



RC AS REGNECENTRALEN

SCANDINAVIAN INFORMATION PROCESSING SYSTEMS

RCSL no: 31-D134
Edition: April 1972
Author: Kirsten Mossin

Title: Plottingsystem RC 4000

Keywords: RC 4000, Software, Plotter, Algol 5, Fortran

Abstract: The Plotting system enables the user to control the plotting of vectors and texts from Algol- and Fortran-programs. The system is independent of the actual kind of plotter from the users point of view. At present it is implemented for on-line use of the Calcompplotter and off-line use of the Kingmaticplotter via punched tape. Danish edition. 31 pages.



SYSTEM LIBRARY

DK-2500 VALBY · BJERREGAARDSVEJ 5 · TELEPHONE: (01) 46 08 88 · TELEX: 64 64 rcinf dk · CABLES: INFOCENTRALEN

Indholdsfortegnelse

1. Funktion	2
2. Kørsel med plottersystemet	9
3. Fejludskrifter	11
4. Diskussion	12
5. Referenceliste	14
App. A Eks. paa programmer, der anvender plottersystemet	15
App. B Mønsterlagre	21
App. C Kørsel med magnetbaand	28

Plottersystemet er programmeret af Peter Jørgensen, Vilhelm Rosenqvist, Henning Høgh Olesen og Kirsten Mossin.

Plottersystem RC 4000.

1. Funktion

Systemet anvendes paa følgende maade:

- I) Kørsel af brugerprogrammet med initialisering af 8 standardvariable efterfulgt af kald af diverse plotterprocedurer, der bevirker, at der pakkes og lagres noget delvist bearbejdet plotterinformation paa et af brugeren opgivet dokument.

- II) Kørsel af et af to mulige selvstændige programmer med input fra det førømtalte dokument. Enten
 - IIa) program calcomp, der bevirker udtegning paa en Calcompplotter eller

 - IIb) program kingmatic, der bevirker hulning af en strimmel med tegneinformation. En strimmel, der kan læses direkte af en RC 1600 og udtegnes paa en Kingmaticplotter.

De 8 standardvariable, der alle er af type integer, er følgende:

xupper, xlower, yupper, ylower,
xzero, yzero,
xunit, yunit.

Deres funktion er beskrevet i afsnit 1.1.

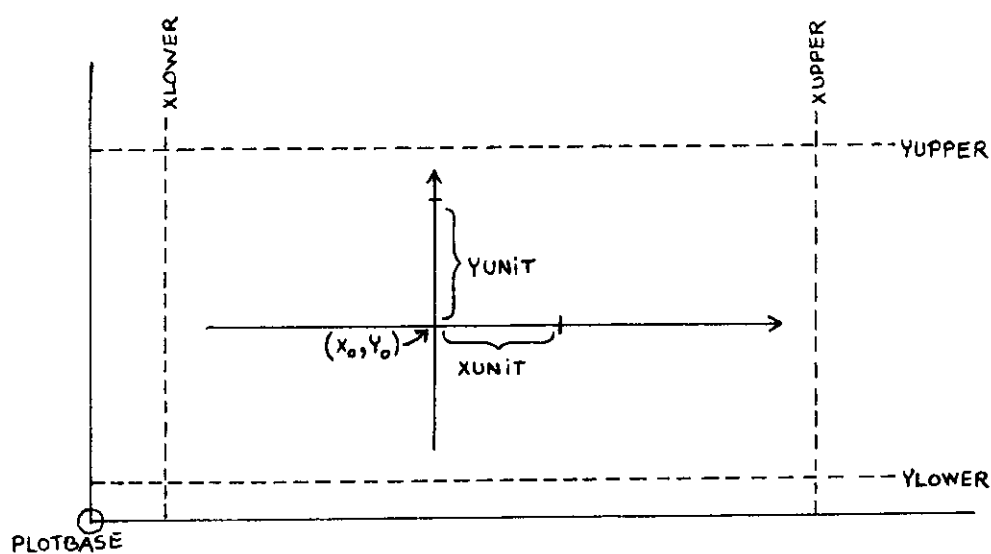
Plotterprocedurerne, der er external procedures, omfatter:

plot	afsn.	1.2.1.
textbegin	-	1.2.2.
textend	-	1.2.2.
plotstop	-	1.2.3.
plotend	-	1.2.4.

1.1. Standardvariable

De 8 standardvariable, der skal initialiseres af brugeren inden plotter-procedureerne benyttes, definerer tegnearealet, nulpunktet for koordinatsystemet samt enhederne paa koordinatakserne. Koordinatsystemet forudsættes at være et normalt retvinklet, to-dimensionalt højre-koordinatsystem.

Plotbase er pennens placering ved start af udtegningen. Det vil for tromleversionen af calcompplotteren sige 5 mm fra papirets højre kant, mens det for de øvrige plottere vil sige i nederste venstre hjørne. For tromleversionen peger x-aksens positive retning mod papirforsyningsrullen, mens den paa de øvrige plottere peger mod højre vinkelret paa tegnebroen. (Se ref. 1.).



liste

xlower	} angiver tegnearealets grænser maalt i 1/40 mm fra plotbase.
xupper	
ylower	
yupper	
xzero	} koordinaterne til nulpunktet for koordinatsystem maalt i 1/40 mm fra plotbase.
yzero	
xunit	} enhederne paa henholdsvis x- og yaksen maalt i 1/40 mm.
yunit	

Standardvariable, som brugeren ikke initialiserer, har værdien 0.

Plotteret systemet undersøger ikke, om de 8 standardvariable er tilknyttet rimelige værdier, inden de benyttes.

Den største tegning, systemet kan behandle, er 1.63 m × 1.63 m. Det svarer til at xupper og yupper skal være mindre end $2^{16} = 65536$.

Er tegnearealet defineret større end det kan tillades paa den plotter, der senere ønskes anvendt, opdages dette ikke, før man under selve udtegningen forsøger at tegne udenfor papirets grænser. Saa stoppes udtegningen med en fejludskrift. Det eneste, der tages højde for, er, at en bruger ikke ødelægger den foregaaende brugers tegning paa Calcompplotteren.

1.2. Plotterprocedurerne

Kald af plotterprocedurerne kræver for alle procedurernes vedkommende som første parameter en zone z. z skal erklæres med kun en share og en bloklængde, der er et multiplum af 128.

Zonen skal være aabnet til et dokument, der enten kan være et baggrundsareal, en fil paa magnetbaand eller eventuelt en strimmel, der saa skal punches i no parity mode.

Alle procedurerne pakker og lagrer delvist bearbejdet tegneinformation paa det paagældende dokument.

1.2.1. Plot

Plot lagrer information om tegning af en vektor fra pennens aktuelle position til punktet x,y med en opgiven stregtype.

Kald:

plot(z,x,y,line);

plot (returnvalue,boolean) true, hvis (x,y) ligger indenfor tegnearealet ellers false.

z (call- and returnvalue,zone) specificerer dokumentet.

x (callvalue,real) x-koordinaten til vektorens endepunkt.

y (callvalue,real) y-koordinaten - - -

line (callvalue,integer) vektorens stregtype

calcompplotter:	line=0	ingen streg (m.løftet pen).
	line=1	fuldt optrukken streg
kingmaticplotter:	line=0	ingen streg
	line=1	fuldt optrukken streg
	line=2,3,4,5,6,7	forskellige former for stipling og punktering. (Se side 38).

Hvis $line < 0$ eller $line > 1$ for Calcompplotteren sættes $line=1$.
Hvis $line < 0$ eller $line > 7$ for Kingmaticplotteren sættes $line=4$,
da denne linietype anses for at være den mindst anvendte og derfor den mest iøjnefaldende ved plotningen.

x- og y-koordinaten skal opgives i det koordinatsystem, som er fastlagt ved initialiseringen af $xzero, yzero, xunit$ og $yunit$. Den ønskede vektor vil kun blive tegnet ud i plotterprogrammet, hvis pennen staar indenfor tegnearealet samtidig med, at vektorens endepunkt ligger indenfor.

Et punkt ligger inden for tegnearealet, hvis

$$xlower \leq x \text{ and } x \leq xupper \text{ and } ylower \leq y \text{ and } y \leq yupper.$$

I modsat fald vil der komme en af følgende reaktioner ved udtegningen:

pennen indenfor-(x,y) udenfor: Den del af vektoren, der ligger indenfor tegnearealet, tegnes. Samtidig kommer der paa current output en fejlmeddelelse indeholdende koordinaterne (x,y).

pennen udenfor-(x,y) indenfor: Den del af vektoren, der ligger indenfor tegnearealet, tegnes.

pennen udenfor-(x,y) udenfor, men vektoren skærer ind over tegnearealet: Den del af vektoren, der ligger indenfor tegnearealet, tegnes. Samtidig kommer der paa current output en fejlmeddelelse indeholdende koordinaterne (x,y).

pennen udenfor-(x,y) udenfor, og vektoren skærer ikke tegnearealet: Slet ingen reaktion.

1.2.2. Textbegin og textend

Textbegin bevirker, at al output fra efterfølgende kald af write til zonen z lagres paa det til zonen knyttede dokument til senere udtegnning af de paagældende symboler med et i textbegin fastlagt format. Ved kald af textend ophører denne virkning.

Kald:

```
textbegin (z, format);
```

```
write(z, ...);
```

```
textend (z);
```

z (call- and returnvalue, zone) specificerer dokumentet.

format (callvalue, integer) definerer tekststørrelse og retning. Textstørrelsen fremkommer ved at multiplicere grundsymbolerne med en skalafaktor (se tabellen nedenfor). Der er mulighed for 16 forskellige formater. Paa Kingmaticplotteren er der endvidere mulighed for at tegne kursiv skrift, dvs. at alle symboler underkastes en hældning paa 14 grader. Dette opnaas ved at forøge format med 16.

Hvis man ønsker en anden textretning end den normale pos. x, skal man forsøge format med en retningsfaktor:

pos. y-retn.: format:= format + 32

neg. x-retn. ('paa hovedet'): format:= format + 64

neg. y-retn.: format:= format + 96

Tabel over textskalafaktor:

format	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
faktor	1	2	3	4	6	8	12	16	24	32	48	64	96	128	192	384

(Se iøvrigt ref. 2, bilag 7 side 2, idet det bemærkes, at der i forhold til dette system er foretaget en ombytning af faktorerne for neg. x- og neg. y-tegneretning.)

Mønsterlagre:

Til saavel Calcomp- som Kingmaticplotteren hører et standardmønsterlager, hvis grundsymboler har følgende størrelse:

Liniehøjde:	0.6 mm
Store bogstaver:	0.4 mm
Smaa bogstaver:	0.3 mm
Typisk bogstavbredde:	0.5 mm

(Se Appendix B samt ref. 2, bilag 6)

Til Calcompplotteren hører desuden et mønsterlager, som følger ISO-alfabetet (se Algol 5, user's manual, 2.0.1). I dette mønsterlager er grundsymbolerne 4 gange større end i standardmønsterlageret:

Liniehøjde:	2.4 mm
Store bogstaver:	1.6 mm
Smaa bogstaver:	1.2 mm
Typisk bogstavbredde:	2.0 mm

(se Appendix B)

Der er til Calcompplotteren mulighed for, at en bruger kan definere og programmere sit eget mønsterlager. (Se ref. 4).

Udtegningen af symboler starter fra pennens aktuelle position. Textend bevirker, at pennen føres tilbage til denne position. Det er tilladt at kalde textbegin flere gange uden kald af textend ind imellem. Dette kan benyttes til at ændre textstørrelsen midt i udtegningen af en række symboler. Ændring af textretning har absolut ingen virkning her.

Standardproceduren write kan med hensyn til parametre benyttes i sin fulde udstrækning. Kald af write til zonen z uden paa forhaand at have kaldt textbegin vil medføre en af følgende to reaktioner:

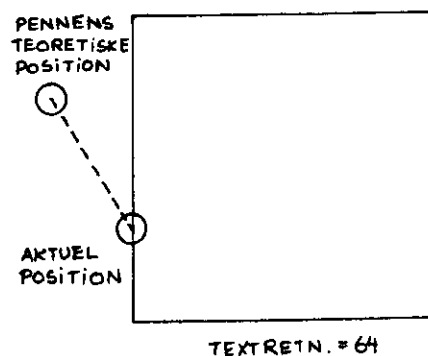
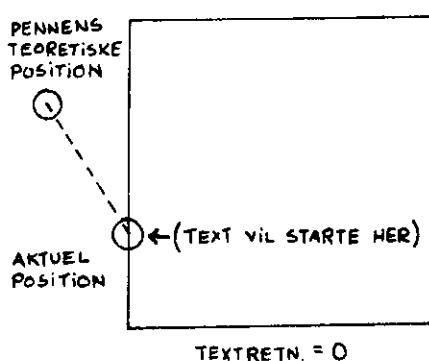
Kald af write efter kald af plot, plotstop eller plotend bevirker, at brugerprogrammet stoppes med en fejludskrift.

Kald af write før kald af nogen af de 5 plotterprocedurer bevirker, at parameterlisten til write skippes under plotningen.

Kald af plot imellem textbegin og textend bevirker, at brugerprogrammet stoppes med en fejludskrift.

Paa Calcompplotteren tegnes kun de symboler, hvis startpunkt ligger indenfor tegnearealet. Symboler, der starter udenfor, kommer sammen med en fejludskrift paa current output under selve udtegningen. Paa Kingmaticplotteren udtegnes hele symbolrækken, selvom dele af denne falder udenfor tegnearealet.

Hvis parametrene til nærmest foregaaende kald af plot har haft saadanne værdier, at det til parametrene svarende punkt - kaldet pennens teoretiske position - ligger udenfor tegnearealet, er pennen kun blevet flyttet fra startpunktet for vektoren til endepunktet for den del af vektoren, som falder indenfor tegnearealet. Dette punkt kaldes pennens aktuelle position. Falder pennens teoretiske position ikke sammen med dens aktuelle position ved start af symboltegning, starter symboltegningen i pennens aktuelle position, og dette vil, afhængig af tekstretning og plotter, bevirke, at man enten faar den paagældende tekst skubbet ind i tegnearealet eller faar hele teksten ud som fejludskrift, og dermed ikke faar noget ud paa plotteren. (se fig. 2).



CALCOMP: AL TEXT SOM FEJLUDSKRIFT.
KINGMATIC: TEXTEN PLOTTES, MEN UDENFOR TEGNEAREALET.

FIGUR 2

1.2.3. Plotstop

Plotstop bevirker, at pennen føres løftet tilbage til plotbase (xbase, ybase). Paa Kingmaticplotteren bevirker plotstop endvidere, at denne standses midlertidigt, saaledes at det er muligt at skifte pen, hvorefter plotteren kan startes manuelt.

Kald:

```
plotstop(z);  
z          (call- and returnvalue, zone) specificerer dokumentet.
```

1.2.4. Plotend

Plotend bevirker, at udtegningen stopper. Paa Calcompplotteren føres pennen først løftet frem til startpunktet for næste tegning (dvs. til punktet xupper + 10 cm, ybase). Efter diverse kald af de øvrige plotterprocedurer, men inden z closes, skal der forekomme et kald af plotend.

Kald:

```
plotend(z);  
z          (call- and returnvalue, zone) specificerer dokumentet.
```

2. Kørsel med Plottersystemet.

Plottersystemet foreligger paa 4 strimler,

Str. 1 Plotterprocedure RC 4000: indeholder de 5 external procedurer, samt en external procedure initiate, der, sammen med nogle FP-kommandoer, sørger for oprettelsen af de nødvendige standardvariable. Strimlen indlæses med FP-kommandoen i trf.

Str. 2 calcomp: algolprogram, der styrer Calcompplotteren. Forrest paa strimlen ligger de nødvendige FP-kommandoer, der gør det muligt at indlæse denne ved hjælp af i trf.

Str. 3 symbolstore: Binær udskrift af mønsterlageret til calcomp. I øjeblikket foreligger der to versioner af mønsterlageret, som man kan vælge imellem (se app. B). Strimlerne er mærket henholdsvis symbolstore 1 og symbolstore 2 og indlæses v.h.j.a. FP-kommandoen binin tro.

Str. 4 kingmatic: algolprogram, der styrer Kingmaticplotteren. Strimlen er opbygget som Str. 2 og indlæses som denne v.h.j.a. FP-kommandoen i trf.

Alt efter hvilken plotter, der skal benyttes indlæses str.1 + str.2 + str.3 eller str.1 + str.4.

Anvendes et baggrundsareal til den midlertidigt bearbejdede plotterinformation, kan det af hensyn til oprettelsen af baggrundsarealet oplyses, at der stilles følgende pladskrav: Hvert kald af textbegin, plotstop og plotend fylder et dobbeltord paa baggrundsarealet.

Kald af textend fylder ingenting.

Tekststrengene fra kald af write til zonen, der er tilknyttet baggrundsarealet lagres paa normal vis med 6 tegn pr. dobbeltord.

Hvert kald af plot vil normalt fylde et dobbeltord paa baggrundsarealet (ligger vektorens endepunkt udenfor tegnearealet vil informationen fylde 3 el. 4 dobbeltord).

Det kan anbefales, at man danner sig et løseligt skøn over hvor meget plotterinformation, der vil blive lagret og dermed hvor stort et baggrundsareal, der skal oprettes.

Et simpelt algolprogram, der beregner og derefter udtegner nogle resultater kan f.eks. se saaledes ud:

```
begin zone z(128,1,stderror);  
  ---  
  xlower:= ylower:= 0;  
  xupper:= ---;  
  ---  
  yzero:= ---;  
  open(z,4,<:areaname:>,0);
```

```
beregninger

plot(z,---);
---
plot(z,---);
textbegin(z,4);
write(z,<:plottext:>);
---
textend(z);
plotend(z);
close(z,true)
end
```

Afviklingen af kørslen med dette program paa calcompplotteren kan ske med følgende FP-kommandoer:

```
i trf          ; str.1 indlæses
i trf          ; - 2 -
binin tro      ; - 3 -
areaname=set 3 ; baggrundsareal til midlertidigt bearbejdet plotteroutput
r=algol trf    ; indlæsning og oversættelse af brugerprogrammet
r              ; kørsel af brugerprogrammet
calcomp areaname ; kørsel af program calcomp med input fra areaname
```

I Appendix A er vist et fuldstændigt programeksempel.

3. Fejludskrifter

Ved anvendelsen af plottersystemet kan der optræde to typer af fejl. Den ene type skyldes fejlagtig anvendelse af plotterprocedurerne og bevirker at brugerprogrammet stoppes med en fejludskrift. Den anden skyldes uacceptable oplysninger til plotterproceduren og bevirker at der under kørslen med program calcomp eller kingmatic kommer fejludskrifter. Begge typer fejludskrifter kommer paa current output.

Fejludskrifter under kørsel med brugerprogrammet:

plotcall 3
called from <alarm address>
called from

plot er kaldt imellem textbegin og textend. Kørslen stoppes

z.state 6
called from <alarm address>
called from

Denne standardfejl udskrift kan skyldes at write er kaldt før textbegin, men efter plot, plotstop el. plotend. Kørslen stoppes.

z.state 3
called from <alarm address>
called from

Kan skyldes at plotstop eller plotend er kaldt imellem textbegin og textend. Kørslen stoppes.

Fejludskrift under kørsel med program calcomp el. program kingmatic:

Outside Limits of Drawing
<x>, <y>

plot er kaldt med parametrene x og y, men x, y ligger udenfor tegnearealet. Den del af vektoren, der falder indenfor, bliver tegnet.

no input-source specified
notes not allowed as input-source
<source> illegal type mode.kind = <mode>.<kind>

} Alle tre udskrifter indikerer at input-source ikke er legalt specificeret. Kørslen stoppes.

Yderligere fejludskrifter under kørsel med program calcomp:

Text Outside Limits
<streng>

Den del af symbolstrengen, der er udenfor tegnearealet, skrives ud sammen med fejludskriften. <NL> i symbolstrengen markeres i fejludskriften ved en skraastreg.

Pen Outside Paper

Tegnearealets grænser overstiger Calcompplotterens tegneareal. Udtegningen stoppes.

Yderligere fejludskrift under kørsel med program kingmatic:

Unidentified Plotinformation Skipped Illegal plotterinformation læst og ignoreret. Programmet forsøger at fortsætte.

4. Diskussion

Metoden med at lade selve udtegningen foregaa i et selvstændigt program, er valgt af flere aarsager.

- 1) I brugerens program laaser man sig ikke fast paa hvilken plotter, der skal benyttes.
- 2) Brugeren behøver ikke at have adgang til en plotter i det øjeblik, hvor hans program kører, og han har mulighed for at checke sit øvrige output for fejl, inden udtegningen startes.
- 3) Det er ogsaa en fordel, at man kan genstarte selve udtegningen, hvis f.eks. pennen svigter, uden at køre et maaske tidskrævende brugerprogram endnu engang.
- 4) Endelig skal det nævnes, at systemet let kan udvides med plotterprogram til nye typer af plottere. Systemet kan ligeledes udvides med specielle plotterfunktioner i form af nye external procedures.

Det har været overvejet at lave en version af plottersystemet, der bevirker, at udtegningen sker efterhaanden som plotterinformationen dannes af det kørende brugerprogram. Det er en forholdsvis simpel sag at ændre det bestaaende system, saaledes at dette kan lade sig gøre, men det skal understreges, at det eneste, der taler for en saadan version, er, at brugeren saa kan standse sit eget program, hvis tegningen paa plotteren ser forkert ud.

5. Rerefenceliste

- [1] Sandvang, Tom: RC 4000 Peripheral Devices, RC 4191 Plotter Controller Reference Manual, RCSL NO 55-D56, Nov. 1969.
- [2] Suwalski, Henning: A/S Regnecentralens numerisk styrede tegneanlæg, Nov. 1968.
- [3] RC 4000 Computer Specifications. RC 4191 incremental plotter controller, RCSL 82-H0015, RC 1600 graphic converter, RCSL 82-H0017.
- [4] Olesen, Henning Høgh: Symbolstore Assembler. RCSL NO 55-D146, ca. juni 1971.

Appendix A: Programeksemppler

Eks. paa program, der anvender plottersystemet paa en calcompplotter samt tegningen der kommer ud af kørslen med dette. (Str. 1+2+3 af plottersystemet forudsættes indlæst.)

```
(coor = set 2
s = algol trf
s
calcomp coor
end)
```

```
begin
  integer i;
  zone z(128,1,stderr);

  xlower:= ylower:=0;
  xupper:=10000;
  yupper:=7200;
  xunit:= yunit:= 400;
  xzero:= 5000;
  yzero:= 3600;

  open(z,4,<:coor:>,0);
  plot(z,-12.5,9,1);
  plot(z,12.5,9,1);
  plot(z,12.5,-9,1);
  plot(z,-12.5,-9,1);
  plot(z,-12,0,0);
  plot(z,12,0,1);
  plot(z,11.75,0.25,1);
  plot(z,12,0,0);
  plot(z,11.75,-0.25,1);
  plot(z,10.8,0.25,0);
  textbegin(z,0);
  write(z,<:x-akse:>);
  textend(z);

  for i:= 11 step -1 until 1,-1 step -1 until -11 do
    begin
      plot(z,i,0.1,0);
      plot(z,i,-0.1,1);
      plot(z,i-0.4,-0.3,0);
      textbegin(z,0);
      write(z,<<-dd>,i);
      textend(z)
    end;
```



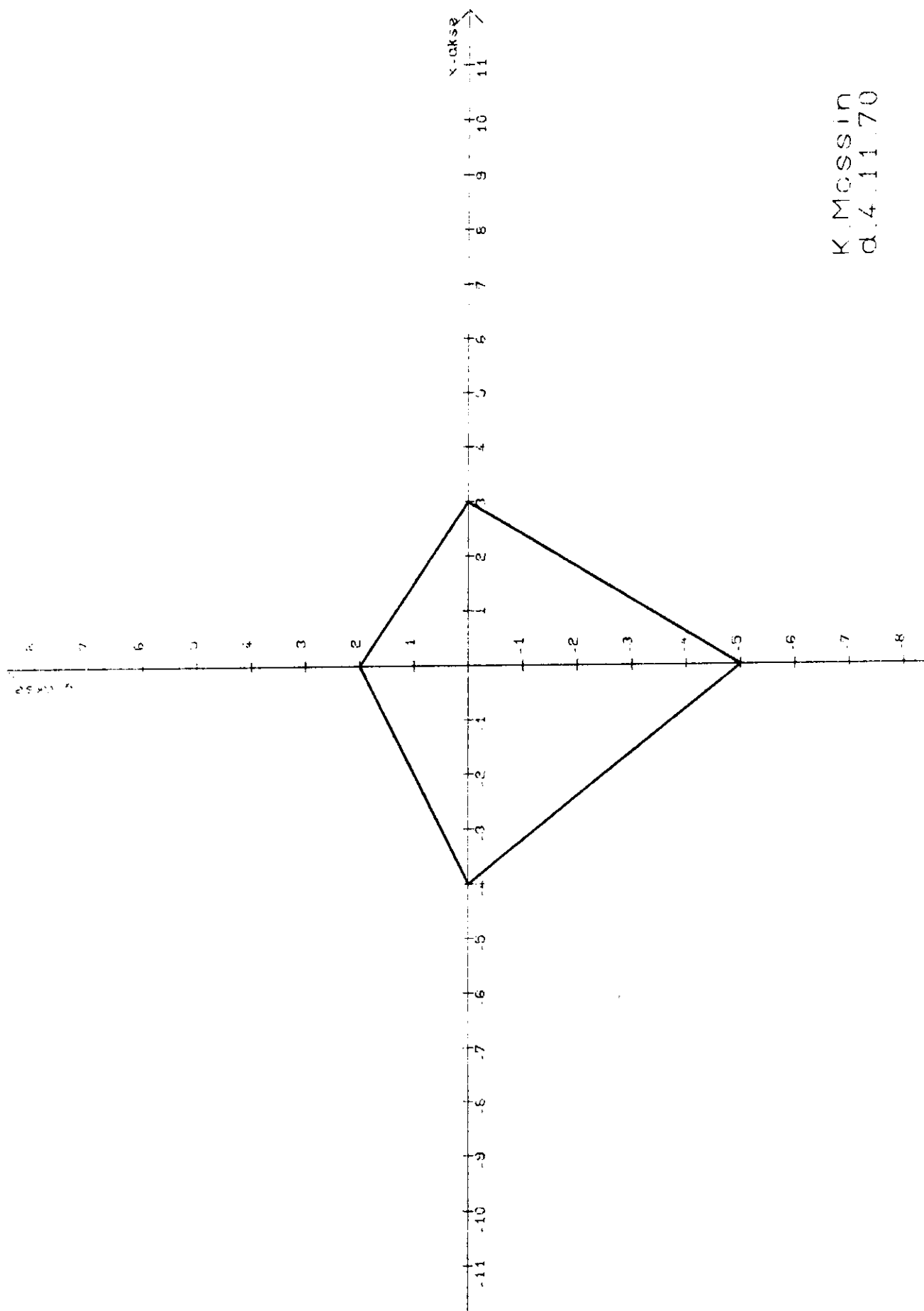
```
plot(z,0,-8.5,0);
plot(z,0,8.5,1);
plot(z,0.25,8.25,1);
plot(z,0,8.5,0);
plot(z,-0.25,8.25,1);
```

```
plot(z,-0.3,7.3,0);
textbegin(z,32);
write(z,<y-akse:>);
textend(z);
```

```
for i:=8 step -1 until 1,-1 step -1 until -8 do
  begin
    plot(z,-0.1,i,0);
    plot(z,0.1,i,1);
    plot(z,0.2,i,0);
    textbegin(z,0);
    write(z,<<-d>,i);
    textend(z)
  end;
```

```
plot(z,0,-5,0);
for i:=1 step 1 until 10 do
  begin
    plot(z,-4,0,1);
    plot(z,0,2,1);
    plot(z,3,0,1);
    plot(z,0,-5,1);
  end;
plot(z,7.5,-7,0);
textbegin(z,1);
write(z,<:K.Mossin<10>d.4.11.70:>);
textend(z);
plotend(z);
close(z,true)
end;
```

Resultatet fra kørsel med program calcomp og symbolstore 2.



K. Messin
d. 4. 11. 70

Eks. paa program, der anvender plottersystemet paa en kingmatic plotter samt tegningen (en planche over alle tegn og stregtyper) der kommer ud af kørslen med dette (Str. 1+4 af plottersystemet forudsættes indlæst).

```
(pejplot = set 5
```

```
s = algol trf
```

```
s
```

```
kingmatic pejplot
```

```
end)
```

```
begin
```

```
comment
```

```
PLANCHE OVER STREGER OG TEGN (KINGMATIC) ;
```

```
integer i,form ; real j ;  
zone tegn(128,1,stderr);  
open(tegn,4,<:pejplot:>,0);
```

```
xupper:=yupper:=20000 ;  
xlower:=ylower:= 800 ;  
xzero :=yzero := 1000 ;  
xunit :=yunit := 400 ;
```

```
plot(tegn,0,28,0);  
plot(tegn,18,28,1);  
plot(tegn,18,0,1);  
plot(tegn,0,0,1);  
plot(tegn,0,28,1);  
plot(tegn,0.5,27.5,0);  
plot(tegn,17.5,27.5,1);  
plot(tegn,17.5,23,1);  
plot(tegn,0.5,23,1);  
plot(tegn,0.5,27.5,1);
```

```
plot(tegn,1,25,0);  
textbegin(tegn,8);  
write(tegn,false add 13,1);  
textbegin(tegn,9);  
write(tegn,<: 4000:>);  
textend(tegn);  
plot(tegn,1,23.5,0);  
textbegin(tegn,5);  
write(tegn,<:PLOTING SYSTEM  
textend(tegn);
```

```
KINGMATIC VERSION:>);
```

```
plot(tegn,0.5,22,0);
textbegin(tegn,5);
write(tegn,<:LINE TYPES :>);
textend(tegn);
j:=22.5;
for i := 0 step 1 until 7 do
begin
  j:=j-0.5;
  plot(tegn,5,j,0);
  textbegin(tegn,4);
  form:=i+48;
  write(tegn,false add form,1);
  textend(tegn);
  plot(tegn,6,j+0.25,0);
  plot(tegn,15,j+0.25,i);
end;
```

```
plot(tegn,0.5,16,0);
textbegin(tegn,5);
write(tegn,<:SYMBOLS :>);
textend(tegn);
plot(tegn,5,16,0);
for form:=4,20 do
begin
  textbegin(tegn,form);
  for i := 65 step 1 until 93 , 97 step 1 until 125, 48 step 1 until 57,
    40 step 1 until 47, 58 step 1 until 62 do
    if i = 93 or i = 125 or i = 57 or i = 62 then write(tegn,false add i,1,<:<10><10>:>)
      else write(tegn,false add i,1);
  for i := 2,3,6,7,8,9,13 step 1 until 22 ,
    26,28,29,30 do write(tegn,false add i,1,<: :>);
    write(tegn,false add 10,6);
end;
textend(tegn);
```

```
plot(tegn,5,1.5,0);
textbegin(tegn,4);
write(tegn,<:16 DIFFERENT TEXTSIZES:>,
<:<10>4 POSSIBLE DIRECTIONS:>);
textend(tegn);
plotend(tegn);
close(tegn,true);
end
```

IC 4000

PLOTTING SYSTEM

KINGMATIC VERSION

LINE TYPES : 0

- 1 _____
- 2 (dotted)
- 3 - - - - - (long dashed)
- 4 - - - - - (short dashed)
- 5 - - - - - (shorter dashed)
- 6 - - - - - (shorter dashed)
- 7 - - - - - (shorter dashed)

SYMBOLS:

ABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅ

abcdefghijklmnopqrstuvwxyzæøå

0123456789

() * + , - . / ; < = >



+ x [O . |  ^ v  A < V] . - o , o

ABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅ

abcdefghijklmnopqrstuvwxyzæøå

0123456789

() * + , - . / ; < = >

+ x [O . |  ^ v  A < V] . - o , o

16 DIFFERENT TEXTSIZES
4 POSSIBLE DIRECTIONS

Appendix B.

Mønsterlager nr. 1 til Calcompplotteren.

De i mønsterlageret beskrevne symboler er vist paa næste side sammen med de tilsvarende ISO-værdier. (Selve symbolerne er tegnet med format 6, resten med format 5).

Grundsymbolernes størrelse:

Liniehøjde:	0.6 mm
Store bogstaver:	0.4 mm
Smaa bogstaver:	0.3 mm
Typisk bogstavbredde:	0.5 mm

Afvielser fra den typiske bogstavbredde:

0.4 mm:	SP ! " ' () , . : ;
	I _ f i j k l r t v
0.6 mm:	M T V W Y Æ m w ø

Specielle symboler:

<12> (FF) har samme virkning som <10> (NL).
<126> har samme virkning som <10> (NL) bortset fra, at der tegnes en optrukken streg umiddelbart under den sidst tegnede linie, d.v.s. en understregning.

Bemærk:

Dette mønsterlager har stort set samme format som standardmønsterlageret til Kingmaticplotteren. Dette bevirker, at tekst, der tegnes, giver samme resultat paa de to plottere.

Appendix B.

Mønsterlager nr. 1 til Calcompplotteren.

0		32	SP	64		96	
1		33	!	65	A	97	a
2		34	"	66	B	98	b
3		35		67	C	99	c
4		36		68	D	100	d
5		37		69	E	101	e
6		38	,	70	F	102	f
7		39	'	71	G	103	g
8		40	(72	H	104	h
9		41)	73	I	105	i
10	NL	42	*	74	J	106	j
11		43	+	75	K	107	k
12	NL	44	,	76	L	108	l
13		45	-	77	M	109	m
14		46	.	78	N	110	n
15		47	/	79	O	111	o
16		48	0	80	P	112	p
17		49	1	81	Q	113	q
18		50	2	82	R	114	r
19		51	3	83	S	115	s
20		52	4	84	T	116	t
21		53	5	85	U	117	u
22		54	6	86	V	118	v
23		55	7	87	W	119	w
24		56	8	88	X	120	x
25		57	9	89	Y	121	y
26		58	:	90	Z	122	z
27		59	;	91	Æ	123	æ
28		60	<	92	Ø	124	ø
29		61	=	93	Å	125	å
30		62	>	94		126	NL
31		63		95	-	127	

Appendix B.

Mønsterlager nr. 2 til Calcompplotteren.

De i mønsterlageret beskrevne symboler er vist paa næste side sammen med de tilsvarende ISO-værdier. (Selve symbolerne er tegnet med format 2, resten med format 1).

Grundsymbolernes størrelse:

Liniehøjde:	2.4 mm
Store bogstaver:	1.6 mm
Smaa bogstaver:	1.2 mm
Typisk bogstavbredde:	2.0 mm

Afvigelser fra den typiske bogstavbredde:

1.2 mm:	, . : ;
1.6 mm	SP ! " ' () * + - =
	I _ f i j k l v t
2.4 mm:	M W Æ m w æ

Specielle symboler:

<12> (FF) har samme virkning som <10> (NL).
<126> har samme virkning som <10> (NL) bortset fra, at der tegnes en optrukken streg umiddelbart under den sidst tegnede linie, d.v.s. en understregning.

Appendix B.

Mønsterlager nr. 2 til Calcompplotteren.

0		32	SP	64	@	96	
1		33	!	65	A	97	a
2		34	"	66	B	98	b
3		35	£	67	C	99	c
4		36	\$	68	D	100	d
5		37	%	69	E	101	e
6		38	&	70	F	102	f
7		39	'	71	G	103	g
8		40	(72	H	104	h
9		41)	73	I	105	i
10	NL	42	*	74	J	106	j
11		43	+	75	K	107	k
12	NL	44	,	76	L	108	l
13		45	-	77	M	109	m
14		46	.	78	N	110	n
15		47	/	79	O	111	o
16		48	0	80	P	112	p
17		49	1	81	Q	113	q
18		50	2	82	R	114	r
19		51	3	83	S	115	s
20		52	4	84	T	116	t
21		53	5	85	U	117	u
22		54	6	86	V	118	v
23		55	7	87	W	119	w
24		56	8	88	X	120	x
25		57	9	89	Y	121	y
26		58	:	90	Z	122	z
27		59	;	91	Æ	123	æ
28		60	<	92	Ø	124	ø
29		61	=	93	Å	125	å
30		62	>	94	ˆ	126	NL
31		63	?	95	˘	127	

Appendix B.

Standardmønsterlageret til Kingmaticplotteren.

De i mønsterlageret beskrevne symboler er vist paa næste side sammen med de tilsvarende ISO-værdier. (Selve symbolerne er tegnet med format 6, resten med format 5).

Grundsymbolernes størrelse:

Liniehøjde:	0.6 mm
Store bogstaver:	0.4 mm
Smaa bogstaver:	0.3 mm
Typisk bogstavbredde:	0.5 mm

Specielle symboler:

<2>, <3>, <7>, <8>, <16>, <28>, <29> og <30> er markeringssymboler, der har udgangspunkt ca. i midten af symbolet.

<17>, <18>, <19> og <20> er pilespidser, der har udgangspunkt i selve spidsen.

<6> ([), <9> (|), <14> (^), <15> (v), <21> (l) og <25> () er symboler, der findes paa en flexowriter men ikke i ISO-alfabetet.

<13> (RC-mærket) har udgangspunkt i nederste, venstre hjørne.

Appendix B.

Standardmønsterlager til Kingmaticplotteren.

0		32	SP	64	96	
1		33		65	97	a
2	+	34		66	98	b
3	x	35		67	99	c
4		36		68	100	d
5		37		69	101	e
6	[38		70	102	f
7	○	39		71	103	g
8	┌	40	(72	104	h
9		41)	73	105	i
10	NL	42	x	74	106	j
11		43	+	75	107	k
12		44	,	76	108	l
13	└	45	-	77	109	m
14	∧	46	.	78	110	n
15	∨	47	/	79	111	o
16	⊙	48	0	80	112	p
17	∇	49	1	81	113	q
18		50	2	82	114	r
19	↖	51	3	83	115	s
20	↗	52	4	84	116	t
21]	53	5	85	117	u
22		54	6	86	118	v
23		55	7	87	119	w
24		56	8	88	120	x
25		57	9	89	121	y
26		58	:	90	122	z
27		59	;	91	123	ø
28	□	60	<	92	124	ø
29	°	61	=	93	125	å
30	○	62	>	94	126	
31		63		95	127	

Appendix B


Udskrift af de mulige formater i mønsterlager 1 samt i standardmønsterlageret til Kingmaticplotteren.


FORMAT 0 : -

FORMAT 1 : 

FORMAT 2 : 


FORMAT 3 : 


FORMAT 4 : 

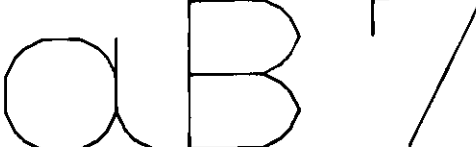
FORMAT 5 : 

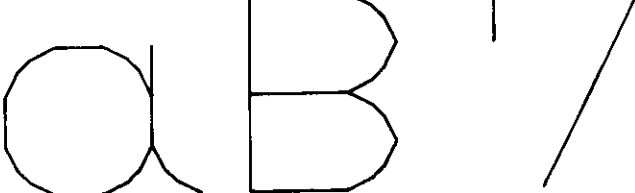
FORMAT 6 : 

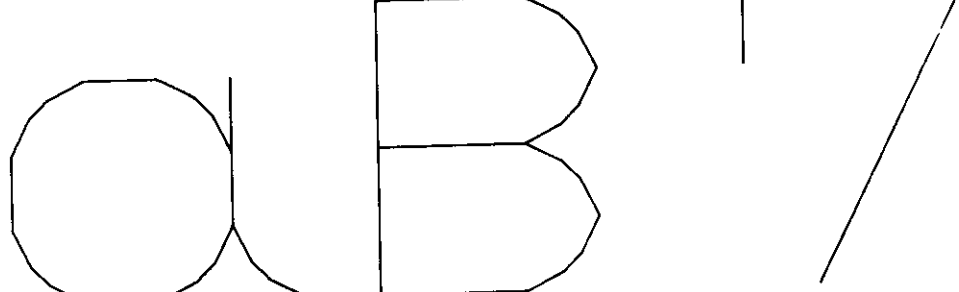
FORMAT 7 : 

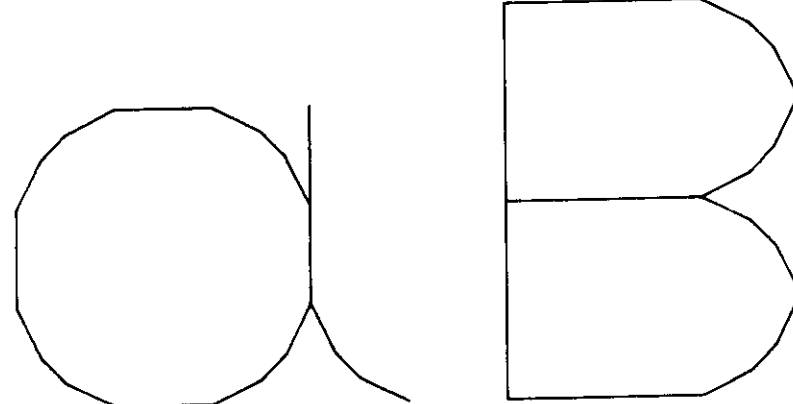
FORMAT 8 : 

FORMAT 9 : 

FORMAT 10 : 

FORMAT 11 : 

FORMAT 12 : 

FORMAT 13 : 

APPENDIX C: Kørsel med Magnetbaand

Paa RC 4000 installationer med magnetbaandsstationer kan man med fordel anvende et magnetbaand til lagring af midlertidigt bearbejdet plotterinformation fra mange brugerprogrammer, idet man saa paa passende tidspunkter kan afvikle kørslen af program calcomp eller kingmatic.

Til dette brug er udviklet :

- 1) external boolean procedure getfile.
- 2) external procedure eoftest som er en blockprocedure.
- 3) drive strimmel til styring af program calcomp eller program kingmatic.
drive er en samling fp-commands.

eoftest og getfile er to procedurer som brugeren skal anvende i sit program, hvis hans tegneoutput skal gemmes paa et magnetbaand evt. sammen med andre brugeres tegninger. De to procedurer sikrer at brugerens tegning anbringes i en fri fil paa det magnetbaand, han har aabnet sin outputzone til, og at næste brugers tegning kommer i den efterfølgende fil, saaledes at ens egen tegning ikke ødelægges.

Naar en række tegninger ved hjælp af de to ovennævnte procedurer samt plotterprocedurerne er placeret efter hinanden paa et magnetbaand, tages drive strimmelen og enten program kingmatic eller program calcomp i brug for den endelige bearbejdning.

drive strimmelen er en samling fp-commands, som gør det muligt at køre enten program calcomp eller program kingmatic flere gange efter hinanden uden operatørrindgreb, og paa en saadan maade at programmet for hver kørsel tager input fra en ny fil paa magnetbaandet. Ved at sætte styreinformation ind i en note og i en katalogindgang er det muligt at bestemme hvilke filer (i et interval) der skal udtegnes (se brugerexemplet).

Brugeren skal skrive flg. i sit program (idet han dog maa bruge sine egne navne paa zonen og baandet) :

```
begin
.
.
.
zone tegning(128,1,eoftest) ;
.
.
open(tegning,18,<:plottape:>,1 shift 16) ;
if -, getfile(tegning) then goto FEJL ;
comment setposition skal ikke kaldes idet dette varetages af
getfile ;
.
```

```
close(tegning,true) ;  
comment ved close skrives eof paa baandet og blockproceduren  
eoftest kaldes da zonen tegning blev aabnet med give-  
up mask 1 shift 16. eoftest sørger nu for at næste  
program ved kald af getfile faar tildelt næste fil  
til output ;
```

```
.  
.  
end
```

getfile og eoftest holder øje med antallet af filer ved at benytte en catalogindgang kaldet plotfile til at skrive det aktuelle fil-nummer i. Ved begyndelsen af en magnetbaandskørsel maa brugerne derfor sætte denne catalogindgang, saa procedureerne kan finde den. Catalogindgangen skal blot beskrive den fil som brugerne vil lægge den første tegning i (ved et nyt magnetbaand fil nr. 0), og de efterfølgende tegninger vil da automatisk komme i de efterfølgende filer (se brugereksemplet) .

Bemærk 2 ting:

- a) getfile, der skal have outputzonen som parameter, er af boolean type. Den returnerer med værdien true hvis alt er OK , eller med false hvis catalogindgangen plotfile ikke var oprettet.
- b) eoftest er en blockprocedure der bliver kaldt naar et eof skrives paa outputbaandet. Dette sker naar outputzonen closes (eller ved setposition i utide) .Hvis man ikke closer eller hvis programmet afbrydes paa grund af index-fejl eller lignende vil næste brugers tegning overskrive den foregaaende. Dette er tilsigtet da den foregaaende tegning alligevel ikke kan udtegnes fornuftigt af program calcomp eller program kingmatic, da tegningen ikke er korrekt afsluttet.

Følgende navne er standard paa de udleverede procedurer mm.

```
plotfile , der er navnet paa en catalogindgang.  
slut      , - - - - -  
s         , - - - - - note.  
drive    , - - - - - baggrundslageromraade  
desuden skal zonens størrelse være som angivet, og  
giveup-mask i openkaldet skal være som angivet.
```

BRUGEREKSEMPEL:

Naar et antal brugere afleverer deres programmer til kørsel (eller kører dem selv) gøres som følger.

```
att s  
call 8 plottape      ;(hvis det er baandstation 8 som anvendes)  
plotfile=set 0 0 0 0 ;(parameter nr. 4 angiver at første frie  
; fil er fil nr. 0 )
```

Herefter oversættes og køres programmerne som normalt. Naar alle programmer er kørt, gemmes den værdi som nu staar i plotfile, ved at plotfile permanentes (eller ved at lave lookup, og huske værdien for senere genoprettelse).

genstart:

Ønsker man senere at tegne videre paa samme baand, beskriver plotfile, hvis den blev permanentet, den første frie fil paa magnetbaandet. (huskede man værdien kan plotfile sættes til den gamle værdi). Man monterer da blot baandet og kører som før.

Naar et vist antal tegninger har ophobet sig paa magnetbaandet ønsker man at faa disse tegnet ud paa plotteren. Da gøres som følger:

```
lookup plotfile          ; (og lookup svarer f.ex. )
plotfile=set 0 0 0 7 0 0 0
```

Der ligger da 7 tegninger i fil 0-6 fordi fil 7 er den første frie fil paa plottape.

Videre fortsættes :

```
i trf                      ; drive strimmelen indlæses og
                           ; paa current output udskrives :
edit begin
lad s beskrive første fil
lad slut beskrive sidste fil
tast derefter fp-commandoen i drive
edit end
```

Den givne instruktion udføres, idet man ogsaa husker at navngive baandstationen (her : plottape) :

```
att s
call 8 plottape
s = set mto plottape 0 0 ; (første tegning i fil 0)
slut = set 0 0 0 6      ; (kun filnummeret er her af betydning)
i drive
```

Bemærk at filnr. i catalogindgangen slut er 6, dvs. den sidste fil hvori der ligger en tegning. Plotteren vil nu køre automatisk under styring af calcomp programmet (eller for kingmatic programmet : punch) og fp vender først tilbage til konsollen naar alle filer er bearbejdet. Ønsker man at standse inden dette tidspunkt, kan processen breakes ved hjælp af operativsystemet s. Da drive strimmelen før hver tegning udskriver hvilken fil den paagældende tegning ligger i er det senere let at foretage genstart, hvor første tegning nu er den der før blev afbrudt.

Foruden noten s, catalogindgangene plotfile og slut, anvendes af forskellige grunde ogsaa en catalogindgang kaldet start, under plotningen. Denne beskriver imidlertid altid det samme som s, og man behøver ikke at bekymre sig om den, bortset fra, at den kan ligge i cataloget fra en tidligere kørsel under et andet catalognummer, end det man selv kører under, hvorfor monitoren protesterer mod anvendelse af denne. Dette problem er generelt og kan ogsaa forekomme ved de andre catalogindgange, hvis disse permanentes for senere brug (f.ex. plotfile).

Eks. paa program, der anvender plottersystemet i forbindelse med de til magnetbaandskørsel udarbejdede strimler. Programmet, som tegner alle formater, kan benyttes i forbindelse med Kingmaticplotteren (str. 1+4+eofstest+getfile forudsættes indlæst) eller i forbindelse med calcompplotteren med mønsterlager nr. 1 (str. 1+2+3+eofstest+getfile forudsættes indlæst). Output findes i App. B.

Kald af hhv. program calcomp eller program kingmatic varetages af str. drive calcomp eller str. drive kingmatic. Det forudsættes at baandstationen i dette tilfælde er døbt plottape.

```
(plotfile = set 0 0 0 0
s = algol trf
s
i trf          ; drive str.
s = set mto plottape 0 0
slut = set 0 0 0 0
i drive
end)
```

ALLE FORMATER KINGMATIC ELLER CALCOMP

```
begin real y ; integer i ;
  zone form(128,1,eofstest);
  open(form,18,<:plottape:>,1 shift 16);
  if -, getfile(form) then goto FEJL;
  xupper:=yupper:=30000;
  xlower:=ylower:= 100;
  xzero :=yzero := 500;
  xunit :=yunit := 400;

  for i := 0 step 1 until 15 do
  begin
    y:=case i+1 of(60.0, 59.0, 58.0, 57.0, 56.0, 55.0,
                  54.0, 53.0, 51.5, 50.0, 47.5, 44.5,
                  40.0, 34.0, 25.5, 4.5);
    plot(form,0,y,0);
    textbegin(form,5);
    write(form,<:FORMAT :>,<<dd>,i,<: : :>);
    textbegin(form,i);
    write(form,<:aB7:>);
    textend(form)
  end;
  plotend(form);
  close(form,true);
FEJL:
end
```