruksanvisning gäller till alla delar för båda typerna.

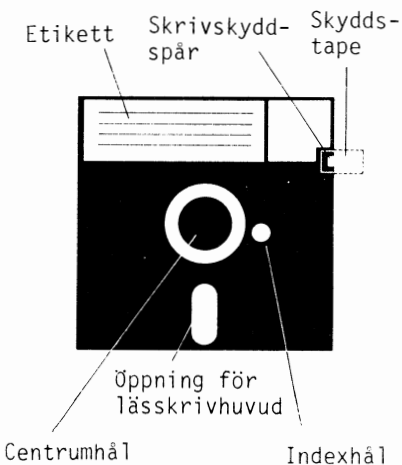
Svenska (och engelska) benämningar



Flexskivenhet FD 2
(Floppy disc unit)

(DRO:,DR1:) Drivenhet
(Floppy driver)

Flexskivenhet FD 2II
(Floppy disc unit)



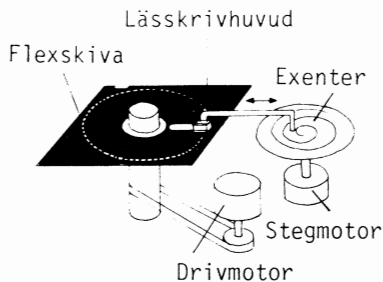
Flexskivenheten FD används för att snabbt lagra stora mängder information, i form av program som skall köras, eller av data som skall bearbetas i ABC 80.

Informationen lagras genom magnetisering av ett tunnt järnoxidskikt på en tunn, plan, snurrande skiva - den s k flexskivan (flexible diskette).

På en sådan skiva kan man lagra ungefär 72 000 tecken (siffror eller bokstäver). Flexskivenheten har plats för 2 skivor samtidigt och genom att man lätt kan byta skivorna, är lagringskapaciteten i praktiken obegränsad.

Jämfört med en bandspelare har flexskivan den stora fördelen, att man kan hitta en viss information enormt mycket snabbare; på bräkdelen av en sekund.

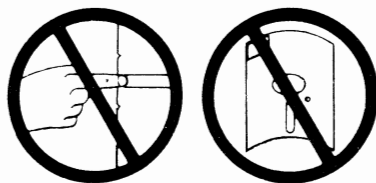
Beträffande olika flexskivor till ABC 80 se Appendix B.



Livslängden för lässkrivhuvudet, som sköter in- och avspelningen av informationen, är ca 20 000 timmar. För själva flexiskivan uppges en livslängd på 3 miljoner passager av lässkrivhuvudet på ett spår.

Flexiskivan roterar inuti sitt pappersomslag. För att minska friktionen mellan omslag och skiva är omslagets insida klädd med en specialplast.

Hantering av flexiskiva

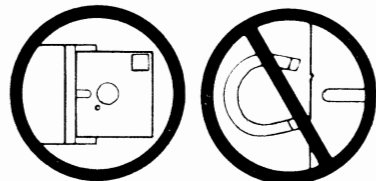


Flexiskivan är ömtålig och måste hanteras med största varsamhet:

Rör ej det magnetiska skiktet i skivans öppningar

Böj eller vik inte skivan

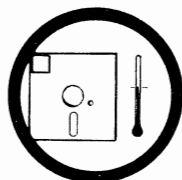
Sätt genast tillbaka skivan i dess omslag när den inte används



Håll skivan borta från starka magnetfält

Placera inte skivan i direkt solljus

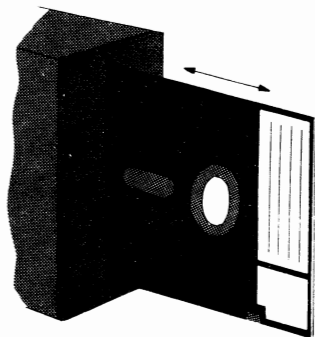
Lagra skivan helst vertikalt. Tillåten lagringstemperatur $+10^{\circ}\text{C}$ - $+52^{\circ}\text{C}$



För att skriva på etiketten använder man en filt-penna och trycker löst. Kulpenna eller dylikt får absolut inte användas, eftersom den kan ge tryckmärken i skivan.

Om man vill skrivskydda flexiskivan, d v s göra det omöjligt att ändra den information som finns på skivan, sätter man en tape-bit över skrivskyddspåret.

Inkoppling och start



1. Stäng av ABC 80 och flexskivenheten. Anslut flexskivenhetens flatkabel till uttaget på baksidan av tangentbordet till ABC 80.

2. Öppna båda luckorna på flexskivenheten.

3. Slå på flexskivenheten och ABC 80.

4. Tryck på RESET-knappen på tangentbordets baksida och kontrollera att flexskivenhetens röda lampor tänds för ett ögonblick, en i taget.

5. Sätt i flexiskivan med demonstrations- och systemprogrammet (FLEXDEMO), som följer med flexskivenheten i den vänstra drivenheten (DRO:). Skivan skall vändas med spåret på skrivskyddet nedåt, så att öppningen för lässkrivhuvudet kom-

mer först. Tvinga aldrig in skivan med våld; den skall glida in lätt när den stoppas in på rätt sätt.

6. Stäng försiktigt luckan på flexskivenheten och kontrollera att skivan inte böjs av luckan. Tryck eventuellt på RESET, varvid flexskivenhetens motor startar. Härigenom går det lättare att få flexskivan på rätt plats så att luckan kan stängas ordentligt.

7. Tryck på RESET igen för att flexskivan skall vara aktiv. Skriv något (felaktigt) på tangentbordet och tryck RETURN och ABC 80 svarar med "FÖRSTÄR EJ" i stället för "ERR 11" om en systemskiva skulle finnas i någon av drivenheterna.

LIB-programmet

RUN LIB

Listning

För att lista vad som finns på flexskivorna, kör Du programmet LIB genom att skriva RUN LIB (och RETURN). Sedan svarar Du på frågan "Vill Du ha en printerlista (J/N)" med N (utan RETURN) om listan skall upp på bildskärmen, och med J (utan RETURN) om Du vill ha listan utskrivna på printern.

Det senare alternativet förutsätter att printern är inkopplad, annars svarar ABC 80 med "FINNS EJ I DETTA SYSTEM LINE 30" eller "ERR 8 LINE 40". I så fall får Du starta om med RUN LIB.

Följande listning sker på bildskärm eller printer:

```
**** Drive: 0 Flexskiva=FLEXDEMO ****
LIB      .BAC BASICERR.SYS CMDINT  .SYS
COPYLIB .ABS COPY      .ABS DOSGEN .ABS
MEM      .ABS SPACE    .ABS MEDLEM .BAC
FÖRENNY .BAC MEDLNYA  .BAC MEDLLIST.BAC
MEDLKORR.BAC FÖRENING.DAT NAMN    .DAT
LAGER    .BAC LAGPREP  .BAC LAGERNYA.BAC
LAGLIST  .BAC LAGTRANS.BAC LAGBORT .BAC
Kvarvarande utrymme 23 sektorer
```

```
**** Drive: 1 Flexskiva=VAROR ****
LIB      .BAC BASICERR.SYS VARUNR  .DAT
VAROR    .DAT
Kvarvarande utrymme 235 sektorer
```

Först skrivs ett huvud med numret för drivenheten. Om första filen slutar med .LBL blir den skivans namn som skrivs i huvudet. Därefter följer en lista på alla filnamn och slutligen skrivs antalet sektorer som är kvar. Samma uppgifter skrivs för flexskivenhet 1 (DRI:) om en skiva är inlagd där.

.BAC .BAS

Program lagras med filtyp .BAC (komprimerad form) eller .BAS (textformat).

LIB.BAC är det program, som Du kör för att få listan över de filer, som finns på flexskivorna.

- .DAT Data lagras med valfri filtyp, i detta fall med .DAT.
- BASICERR.SYS BASICERR.SYS är en systemfil knuten till ABC 80 och den läses in vid påslag eller RESET. Det är denna fil, som ger felutskrift i klartext i stället för med siffror, som endast talar om feltyp. Detta gäller felena 1 - 66.
- CMDINT.SYS CMDINT.SYS (Command Interpreter) är en systemfil, som är ett operativsystem med vissa kommandon. Under detta kan Du utföra alla kommandon med filtyp .ABS (se vidare kapitel 4).

Kopiering av LIB.BAC

Programmet LIB.BAC kan bara köras med RUN LIB. Det går alltså inte att lista eller kopiera. Det är en sk spärerad programvara, som gäller en del program från leverantören.

Du kan dock kopiera LIB.BAC genom att skriva "S" när ABC 80 frågar efter printerlista, och sedan svara på vilken driv-enhet programmet skall lagras i.

Allmänna tips

- Skivbyte Vid byte av flexskiva måste Du se till, att det inte finns några öppna filer på skivan när Du tar ut den. Om Du inte är säker att alla filer är stängda, skriv då kommandot CLEAR innan Du tar ut skivan.
- Kopior För att gardera sig mot att filer förstörs genom mekaniska fel på flexskivan, bör Du alltid ha en reservkopia på en annan flexskiva ("backup kopia"). Se kapitel 4 under "COPYLIB".
- Skivnamn När Du har gjort i ordning en ny skiva, kan Du lägga in den fil som skall bli skivans namn, när Du kör RUNLIB. Du gör på följande sätt om skivan sitter i DR1:
NEW för rensning av programminnet,
SAVE DR1:NAMN.LBL där NAMN är skivans beteckning.
- Avstängning När utrustningen stängs av, antingen ABC 80 eller flexenheten, måste luckorna absolut öppnas så att inte informationen på flexskivorna förstörs av misstag.

Beskrivning av ett program för MEDLEMSREGISTRERING

MEDLEM är ett demonstrationsprogram på flexskivan FLEXDEMO. Programmet har till uppgift att hålla reda på ett tänkt medlemsregister.

4 uppgifter skrivs in för varje medlem, nämligen namn, adress, postadress och medlemsnummer.

Programmet är inte ett komplett personregister, utan vill visa hur Du hanterar flexskivenheten med s k sekvensiella filer.

RUN MEDLEM Starta programmet med att skriva RUN MEDLEM. Följande text visas då på bildskärmen:

```

                MEDLEMSREGISTER
Rutinnr
1  NYA MEDLEMMAR
2  LISTA MEDLEMMAR
3  KORRIGERA MEDLEMMAR
4  SLUT
Skriv rutinnummer (utan RETURN) ■
```

Här väljer Du vilken rutin (= underprogram) Du vill använda. Alla rutiner avslutas alltid med att gå tillbaka hit till huvudrutinen.

NYA MEDLEMMAR Välj den första rutinen (1) NYA MEDLEMMAR. Denna rutin används för att föra in nya medlemmar i registret:

```

                NYA MEDLEMMAR
FÖRENINGEN      : ABC 80 KLUBBEN
LÖPNUMMER       : 4
EFTER- & FÖRNAMN: ■
GATUADRESS      :
POSTADRESS      :
MEDLEMSNUMMER   :
```

Här skriver Du in uppgifterna för varje medlem, t ex
Nilsson Lennart
Järnvägsgatan 2
567 89 GLESBYGDEN
127

Efter att ha skrivit in medlemsnumret kommer frågan "AR UPPGIFTERNA RIKTIGA (J/N)" upp. Nu har Du möjlighet att skriva om uppgifterna genom att svara med N (nej).

Svarar Du med J (ja), blir nästa fråga på skärmen "MERA NAMN (J/N)?" Du kan nu fortsätta med fler medlemmar genom att svara J - eller avsluta inskrivningen med N. Då läggs de nya namnen upp på flexskivan och programmet återgår till huvudrutinen i MEDLEM.

LISTA MED-
LEMMAR

Välj nu den andra rutinen LISTA MEDLEMMAR. Denna rutin
listar alla medlemmar eller vissa delar av registret:

```

                LISTA MEDLEMMAR
FÖRENINGEN: ABC 80 KLUBBEN
SKALL LISTNING SKE PÅ PRINTER? (J/N)
SKALL ALLA MEDLEMMAR LISTAS? (J/N)

```

Svara N (och RETURN) på första frågan om Du vill ha list-
ningen på bildskärmen. Svara J (RETURN) på den andra frå-
gan om alla medlemmar skall med på listan. Svarar Du N
(RETURN), ange då mellan vilka löpnummer listningen skall
ske.

```

                LISTA MEDLEMMAR
LÖPNUMMER:
Tryck en tangent för varje medlem
MEDLEMSLISTA: ABC 80 KLUBBEN
LÖPNUMMER :
NAMN      :
ADRESS    :
POSTADRESS:
MEDLEMSNR :
1
KARLSSON KALLE
KUNGSGATAN 12
111 11 ASTAD
274

```

Listningen fortsätter så länge Du trycker ned en tangent
till och med sista medlemmen, då programmet återgår till
huvudrutinen.

KORRIGERA
MEDLEMMAR

Välj den tredje rutinen KORRIGERA MEDLEMMAR om Du vill rät-
ta felaktigheter i medlemsregistret:

```

                KORRIGERA MEDLEMMAR
FÖRENINGEN: ABC 80 KLUBBEN
LÖPNUMMER      :
1 EFTER- & FÖRNAMN:
2 GATUADRESS   :
3 POSTADRESS   :
4 MEDLEMSNUMMER :
----- SÖK MEDLEM -----
SKRIV SÖKBEGREPPET FÖR MEDLEMEN
■

```

SÖK MEDLEM

Skriv in ett sökbegrepp för medlemmen, som är signifikativt för medlemmen i fråga. Vill Du söka den nyinskrivna medlemmen, räcker det inte med "LE" om namnet "KALLE" finns före, medan "LEN" däremot går bra. Ange med 1 att sökbegreppet finns i uppgiften EFTER- & FÖRNAMN. Nu sker sökningen av medlemmen och Du kan övergå till att rätta felet.

<u>KORRIGERA MEDLEMMAR</u>	
FÖRENINGEN: ABC 80 KLUBBEN	
LÖPNUMMER	:4
1 EFTER- & FÖRNAMN:	NILSSON LENNART
2 GATUADDRESS	:JÄRNVÄGSGATAN 2
3 POSTADDRESS	:567 89 GLESBYGDEN
4 MEDLEMSNUMMER	:127
5 BORTTAGNING AV DENNA MEDLEM	
_____ RÄTTA MEDLEM _____	
Ange 0 för ingen mer rättning och 5 för borttagning	
VILKEN UPPGIFT SKALL RÄTTAS (0-5)?4	

**RÄTTNING
MEDLEM**

Medlemsuppgifterna skrivs ut på bildskärmen och Du anger med en siffra vilken uppgift som skall rättas (1-4) eller om medlemmen skall strykas ur registret (5).

Den felaktiga uppgiften skriver Du om helt (plus RETURN). När medlemsuppgiften är korrekt, anger Du med 0 att inga fler rättningar behövs. För säkerhets skull svarar Du en extra gång på frågan om medlemsuppgiften är riktig. Med den sista frågan "SKALL FLER MEDLEMMAR RÄTTAS (J/N)?" anger Du fler rättningar eller återgång till huvudrutinen.

SLUT

Välj den fjärde (4) huvudrutinen om Du vill tillbaka till BASIC för att göra något annat.

NY FÖRENING

Det finns ett underprogram knutet till MEDLEM, som genererar en ny föreningen eller byter föreningens namn. Starta med att skriva RUN FÖRENNY:

<u>NYTT FÖRENINGSNAMN</u>	
Ny förening eller ändring av namn på den gamla föreningen.	
Finns redan föreningsnamn (ja/nej)? JA	
Gamla namnet är: ABC 80 KLUBBEN	
Skriv det nya namnet: <u>DATORKLUBBEN BAJT</u>	
Är det nya namnet riktigt (ja/nej)? JA	
Föreningen	: ABC 80 KLUBBEN
har bytt namn till: DATORKLUBBEN BAJT	

Om Du endast vill byta föreningens namn, svara "JA" på frågan "FINNS REDAN FÖRENINGSNAMN (JA/NEJ)?"

Skriv in det nya föreningsnamnet och bekräfta att det är rätt. Alla gamla medlemsuppgifter blir i så fall oförändrade och Du kan fortsätta programmet som vanligt.

NYTT FÖRENINGSNAMN

Ny förening eller ändring av
namn på den gamla föreningen.

Finns redan föreningsnamn (ja/nej)? NEJ

Vad är det nya namnet? ABC 80 KLUBBEN

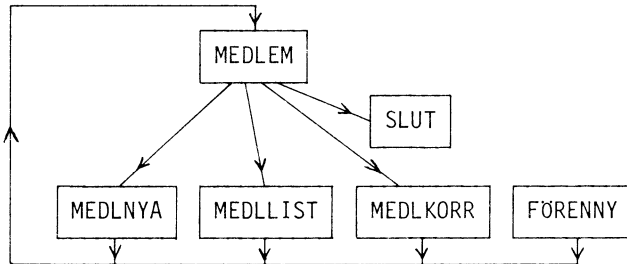
Är det nya namnet riktigt (ja/nej)? JA

FÖRENINGEN ABC 80 KLUBBEN ÄR UPPLAGD

Om Du i stället svarar "NEJ" på första frågan, förstörs alla gamla uppgifter som finns på skivan. Du skriver föreningsnamnet, bekräftar det och fortsätter med MEDLEM.

Kommentarer till programmet MEDLEM

Programmet MEDLEM består egentligen av 5 delprogram och 2 datafiler. Huvudprogrammet växlar ned till underprogrammen MEDLNYA, MEDLLIST eller MEDLKORR som återgår till huvudprogrammet när de avslutas.



Alla dessa program skulle kunna få plats samtidigt, men de är uppdelade, för att ge en enkel struktur och vissa möjligheter att koppla ihop olika delprogram.

Alla underprogram använder datafilerna FÖRENING.DAT och NAMN.DAT. Den första filen innehåller först föreningsnamnet och sedan antalet upplagda namn. Filen NAMN.DAT innehåller data (namn, adress, postadress och medlemsnummer) för alla medlemmarna i sekvensiell ordning. Detta innebär, att om man lägger upp en ny medlem i underprogrammet MEDLNYA måste man gå förbi alla gamla namn och där lagra den nya medlemmen.

Tar man bort eller rättar en medlem med MEDLKORR, flyttar programmet även om alla bakom liggande medlemmar. Det innebär, att man antingen lägger upp alla medlemmar i minnet eller - som i detta fall - läser en medlem, testar om den skall rättas, och om inte, skriver in den igen. (se programlistningen för MEDLKORR raderna 520 - 670).

Detaljerade flödesscheman för de olika programmen samt fullständig programlistning finns i Appendix.

OBS: Det finns olika lagar som reglerar hur personuppgifter får registreras med hjälp av dator. Ovanstående medlemsregister är i princip ett personregister i Datalagens mening.

Tillstånd att upprätta dataregister ges av:
Datainspektionen
Box 12050
102 22 STOCKHOLM
Tfn 08-22 79 80

KAPITEL 2

PROGRAMKONTROLL

Några enkla programexempel

Flexskiva i DRO:

1. Sätt in en flexskiva som inte är full och starta ABC 80, t ex flexskivan "VAROR" i den vänstra drivenheten (DRO:)
2. Skriv ett kort program, t ex:
NEW
10 FOR A = 1 TO 10
20 PRINT A;
30 NEXT A
Provkör därefter programmet med RUN. Lagra sedan programmet på den vänstra flexskivan (DRO:) under namnet PROG1 genom att använda kommandot
SAVE DRO:PROG1
3. Skriv ett annat kort program, t ex:
NEW
70 FOR B%11 TO 20
80 PRINT B;
90 NEXT B
Provkör programmet med RUN. Lagra sedan programmet under namnet PROG2 med kommandot
SAVE DRO:PROG2
4. Hämta det första programmet med kommandot
LOAD DRO:PROG1
Kontrollera med kommandot RUN att det är rätt (det går även direkt med kommandot RUN PROG1).
5. Komplettera PROG1 med PROG2 genom kommandot
MERGE DRO:PROG2
Provkör med RUN.
6. Lagra detta nya program (PROG1+PROG2) under namnet PROG3 i okompilerad */ form eller i textformat med kommandot LIST DRO:PROG3. Se sedan vilka program som finns på flexskivan med RUN LIB (jfr Kap 1).
7. Hämta och kör PROG1 med kommandot RUN PROG1 (DRO: kan utelämnas om det bara finns ett program med namnet PROG1).
8. Byt namn på programmet PROG1 med kommandot
NAME "DRO:PROG1.BAC" AS "PROG4.BAC"
9. Tag bort programmet PROG2 med kommandot
UNSAVE DRO:PROG2
Provkör PROG1, PROG2, PROG3 och PROG4.

*/ Om kompilering, se "Mikrodatorns ABC", kap 7.2.7


```
CHAIN"DRO:PROG4"      10. Skriv ett program som innehåller CHAIN, t ex:  
NEW  
10 PRINT "LÄGE1"  
20 CHAIN"DRO:PROG4"  
30 PRINT "LÄGE2"  
40 END
```

Kör programmet med RUN. Vad händer?

Gör rent 11. Gör rent på flexskivan genom att ta bort alla program som inte behöver sparas med UNSAVE.

Om filnamn och enheter

Självva filnamnet består av filnamn, punkt och filtyp. T ex ABCD1234.EFG, där ABCD1234 är filnamnet och EFG är filtypen.

Filnamn kan vara upp till 8 tecken långt. Det måste alltid börja med en bokstav och får sedan innehålla både siffror och bokstäver.

Filtyp kan vara upp till 3 tecken. Operativsystemet i ABC 80 sätter ut filtyperna .BAC och .BAS på programfiler om man inte skriver någon filtyp.

På systemflexskivan används filtypen .SYS för att beteckna systemfiler och filtypen .ABS för att beteckna systemunderprogram (absolutadresserade filer). För övrigt kan man välja filtyp-beteckningen fritt, lämpligen så att den talar om vad det är för sorts fil; t ex .DAT för en fil som innehåller data, .TXT för en fil som innehåller text.

Filnamn kan även vara stränguttryck, t ex A α + B α ".BAS".

I den följande beskrivningen används "enhet" för att visa hur man kan adressera en bestämd utrustning till ABC 80. De enheter som kan adresseras av ABC 80 är:

DRO: DR1:	flexskivenhetens båda drivenheter
CAS:	kassettspelaren
PR:	skrivaren (printern)

Utelämnas enheten, provar ABC 80 först automatiskt kommandot på flexskivenheten DRO:.. Går kommandot då ej att utföra (filen finns ej, skivan ej klar), provas kommandot på DR1:.. Går inte heller det, får man någon form av felmeddelande.

Vill man adressera kassetbandspelaren (och flexskivenheten är inkopplad) måste enheten CAS: absolut anges i kommandot.

Kommandobeskrivning

Förkortningar

(p)	kan endast användas i program
(k)	kan endast användas som kommando
(pk)	kan användas både i program och som kommando

LOAD enhet: filnamn .filtyp

(k) söker efter programfilen "filnamn". Om programfilen hittas, nollställs ABC 80's interna programminne och inläsning sker.

Utelämnas filtyp, söker ABC 80 först efter filnamn .BAC på båda drivenheterna och sedan efter filnamn .BAS. Om programfilen inte hittas, svarar ABC 80 med "ERR 21" eller i klartext om en skiva med BASICERR.SYS är isatt och aktiverad när utrustningen startades.

Finns 2 programfiler med samma namn på drivenheten DRO: respektive DRI:, kommer alltså programfilen DRO: att läsas in om enheten utelämnas.

Att utelämnas filnamn fungerar endast när enheten CAS: anges.

Exempel

LOAD DRO: SILL.BAC	Hämtar programmet SILL.BAC på DRO:
LOAD DRI: HERRIN.BAS	Hämtar programmet HERRIN.BAS på DRI:
LOAD DEMO	Hämtar programmet DEMO (.BAC eller .BAS) på DRO: eller DRI:
LOAD CAS: DEMO	Hämtar programmet DEMO från kassetten
LOAD CAS:	Hämtar nästa program från kassetten
LOAD TOK	Hämtar programmet TOK (som inte finns) och man ser hur lamporna på drivenheterna tänds vardera 2 gånger under sökning efter .BAC respektive .BAS

RUN enhet: filnamn .filtyp

(k) läser in och startar körning av programmet "filnamn". Kommandot gör alltså samma sak som LOAD följt av RUN. (Se LOAD för mera information).

SAVE
LIST enhet: filnamn .filtyp

(k) skapar en ny fil med namnet "filnamn" och lagrar det program, som finns i ABC 80's interna minne på denna fil.

Med SAVE lagras programmet i kompilerad form med filtypen .BAC tillagd, om ingen filtyp skrivs.

Med LIST lagras det okompilerade programmet i textformat med filtypen .BAS tillagd.

Exempel

SAVE EBO	Lagrar programmet EBO.BAC på en skiva i komprimerad form
SAVE DRI: EBO	Lagrar programmet EBO på DRI:
LIST DRO: EBO	Lagrar programmet EBO.BAS på DRO: i textformat
LIST DERIVE.RA	Lagrar programmet DERIVE.RA

UNSAVE enhet: filnamn . filtyp

(k) tar bort filen "filnamn" från flexskivan. Utelämnas filtyp, söker ABC 80 först efter filnamn .BAC på båda drivenheterna och sedan efter filnamn .BAS.

Exempel

UNSAVE EBO Tar bort EBO.BAC eller EBO.BAS på en av enheterna
UNSAVE DRO:EBO Tar bort EBO.BAC eller EBO.BAS på DRO:
UNSAVE EBO.BAS Tar bort EBO.BAS på en av enheterna
UNSAVE DERIVE.RA Tar bort DERIVE.RA

KILL "enhet: filnamn .filtyp"

(pk) tar bort filen "filnamn" från flexskivan. Strängparenteser och filtyp måste vara med!

KILL används för borttagning av filer (huvudsakligen datafiler) under programkontroll.

Exempel

KILL "EBO.BAS" Tar bort EBO.BAS
KILL "DR1:DERIVE.RA" Tar bort DERIVE.RA ur DR1:

NAME "enhet: filnamn1 .filtyp"AS"filnamn2 .filtyp"

(pk) byter namn på en fil från filnamn1 till filnamn2. Strängparenteser måste vara med! Filtyp måste anges för både filnamn1 och filnamn2.

Exempel

NAME "DR1:IDAG.BAS" AS "IGÄR.BAS"

MERGE enhet: filnamn .filtyp

(k) läser in programfilen "filnamn" utan att ta bort vad som redan finns lagrat i minnet. Om samma radnummer finns både på filen och i minnet, kommer raden från filen att användas.

Exempel

MERGE SUBRUT Hämtar in delprogrammet SUBRUT

CHAIN "enhet: filnamn .filtyp"

(p) läser in programfilen "filnamn" och startar körning av detta program. Kommandot har samma effekt som RUN "filnamn", men kan användas som sats i ett program. Strängparenteser måste vara med!

Exempel

100 CHAIN "DEL2" Hämtar in och startar programmet
DEL 2

PREPARE "enhet: filnamn .filtyp" AS FILE filnummer

(pk) skapar en ny fil med namnet "filnamn" och ger den numret "filnummer". Filen öppnas samtidigt för skrivning. Stränguttryck kan användas som filnamn. Filnumret kan ligga mellan 1 och 255 (0 är bildskärmen).

Exempel

```
PREPARE "DATA.DAT" AS FILE 7
PREPARE Aø AS FILE 3
PREPARE Bø + ".BAS" AS FILE 15
PREPARE "DR1: TIDN.TXT" AS FILE 2
```

OPEN "enhet: filnamn .filtyp" AS FILE filnummer

(pk) öppnar en redan existerande fil för läsning. Stränguttryck kan användas som filnamn. Filnummer från 1 till 255. Filtyp måste anges. Max 6 filer kan vara öppna samtidigt.

Exempel

```
OPEN "DATA.DAT" AS FILE 1
OPEN Aø AS FILE 147
OPEN Bø + ".BAS" AS FILE 13
OPEN DR1: "TIDN.TXT" AS FILE 15
```

CLOSE filnummer

(pk) stänger en fil, d v s avslutar läsning eller skrivning av filen. Om filen öppnats med PREPARE, skrivs ett filslutmärke.

OBS! En fil på flexskiva, som har öppnats för skrivning, måste ovillkorligen stängas vid flexskivbyte!

Överföring från kassett till flexskiva

Observera att flexskivenheten använder en del av datorns RAM-minne (se Kap 3), varför det för användaren tillgängliga minnet är mindre än det där endast kassetminnet används.

Flexskivenheten använder ca 2 kbyte och det tillgängliga minnesutrymmet är alltså ca 2 kbyte mindre. Detta kan betyda, att en mycket stor fil, som nästan fyller datorns minne och som går bra att spela in och av med kassetminnet, inte får plats när flexskivenheten är inkopplad. Försök då att dela upp filen i flera små filer och använd kommandot CHAIN (se sid och demoprogrammet MEDLEM).

För icke för stora filer går överföringen till så att filen på kassetminnet spelas över till ABC 80's minne på vanligt sätt och sedan till flexskivan med någon av de kommandon, som visats tidigare.

KAPITEL 3

DATAKONTROLL

Begreppsdefinitioner

(engelska namn inom parentes)

Fil (file) = en följd av poster
Post (record) = en datauppsättning om en individ

Exempel

Individ: bankkonto
Post : Data om kontot (nr, kontohavare, saldo, etc)
Fil : Data om alla konton i en viss bank

Individ: telefonabbonent
Post : data i telefonkatalogen (namn, adress, telefonnummer)
Fil : telefonkatalogen för A-stad

Datafil En fil som innehåller data (siffror och/eller bokstäver)

Programfil En fil som innehåller instruktioner (med radnummer) till datorn

Sekvensiell fil Fil med poster, lagrade efter varandra i lång rad med ett skiljetecken mellan varje post (RETURN)

Fil med direkt åtkomst (random access) = fil där poster är lagrade på bestämt ställe oberoende av andra poster

INPUT- och INPUTLINE-strängar

Vid läsning från en fil eller tangentbordet används instruktionerna INPUT eller INPUTLINE.

INPUT packar den sträng som läses, d v s alla mellanslag (blanka) tas bort ur strängen.

INPUTLINE däremot läser hela strängen, inklusive alla mellanslag samt RETURN och nya rader. Detta kan ge en del överraskande effekter. Prova gärna genom att köra nedanstående enkla exempel:

Skriv ett program, som läser från filen NAMN, som finns på flexskivan FLEXDEMO:

```
10 OPEN "NAMN.DAT" AS FILE 1
20 INPUT #1, A$
30 PRINT A$
40 GET X$
50 IF X$ <> "S" THEN 20
60 CLOSE 1
70 END
```

Starta programmet med RUN och stega fram det genom att trycka på vilken tangent som helst. Tangenten S ger avslutning av programmet. Lägg märke till att alla mellanslag har försvunnit.

Ändra nu rad 20 i programmet till:

```
20 INPUTLINE 1, Aǻ
```

Kör programmet och observera att mellanslagen nu finns med. Det blir också dubbla radframmatningar, vilket inte alltid är önskvärt. Tag bort den radframmatningen, som printsatsen ger genom att ändra rad 30 till:

```
30 PRINT Aǻ;
```

Kör programmet och lägg märke till att det nu blir enkel radframmatning.

Om Du har läst en sträng med INPUTLINE från en fil eller tangentbordet, och skall skriva den på en annan fil, skall Du först kapa bort de 2 sista karaktärerna (= vagnretur + ny rad).

När Du sedan skriver på filen, läggs ett RETURN ändå på i slutet för att skilja posterna från varandra. Kapningen av de 2 sista karaktärerna gör Du på följande sätt:

```
Aǻ = LEFTǻ(Aǻ,LEN(Aǻ)-2%)
```

Några exempel med sekvensiella filer

Datafil

1. Sätt in en (icke full) flexskiva i den vänstra drivenheten (DRO:), t ex flexskivan VAROR

2. Preparera och öppna en datafil, som vi här kallar "DATA.DAT" och tilldelar filnummer ett (1) med kommandot
PREPARE "DRO:DATA.DAT" AS FILE 1

3. Lagra posterna 50, 70, potatis i den fil, som heter DATA
PRINT ~~ǻ~~ 1, 50
PRINT ~~ǻ~~ 1, 70%
PRINT ~~ǻ~~ 1, "potatis"

4. Stäng filen DATA med kommandot
CLOSE 1

5. Öppna åter filen med
OPEN "DRO:DATA.DAT" AS FILE 3

6. Hämta informationen från filen DATA med kommandot
INPUT ~~ǻ~~ 3,A,B%,Cǻ

7. Skriv ut data med kommandot
PRINT A,B%,Cǻ

8. Stäng filen med kommandot
CLOSE 3

9. Du måste alltid städa upp filerna efter Dig:
KILL "DATA.DAT"

10. Om Du samtidigt vill hantera data till eller från kassett, skall enhetsnamnet "CAS:" stå före filnamnet i stället för "DRO:" vid PREPARE och OPEN:

Programfil

1. Följande program preparerar filen "TEXT" och matar in valfria textsträngar i en sekvens:

```
10 REM ***** INSKRIFT *****
20 PRINT " INMATNING TILL DATAFILEN "TEXT"
30 PRINT "ANGE ANTAL STRÄNGAR SOM SKALL INMATAS";
40 INPUT S%
50 PREPARE "TEXT" ASFILE 1
60 FOR A%=1% TO S%
70 PRINT A% " ";
80 INPUT A$
90 PRINT #1,A$
100 NEXT A%
110 PRINT "KLART"
120 CLOSE 1
130 END
```

Skriv in programmet och kör det. Ange 3 strängar och skriv in dessa.

2. Följande utskriftsprogram skriver ut ett antal poster, som finns i filen "TEXT":

```
10 REM ***** UTSKRIFT *****
20 PRINT " UTMATNING FRÅN DATAFILEN "TEXT"
30 PRINT "ANGE ANTAL STRÄNGAR SOM SKALL UTMATAS";
40 INPUT S%
50 OPEN "TEXT" ASFILE 1
60 FOR A%=1% TO S%
70 INPUT #1,A$
80 PRINT A% " "A$
90 NEXT A%
100 PRINT "KLART"
110 CLOSE 1
120 END
```

Skriv in programmet och kör det. Ange 3 strängar, som skall skrivas ut. Om Du anger fler än filen innehåller, kommer Du att få fel i rad 70 SLUT PÅ FILEN. I stället för att ange antalet strängar som skall skrivas ut, borde Du då ta med alla strängar fram till den sista.

3. Följande utskriftsprogram skriver ut alla poster i filen "TEXT" tills något fel inträffar.

```
10 REM ***** UTSKRIFT2 *****
20 PRINT "UTMATNING FRÅN DATAFILEN "TEXT"
30 ONERRORGOTO 90
40 OPEN "TEXT" ASFILE 1
50 A%=A%+1%
60 INPUT #1,A$
70 PRINT A% " "A$
80 GOTO 50
90 PRINT ERRCODE="ERRCODE"
100 CLOSE 1
110 END
```

Med rad 30 sätter Du villkoret, att om ett fel inträffar skall programmet fortsätta med rad 90. I rad 90 skrivs feltypen ut med PRINT ERRCODE. Du kan även testa på ERRCODE som ett numeriskt värde för att se om den är 34 (= slut på filen) eller något annat fel.

Filer med direkt åtkomst (random access)

Kommandon i BASIC för att hantera filer med direkt åtkomst finns inte, utan hanteringen görs med CALL-instruktioner. Varje post omfattar då en hel sektor på flexskivan (minus identifikation), alltså 253 tecken totalt. Lämpligen delar man in denna post i delposter eller underposter beroende på applikationen.

För att hantera filer med direkt åtkomst, finns följande CALL-rutiner:

CALL (28666%,F%) Sätter internpekare för fil F% på strängen Q0\$, då denna kan förstöras

CALL (28668%,P%) Läser av post P% till strängen Q0\$

CALL (28670%,P%) Skriver post P% från strängen Q0\$

F% = filnummer

P% = postnummer i filen

Postnumret är således relativt inom filen och börjar med värdet 0 för första posten.

För att läsa en post från en öppen fil, gör Du så här:

```
10 Z%=CALL (28666%,F%) + CALL(28668%,P%):A$=Q0$
```

Här har värdet i Z% ingen betydelse, och A\$ är den sträng, som posten hamnar i.

För att skriva en post från en öppen fil, gör Du följande:

```
20 Z%=CALL(28666%,F%):Q0$=A$:Z%=CALL(28670%,P%)
```

där A\$ är den sträng, som skall skrivas.

Exempel på läsning från filen VAROR.DAT:

```
10 REM *** LÄSRND ***
20 OPEN "VAROR.DAT" AS FILE 1
30 ; "ANGE POSTNUMMER SOM SKALL LÄSAS";
40 INPUT P%
50 Z%=CALL (28666%,1%) + CALL(28668%,P%)
60 ;Q0$
70 GOTO 30
```

Med detta program kan Du se vad filen VAROR.DAT innehåller. Om Du anger för högt värde, hamnar Du naturligtvis utanför filen och får då felmeddelande.

Vid preparering av en ny fil, som skall användas för direkt åtkomst, bör Du lämpligen fylla den med blanka tecken (mellanslag), för Du kan inte skriva en valfri post direkt efter prepareringen.

Exempel på preparering av filen UPPGIFT.TXT för 2 000 tecken:

```
10 REM *** PRE2000 ***
20 PREPARE "UPPGIFT.TXT" AS FILE2
30 P%=2000/253 : REM Antal poster
40 A$=SPACE$(253):REM Blank sträng
50 FOR P1%=0% TO P%
60 Z%=CALL(28666,2%):Q0$=A$:Z%=CALL(28670%,P1%)
70 NEXT P1%
80 CLOSE 2
90 END
```

Beskrivning av ett lagerhållningsprogram

LAGER är ett demonstrationsprogram på flexskivan FLEXDEMO. Programmet behandlar ett varuregister, som finns på flexskivan VAROR. Varje vara omfattar här följande utrymme på flexskivan:

Varunummer	0 tecken (styrfunktion)
Varunamn	19 tecken
Antal varor i lager	4 tecken
Beställningspunkt	4 tecken
Varutyp	4 tecken

Programmet kan ta upp nya varor, lista varor, utföra transaktioner och ta bort varor.

Programmet LAGER är inte ett komplett lagerhanteringssystem, utan vill visa hur Du hanterar filer med direkt åtkomst samt hur uppläggnigen av ett program kan se ut.

LAGER Sätt in flexskivan FLEXDEMO i enhet 0 (DRO:) och VAROR i enhet DRI: Starta programmet med RUN LAGER.

<u>LAGERREGISTER</u>	
VARUGRUPP: ABC 80 SYSTEM	
Rutinnr	
1	NYA VAROR
2	LISTNING/BESTÄLLNING
3	TRANSAKTION (IN/UT-HANTERING)
4	BORTTAGNING AV VAROR
5	SLUT
Skriv rutinnummer (utan RETURN)	

Nu väljer Du vilken underrutin Du vill köra. Alla underrutiner återvänder hit till huvudrutinen vid avslut.

Har Du ingen flexskiva med ett varuregister i drivenhet
DRI: (eller luckan inte är stängd), visas följande:

```

                LAGERREGISTER
VARUFLEXSKIVA SAKNAS I DRIVENHET 1
ELLER VARUFIL SAKNAS PA FLEXSKIVAN

SKALL NY VARUFIL LÄGGAS UPP (J/N) ?N

FINNS VARUFLEXSKIVA I DRIVENHET 1 ?J

```

Om Du vill ha ett nytt varuregister på en annan skiva än
flexskivan VAROR, svarar Du J på första frågan. (Se vida-
re "LAGERPREP" längre fram).

LAGERNYA

Välj rutin 1, NYA VAROR, för att ange en ny vara i re-
gistret.

```

                UPPLÄGGNING NYA VAROR
VARUGRUPP: ABC 80 SYSTEM
UPPG.NR

1   VARUNUMMER      : ■
2   VARUNAMN       :
3   ANTAL I LAGER   :
4   BESTÄLLNINGSPUNKT:
5   TYPBETECKNING  :
    ANGE VARUNUMMER

MINSTA VARUNR = 1000  HÖGSTA = 1200

Tryck bara RETURN för avslut

```

Här anger Du varans namn, antal varor i lager från bör-
jan, beställningspunkt och typbeteckning. För varje upp-
gift står det längst ner på skärmen vilka restriktioner
som gäller.

Ange uppgifterna så här:

```

                UPPLÄGGNING NYA VAROR
VARUGRUPP: ABC 80 SYSTEM
UPPG.NR

1   VARUNUMMER      :1018
2   VARUNAMN       :80 Kol.printer P80:
3   ANTAL I LAGER   :0
4   BESTÄLLNINGSPUNKT:20
5   TYPBETECKNING  :AP40

ÄR UPPGIFTERNA RIKTIGA (J/N)? J

```

När alla uppgifterna är korrekta, svarar Du med J på sista frågan, och varan läggs upp i registret. För att avsluta inmatningen, trycker Du RETURN när frågan efter varunamnet kommer upp, samt svarar N på frågan om mera varor skall läggas upp.

LAGLIST

Välj rutin 2, LISTNING/BESTÄLLNING, ur huvudprogrammet för att få ut en lista av varuregistret:

```

                LISTNING AV VAROR
VARUGRUPP: ABC 80 SYSTEM
Skall listningen ske på PRINTER (J/N) ?N
LISTNING AV
1  ALLA VAROR
2  BESTÄLLNINGSVAROR
3  VAROR AV VISS TYP
   VILKEN LISTNING SKALL SKE? 1

```

Först väljer Du om listningen skall ske på eventuellt inkopplad PRINTER. Sedan väljer Du listningsvillkor. Prova alla 3 alternativen! När skärmen blir full, trycker Du själv en tangent för nästa uppgift. När listningen är avslutad, svarar Du på frågan om fler listningar skall göras.

Om Du väljer alternativ 3, skall Du även ange en typbeteckning, för att skilja ut de varor som skall listas.

```

VILKEN LISTNING SKALL SKE?3
SKRIV DEL AV ELLER HELA TYPBETECKNINGEN
SOM SKALL LISTAS?HI
TYPLISTA MED TYP :HI
VARUGRUPP :ABC-80 SYSTEM
=====
VARUNR   VARUNAMN           LAGER BEST TYP
=====
1028 Houston HI-PLOT      0023 0015 HIPL
1029 Houston HI-PAD       0020 0004 HIPD

```

I detta fall är varutypen "HI" och sorterar därmed ut alla varor med den beteckningen. För att få en lista på tillhörande litteratur, väljer Du typbeteckningen "ALIT". Avsluta listningen med att svara N på frågan om ytterligare listningar.

LAGTRANS

Välj underrutin 3 för TRANSAKTION av varor:

```

          TRANSAKTION AV VAROR
VARUGRUPP: ABC 80 SYSTEM
UPPG.NR           (U/I/A)
1   VARUNUMMER   :
2   VARUNAMN    :
3   ANTAL I LAGER :
4   BESTÄLLNINGSPUNKT:
5   TYPBETECKNING :
ANGE UTTAG, INSÄTTNING ELLER AVSLUT
```

Du börjar med val av uttag, insättning eller avslut med en bokstav (utan RETURN).

Gör ett uttag på varunummer 1051 (Flexskivor, ej formaterade) på 500 st:

```

          TRANSAKTION AV VAROR
VARUGRUPP: ABC 80 SYSTEM
UPPG.NR           ===UTTAG===
1   VARUNUMMER   :1051
2   VARUNAMN    :FLEXSKIVOR ej form
3   ANTAL I LAGER :700
4   BESTÄLLNINGSPUNKT:240
5   TYPBETECKNING :AFLX
ANTAL UTTAG?     :500
MIN = 0, MAX = 700
ÄR UPPGIFTERNA RIKTIGA (J/N)? J
GÖR BESTÄLLNING
TRANSAKTION SKER AV VARAN
```

Ange då U (för uttag), varunummer 1051 och 500 vid antal uttag. Programmet kontrollerar hela tiden att Du håller Dig inom den givna intervallen. Är uppgifterna korrekta, svarar Du J vid motsvarande fråga. Om beställningspunkten underskrids genom uttaget, blir Du påmind att göra ny beställning. Avslut sker genom att Du anger A i första valet och svarar med N på frågan om fler transaktioner.

LAGBORT

Välj rutin 4 för BORTTAGNING AV VAROR.

Du anger varunumret som skall tas bort (inom angivna nummer) och får upp varan i fråga för kontroll.

BORTTAGNING AV VAROR

VARUGRUPP: ABC 80 SYSTEM

1 VARUNUMMER :1018
2 VARUNAMN :80 Kol.printer P 80
3 ANTAL I LAGER :0
4 BESTÄLLNINGSPUNKT:0000
5 TYPBETECKNING :0020

SKALL UPPGIFTEN TAS BORT (J/N)? J

Ange att den tidigare inlagda varan 1018, 80 kol. printer, skall bort genom att skriva varunumret.

Svara J på frågan om varan skall bort. Avslut görs genom att endast trycka RETURN vid varunumret samt svara med N på frågan om fler borttagningar.

LAGPREP

Om Du vill ha ett nytt varulager, sätter Du in en flexskiva, som inte redan innehåller ett varulager, i drivenhet 1 (DRT:) och skivan FLEXDEMO i drivenhet 0 (DRO:).

Starta med RUN LAGER och svara J på frågan om nytt varulager.

NYTT LAGER

STOPPA IN EN FLEXSKIVA I DRIVENHET 1 SOM
SKALL INNEHÅLLA DET NYA VARUREGISTRET

ÄR DETTA KLART (SVARA JA)? JA

Svara JA på frågan om skivan är klar, skriv sedan varugrupsnamnet, antalet varor som skall få plats och förs- ta varunumret.

Uppgifterna kontrolleras och därefter prepareras en ny varugrupp på skivan i enhet 1.

NYTT LAGER

PREPARERING SKER AV SKIVAN

VARUNUMMER :6496

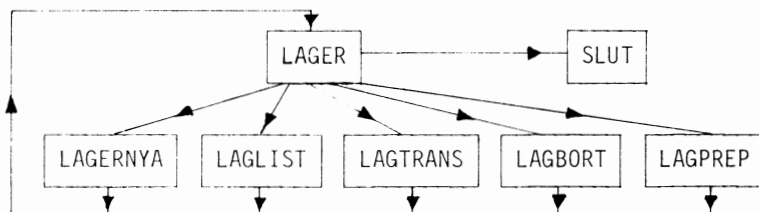
ANTAL VAROR BLIR 500 STYCKEN

Under prepareringen ser Du på skärmen vilka nummer som läggs upp. Efter prepareringen fortsätter Du med huvud- rutinen som tidigare.

Kommentarer till programmet LAGER

Programmet består av huvudprogrammet LAGER och 5 underprogram: LAGERNYA, LAGLIST, LAGTRANS, LAGBORT och LAGPREP.

Dessa underprogram påverkar filerna VARUNR.DAT och VAROR.DAT.



Varje vara tar 31 byte (tecken) och Du får därför plats med 8 st per post ($8 \times 31 = 248$ byte där en sektor är på 253 byte).

Du kan maximalt få plats med 2264 varuposter på en flexskiva eftersom filen VAROR.DAT är max 283 sektorer lång (283 sektorer = 8 delposter/ sektor = 2264 delposter). Ett detaljerat flödesschema (samt programlistning) hittar Du i Appendix D.

Det första som sker när Du skall köra programmet, är en kontroll att det finns en lämplig skiva i drivenhet 1. Detta sker med en programmerad felhantering. Satsen 110 ONERROR GOTO 370 i programfilen LAGER medför att om varufiler eller flexskiva saknas i drivenhet DR1: hoppar programmet till rad 370, och Du får tillfälle att starta ett nytt lager eller att sätta in en flexskiva.

Varunumren lagras inte i varuposten. Numren är egentligen inte nödvändiga, men är medtagna för att Du lättare skall se hur programmet arbetar med filerna. Vid listning av lagret kan man t ex tydligt se när varunumren räknas upp, hur datorn läser 8 poster i taget.

Namnet på varugruppen tillsammans med första och sista varunummer lagras på en särskild fil "VARUNR.DAT" och varuposterna på en annan fil "VAROR.DAT". Denna uppdelning gör programmet enklare.

I programfilen LAGPREP på raderna 570 - 630 reserveras plats på skivan för varulagret. Du får då också en kontroll på att utrymmet på skivan räcker till samtidigt med att varunumren visas på bildskärmen. Om utrymmet ej räcker, ger programmet uppgift om högsta varunumret som får plats.

KAPITEL 4

DISK OPERATIV SYSTEMET

Kommandointerpretator

För att sköta överföringen mellan flexskivan och ABC 80 finns ett kontrollkort och ett styrprogram. Programmet kallas DOS (Disk Operativ System) och ligger i ROM från adress 24576 -28672. Till systemet finns en uppsättning kommandon, som förutom tidigare nämnda funktioner, kan hantera operativsystemet.

Huvudprogrammet för hanteringen är CMDINT.SYS (från eng. command interpreter.system file).

När Du skall starta huvudprogrammet måste Du ha en systemskiva i en av drivenheterna och skriver BYE. Därmed lämnar Du helt och hållet BASIC:en.

Följande utskrift kommer upp:

```
ABC 80 DISC OPERATING SYSTEM
VERS 2.1 MAY '79
* R E A D Y *
```

Om filen CMDINT.SYS saknas på flexskivan, återgår Du till BASIC.

Kommandon på systemskivan

Följande kommandon finns på systemskivan:

SPACE \ När Du arbetar med en skiva fylls den alltmer och det är
CHT då bra att veta hur mycket plats det finns kvar för nya
filer, d v s hur många sektorer om 253 byte återstår.
Det är också bra att veta hur många nya filnamn (directory
entries) som kan skapas (max 120).

Exempel

```
REMAINING SPACE ON DRIVE 0:
  23 SECTORS  98 DIRECTORY ENTRIES
REMAINING SPACE ON DRIVE 1:
  237 SECTORS 116 DIRECTORY ENTRIES
```

MEM Kommandot ger en minneskarta över ROM- och RAM-minne i
ABC 80 systemet. Skriver Du kommandot MEM, visar bild-
skärmen följande:

```
MEM
ABC 80 MINNESKARTA (VER 1.1)
      FÖRSTA  SISTA  CHECKSUMMA
ROM   0000   3FFF   2C09
ROM   6000   6FFF   D8F1
RAM   7000   7FFF
RAM   0000   FFFF
TOTAL ROM-AREA:  20 K-BYTE
TOTAL RAM-AREA:  17 K-BYTE
```

Alla sifferuppgifter är hexadecimala. Räkna Du om första checksumman till decimalt värde, skall den bli densamma som checksumman på kortet under ABC 80's tangentbord.

(COPY

LIBRARY)

enhet: filnamn .filtyp ,kopians filnamn .kopians filtyp

Detta kommando går att använda utan att man vet filstorlek, filtyp, etc, vilket är svårt med andra kommandon. Det ger också en mycket snabb kopiering.

Utelämnas kopians filtyp, får kopian samma filtyp som originalet. Skall många filer kopieras, kan det vara bekvämare att använda COPYLIB.

Exempel

COPY DRO: KLOKER.DAT , DR1: TOKER.DAT

COPYLIB

originalenhet, kopieenhet

Detta kommando kopierar hela eller delar av alla de filer, som finns på originalenhetens flexskiva till kopieenhetens skiva.

Kommandot svarar med:

COPY FILES UNDER DIRECTORY CONTROL V2.1

LEGAL RESPONSES ARE :

A - COPY ALL REST OF LIBRARY (= kopierar resten av biblioteket)

X - EXIT DON'T COPY ANYTHING (= återgår, kopierar ingenting)

I - IGNORE REST OF LIBRARY (= låt bli resten av biblioteket)

Y - COPY THIS FILE (= kopiera denna fil)

Y=NEW.FIL COPY USING THE NEW NAME

(= kopiera och ge kopian ett nytt namn "NEW.FIL")

COPY filnamn?

Datorn kommer nu att räkna upp alla filer som finns i originalenhetens bibliotek och väntar sig för varje fil något av svaren enligt listan ovan. Endast RETURN ger nästa filnamn i uppräknings och ingen kopiering.

Exempel

COPYLIB DRO:, DR1: kopierar från drivenhet 0 till 1.

När alla filer är uppräknade, startar kopieringen och ABC 80 skriver:

PHASE 2 DO THE COPY

Filnamn *** RECORDS COPIED

Du får nu en lista över alla filer vartefter de kopieras med antalet records (= sektorer) som filen upptar. Kommer i stället på bildskärmen upp "ABT CODE 41" betyder det att kopieenhetens flexskiva är full (se felmeddelandelistan).

Kommandot gör i ordning en flexskiva för användning. För att Du skall kunna använda en viss flexskiva, är det nödvändigt att en del grundläggande information finns inskriven på skivan. Denna information består av grunderna till DOS filkatalog, där filnamnen lagras tillsammans med de delar av skivan, som är dåliga och som inte skall användas för inskrivning.

Under iordningsställandet av en skiva raderas alla inspelade filer på skivan ut. För att förhindra, att detta sker av misstag, ställer programmet Dig ett antal frågor efter kommandot DOSGEN för att få bekräftat, att det inte är frågan om något misstag.

DOSGEN kontrollerar alla sektorer på skivan och ger ett meddelande om vilka sektorer som är dåliga.

Exempel

DOSGEN,F DR1: innebär att flexskivan i drivenhet DR1: skall formateras och testas för filhantering.

```
Svara på de frågor som ställs. På bildskärmen skrivs
följande: FLOPPYN IDRIVE 1 KOMMER ATT RENSAS HELT
VID INITIERINGEN. ÄR DET OK? <J/N>J
ABSOLUT SÄKER?? (J/N) J
JAG FORMATTERAR FLOPPYN!!!
JAG TESTAR FLOPPYN (HAV TALAMOD) !
SPAR 39 ÄR DÄLIGT (BORTTAGET) !
NU ÄR DET KLART!!!
```

* R E A D Y *

I detta fall har det blivit ett fel vid spår 39 och därmed reduceras antalet totala sektorer med 8 st.

DOSGEN DR0: innebär att flexskivan i drivenhet DR0: rensas på allt och testas för filhantering.

Svara på de frågor som ställs. Utskriften på bildskärmen blir som ovan, med undantag av raden om formatering och eventuellt dåligt spår.

Efter att ha preparerat en flexskiva, är det lämpligt att först lägga upp flexskivans namn och LIB-programmet från den andra skivan. Detta gör Du i BASIC på följande sätt:

Skriv NEW för att rensa eventuella program. Skriv sedan SAVE DR1: SKIVNAMN.LBL om den nya skivan skall heta så och finns i drivenhet DR1:

Skriv RUN LIB och svara med S på frågan om printerlista, och J på frågan om Du skall lagra LIB-programmet samt ange drivenheten (DR0: eller DR1:).

Härefter bör Du kopiera över BASICERR.SYS och övriga systemprogram med hjälp av COPY LIB (se sid 29). Se vidare Appendix B för vilka filer som ingår som systemprogramvara.

Återgång till BASIC

Du återgår till BASIC genom att skriva xBAS. Detta kommando utför samma sak, som när Du trycker på RESET-knappen. Tidigare program är således borta och även extra inlagda drivrutiner för t ex printer-V24:.

APPENDIX

TEKNISK BESKRIVNING

Flexskivan drivs av en likströmsmotor med varvet 300 rpm. Varvtalet hålls konstant med en tachometeråterföring.

Inspelning på skivan sker på 40 spår (track), som vardera är 0,047 mm breda.

Skrivhuvudet flyttas mellan spåren av en stegmotor och en flyttning tar ca 25 ms.

Varje spår är indelat i 8 sektorer (sector) och i en sektor kan 256 byte lagras. Totalt kan alltså 81920 byte lagras. Av dessa går en del åt till biblioteksrutiner, varför Du disponerar över ca 72000 byte.

Accesstiden utgörs av den tid det tar att flytta lässkrivhuvudet till rätt spår, dels av den tid det dröjer tills rätt sektor har kommit under lässkrivhuvudet. Om man antar, att lässkrivhuvudet skall flyttas 10 spår och att skivan skall vrida sig 1/2 varv blir accesstiden:

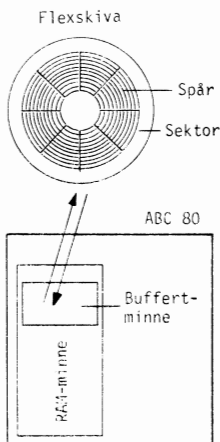
$$10 \cdot 25 + \frac{60 \cdot 1000}{300} \cdot 1/2 = 350 \text{ ms}$$

Överföringshastigheten är 125 kbit/sek.

Att läsa en sektor tar således

$$\frac{256 \cdot 8}{125} = 16 \text{ ms}$$

Inspelning på skivan sker med frekvensmodulering (se "Mikrodatorns ABC", sid 163). Felfrekvensen uppges till 1 per 10⁹.



Vid kommunikation med ABC 80 används en del av RAM-minnet av flexskivenheten som buffertminne (DOSBUF).

Det finns 8 buffertminnen och varje buffertminne rymmer 256 byte.

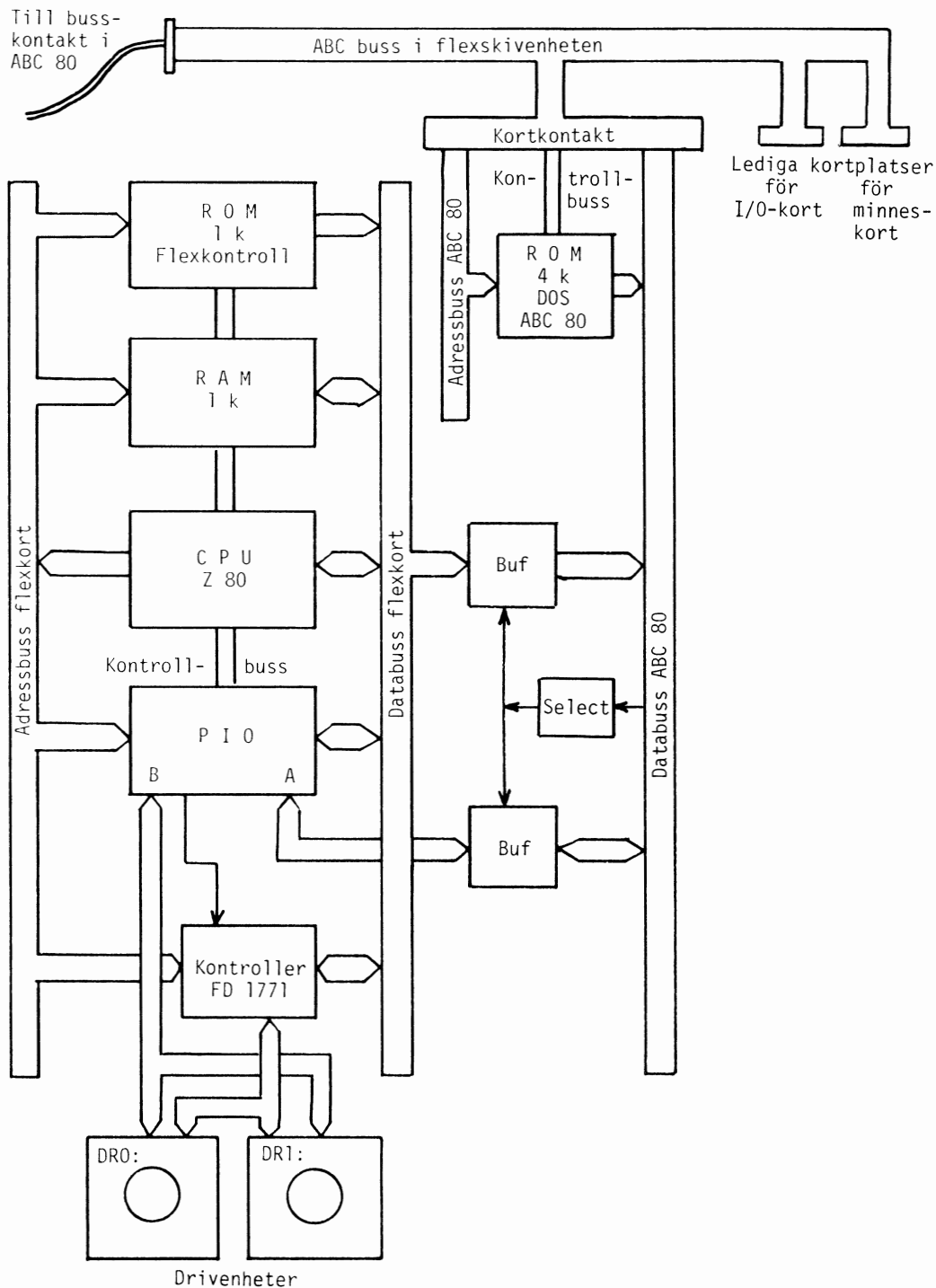
Vid skrivning på en fil lagras först de data som filen skall innehålla i buffertminnet. När buffertminnet är fullt, överförs hela buffertminnets innehåll på en gång till flexskivan, varvid således en hel sektor skrivs på en gång.

Läsning från en fil sker på motsvarande sätt. Begäran om läsning av en viss post medför att hela den sektor, där posten finns, överförs till buffertminnet. Om nästa post, som skall läsas, finns i samma sektor, sker ingen överföring från flexskivan, utan posten läses från buffertminnet direkt.

TEKNISK SPECIFIKATION av flexskivenhet till ABC 80

Typbeteckning	FD2	FD2U
Beskrivning:	Flexskivenhet med plats för 2 flexskivor Djup modell	Flexskivenhet med plats för 2 flexskivor Bred modell med plats för bildsärm ovanpå
Dimensioner: Bredd Djup Höjd Vikt	17,2 cm 42 cm 16,3 cm 8,8 kg	57,5 cm 25 cm 17,7 cm vid drivenheten 9,6 cm vid bildskärm 11,6 kg
Kraftförsörjning: Nättaggregat Strömförbrukning nominell	220V 150 VA 12V 3A 5V 3A 12V 0,8A 5V 0,3A per drivenhet 12V 0,04A 12V 0,03A Flexkontrollkortet 5V 0,8A	220V 200VA 12V 6A 5V 6A Fläktkyld
Kortplatser, varav en plats till flexkontrollkortet Flatkabel-längd till 4680-kontakten i ABC 80, minst ca	3 st 60 cm	6 st 10 cm
Drivenhet-typ:	Pertec FD 200 Micro Floppy Disk, anpassad för 5 1/4" flexskivor, mjukvarusektorerade	
Kapacitet på en drivenhet: Överföringshastighet Söktid Antal spår Antal sektorer Antal bytes	125k bits per sek 25 msec spår till spår 40 st 8 per spår, 320 totalt 256 per sektor, 81920 totalt, varav ca 72000 för användaren	

BLOCKSCHEMA ÖVER FLEXSKIVENHETEN



FLEXSKIVOR TILL ABC 80

Till ABC 80 skall standard flexskivor 133 mm (5,25 tum) användas. Dessa finns i princip i 3 utföranden:

1. Blanka, d v s helt oinspelade skivor. Dessa måste före användning genomgå s k generering med systemprogrammet DOSGEN (se Kap 4).
2. DOSgenererade. Dessa skivor är genererade från fabrik med extra hög precision, vilket är nödvändigt för att garantera kompatibilitet, d v s att en skiva som är inspelad på en viss flexskivenhet, går att spela av på en helt annan skiva
3. Formatterade, d v s genererade på en icke ABC-utrustning. Funktion och kompatibilitet kan inte garanteras!

Filer i systemprogramvaran

En flexskiva med systemprogramvaran innehåller:

SKIVNAMN.LBL	Flexskivans namn
LIB .BAC	Biblioteklistningsprogram
BASICERR.SYS	Felkommandon i klartext
CMDINT .SYS	Kommandointerpretator för .ABS program
COPYLIB .ABS	Kopiering av många filer
COPY .ABS	Kopiering av en fil
DOSGEN .ABS	Dosgenerering av flexskiva
MEM .ABS	Minneskarta över ABC 80
SPACE .ABS	Resterande utrymme på flexskivorna

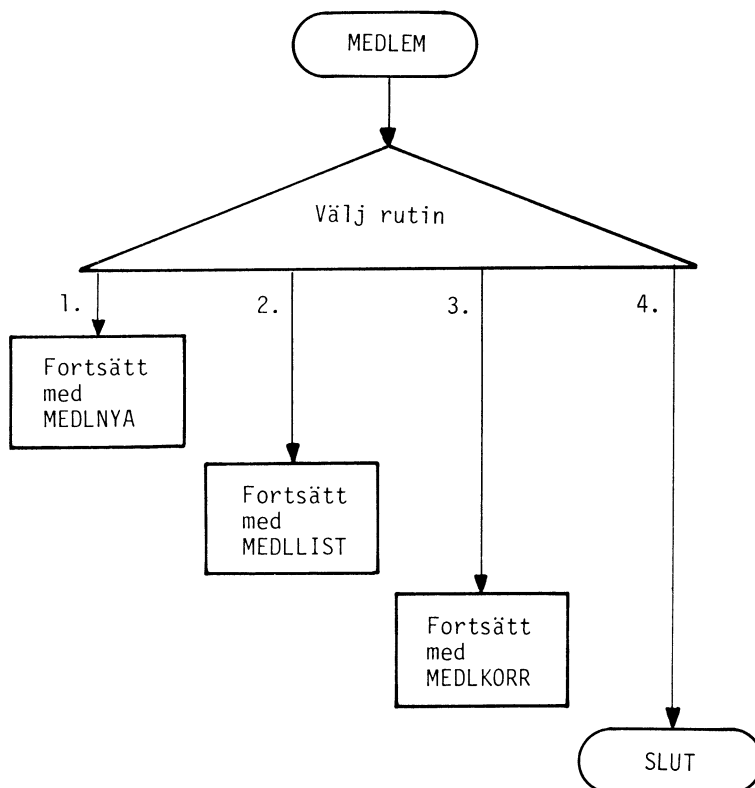
FLÖDESSCHEMA OCH PROGRAMLISTNING "MEDLEM"

Huvudprogram

```

10 REM ***** MEDLEM *****
20 REM 790220 HM
30 ; CHR$(12%)
40 ; "      MEDLEMSREGISTER"
50 ; "      ";CHR$(151%);STRING$(15%,44%)
60 ; CUR(4,0);"Rutinnr."
70 ; CUR(6%,3%);"1      NYA MEDLEMMAR"
80 ; CUR(8%,3%);"2      LISTA MEDLEMMAR"
90 ; CUR(10%,3%);"3      KORRIGERA MEDLEMMAR"
100 ; CUR(12%,3%);"4      SLUT"
110 REM ----- VAL AV RUTIN -----
120 ; CUR(18%,0%);"Skriv rutinnummer (uten RETURN) ";
130 GET X$: ; X$
140 IF X$<"1" OR X$>"4" THEN 120
150 ; CUR(2%*VAL(X$)+4%,6%);CHR$(127%)
160 ON VAL(X$) GOTO 170,180,190,200
170 CHAIN "MEDLNYA"
180 CHAIN "MEDLLIST"
190 CHAIN "MEDLKORR"
200 END

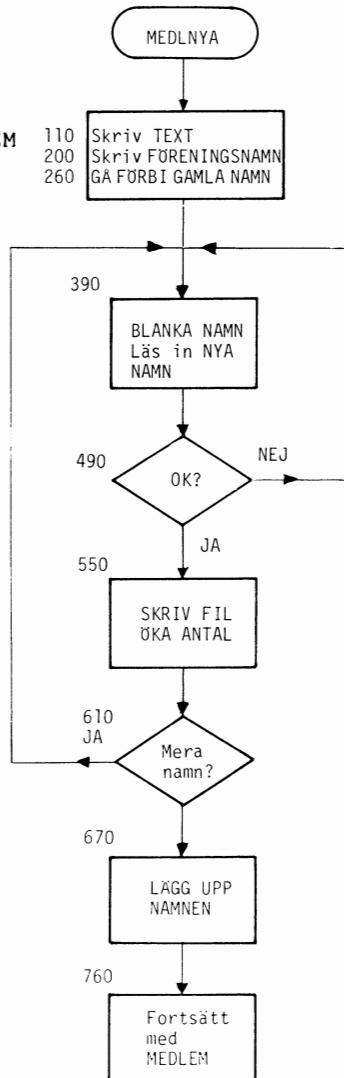
```




```

550 REM ----- SKRIV FIL -----
560 FOR B%=1% TO 4%
570 ; #3,B$(B%)
580 NEXT B%
590 REM ----- ÖKA ANTAL -----
600 A%=A%+1%
610 ; CUR(17%,0%)"MERA NAMN (J/N) ?";
620 GET X$: ; X$
630 ; CUR(17%,0%)SPACE$(40%)
640 IF X$="J" OR X$="j" THEN 390
650 IF NOT (X$="N" OR X$="n") THEN 610
660 ; CUR(17%,0%)SPACE$(40%)
670 REM --LÄGG UPP NAMNEN--STÄNG FILEN-
680 ; CUR(17%,0%) "NYA NAMN LÄGGS UPP PÅ FLEXSKIVAN"
690 CLOSE 1
700 CLOSE 2
710 CLOSE 3
720 OPEN "FÖRENING.DAT" ASFILE 1
730 ; #1,A$
740 ; #1,A% : REM NYA ANTAL MEDLEMMAR
750 CLOSE 1
760 CHAIN "MEDLEM" : REM GÅ TILL MEDLEM

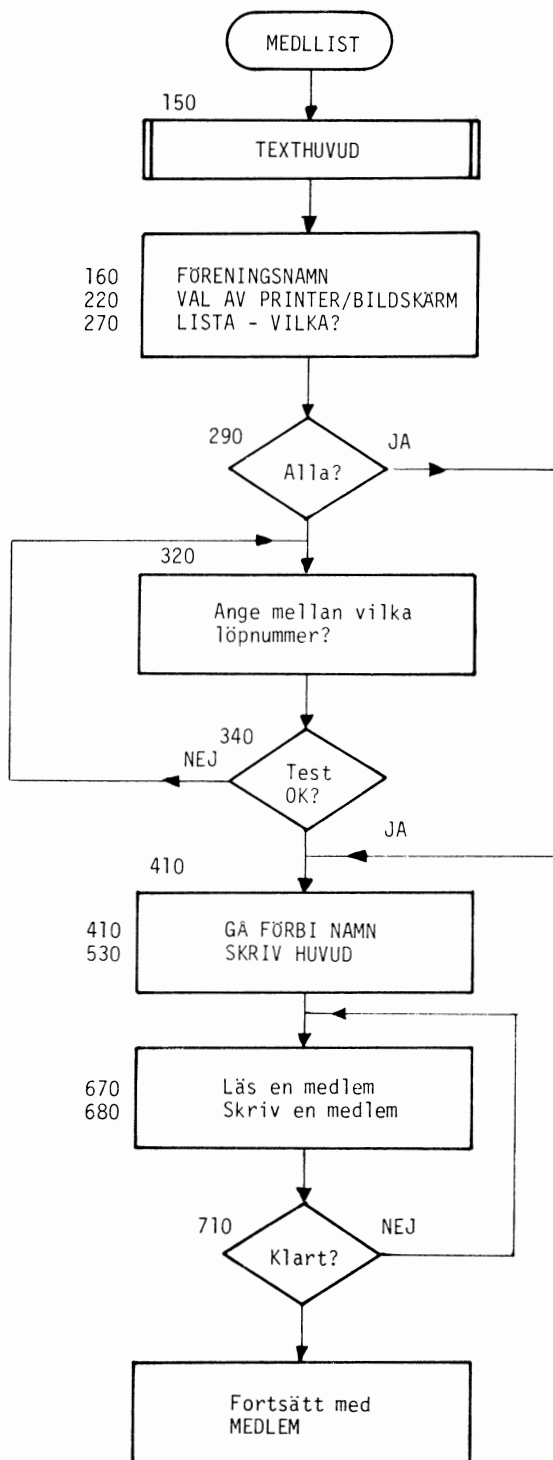
```




```

490 NEXT B%
500 NEXT A1%
510 REM ++++++ LISTNING ++++++
520 IF P%=0% THEN ; "Tryck en tangent för varje medlem" : ;
530 REM ++SKRIV+HUVUD++
540 ; #P%, "MEDLEMSLISTA "A$"
550 ; #P%, SPACE$(13%)STRING$(LEN(A%),61%)
560 ; #P%, "LÖPNUMMER"
570 ; #P%, "NAMN"
580 ; #P%, "ADRESS"
590 ; #P%, "POSTADRESS"
600 ; #P%, "MEDLEMSNR"
610 REM ++++++ UTSKRIFTSLOOP ++++++
620 FOR A1%=A2% TO A%
630 ; #P%
640 ; #P%,A1% : REM Skriv löpnummer
650 REM hämta data, skriv data
660 FOR B%=1% TO 4%
670 INPUTLINE #2,B%
680 ; #P%,LEFT$(B%,LEN(B%)-2%)
690 NEXT B%
700 IF P%=0% THEN GET Z$ : REM VÄNTA
710 NEXT A1%
720 ; : ; "LISTNING SLUT"
730 CLOSE 1
740 CLOSE 2
750 IF P%=4% THEN CLOSE 4
760 CHAIN "MEDLEM"
770 REM ***** TEXTHUVUD *****
780 ; CHR$(12%)
790 ; " LISTA MEDLEMMAR"
800 ; " ";CHR$(151%);STRING$(15%,44%)
810 RETURN

```

Medlemskorrigering

```

10 REM ***** MEDLKORR *****
20 REM 790305 HM
30 REM ""
40 REM A$=Föreningsnamn
50 REM A%=Antal medlemmar
60 REM A1%=Räknare för aktuellt löpnr.
70 REM B$(4%)=4 strängar för data
80 REM B%=Adress till uppgiftsnr. 1 - 4
90 REM C$=Tillfällig lagring av sträng
100 REM F$=Frågetext
110 REM F%=Svar J=> F%=1 , N=> F%=2
120 REM N%=Antal passerade medlemmar
130 REM R%=Rättning uppgiftsnr. 0 - 5
140 REM S$=Sökbegrepp (del av sökord)
150 REM S%=SÖK-UPPGIFTS-NR.
160 REM X%=J/N alternativ
170 REM -----TEXT-----
180 ; CHR$(12%)
190 ; CUR(0%,10%)"KORRIGERA MEDLEMMAR"
200 ; CUR(1%,9%)CHR$(151%)STRING$(19%,44%)
210 ; CUR(4%,0%)" LÖPNUMMER      :"
220 ; : ; "1 EFTER&FÖRNAMN  :"
230 ; : ; "2 GATUADRESS      :"
240 ; : ; "3 POSTADRESS      :"
250 ; : ; "4 MEDLEMSNUMMER  :"
260 REM -----FÖRENINGSNAMN -----
270 DIM B$(4%)=40%
280 OPEN "FÖRENING.DAT" ASFILE 1
290 INPUTLINE #1,A$
300 A$=LEFT$(A$,LEN(A$)-2%)
310 ; CUR(2%,0%)"FÖRENINGEN  ":"A$
320 INPUT #1,A%
330 A2%=A%
340 GOSUB 1470 : REM SÖKRUBRIK
350 REM ----- ÖPPNA NAMNFIL -----
360 OPEN "NAMN.DAT" ASFILE 2
370 OPEN "NAMN.DAT" ASFILE 3
380 A1%=0%
390 REM -----FRÅGA SÖKBEGREPP -----
400 GOSUB 1500 : REM BLANKA DATA+FRÅGA
410 ; CUR(19%,0%)"SKRIV SÖKBEGREPP FÖR MEDLEM MEN"
420 INPUTLINE S$
430 S$=LEFT$(S$,LEN(S$)-2%)
440 REM ---FRÅGA SÖKNR AV SÖKBEGREPP---
450 F$="VAR FINNS SÖKBEGREPPET (1-4) ?"
460 GOSUB 1620 : REM SVAR F% 0 - 5
470 IF F%=0% OR F%>4% THEN 460
480 S%=F%
490 ; CUR(2%*S%+4%,17%)S$
500 GOSUB 1530 : REM BLANKA FRÅGA
510 ; CUR(19%,10%)"SÖKNING PAGAR"
520 REM ---SÖKNING AV SÖKBEGREPP-----
530 N%=0%
540 ; CUR(4%,17%);A1%+1%;"  "
550 FOR B%=1% TO 4%

```

```

560 INPUTLINE #2,C$
570 B$(B%)=LEFT$(C$,LEN(C$)-2%)
580 NEXT B%
590 A1%=A1%+1% : N%=N%+1%
600 IF INSTR(1%,B$(S%),S$)>0% THEN 770
610 FOR B%=1% TO 4%
620 ; #3,B$(B%)
630 NEXT B%
640 REM ----- NÄSTA MEDLEM
650 IF N%=A% THEN 680
660 GOSUB 1760 : REM SISTA MEDLEM ?
670 GOTO 540
680 REM -----INGA MER NAMN FINNS -----
690 GOSUB 1530 : REM BLANKA FRÅGA
700 ; CUR(19%,6%)"SÖKBEGREPPET: "S$
710 ; "FINNS EJ I UPPGIFT:"S%
720 REM ---- SLUTLIG UPPDATERING -----
730 GOSUB 1680 : REM PASSERA TILL SISTA
740 F$="ÖNSKAS FLER SÜKNINGAR (J/N) ?"
750 GOSUB 1570 : REM SVAR F%
760 ON F% GOTO 360,1370
770 REM ----- MEDLEM ÄR FUNNEN -----
780 REM ----- RÄTTNING AV MEDLEM -----
790 ; CUR(16%,0%);CHR$(151%);
800 ; ",,,,,,,,,,,,,,,,,,,,,, RÄTTA MEDLEM ,,"
810 ; SPACE$(279%); : REM BLANKA
820 FOR B%=1% TO 4%
830 ; CUR(2%*B%+4%,17%)B$(B%)
840 NEXT B%
850 ; CUR(14%,0%);"5 BORTTAGNING AV DENNA MEDLEM"
860 ; CUR(18%,0%)"Ange 0 för ingen mer rättning "
870 ; " och 5 för borttagning "
880 F$="VILKEN UPPGIFT SKALL RÄTTAS (0-5) ?"
890 GOSUB 1620 : REM SVAR F% 0 -- 5
900 R%=F%
910 IF R%=0% THEN 980
920 IF R%=5% THEN 1240
930 ; CUR(2%*R%+4%,17%)SPACE$(23%)
940 ; CUR(2%*R%+4%,17%); : INPUTLINE C$
950 B$(R%)=LEFT$(C$,LEN(C$)-2%)
960 GOTO 850
970 REM ----- RÄTTNING OK? -----
980 GOSUB 1530 : REM BLANKA FRÅGA
990 F$="ÄR MEDLEMSUPPGIFTEN RIKTIG (J/N) ?"
1000 GOSUB 1560 : REM SVAR F%
1010 GOSUB 1530 : REM BLANKA FRÅGA
1020 IF F%=2% THEN 850
1030 REM ----- UPPDATERING -----
1040 FOR B%=1% TO 4%
1050 ; #3,B$(B%)
1060 NEXT B%
1070 REM ----- MER RÄTTNINGAR ? -----
1080 GOSUB 1530 : REM BLANKA FRÅGA
1090 F$="SKALL FLER MEDLEMMAR RÄTTAS (J/N) ?"
1100 GOSUB 1560 : REM SVAR F%
1110 IF F%=2% GOTO 1370
1120 REM ---- SAMMA SÖKBEGREPP -----
1130 ; CUR(14%,0%)SPACE$(399%);

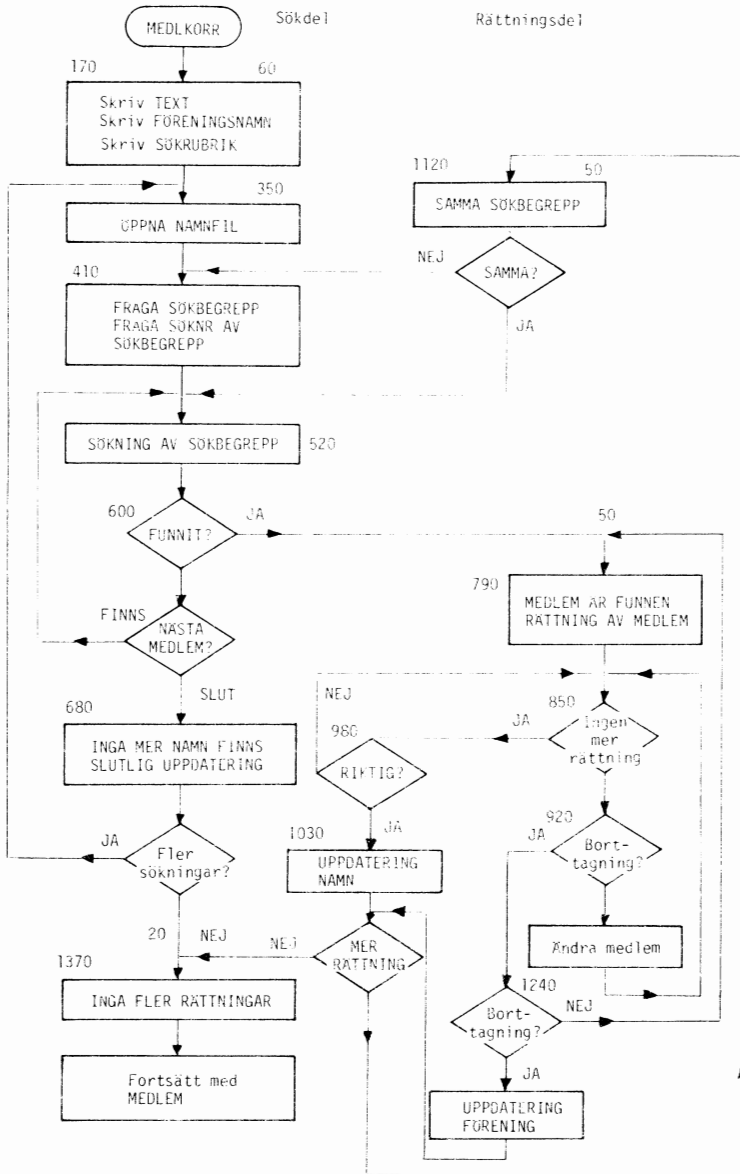
```



```

1720 NEXT B%
1730 ; CUR(4%,17%)A1% " "
1740 NEXT A1%
1750 GOTO 1780
1760 REM ## SISTA+1 = FÖRSTA MEDLEM ###
1770 IF A1%<A% THEN RETURN
1780 CLOSE 2
1790 CLOSE 3
1800 OPEN "NAMN.DAT" ASFILE 2
1810 OPEN "NAMN.DAT" ASFILE 3
1820 A1%=0%
1830 ; CUR(4%,17%)"1 " "
1840 RETURN

```



Nytt föreningsnamn

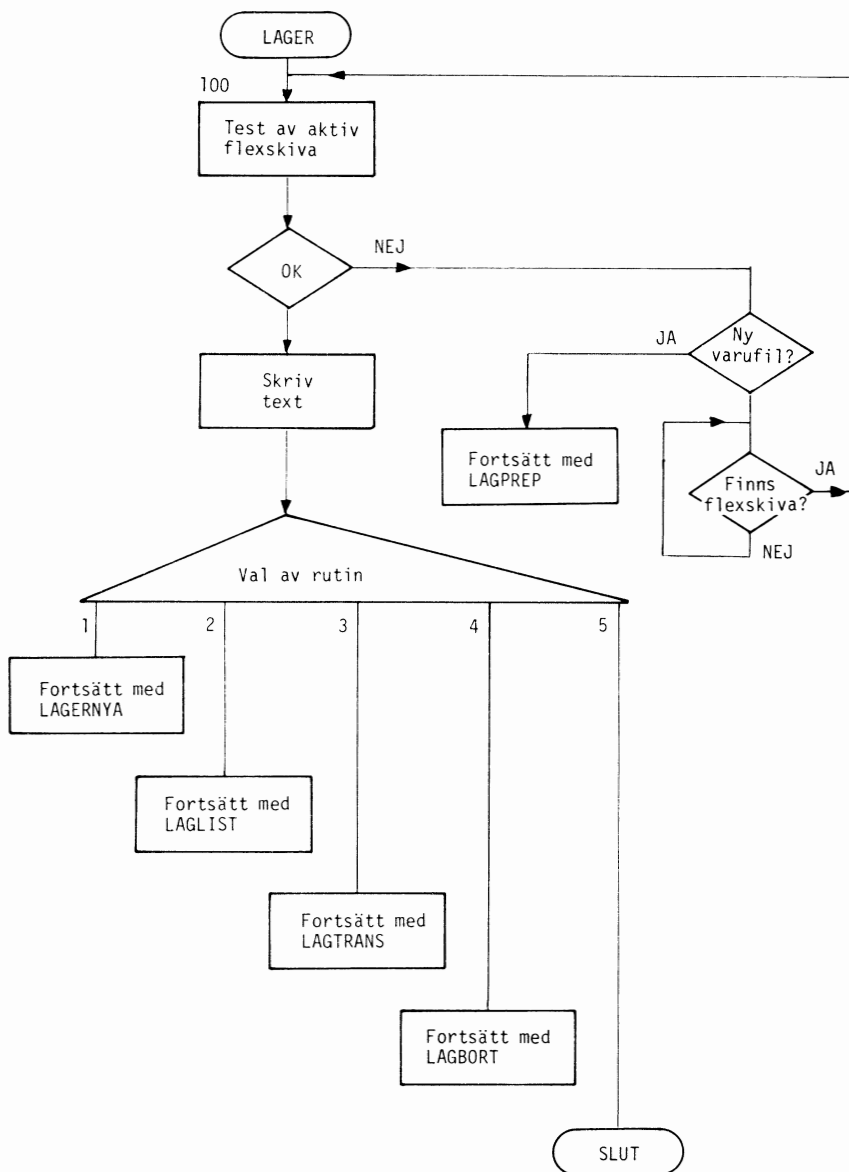
```

10 REM ***** FÖRENNY *****
20 REM 790223 HM
30 ; CHR$(12%)
40 ; "      NYTT FÖRENINGSNAMN"
50 ; "      ";CHR$(151%);STRING$(18%,44%)
60 ;
70 ; "Ny förening eller ändring av"
80 ; "namn på den gamla föreningen."
90 ; "Finns redan föreningsnamn (ja/nej)";
100 INPUT X$
110 IF X$="ja" OR X$="JA" THEN 310
120 IF NOT (X$="nej" OR X$="NEJ") THEN 90
130 REM ----- GÖR NY FÖRENING -----
140 ; "Vad är det nya namnet ?"
150 INPUTLINE A$
160 A$=LEFT$(A$,LEN(A$)-2%)
170 ;
180 ; "Är det nya namnet riktigt(ja/nej):";
190 INPUT X$
200 IF NOT (X$="ja" OR X$="JA") THEN 10
210 PREPARE "FÖRENING.DAT" ASFILE 1
220 ; #1,A$
230 ; #1,0 : REM ANTAL MEDLEMMAR=0
240 CLOSE 1
250 PREPARE "NAMN.DAT" ASFILE 1
260 CLOSE 1
270 ;
280 ; "FÖRENINGEN ";A$;" ÄR UPPLAGD "
290 GOTO 560
300 REM ++++++
310 REM ++++ BYT NAMN PÅ FÖRENING ++++++
320 OPEN "FÖRENING.DAT" ASFILE 1
330 INPUTLINE #1,A$
340 A$=LEFT$(A$,LEN(A$)-2%)
350 INPUT #1,A
360 CLOSE 1
370 ; "Gamla namnet är :";A$
380 ; STRING$(16%,32%);CHR$(151%);STRING$(LEN(A$),44%)
390 ; "Skriv det nya namnet:";
400 INPUTLINE B$
410 B$=LEFT$(B$,LEN(B$)-2%)
420 ;
430 IF LEN(B$)=0% THEN ; "Inget nytt namn" : GOTO 560
440 ; STRING$(20%,32%);CHR$(151%);STRING$(LEN(B$),44%)
450 ; "Är det nya namnet riktigt(ja/nej):";
460 INPUT X$
470 IF NOT (X$="ja" OR X$="JA") THEN 10
480 REM ++++++ INSKRIFT NYTT NAMN ++++++
490 OPEN "FÖRENING.DAT" ASFILE 1
500 ; #1,B$
510 ; #1,A
520 CLOSE 1
530 ; "Föreningen      :";A$
540 ; "har bytt namn till :";B$

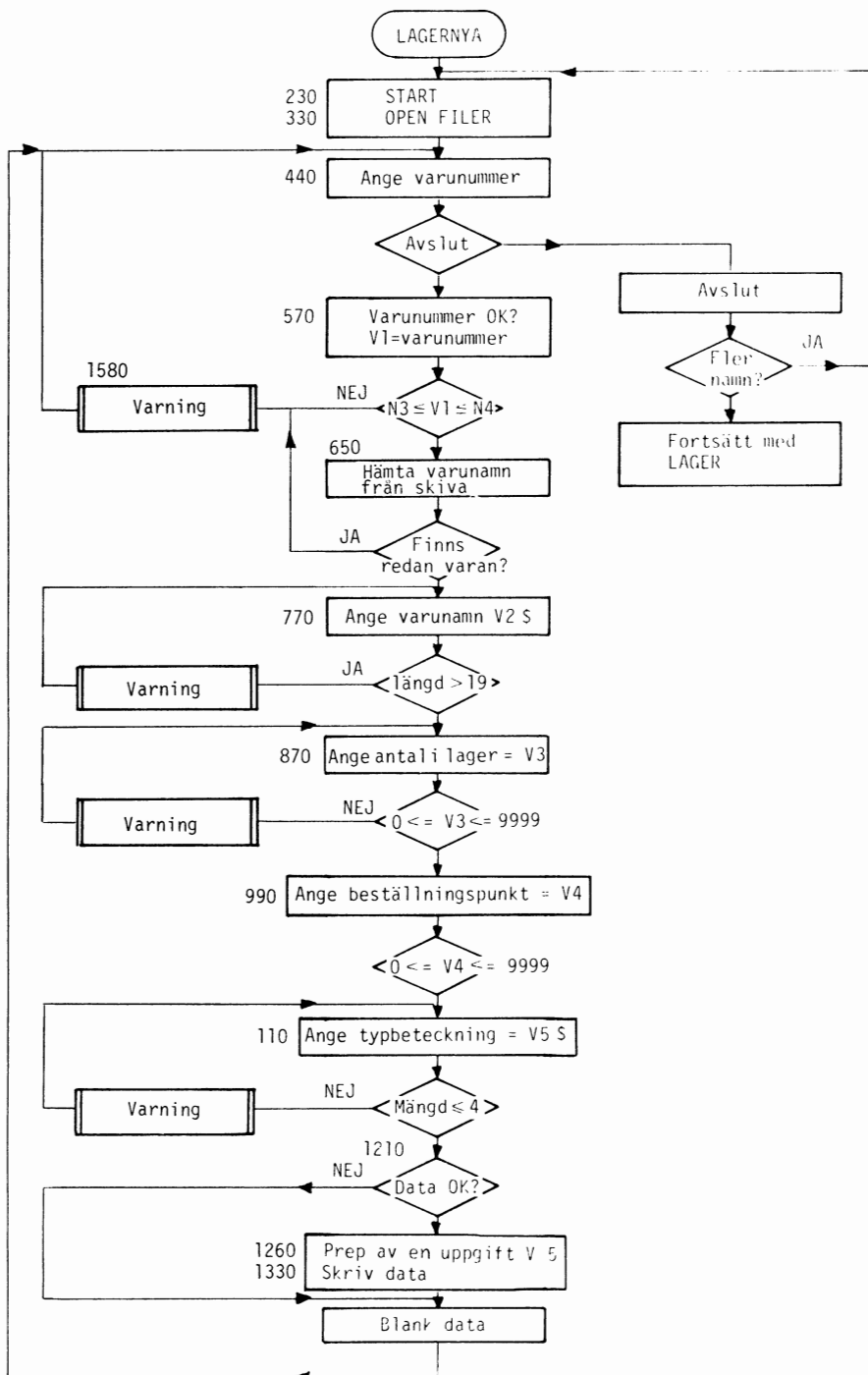
```


FLÖDESSCHEMAN "LAGER"

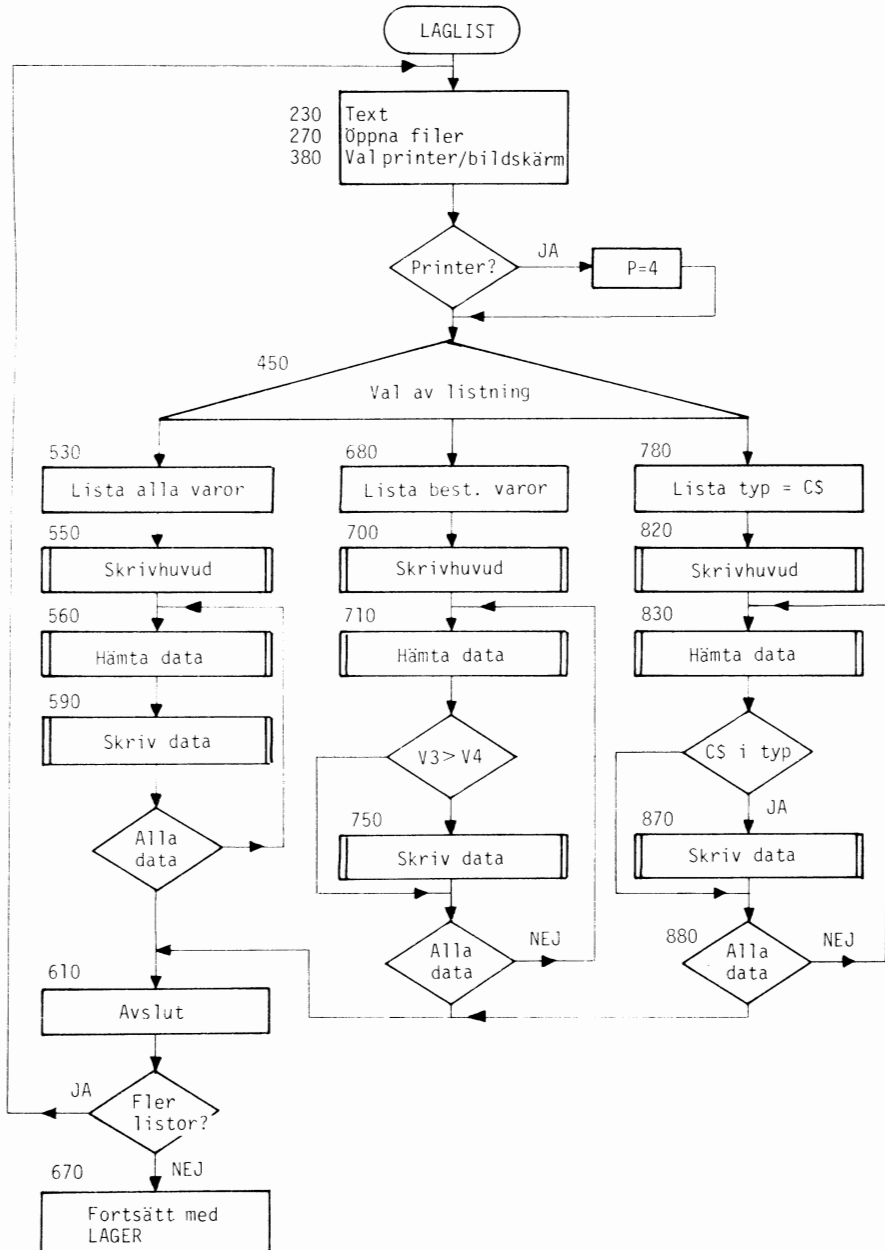
Huvudprogram (Lagerregister)



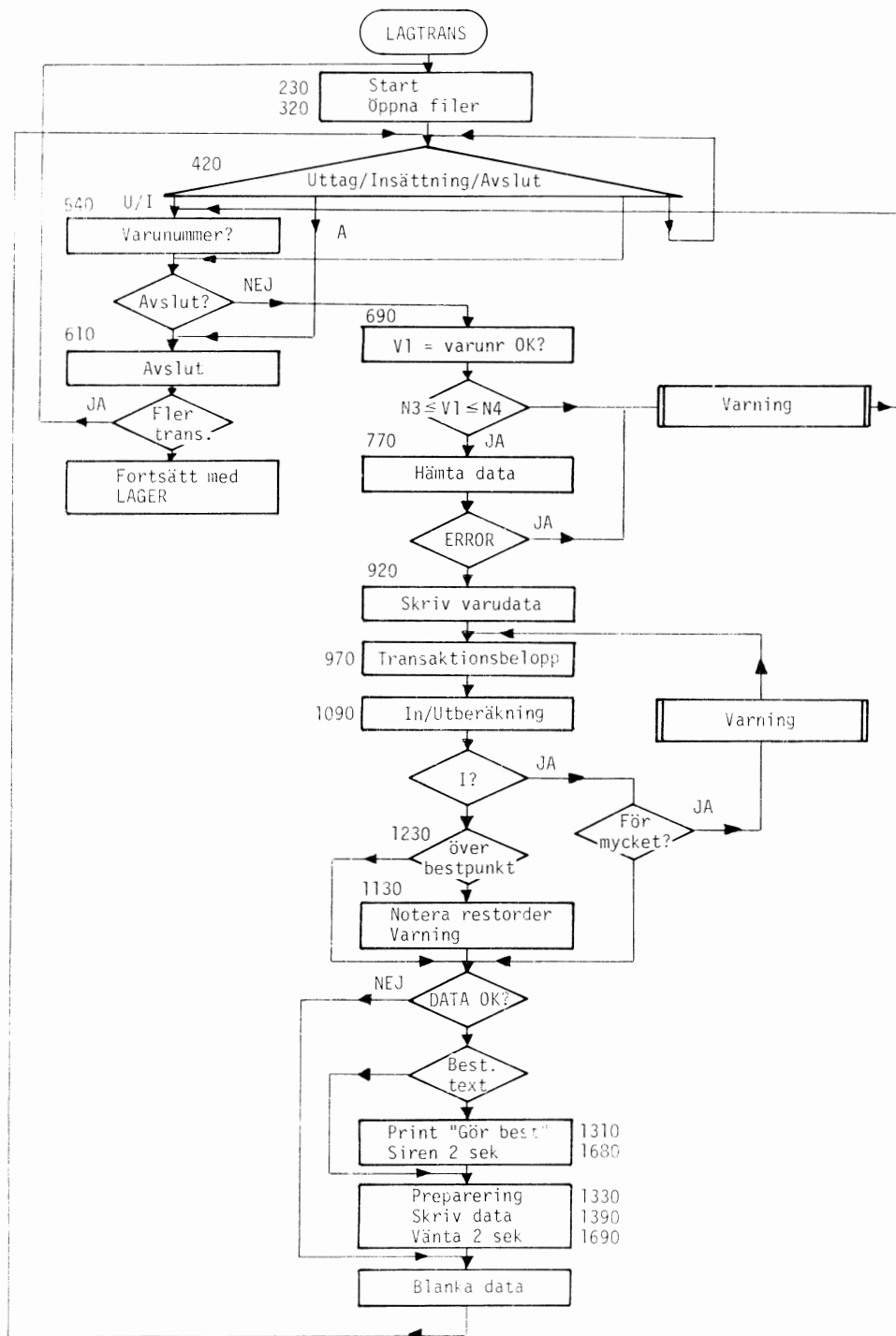
Uppläggnig av nya varor



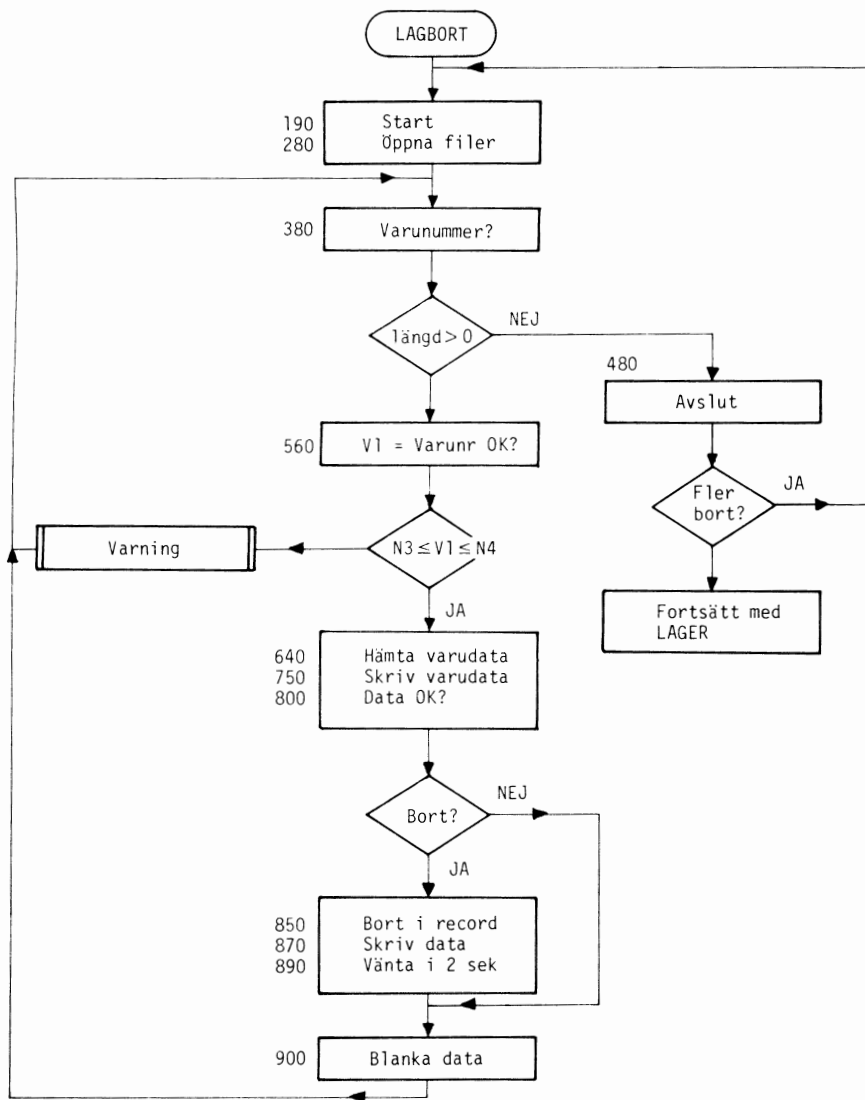
Listning av varor



Transaktion av varor



Borttagning av varor



Nytt lager

