

Appendix A

Acknowledgements

Many of the improvements and bug fixes have been prompted by my CLEO colleagues at Cornell. Dave Brown has tested the new versions that I sent him, fixed a number of bugs and implemented some new features. Jon Lewis has implemented new commands and is always a source of new ideas. He has also read through an older version of the manual several times and made a number of suggestions. Bijan Nemati used to support Mn_Fit at Cornell and was always ready to try out new versions and try to get them to work! Dan Bliss then took over that responsibility. The current Mn_Fit tsar for CLEO is Horst Severini. Roy Wang suggested the list of commands to help newcomers get started.

While Mn_Fit can probably be seen as competition for PAW, I have made use of a number of features built into PAW, particularly COMIS and the lego plotting code. I also use HIGZ for the graphics interface, to minimize the number of changes I have to make when I want to implement a new interface.

The TYPSCN package, originally written by D.G. Cassel and T.J. Killian, later updated by Chris Bebek, is straightforward to use and relieves the programmer of many of the usual chores associated with deciding which command has been given, or interpreting numbers.

Most of the code for calculating expressions and also the use of registers, parameters etc. came from Paul Avery's MULFIT package.

Jon Lewis has implemented the DI3000 interface using HIGZ/DI3000.

Many other Mn_Fit users have sent in their comments and bug reports. The list of good ideas for new features to implement never seems to get shorter!

Appendix B

Examples of Symbols, Fonts, Hatching and Keys

The complete set of Mn_Fit symbols is shown in Fig. B.1 and the standard Mn_Fit registers are given in Table B.1.

The same fonts are available in Mn_Fit as in PAW. For some examples see Fig. B.4. Note that the font and precision are specified with the form **spfff**, where **s** is the sign of the font, **p** is the precision and **fff** is the font, e.g. -1013 to get font -13 with precision 1. The fonts available depend on the version of the graphics package used. In Fig. B.4 HIGZ Postscript fonts are shown. In Fig. B.5 those for GKSGRAL and GKSGRAL Postscript are shown.

The complete character set for the device independent HIGZ font (font 0) is shown in Fig. B.2, while that for HIGZ Postscript font -13 is shown in Fig. B.3. Note that you only see the Zapfdingbats fonts when you look at or print the Postscript file. Use the **HARDCOPY** or **CAPTURE** command to make such a file.

HIGZ Postscript fonts can be obtained using precision 0,1, or 2. They are identical when you make a hardcopy. The differences are how they appear on the screen. If you use Postscript fonts and precision 2 you get the hardware default font on the screen. If you use precision 1 then the standard IGTEXT font (font 2000) is used on the screen and the Postscript font is used when printing. If you use precision 0, you get the Postscript fonts on the screen, but not superscripts, subscripts, Greek characters etc.

The default font everywhere is now (Version 4.06) -1004 i.e. IGTEXT on the screen and Helvetica Postscript font when printing.

Note that there is an offset between the HIGZ/Postscript font numbers and those in GKSGRAL. Times-Roman is -1 in GKSGRAL and -13 in HIGZ/Postscript, Times-Italic is -2 and -1 respectively.

Device independent hatchings are available in all HIGZ interfaces and these are shown in Fig. B.6. Each graphics package usually has its own hatchings and patterns (DECGKS) and these are often different for screen and hardcopy devices. Some examples for GKSGRAL are shown in Fig. B.7. Consult the relevant graphics package documentation for more details.

Hatch -3 is often used, because it gives a nice shaded overlay. However note that this is really a pattern (because you cannot see through it) and if you use it in an overlay it obscures the ticks. The solution is to do a **plot/noclear** on the 1st id again to get the scale redrawn. The solid fill patterns (100) and examples of hatch -3 are shown in Fig. B.8.

The **KEY** command can be used to provide a legend of the different symbols and shading used in a plot. Some examples are given in Fig. B.9. See Section 4.70 on page 121 for more details.

A picture showing the standard Mn_Fit sizes and the appropriate commands is shown in Fig. B.10.

A complete list of the Postscript characters that are available is shown in Fig. B.11. The first column shows the character code. The second shows what you get if you give the character code using the form `\nnn`. The other columns show what you get in Greek (`[\nnn]`), Special (`"\nnn#`) and Zapf Dingbats (`~\nnn#`) modes respectively. `nnn` indicates the character code given in the first column.

Histogram mode	Line drawing mode	2-D Histogram mode
Symbol 1	Symbol -1	1,2... Symbol -2
Symbol 2	Symbol -2	1...Z Symbol -1
Symbol 3	Symbol -3 Symbol 1
Symbol 4	Symbol -4	
Symbol 5	Symbol -5	
Symbol 6	Symbol -6	
Symbol 7	Symbol -7	
Symbol 8	Symbol -8	
Symbol 10		Symbol -10
Symbol 11		Symbol -11
Symbol 12		Symbol -12
Symbol 13		Symbol -13
Symbol 14		Symbol -14
Symbol 15		Symbol -15
Symbol 16		Symbol -16
Symbol 17		Symbol -17
Symbol 18		Symbol -18
Symbol 19		Symbol -19
1-dimensional plots:		
Symbol 1n Show no errors (as above)		
Symbol 2n Show x errors		
Symbol 3n Show y errors		
Symbol 4n Show x and y errors		
Symbol 6n x errors with line at end		
Symbol 7n y errors with line at end		
Symbol 8n x and y errors with lines at end		
2-dimensional plots:		
Symbol $\geq 1n$ Symbol area \sim Number of entries		

Figure B.1: Mn_Fit Symbols

101	The result of a SUM or INTEGRATE command
111	The χ^2 or likelihood from the fit
112	The confidence level of a fit
Registers 121 to 200 and 231 to 257 are filled if you give the command SET PLOT id [&idb] DEFAULT	
121	The plot identifier
122	The secondary plot identifier
123	The number of entries (histograms) or points
124	The dimension of the plot (positive for histograms, negative for Ntuples and a series of points).
125	The area under the plot (i.e. sum of weights)
126	The minimum number of entries (weight)
127	The maximum number of entries (weight)
128	The creation date of the histogram (yymmdd)
129	The creation time of the histogram (hhmm)
131	The number of bins on x-axis (0 for Ntuples and points)
132	The lower limit of the x-axis
133	The upper limit of the x-axis
134	The mean value for the x-axis
135	The sigma for the x-axis
136	The number of bins on y-axis (0 for Ntuples and points)
137	The lower limit of the y-axis
138	The upper limit of the y-axis
139	The mean value for the y-axis
140	The sigma for the y-axis
	etc. up to 14th dimension of an Ntuple.
231	Underflows x-axis
232	Contents x-axis
233	Overflows x-axis
For 2 dimensional histograms 9 registers are filled (contents in register 235), while for 3-dimensional histograms 27 registers are filled (contents in register 244).	
Registers 201-204 contain the positions of the corners of the current plot in cm	
201	x position left
202	x position right
203	y position bottom
204	y position top
Registers 205-210 contain the limits used for the drawing of each of the axes They are in plot co-ordinates	
205	x minimum
206	x maximum
207	y minimum
208	y maximum
209	z minimum
210	z maximum
Registers > 300 contain extra variable names that you have defined. Up to 200 variables are allowed.	

Table B.1: The list of standard Mn_Fit registers. See Section 1.7 for more details on what you can give when Mn_Fit asks for a number.

Font 2000, i.e. IGTEXT

Upper Roman	Lower Roman	Upper Greek	Lower Greek	Upper Special	Lower Special
A	a	Α	α	±	±
B	b	Β	β	∓	∓
C	c	Γ	γ	⊕	⊗
D	d	Δ	δ	\$	\$
E	e	Ε	ε	!	!
F	f	Φ	φ	#	#
G	g	Γ	γ	>	>
H	h	Χ	χ	?	?
I	i	Ι	ι	∫	∫
J	j	Ι	ι	∴	∴
K	k	Κ	κ	∠	∠
L	l	Λ	λ	<	<
M	m	Μ	μ	[[
N	n	Ν	ν	≡	≡
O	o	Ο	ο	{	{
P	p	Π	π	}	}
Q	q	Θ	θ	√	√
R	r	Ρ	ρ	⊙	⊙
S	s	Σ	σ	♥	♥
T	t	Τ	τ	◇	◇
U	u	Χ	χ	⊗	⊗
V	v	Ω	ω	⊕	⊕
W	w	Ω	ω	&	&
X	x	Ψ	ψ	×	×
Y	y	Ζ	ζ	%	%
Z	z	Ζ	ζ	∞	∞
0	0	0	0	⊙	⊙
1	1	1	1	▣	▣
2	2	2	2	▴	▴
3	3	3	3	▾	▾
4	4	4	4	★	★
5	5	5	5	→	→
6	6	6	6	↑	↑
7	7	7	7	←	←
8	8	8	8	↓	↓
9	9	9	9	↷	↷
.	.	.	.	◇	□
,	,	,	,	↷	↷
+	+	+	+	+	+
-	-	-	-	-	-
*	*	*	*	*	*
/	/	/	/	/	/
=	=	=	=	=	=
((((((
))))))

[Switch to Greek]	Switch back from Greek
"	Switch to special	#	Switch back from special
?	Switch to subscript	!	Switch back from subscript
^	Switch to superscript	!	Switch back from superscript
&	Backspace one character	@	Print an escape character

Figure B.2: HIGZ Portable Software Characters and the escape characters to switch modes.

Font -1004, i.e. Postscript - Helvetica

Upper Roman	Lower Roman	Upper Greek	Lower Greek	Upper Special	Lower Special	Upper Zapf	Lower Zapf
A	a	A	α	±	≈	☆	✿
B	b	B	β	—	≡	✿	✿
C	c	H	η	☐	⊥	✿	✿
D	d	Δ	δ	√	∂	✿	✿
E	e	E	ε	!	f	✿	✿
F	f	Φ	φ	#	∪	◆	✿
G	g	Γ	γ	>	∪	◇	✿
H	h	X	χ	?	∪	★	✿
I	i	I	ι	↓	∪	☆	✿
J	j	I	ι	⋮	∪	⊕	✿
K	k	K	κ	;	∪	☆	✿
L	l	Λ	λ	<	∪	☆	●
M	m	M	μ	⌈	ε	☆	○
N	n	N	ν	⌋	≠	☆	■
O	o	O	ο	≡	∇	☆	□
P	p	Π	π	~	∧	☆	□
Q	q	Θ	θ	↙	↕	✱	□
R	r	Σ	ρ	↘	↕	✱	□
S	s	T	σ	♠	↕	✱	▲
T	t	Υ	τ	♥	↕	✱	▼
U	u	X	υ	♦	↕	✱	◆
V	v	Ω	ω	♣	↕	✱	◇
W	w	Ξ	ξ	≤	&	✱	◐
X	x	Ψ	ψ	×	⊗	✱	—
Y	y	Z	ζ	%	⊗	✱	■
Z	z	0	0	8	⊗	✱	■
0	0	1	1	⊗	⊗	✱	■
1	1	2	2	⊗	⊗	✱	■
2	2	3	3	⊗	⊗	✱	■
3	3	4	4	⊗	⊗	✱	■
4	4	5	5	⊗	⊗	✱	■
5	5	6	6	⊗	⊗	✱	■
6	6	7	7	⊗	⊗	✱	■
7	7	8	8	⊗	⊗	✱	■
8	8	9	9	⊗	⊗	✱	■
9	9	.	.	↔	↔	✱	■
.	.	,	,	.	.	✱	■
,	,	+	+	,	,	✱	■
+	+	-	-	+	+	✱	■
-	-	*	*	<	<	✱	■
*	*	/	/	>	>	✱	■
/	/	=	=	÷	÷	✱	■
=	=	((≠	≠	✱	■
(())	≡	≡	✱	■
))			”	”	✱	■

[Switch to Greek]	Switch back from Greek
"	Switch to special	#	Switch back from special
?	Switch to subscript	!	Switch back from subscript
^	Switch to superscript	!	Switch back from superscript
~	Switch to Zapf Dingbats	#	Switch back from Zapf Dingbats
&	Backspace one character	@	Print an escape character

Figure B.3: Postscript Version of HIGZ Portable Software Characters and the escape characters to switch modes.

GKSGRAL Software Fonts		Postscript
Font/Prec	Font/Prec	Font/Prec
– 1/2 abcABC 123	–201/2 abcABC 123	– 1/0 abcABC 123
– 2/2 abcABC 123	–202/2 abcABC 123	– 2/0 <i>abcABC 123</i>
– 3/2 abcABC 123	–203/2 abcABC 123	– 3/0 abcABC 123
– 4/2 abcABC 123	–204/2 abcABC 123	– 4/0 <i>abcABC 123</i>
– 5/2 <i>abcABC 123</i>	–205/2 <i>abcABC 123</i>	– 5/0 abcABC 123
– 6/2 <i>abcABC 123</i>	–206/2 <i>abcABC 123</i>	– 6/0 <i>abcABC 123</i>
– 7/2 <i>abcABC 123</i>	–207/2 <i>abcABC 123</i>	– 7/0 abcABC 123
– 8/2 <i>abcABC 123</i>	–208/2 <i>abcABC 123</i>	– 8/0 <i>abcABC 123</i>
– 9/2 abcABC 123	–209/2 abcABC 123	– 9/0 abcABC 123
–10/2 <i>abcABC 123</i>	–210/2 <i>abcABC 123</i>	–10/0 <i>abcABC 123</i>
–11/2 abcABC 123	–211/2 abcABC 123	–11/0 abcABC 123
–13/2 αβχABX 123	–213/2 αβχABX 123	–12/0 <i>abcABC 123</i>
–101/2 <i>abcABC 123</i>	–301/2 <i>abcABC 123</i>	–13/0 αβχABX 123
–102/2 <i>abcABC 123</i>	–302/2 <i>abcABC 123</i>	
–103/2 <i>abcABC 123</i>	–303/2 <i>abcABC 123</i>	
–104/2 <i>abcABC 123</i>	–304/2 <i>abcABC 123</i>	
–105/2 <i>abcABC 123</i>	–305/2 <i>abcABC 123</i>	
–106/2 <i>abcABC 123</i>	–306/2 <i>abcABC 123</i>	
–107/2 <i>abcABC 123</i>	–307/2 <i>abcABC 123</i>	
–108/2 <i>abcABC 123</i>	–308/2 <i>abcABC 123</i>	
–109/2 <i>abcABC 123</i>	–309/2 <i>abcABC 123</i>	
–110/2 <i>abcABC 123</i>	–310/2 <i>abcABC 123</i>	
–111/2 abcABC 123	–311/2 abcABC 123	
–113/2 αβχABX 123	–313/2 αβχABX 123	

Figure B.5: Examples of GKSGRAL Fonts

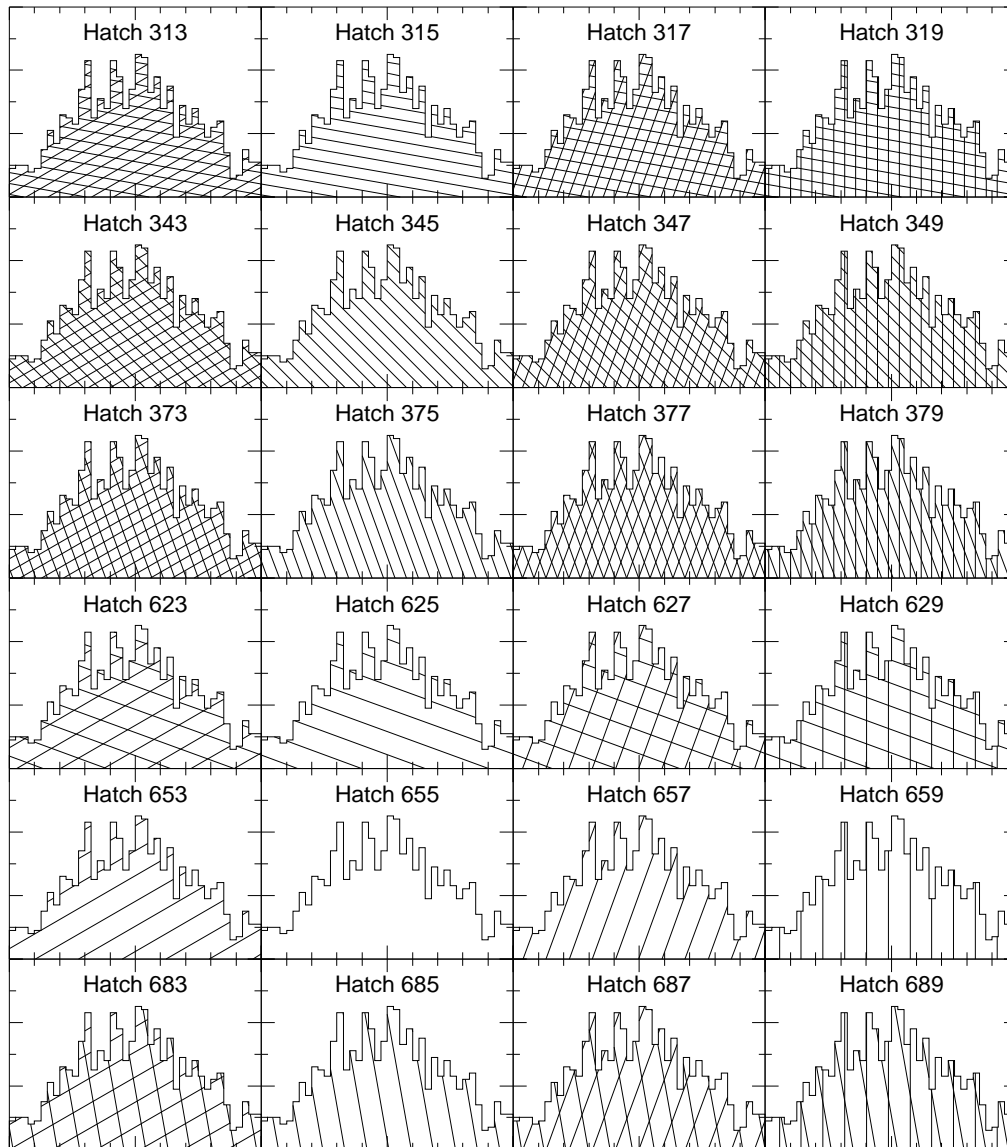


Figure B.6: Examples of HIGZ Hatching

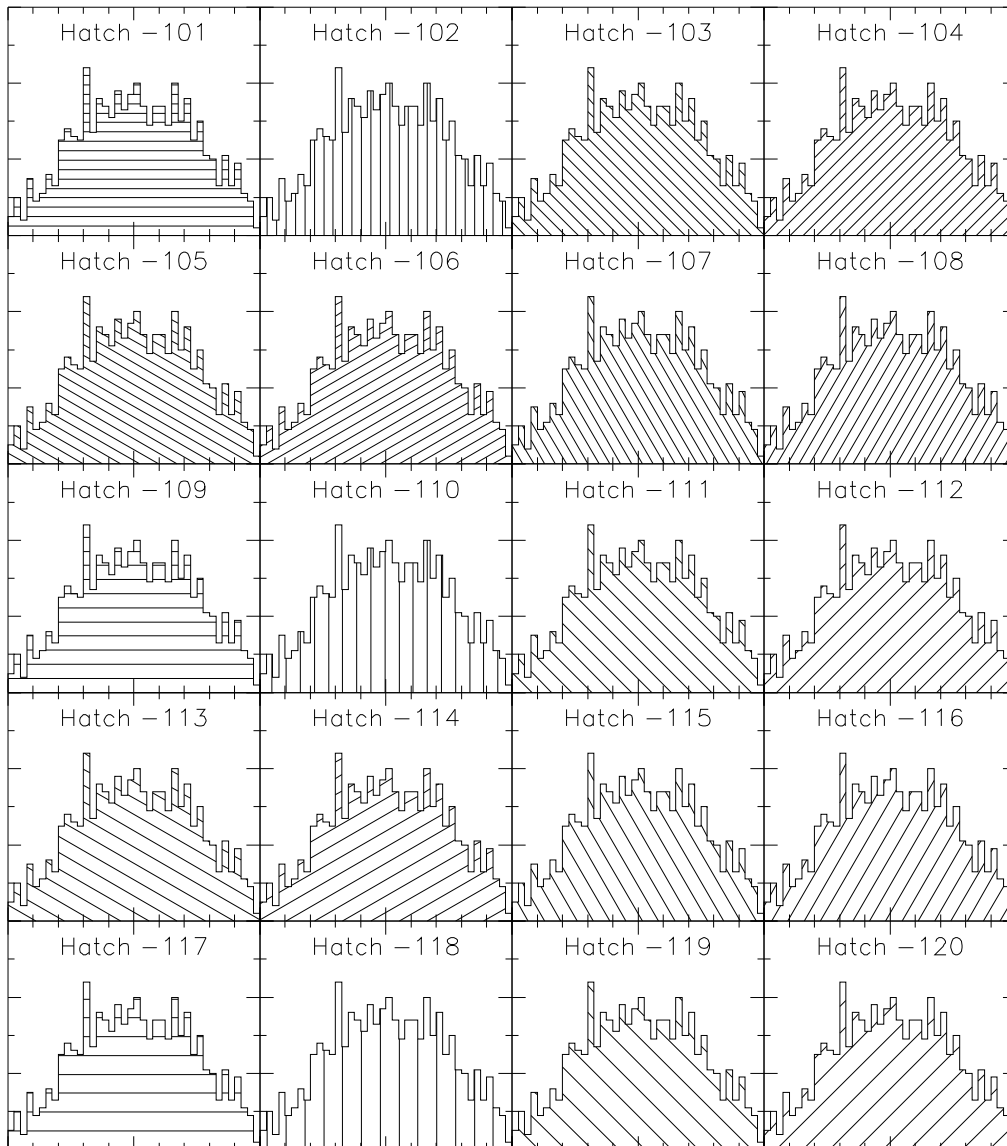


Figure B.7: Examples of GKSGRAL Hatching

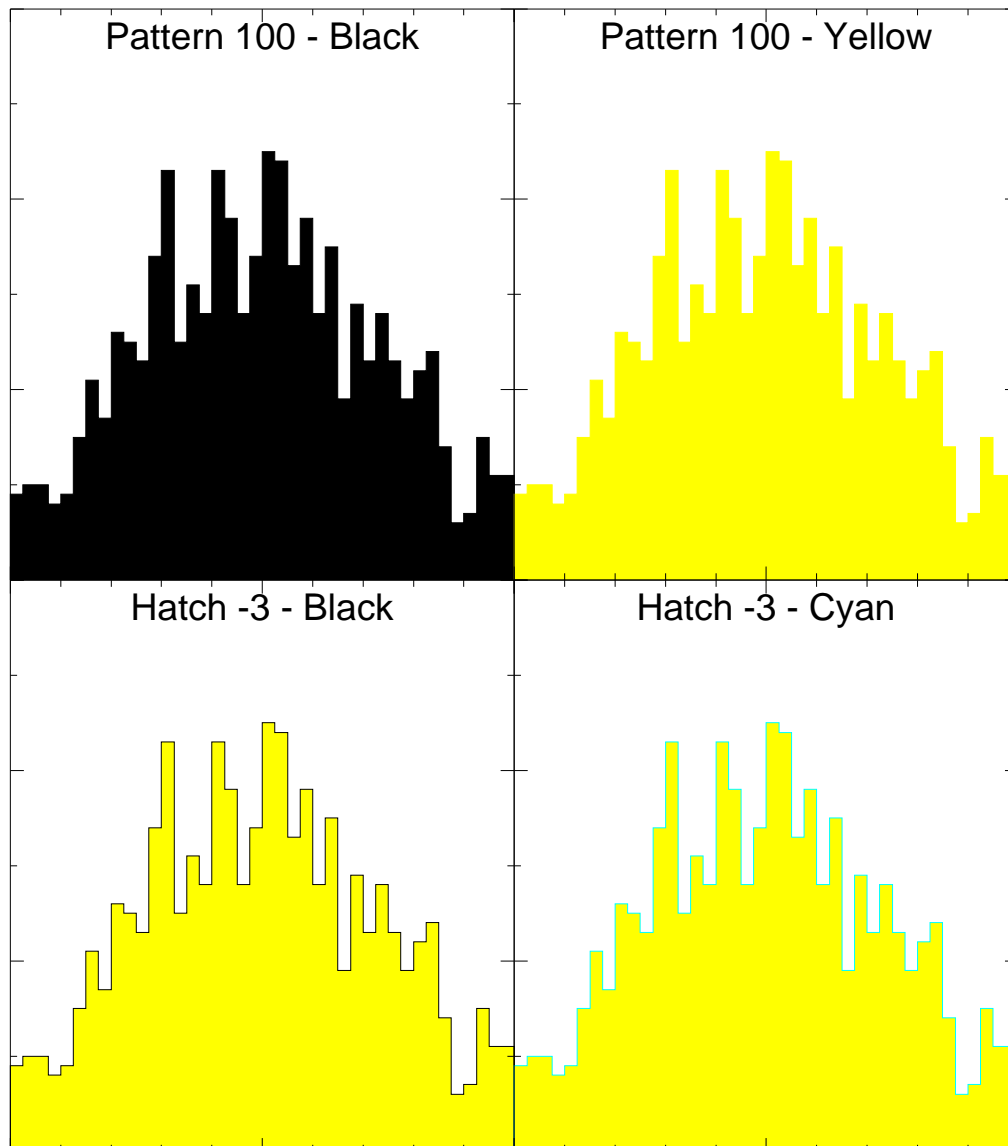


Figure B.8: Examples of Patterns

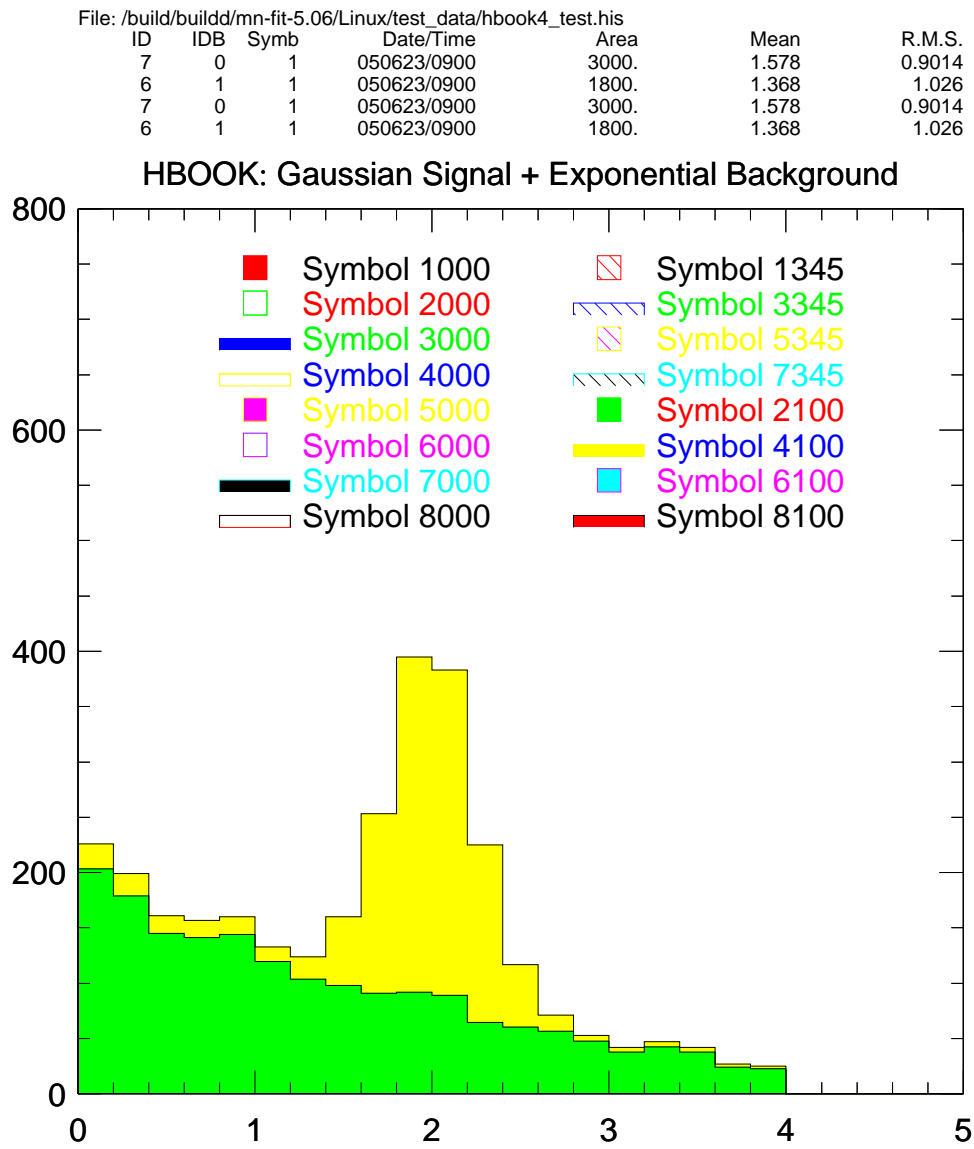


Figure B.9: Examples of Keys

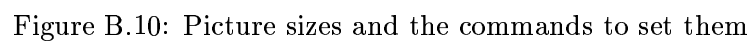


Figure B.10: Picture sizes and the commands to set them

Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf
\0	\0	∴0	∴⊗	*✍	\25	\25	∴25	∴∅→	*✍✕	\50	((≡	✍✍
\1	\1	∴1	∴⊕	*✍	\26	\26	∴26	∴∅↑	*✍✕	\51))	"	✍✍
\2	\2	∴2	∴∅	*✍	\27	\27	∴27	∴∅←	*✍✕	\52	*	*	<	✍✍
\3	\3	∴3	∴∅	*✓	\28	\28	∴28	∴∅↓	*✍✕	\53	+	+	+	✍✍
\4	\4	∴4	∴•	*✓	\29	\29	∴29	∴∅↔	*✍✍	\54	,	,	,	✍✍
\5	\5	∴5	∴→	*✕	\30	\30	∴30	∴∅⊗	*✓✍	\55	-	-	∠	✍✍
\6	\6	∴6	∴↑	*✕	\31	\31	∴31	∴∅⊕	*✓✍	\56	.	.	.	✍✍
\7	\7	∴7	∴←	*✕	\32	\32	∴32	∴∅∅	*✓✍	\57	/	/	÷	≡
\8	\8	∴8	∴↓	*✕	\33	\33	∴33	∴∅∅	*✓✓	\58	8	8	↓	✕
\9	\9	∴9	∴↔	*✍	\34	\34	∴34	∴∅•	*✓✓	\59	9	9	↔	✍
\10	\10	∴10	∴⊕⊗	*✍✍	\35	\35	∴35	∴∅→	*✓✕	\60	0	0	⊗	✍
\11	\11	∴11	∴⊕⊕	*✍✍	\36	\36	∴36	∴∅↑	*✓✕	\61	1	1	⊕	✍
\12	\12	∴12	∴⊕∅	*✍✍	\37	\37	∴37	∴∅←	*✓✕	\62	2	2	∅	✍
\13	\13	∴13	∴⊕∅	*✍✓	\38	\38	∴38	∴∅↓	*✓✕	\63	3	3	∅	✓
\14	\14	∴14	∴⊕•	*✍✓	\39	\39	∴39	∴∅↔	*✓✍	\64	4	4	•	✓
\15	\15	∴15	∴⊕→	*✍✕	\40					\65	5	5	→	✕
\16	\16	∴16	∴⊕↑	*✍✕	\41	!	!	!	✍	\66	6	6	↑	✕
\17	\17	∴17	∴⊕←	*✍✕	\42	"	∇	∇	✍	\67	7	7	←	✕
\18	\18	∴18	∴⊕↓	*✍✕	\43	#	#	#	✍	\68	8	8	↓	✕
\19	\19	∴19	∴⊕↔	*✍✍	\44	\$	∃	∃	✍	\69	9	9	↔	✍
\20	\20	∴20	∴∅⊗	*✍✍	\45	%	%	%	✍	\70	8	8	↓	✕
\21	\21	∴21	∴∅⊕	*✍✍	\46	&	&	&	✍	\71	9	9	↔	✍
\22	\22	∴22	∴∅∅	*✍✍	\47	'	ə	ə	✍	\72	:	:	:	✍
\23	\23	∴23	∴∅∅	*✍✓	\48	8	8	↓	✕	\73	;	;	;	✍
\24	\24	∴24	∴∅•	*✍✓	\49	9	9	↔	✍	\74	<	<	<	✍

Figure B.11: Postscript characters - Codes 0 to 75

Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf
\75	=	=	≠	†	\100	@	≡	≡	⌘	\125	U	Υ	◆	⌘
\76	>	>	>	‡	\101	A	Α	±	☆	\126	V	ϛ	♣	✱
\77	?	?	?	‡	\102	B	Β		⌘	\127	W	Ω	≤	✱
\78	\78	∴78	∴←↓	*XX	\103	C	Χ	∃	⌘	\128	8	8	↓	✱
\79	\79	∴79	∴←↔	*X+	\104	D	Δ	∇	⌘	\129	9	9	↔	+
\80	\80	∴80	∴↓⊗	*X/	\105	E	Ε	!	⌘	\130	X	Ξ	×	✱
\81	\81	∴81	∴↓⊕	*X/	\106	F	Φ	#	◆	\131	Y	Ψ	%	✱
\82	\82	∴82	∴↓⊖	*X/	\107	G	Γ	>	◇	\132	Z	Ζ	∞	⌘
\83	\83	∴83	∴↓⊘	*X/	\108	8	8	↓	✱	\133	[[[✱
\84	\84	∴84	∴↓•	*X/	\109	9	9	↔	+	\134	\	∴	∴	✱
\85	\85	∴85	∴↓→	*XX	\110	H	Η	?	★	\135]]]	✱
\86	\86	∴86	∴↓↑	*XX	\111	I	Ι	∫	☆	\136	^	⊥	⊥	⌘
\87	\87	∴87	∴↓←	*XX	\112	J	ϑ	:	⊗	\137	_	_	_	⌘
\88	\88	∴88	∴↓↓	*XX	\113	K	Κ	;	★	\138	8	8	↓	✱
\89	\89	∴89	∴↓↔	*X+	\114	L	Λ	<	★	\139	9	9	↔	+
\90	\90	∴90	∴↔⊗	*+/	\115	M	Μ	[★	\140	'	_	_	⌘
\91	\91	∴91	∴↔⊕	*+/	\116	N	Ν]	★	\141	a	α	≈	⌘
\92	\92	∴92	∴↔⊖	*+/	\117	O	Ο	≥	★	\142	b	β	≡	✱
\93	\93	∴93	∴↔⊘	*+/	\118	8	8	↓	✱	\143	c	χ	⊥	⌘
\94	\94	∴94	∴↔•	*+/	\119	9	9	↔	+	\144	d	δ	∂	⌘
\95	\95	∴95	∴↔→	*+X	\120	P	Π	{	☆	\145	e	ε	f	⌘
\96	\96	∴96	∴↔↑	*+X	\121	Q	Θ	}	★	\146	f	φ	∩	⌘
\97	\97	∴97	∴↔←	*+X	\122	R	Ρ	√	★	\147	g	γ	∪	⌘
\98	\98	∴98	∴↔↓	*+X	\123	S	Σ	♠	✱	\148	8	8	↓	✱
\99	\99	∴99	∴↔↔	*++	\124	T	Τ	♥	✱	\149	9	9	↔	+

Figure B.12: Postscript characters - Codes 75 to 149

Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf	Input	Roman	Greek	Spec	Zapf
\150	h	η	⊃	⌘	\175	}	}	}	“	\200	\200	∅	⊕	⌘⌘⌘⌘
\151	i	ι	⊇	⌘	\176	~	~	~	”	\201	\201	∅	⊕	⌘⌘⌘⌘
\152	j	φ	⋈	⌘	\177					\202	\202	∅	⊕	⌘⌘⌘⌘
\153	k	κ	⊂	⌘	\178	\178	∅	⊕	⌘⌘⌘⌘	\203	\203	∅	⊕	⌘⌘⌘⌘
\154	l	λ	⊆	●	\179	\179	∅	⊕	⌘⌘⌘⌘	\204	\204	∅	⊕	⌘⌘⌘⌘
\155	m	μ	∈	○	\180	\180	∅	⊕	⌘⌘⌘⌘	\205	\205	∅	⊕	⌘⌘⌘⌘
\156	n	ν	∉	■	\181	\181	∅	⊕	⌘⌘⌘⌘	\206	\206	∅	⊕	⌘⌘⌘⌘
\157	o	ο	∇	□	\182	\182	∅	⊕	⌘⌘⌘⌘	\207	\207	∅	⊕	⌘⌘⌘⌘
\158	8	8	↓	✕	\183	\183	∅	⊕	⌘⌘⌘⌘	\208	\208	∅	⊕	⌘⌘⌘⌘
\159	9	9	↔	⊕	\184	\184	∅	⊕	⌘⌘⌘⌘	\209	\209	∅	⊕	⌘⌘⌘⌘
\160	p	π	^	□	\185	\185	∅	⊕	⌘⌘⌘⌘	\210	\210	∅	⊕	⌘⌘⌘⌘
\161	q	θ	∇	□	\186	\186	∅	⊕	⌘⌘⌘⌘	\211	\211	∅	⊕	⌘⌘⌘⌘
\162	r	ρ	↔	□	\187	\187	∅	⊕	⌘⌘⌘⌘	\212	\212	∅	⊕	⌘⌘⌘⌘
\163	s	σ	←	▲	\188	\188	∅	⊕	⌘⌘⌘⌘	\213	\213	∅	⊕	⌘⌘⌘⌘
\164	t	τ	↑	▼	\189	\189	∅	⊕	⌘⌘⌘⌘	\214	\214	∅	⊕	⌘⌘⌘⌘
\165	u	υ	⇒	◆	\190	\190	∅	⊕	⌘⌘⌘⌘	\215	\215	∅	⊕	⌘⌘⌘⌘
\166	v	ϖ	↓	❖	\191	\191	∅	⊕	⌘⌘⌘⌘	\216	\216	∅	⊕	⌘⌘⌘⌘
\167	w	ω	&	►	\192	\192	∅	⊕	⌘⌘⌘⌘	\217	\217	∅	⊕	⌘⌘⌘⌘
\168	8	8	↓	✕	\193	\193	∅	⊕	⌘⌘⌘⌘	\218	\218	∅	⊕	⌘⌘⌘⌘
\169	9	9	↔	⊕	\194	\194	∅	⊕	⌘⌘⌘⌘	\219	\219	∅	⊕	⌘⌘⌘⌘
\170	x	ξ	ϖ	⌈	\195	\195	∅	⊕	⌘⌘⌘⌘	\220	\220	∅	⊕	⌘⌘⌘⌘
\171	y	ψ	~	⌈	\196	\196	∅	⊕	⌘⌘⌘⌘	\221	\221	∅	⊕	⌘⌘⌘⌘
\172	z	ζ	⌘	■	\197	\197	∅	⊕	⌘⌘⌘⌘	\222	\222	∅	⊕	⌘⌘⌘⌘
\173	{	{	{	‘	\198	\198	∅	⊕	⌘⌘⌘⌘	\223	\223	∅	⊕	⌘⌘⌘⌘
\174				’	\199	\199	∅	⊕	⌘⌘⌘⌘	\224	\224	∅	⊕	⌘⌘⌘⌘

Figure B.13: Postscript characters - Codes 150 to 224