PostScript Characters and Fonts in Scribe

A. D. Guisewite

CMU-RI-SL-89-00

Intelligent Sensors Laboratory Application Note

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

May 1997

Abstract

Various Scribe procedures for depicting the full range of resident and downloadable characters and fonts available from a laser printer are described along with techniques for creating new Scribe symbols.

Introduction

The present Scribe site implementation does not allow the full usage of the PostScript®[1] resident and downloadable characters and fonts. This application note describes the procedures for obtaining these characters and fonts using a PostScript-compatible laser printer.[2] Later sections explain the design of new Scribe symbols, textforms, and forms. The final sections cover new and modified Scribe environments.

These characters and fonts have been installed in a new document makefile **TECHRE.MAK** created primarily to produce a Carnegie-Mellon University Robotics Institute Technical Report. A listing and explanation of **TECHRE.MAK** is in Appendix A.

PostScript Characters

The PostScript characters shown in this paper are (for the most part) written in the Helvetica family of fonts. The character-mapped symbols in Tables 6 through 11 are available in all of the PostScript-supported fonts.[3]

Access via Library File — Mathematical Symbols and Accented Characters

The following four tables show the translations from lower case and upper case alphabetical, numerical, and punctuation characters to their mathematical symbol and accented character equivalents. For the diacritical marks themselves, see Table 6 on page 9.

Note

- You **must** have @LibraryFile(Accents) and @LibraryFile(Mathematics10) present in your manuscript to access these characters and even with Scribe's **ACCENT.LIB** file you can only access the @act column (roman typeface) characters.
- Notice that the @b(char) and @i(char) environments are concatenated with the @act(char) environment and that there is **no** @p(char) environment.[4]
- Scribe's **ACCENT.LIB** file only has mnemonic form equivalents for the @act character calls[5] and they are shown in the last column. For example, @aac is the same as @act(a) and @uaac is the same as @act(A).
- Senslab's **NEWACC.LIB** has all four facecodes (as shown) and mnemonic form equivalents for the commandstrings as well (i.e., @aac, @uaac, @baac, @buaac, @iaac, @iuaac, @paac, and @puaac).[6] The @act, @bact, @iact, and @pact environments have also been made available as facecodes 1, 2, 3, and 4 respectively (@1(a) is the same as @act(a)).
- The accented characters shown are the ones available as part of the PostScript set —

---

1. PostScript is a registered trademark of Adobe Systems Incorporated.

2. The fonts described herein apply to an Apple LaserWriterPlus™-printed document — other laser printers such as the DEC ScriptPrinter™ will use different fonts. (There are some fonts in common.)

3. The typecase file has to be remapped to the desired font and the @DefineFont command has to select the desired font as well. See the section on the **NEWSYM.TYP** typecase file for details.

4. There is a way to get the @p(char) facecode but it involves copying, renaming, and editing Scribe's **ACCENT.LIB** file. The characters are shown for reference and a listing of the **NEWACC.LIB** library file is in Appendix B.

5. Scribe's Version 5 (now called OldScribe) **has** equivalents for the @act, @bact, and @iact character calls.

6. At present, these accented characters are only available in the Helvetica, Times-Roman, and Courier font families — the others can be easily added as needed.

they are printed as one character after being mapped to an equivalent unaccented character (reencoded) and invoked as either a form (@aac) or commandstring (@act(a)).[7] Since there are 28 accented characters and only 26 letters available, two punctuation marks (per case) are used.

Other accented characters can be created by overprinting (via @Ovp) a diacritical mark.

7. It is possible to create other accented characters with PostScript, but this involves more than mere reencoding.

Lower Case Alphabetical Character Translations

Table 1

The lower case alphabetical translations to mathematical and accented characters

Character	@jsym()	@ksym()	@zsym()	@act()	@bact()	@iact()	@pact()	@form
a	⎪	⎰	a	á	**á**	*á*	***á***	@aac
b	⎯	⎱	b	â	**â**	*â*	***â***	@acr
c	↵	⎪	c	ä	**ä**	*ä*	***ä***	@aum
d	ℵ	⎭	d	à	**à**	*à*	***à***	@agr
e	ℑ	⎫	e	å	**å**	*å*	***å***	@arg
f	ℜ	⎪	f	ã	**ã**	*ã*	***ã***	@atl
g	℘	⎭	g	ç	**ç**	*ç*	***ç***	@ccd
h	⊗	⎤	h	é	**é**	*é*	***é***	@eac
i	⊕	⎪	i	ê	**ê**	*ê*	***ê***	@ecr
j	∅	⎦	j	ë	**ë**	*ë*	***ë***	@eum
k	∩	⎤	k	è	**è**	*è*	***è***	@egr
l	∪	⎬	l	í	**í**	*í*	***í***	@iac
m	⊃	⎭	m	î	**î**	*î*	***î***	@icr
n	⊇	(	n	ï	**ï**	*ï*	***ï***	@ium
o	⊄	)	o	ì	**ì**	*ì*	***ì***	@igr
p	⊂	[	p	ñ	**ñ**	*ñ*	***ñ***	@ntl
q	⊆	]	q	ó	**ó**	*ó*	***ó***	@oac
r	∈	{	r	ô	**ô**	*ô*	***ô***	@ocr
s	∉	}	s	ö	**ö**	*ö*	***ö***	@oum
t	∠	∃	t	ò	**ò**	*ò*	***ò***	@ogr
u	∇	υ	u	õ	**õ**	*õ*	***õ***	@otl
v	®	ϖ	v	š	**š**	*š*	***š***	@sca
w	©	ω	w	ú	**ú**	*ú*	***ú***	@uac
x	™	ξ	x	û	**û**	*û*	***û***	@ucr
y	∏	ψ	y	ü	**ü**	*ü*	***ü***	@uum
z	√	ζ	z	ù	**ù**	*ù*	***ù***	@ugr
{				ÿ	**ÿ**	*ÿ*	***ÿ***	@yum
|				ž	**ž**	*ž*	***ž***	@zca

Upper Case Alphabetical Character Translations

Table 2

The upper case alphabetical translations to mathematical and accented characters

Character	@jsym()	@ksym()	@zsym()	@act()	@bact()	@iact()	@pact()	@form
A	≤	·	*A*	Á	**Á**	*Á*	***Á***	@uaac
B	⁄	¬	*B*	Â	**Â**	*Â*	***Â***	@uacr
C	∞	∧	*C*	Ä	**Ä**	*Ä*	***Ä***	@uaum
D	*f*	∨	*D*	À	**À**	*À*	***À***	@uagr
E	♣	⇔	*E*	Å	**Å**	*Å*	***Å***	@uarg
F	♦	⇐	*F*	Ã	**Ã**	*Ã*	***Ã***	@uatl
G	♥	⇑	*G*	Ç	**Ç**	*Ç*	***Ç***	@uccd
H	♠	⇒	*H*	É	**É**	*É*	***É***	@ueac
I	↔	⇓	*I*	Ê	**Ê**	*Ê*	***Ê***	@uecr
J	←	◊	*J*	Ë	**Ë**	*Ë*	***Ë***	@ueum
K	↑	⟨	*K*	È	**È**	*È*	***È***	@uegr
L	→	®	*L*	Í	**Í**	*Í*	***Í***	@uiac
M	↓	©	*M*	Î	**Î**	*Î*	***Î***	@uicr
N	°	™	*N*	Ï	**Ï**	*Ï*	***Ï***	@uium
O	±	∑	*O*	Ì	**Ì**	*Ì*	***Ì***	@uigr
P	″	⎛	*P*	Ñ	**Ñ**	*Ñ*	***Ñ***	@untl
Q	≥	⎜	*Q*	Ó	**Ó**	*Ó*	***Ó***	@uoac
R	×	⎝	*R*	Ô	**Ô**	*Ô*	***Ô***	@uocr
S	∝	⎡	*S*	Ö	**Ö**	*Ö*	***Ö***	@uoum
T	∂	⎜	*T*	Ò	**Ò**	*Ò*	***Ò***	@uogr
U	•	⎣	*U*	Õ	**Õ**	*Õ*	***Õ***	@uotl
V	÷	⎡	*V*	Š	**Š**	*Š*	***Š***	@usca
W	≠	⎨	*W*	Ú	**Ú**	*Ú*	***Ú***	@uuac
X	≡	⎣	*X*	Û	**Û**	*Û*	***Û***	@uucr
Y	≈	⎜	*Y*	Ü	**Ü**	*Ü*	***Ü***	@uuum
Z	…	⟩	*Z*	Ù	**Ù**	*Ù*	***Ù***	@uugr
[				Ÿ	**Ÿ**	*Ÿ*	***Ÿ***	@uyum
\				Ž	**Ž**	*Ž*	***Ž***	@uzca

Numerical Character Translations

Table 3

The numerical translations to Times-Roman and Times-Italic numbers

Character	@jsym()	@ksym()	@zsym()
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*
0	0	0	*0*

Punctuation Character Translations

Table 4

The punctuation translations to mathematical characters

Character	@jsym()	@ksym()	@zsym()
!	!	!	*!*
@	≅	≅	*@*
#	#	#	*#*
$	∃	∃	*$*
%	*%*	*%*	*%*
^	⊥	⊥	∧
&	&	&	&
*	∗	∗	*
(	(	(	*(*
)	)	)	*)*
–	–	–	–
+	+	+	+
~	~	~	~
-	–	–	-
=	=	=	=
‘	¯	¯	*‘*
[	[	[	*[*
]	]	]	*]*

... continued on following page

Character	@jsym()	@ksym()	@zsym()
\	∴	∴	\
|	|	|	|
;	;	;	;
:	:	:	:
’	ɜ	ɜ	’
"	∀	∀	″
{	«	{	{
}	»	}	}
’	’	’	’
.	.	.	.
/	/	/	/
<	<	<	<
>	>	>	>
?	?	?	?

Extra PostScript Symbols

There are two sets of characters resident in PostScript that are not available through normal keyboard use.

1. A set of characters obtained by using the Zapf Dingbat font.
2. A set of characters obtained by creating a new typecase LibraryFile with ASCII character mapping to the desired PostScript equivalent.

Access via @SpecialFont Command — ZapfDingbat Symbols

The following special characters are invoked by placing the Scribe command

@SpecialFont(F8[8] **ZapfDingbats)**

at the beginning of the manuscript.

8. The number in the string F**8** can be any digit from 0 through 9.

Table 5

The ASCII character translations to special font Zapf Dingbat characters

Character	@F8()	Character	@F8()	Character	@F8()
a	❁	A	✡	!	✁
b	✲	B	✣	@	✠
c	❃	C	✤	#	✂
d	❄	D	✥	$	✄
e	❅	E	✦	%	✆
f	❆	F	✧	^	✇
g	❇	G	✩	&	✍
h	❈	H	★	*	✎
i	❉	I	☆	(	✏
j	❊	J	✪	)	✐
k	❋	K	✫	_	✑
l	●	L	✬	+	✒
m	❍	M	✭	~	✓
n	■	N	✮	]	✔
o	❑	O	✯	|	✕
p	❒	P	✰	:	✖
q	❑	Q	✶	"	✗
r	❒	R	✷	}	✘
s	▲	S	✸	<	✙
t	▼	T	✹	>	✚
u	◆	U	✺	?	✛
v	❖	V	✻	-	✜
w	◗	W	✼	=	✝
x	❘	X	✽	‘	✞
y	❙	Y	✾	[	✟
z	❚	Z	✿	\	✠
1	✁	6	✖	;	✡
2	✂	7	✗	’	✢
3	✓	8	✘	{	✣
4	✔	9	✙	,	✤
5	✕	0	✁	.	✥
				]	✦

Access via Character Mapping

The characters shown in the following tables are obtained through the character mapping procedure embedded in a database file. Octal codes are given for each special character to facilitate creating the map. The equivalent ASCII character is shown for each special character and can be reassigned as desired. An explanation and listing of **NEWSYM.TYP** follows the tables.

Note

- The first five sets of characters are available in all font families but are only shown in **Helvetica** (Tables 6 through 10) as are the first four characters in Table 11.
- The seventh set of symbols (Table 12) requires the ZapfDingbats @LibraryFile command as well.

Diacritical Marks

Table 6

The character translations to diacritical marks with corresponding octal codes

Mark	@N()	Octal Code and Description
´	’	\302 Acute
˘	Z	\306 Breve
¸	z	\313 Cedilla
ˆ	^	\303 Circumflex
¨	w	\310 Diaeresis, trema, or umlaut
`	‘	\301 Grave
˝	"	\315 Hungarian double prime*
¯	-	\305 Macron
˙	.	\307 Over-dot
°	K	\312 Over-ring or Czech krouzek
˛	k	\316 Inverted cedilla
˜	~	\304 Tilde
ˇ	W	\317 Caron, wedge, or Slavic hachek

*This is not a "seconds" mark or double quote — it is much more slanted.

Digraphs, Ligatures, Etc.

Table 7

The character translations to digraphs, ligatures, foreign special characters with corresponding octal codes

Character	@N()	Octal Code and Description
fi	F	\256 Digraph for lower case FI
fl	f	\257 Digraph for lower case FL
ß	s	\373 German digraph for lower case SS
æ	a	\361 Ligature for lower case AE or "ash"*
Æ	A	\341 Ligature for upper case AE
œ	e	\372 Ligature for lower case OE
Œ	E	\352 Ligature for upper case OE
ı	i	\365 Turkish or dotless lower case I†
ł	l	\370 Polish lower case slashed L
Ł	L	\350 Polish upper case slashed L
ø	o	\371 Danish/Norwegian lower case slashed O
Ø	O	\351 Danish/Norwegian upper case slashed O

Monetary Marks

The peso symbol is available as a commandstring, see page 24.

Table 8

The character translations to monetary symbols with corresponding octal codes

Symbol	@N()	Octal Code and Description
¢	c	\242 Cent
¤	*	\250 Currency
ƒ	C	\246 Florin
£	$	\243 Pound Sterling
¥	Y	\245 Yen

*The ash is a recognized symbol of the International Phonetic Association.

†The "dotless i" is also used beneath diacritical marks and other places where the dot would intrude.

English and Foreign Punctuation Marks

Table 9

The character translations to English and foreign punctuation marks with corresponding octal codes

Mark	@N()	Octal Code and Description
'	q	\251 Typewritten single quote, matches "
”	Q	\272 English close double quote, matches \252
‹	M	\254 Open single guillemet*
›	m	\255 Close single guillemet
«	N	\253 Open double guillemet
»	n	\273 Close double guillemet
„	G	\271 German open double quote
“	g	\252 German close double quote (and English open double quote)
·		\264 Greek colon, semicolon†
¡	!	\241 Spanish exclamation point
¿	?	\277 Spanish question mark

Reference Marks

Table 10

The character translations to reference marks with corresponding octal codes

Mark	@N()	Octal Code and Description
†	d	\262 Dagger
‡	D	\263 Double dagger
¶	P	\266 Paragraph
§	S	\247 Section

*Guillemets are quotation marks used in French, Italian, Russian, and Spanish literature.

†Since this mark represents both a Greek colon and semicolon it has been mapped to both marks (@N(:) and @N(;)).

Miscellaneous Symbols

Table 11

The character translations to miscellaneous symbols with corresponding octal codes

Symbol	@N()	Octal Code and Description
ª	x	\343 Ordfeminine
º	X	\353 Ordmasculine
⁄	/	\244 Fraction
‰	%	\275 Parts per thousand, especially salinity
φ	p	\151 Script lower case phi
ς	y	\126 Final lower case sigma
ϑ	t	\112 Script lower case theta
ϒ	U	\241 Script upper case upsilon
ϖ	I	\166 Alternate lower case pi
	R	\360 Apple Computer trademark

The NEWSYM.TYP Senslab NewSymbol Typecase File

The complete character mapping database file for the foregoing tables is shown below. If you use a font family other than Helvetica, change the **RawFont Helvetica** subfield accordingly (e.g. **RawFont TimesRoman**).

```
@Marker{TypeCase, NewSymbol, PostScript}

@DefineTypeCase[NewSymbol,
        Entry {Char ASCIIGraphics, RawFont Helvetica},
        Entry {Char "F", MapsTo "8#256", RawFont Helvetica},
        Entry {Char "f", MapsTo "8#257", RawFont Helvetica},
        Entry {Char "s", MapsTo "8#373", RawFont Helvetica},
        Entry {Char "A", MapsTo "8#341", RawFont Helvetica},
        Entry {Char "a", MapsTo "8#361", RawFont Helvetica},
        Entry {Char "E", MapsTo "8#352", RawFont Helvetica},
        Entry {Char "e", MapsTo "8#372", RawFont Helvetica},
        Entry {Char "C", MapsTo "8#246", RawFont Helvetica},
        Entry {Char "U", MapsTo "8#241", RawFont Symbol},
        Entry {Char "X", MapsTo "8#353", RawFont Helvetica},
        Entry {Char "x", MapsTo "8#343", RawFont Helvetica},
        Entry {Char "i", MapsTo "8#365", RawFont Helvetica},
        Entry {Char "L", MapsTo "8#350", RawFont Helvetica},
        Entry {Char "l", MapsTo "8#370", RawFont Helvetica},
        Entry {Char "O", MapsTo "8#351", RawFont Helvetica},
        Entry {Char "o", MapsTo "8#371", RawFont Helvetica},
        Entry {Char "c", MapsTo "8#242", RawFont Helvetica},
        Entry {Char "$", MapsTo "8#243", RawFont Helvetica},
        Entry {Char "/", MapsTo "8#244", RawFont Helvetica},
        Entry {Char "Y", MapsTo "8#245", RawFont Helvetica},
        Entry {Char "d", MapsTo "8#262", RawFont Helvetica},
        Entry {Char "D", MapsTo "8#263", RawFont Helvetica},
        Entry {Char "P", MapsTo "8#266", RawFont Helvetica},
        Entry {Char "S", MapsTo "8#247", RawFont Helvetica},
        Entry {Char "%", MapsTo "8#275", RawFont Helvetica},
        Entry {Char "*", MapsTo "8#250", RawFont Helvetica},
        Entry {Char "R", MapsTo "8#360", RawFont Symbol},
        Entry {Char "Q", MapsTo "8#272", RawFont Helvetica},
        Entry {Char "q", MapsTo "8#251", RawFont Helvetica},
        Entry {Char "G", MapsTo "8#271", RawFont Helvetica},
        Entry {Char "g", MapsTo "8#252", RawFont Helvetica},
        Entry {Char "M", MapsTo "8#254", RawFont Helvetica},
        Entry {Char "m", MapsTo "8#255", RawFont Helvetica},
        Entry {Char "N", MapsTo "8#253", RawFont Helvetica},
        Entry {Char "n", MapsTo "8#273", RawFont Helvetica},
        Entry {Char ":", MapsTo "8#264", RawFont Helvetica},
        Entry {Char "!", MapsTo "8#241", RawFont Helvetica},
        Entry {Char "?", MapsTo "8#277", RawFont Helvetica},
        Entry {Char "’", MapsTo "8#302", RawFont Helvetica},
        Entry {Char "Z", MapsTo "8#306", RawFont Helvetica},
        Entry {Char "z", MapsTo "8#313", RawFont Helvetica},
        Entry {Char "^", MapsTo "8#303", RawFont Helvetica},
        Entry {Char "w", MapsTo "8#310", RawFont Helvetica},
        Entry {Char "‘", MapsTo "8#301", RawFont Helvetica},
        Entry {Char <">, MapsTo <8#315>, RawFont Helvetica},
        Entry {Char "-", MapsTo "8#305", RawFont Helvetica},
        Entry {Char ".", MapsTo "8#307", RawFont Helvetica},
        Entry {Char "K", MapsTo "8#312", RawFont Helvetica},
        Entry {Char "k", MapsTo "8#316", RawFont Helvetica},
        Entry {Char "~", MapsTo "8#304", RawFont Helvetica},
        Entry {Char "W", MapsTo "8#317", RawFont Helvetica},
        Entry {Char "&", MapsTo "&", RawFont ZapfChanceryMediumItalic},
```

```
Entry {Char "B", MapsTo " ", RawFont Helvetica},
Entry {Char "b", MapsTo " ", RawFont Helvetica},
Entry {Char "H", MapsTo " ", RawFont Helvetica},
Entry {Char "h", MapsTo " ", RawFont Helvetica},
Entry {Char "I", MapsTo "v", RawFont Symbol},
Entry {Char "J", MapsTo " ", RawFont Helvetica},
Entry {Char "j", MapsTo " ", RawFont Helvetica},
Entry {Char "p", MapsTo "j", RawFont Symbol},
Entry {Char "r", MapsTo " ", RawFont Helvetica},
Entry {Char "T", MapsTo " ", RawFont Helvetica},
Entry {Char "t", MapsTo "J", RawFont Symbol},
Entry {Char "u", MapsTo " ", RawFont Helvetica},
Entry {Char "V", MapsTo " ", RawFont Helvetica},
Entry {Char "v", MapsTo " ", RawFont Helvetica},
Entry {Char "y", MapsTo "V", RawFont Symbol},
Entry {Char "@", MapsTo " ", RawFont Helvetica},
Entry {Char "#", MapsTo " ", RawFont Helvetica},
Entry {Char "(", MapsTo "(", RawFont Helvetica},
Entry {Char ")", MapsTo ")", RawFont Helvetica},
Entry {Char "_", MapsTo " ", RawFont Helvetica},
Entry {Char "+", MapsTo " ", RawFont Helvetica},
Entry {Char "=", MapsTo " ", RawFont Helvetica},
Entry {Char "[", MapsTo "[", RawFont AvantGardeBook},
Entry {Char "]", MapsTo "]", RawFont AvantGardeBook},
Entry {Char "|", MapsTo " ", RawFont Helvetica},
Entry {Char "\", MapsTo " ", RawFont Helvetica},
Entry {Char ";", MapsTo "8#264", RawFont Helvetica},
Entry {Char "{", MapsTo "[", RawFont Symbol},
Entry <Char "}", MapsTo "]", RawFont Symbol>,
Entry {Char "<", MapsTo " ", RawFont Helvetica},
Entry {Char ">", MapsTo " ", RawFont Helvetica},
Entry {Char ",", MapsTo ",", RawFont Helvetica},
Entry {Char "0", Space 1},
Entry {Char "1", Space 2},
Entry {Char "2", Space 4},
Entry {Char "3", Space 8},
Entry {Char "4", Space 16},
Entry {Char "5", Space 32},
Entry {Char "6", Space 64},
Entry {Char "7", Space 128},
Entry {Char "8", Space 256},
Entry {Char "9", Space 512},
Entry {Char "",  MapsTo "",  RawFont Helvetica},
Entry {Char " ", MapsTo " ", RawFont Helvetica}]
```

More ZapfDingbat Symbols

The following table depicts the upper half of the ZapfDingbat character set — the lower half was shown earlier in Table 5. The typecase database file is already available in the Scribe Database so all that is needed are

```
@DefineFont(BodyFont, E[9] (RawFont "ZapfDingbats"))

@Define(E, FaceCode E, TabExport)
```

commands to select these characters with an @**E**(char) environment command.

Note

- Octal codes are **not** given in this table since the typecase file containing the character mapping already exists in Scribe's Database.

---

9. The choice of E is somewhat arbitrary. Remember, facecodes A, B, C, D, F, G, I, J, K, P, R, T, Y, and Z are already assigned.

Table 12

The ASCII character translations to ZapfDingbats symbols

Character	Symbol	Character	Symbol	Character	Symbol
a	➡	A	②	!	❶
b	➢	B	③	@	①
c	➢	C	④	#	❣
d	➤	D	⑤	$	❤
e	➥	E	⑥	%	❥
f	➦	F	⑦	^	➞
g	➧	G	⑧	&	⑇
h	➨	H	⑨	*	♥
i	➩	I	⑩	(	♣
j	➪	J	❶	)	♦
k	➫	K	❷	_	➟
l	➬	L	❸	+	♠
m	➭	M	❹	~	➲
n	➮	N	❺	]	→
o	➯	O	❻	|	➨
p	N/A	P	❼	:	❺
q	➱	Q	❽	"	❢
r	➲	R	❾	}	➩
s	➳	S	❿	<	❼
t	➴	T	➔	>	❾
u	➵	U	→	?	❿
v	➶	V	↔	-	②
w	➷	W	↕	=	❽
x	➸	X	➘	‘	➟
y	➹	Y	➔	[	→
z	→	Z	➘	\	➜
1	⑥	6	❶	;	❻
2	⑦	7	❷	’	❧
3	⑧	8	❸	{	➔
4	⑨	9	❹	,	①
5	⑩	0	⑤	.	③
				/	④

Table 13

The character translations to publication symbols

Character	@y() Symbol	Description
	in Helvetica only	
B	•	40%* bullet
C	©	Copyright mark
M	—	Em dash
N	–	En dash
	and also in Times-Roman	
R	®	Registration mark
T	™	Trademark symbol

PostScript Fonts

In addition to Scribe's normally available running text fonts (Helvetica, Times-Roman, and Courier) and the modifying @b(), @i(), and @p() environments, the following fonts are available by using the @SpecialFont() command.[10]

Note

- A line each of Helvetica and Times-Roman is included for comparison.
- The paragraph labels are the Scribe descriptor spellings for the @SpecialFont commands while the hyphenated forms are the PostScript names.
- Also, the Scribe descriptors only apply to the roman facecode, not the entire font family. For example, defining a special font as **`@SpecialFont(F1 "AvantGardeBook")` only** allows you to use the roman facecode, not the bold, italic, or bolditalic ones. The way around this restriction is to define an environment with the **`Font FontName`** attribute (for a depiction of all the facecodes see item 35 in Appendix Section A.1).
- One other font (ZapfChancery-MediumItalic) is available from PostScript for the Apple LaserWriterPlus™ but is not in the Scribe database. The procedure for obtaining it is to copy the following file **/../RV/USR/MKB/LIB/ZAPFCH.RAW**[11] to your local directory (or another directory and invoke Scribe's @Use(DataBase) command) — then the ZapfChancery-MediumItalic font can be selected with the @SpecialFont command as per above.
- Note that the ZapfChancery-MediumItalic font is two sizes smaller than the default font, and that this is the **only** typeface available.
- Two different fonts (LubalinGraph and Souvenir) are available on the DEC ScriptPrinter™ which replace the Bookman and Palatino font families.

---

*A 40% bullet is 40% of the nominal character height, i.e. 4 points.

10. See *Scribe User Manual*, 4th ed., 2nd printing, page 177.

11. Thanks to Mike Blackwell, Carnegie Mellon University, Robotics Institute

Font Examples

AvantGardeBook

This text is written in AvantGarde-Book font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

BookmanLight

This text is written in Bookman-Light font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

HelveticaNarrow

This text is written in Helvetica-Narrow font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

NewCenturySchlbkRoman

This text is written in NewCenturySchlbk-Roman font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

PalatinoRoman

This text is written in Palatino-Roman font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

ZapfChanceryMediumItalic

This text is written in ZapfChancery-MediumItalic font -
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

Helvetica

abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

TimesRoman

abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

New Scribe Symbols, TextForms, and Forms

The following forms and environments have been designed to supplement those available from Scribe, with one exception which will be treated first. The exception, **@DoubleBox**, is an environment that is described in the database but not mentioned in the manual. These new forms and environments can be either copied to the top of a manuscript file or placed in a separate **<FILENAME.LIB>** file for inclusion with an @Libraryfile command. The easiest approach for the new mathematics forms is to copy Scribe's Mathematics10 Database file to your own directory, rename it for inclusion with an @LibraryFile command, and then append[12] and integrate the new items. You

12. There are some procedures to follow when doing this — merely appending new forms and environments **won't** work.

can then include Scribe's Mathematics10 (or 12) facility with an @LibraryFile command in your own mathematics file to keep current on any Scribe changes as well. A listing of the Senslab **MATHSL.LIB** database file is in Appendix C.

The @DoubleBox Environment

The @DoubleBox environment draws a closely-spaced double-lined box around text in the same manner as @Box.

This text is enclosed in a double box.

Since the line drawn by @DoubleBox is in reality a single wide black line overlaid with a narrow white line, its appearance can be changed somewhat by reDefining (or Modifying) the Scribe **StandardDoubleLineType** attributes as follows:

- **Current Scribe Definition**

```
@DefineLineType(StandardDoubleLineType, Weight 150, Groove 50)
```

- **Revised Scribe Definition**

```
@DefineLineType(StandardDoubleLineType, Weight 300, Groove 100)
```

This revision gives a new @DoubleBox:

This text is enclosed in the new double box.

The environment can be nested and the margins changed similarly to the @Box one.[13]

CommandStrings, TextForms, and Forms

Scribe's database uses three methods to derive its mathematical forms:

CommandString	a form consisting of a codeword and its definition and used without delimiters
TextForm	a form consisting of a codeword and its definition and generally used with delimited text for single-line expressions
Form	a form consisting of a codeword and its definition and used with delimited text and requiring delimited strings for multi-line expressions

Note

- All of these forms are designed for with either the @Math or @MathDisplay environments — their appearance in running text will probably not be as desired. They can be redefined for running text usage.
- These forms are also size dependent in that they must be redefined for point sizes other

---

13. See the *Scribe User Manual Supplement for Version 5.*, pp 3–5.

than 10. This is true of the mathematics library files in general — a separate library file exists for 8, 10, and 12 point mathematics in PostScript.

Single-Line Forms Update

The following Scribe mathematics facilty single-line forms were previously listed either as "fakes" or "blanks" in the Scribe Manual. Those that appear in the tables below with definitions are now accessable and can be placed either at the beginning of the manuscript or in a modified mathematics library. Most of the updated form definitions require a lot of Scribe's positioning commands and can be defined in many other ways as well.

These single-line forms are created as CommandStrings, e.g.:

```
@CommandString[Add      "#@JSym(+)#"]
```

where the term following the square bracket is defined as the expression in quotes (the octothorps create small spaces in the math environment)

Single-Line Forms: Fakes

The following group of characters replaces the "fake" labelled characters on pages 114 and 115 of the *Scribe User Manual* and are designed to emulate the original Dover characters as closely as possible.

Table 14

Single-line forms and definitions to replace the "fake" characters
in Scribe's database

Command	Output	Definition
@CSet	$\mathbb{C}$	**@Begin(T, Size +2)C @End(T) @Hsp(-5.5pt) @~* @Begin(R, Script +2pt, Size -4) @VBar@End(R)**
@NSet	$\mathbb{N}$	**@Begin(T, Size +2)N @End(T) @Hsp(-7.5pt) @~ @Begin(R, Script +1.4pt, Size -2) @VBar@End(R)**
@QSet	$\mathbb{Q}$	**@Begin(T, Size +2)C @End(T) @Hsp(-5.5pt) @~ @Begin(R, Script +2pt, Size -4) @VBar@End(R)**
@RSet	$\mathbb{R}$	**@Begin(T, Size +2)R @End(T) @Hsp(-7.5pt) @~ @Begin(R, Script +1.4pt, Size -2) @VBar@End(R)**
@ZSet	$\mathbb{Z}$	**@F3(Z) @Hsp(-3pt) @F3(Z)**

Single-Line Forms: Blanks

The following characters replace those listed as "blanks" on pages 114 and 115 of the *Scribe User Manual* and are designed to appear like those listed in the *Scribe Pocket Reference*.

Table 15

Single-line forms and definitions to replace those listed as
"blanks" in Scribe's database

Most of the definitions in this table, as well as the previous one, are simple CommandStrings — the @Qed, @SqInter, and @SqUnion symbols also require the Box definitions. The use of modified boxes makes it easy to change the size and position of these three symbols by merely changing the BoxMargin distances.

Command	Output	Definition
@Circ	$A \circ B$	**# @Begin(R, Script -3pt, Size +2) @Jsym(N) @End(R) #**
@GtLt	$A \gtrless B$	**# @Begin(R, Script -1pt) @Begin(R, Script +4pt) @~ > @End(R) @Hsp(-6pt) < @End(R) #**
@HBar	h	**h@Begin(R, Script +5pt, Slant +75, Size 1) @~ @Hsp(-2pt) | @End(R) #**
@LtGt	$A \lessgtr B$	**# @Begin(R, Script -1pt) @Begin(R, Script +4pt) @~ < @End(R) @Hsp(-6pt) > @End(R) #**
@Mp	$A \mp B$	**@Add @Begin(R, Script +3.5pt) @~ @Hsp(-7pt) @Jsym(-) @End(R)**
@ODiv	$A \oslash B$	**# @Ovp[@Begin(R, Size -1) @F3(O) @End(R)] @~ @Hsp(+3.5pt) @~ @Begin(R, Script +1pt, Size -2) @Jsym(B) @End(R) ###**

... continued on following page

*The @~ is a Scribe command to ignore the blank spaces between it and the next printing character. It is used here to split the TextForm command into multiple lines for clarity.

Command	Output	Definition

@ODot	$A \odot B$	`# @Ovp[@Begin(R, Size -1) @F3(O) @End(R)] @~`
		`@Hsp(+3.3pt) @~`
		`@Begin(R, Script +2.5pt, Size 3) @Bullet @End(R) ###`

@OMinus	$A \ominus B$	`# @Begin(R, Size -1) @F3(O) @End(R) @Hsp(-8pt) @~`
		`@Begin(Transparent, Script +0.75pt) @~`
		`@Sub @End(Transparent) #`

@Qed	$A \square B$	`@DefineBox(QedBox, All StandardLinetype)`
		`@Define(QedBoxMargins,`
		`  BoxTM 8pt, BoxBM 0pt, BoxLM 7pt, BoxRM 7pt)`
		`@Define(Qedsym, Boxtype QedBox, Use QedBoxMargins)`
		`@CommandString[Qed "#@Qedsym()##"]`

@SqInter	$A \sqcup B$	`@DefineBox(SqIBox, Left StandardLinetype,`
		`  Bottom StandardLinetype,`
		`  Right StandardLinetype)`
		`@Define(SqIBoxMargins,`
		`  BoxTM 8pt, BoxBM -0.5pt, BoxLM 7pt, BoxRM 7pt)`
		`@Define(SqInt, Boxtype SqIBox, Use SqIBoxMargins)`
		`@CommandString[SqInter "#@SqInt()##"]`

@SqUnion	$A \sqcap B$	`@DefineBox(SqUBox, Left StandardLinetype,`
		`  Top StandardLinetype,`
		`  Right StandardLinetype)`
		`@Define(SqUBoxMargins,`
		`  BoxTM 8pt, BoxBM 0pt, BoxLM 7pt, BoxRM 7pt)`
		`@Define(SqUnt, Boxtype SqUBox, Use SqUBoxMargins)`
		`@CommandString[SqUnion "#@SqUnt()##"]`

@Top	$A \top B$	`@Begin(Transparent, Script +4.5pt) @~`
		`@Begin(Y, Size -3) M @End(Y) @~`
		`@End(Transparent) @~`
		`@Begin(R, Script +1.25pt, Size -2) @~`
		`@Hsp(-4.25pt) @VBar @End(R) ##`

@UPlus	$A \uplus B$	`# @F3(U) @Hsp(-6pt) @~`
		`@Begin(Transparent, Script +1pt, Size -5) @~`
		`@Add @End(Transparent) ##`

New Mathematical CommandStrings

Two new CommandStrings have been designed:

1. A slashed zero
2. The symbol for geometrical homothesis

The Slashed Zero CommandString

The slashed zero can be designed using the Scribe @Hsp and @Ovp commands but is awkward when used in substantial amounts. It becomes much easier to use when embedded in a Scribe CommandString.

Slashed Zero Definition and Usage

The slashed zero is created with the following CommandString:

```
@CommandString[0="@Ovp(0)@Hsp(+1pt)/#"]
```

and is used in the following manner:

Manuscript Form:

```
22@0.8@0 35
```

Document Form:

220.8035

The Homothesis Form

Referring to Figure 1 below, the two sides AB and DE are homothetic (similar **and** parallel). The similar and parallel aspects are used in the codeword.

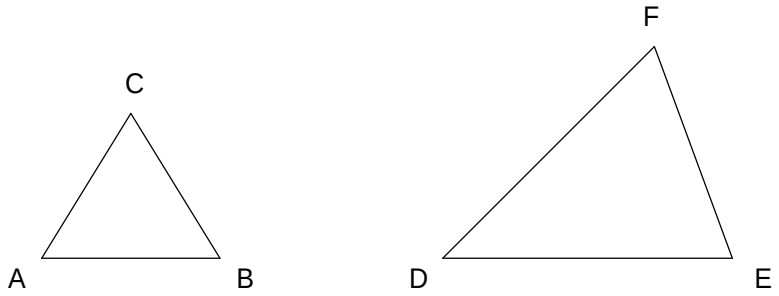


Figure 1. 2-D HOMOTHETIC FIGURES

Side AB of triangle ABC is homothetic with side DE of triangle DEF.

Homotheity can also be extended into three dimensions. The two ellipses in Figure 2 are created by two parallel planes cutting the cone. Ellipse A is therefore homothetic with ellipse B.

Homothesis Form Definition and Usage

The homothesis form [1] is created with the following CommandString:

```
@CommandString[SimPar="#@Ovp(@JSym(~))@Hsp(+1pt)@R(:)##"]
```

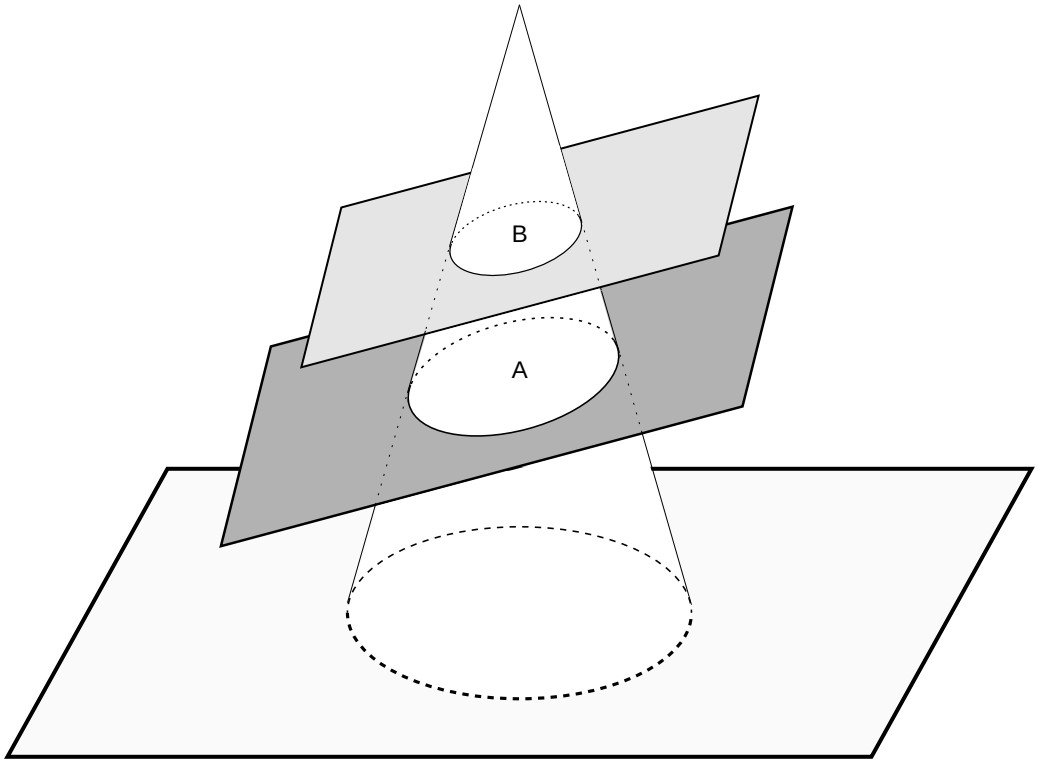


Figure 2. 3-D HOMOTHETIC FIGURES

Ellipses A and B have homotheticity.

and is used in the following manner:

Manuscript Form:

`In these triangles AB@Math(@SimPar)DE.`

Document Form:

In these triangles AB ∻ DE

New Textual CommandString

As mentioned earlier, a commandstring for the Mexican peso has been designed.

@Peso Definition and Usage

The definition of the peso commandstring is:

`@CommandString[Peso "P@Hsp(-7pt)@Begin(JSym, Script +3.5pt, Size -2)b@End(JSym)"]`

and it is used as follows:

```
That would cost @Peso 345,000 in Guadalajara.
```

to give:

That would cost ₱ 345,000 in Guadalajara.

New Single-Line TextForms

So far, two new single-line forms have been created:

1. An overword TextForm which extends the @OverLine TextForm to multiple characters.
2. An overword TextForm which extends the @OverLineCap TextForm to multiple characters.

The OverWordCap TextForm

The @OverWordCap TextForm was created to enable the printing of overlined multicharacter expressions and will for the most part supersede the Scribe @OverLineCap TextForm as it can be used in any environment and does not affect intercharacter spacing as does @OverLineCap.

OverWordCap TextForm Definition and Usage

It is defined as follows:

```
@TextForm[OverWordCap=@~
"@Begin(MathemArea)@MathBar@Parm(Text)@End(MathemArea)"]
```

and is used similarly to the @OverLineCap command:

Manuscript Form: Running Text

```
The perimeter of figure @OverWordCap(ABCD) is given as ...
```

Document Form: Running Text

The perimeter of figure $\overline{ABCD}$ is given as ...

Manuscript Form: @MathDisplay

```
@Sr(The perimeter of figure )@OverWordCap(ABCD)@Sr( is given as ...)
```

Document Form: @MathDisplay

The perimeter of figure $\overline{ABCD}$ is given as ...

The @OverWord TextForm

The @OverWord TextForm is actually more of an environment (since it is presently a modified @Box) but is included here because of its similarity to the @OverWordCap TextForm in usage and appearance. It was designed (again, because of Scribe's @OverLine TextForm inadequacies) to overline multiple lower-case character expressions. Notice that the @OverWord line is lighter than the @OverWordCap to match Scribe's @OverLine. Both the thickness of the line and its height above the characters can be easily changed via the **Weight** and **BoxTM** attributes in the definition below.

OverWord TextForm Definition and Usage.

```
@DefineBox(OverBox, Top StandardLineType)
@DefineLineType(StandardLineType, Weight 25)
@Define(OverWord, BoxType OverBox, BoxTM 1pt)
```

Manuscript Form: Running Text

```
The following @OverWord(words) @OverWord(are) @OverWord(@OverWorded).
```

Document Form: Running Text

The following $\overline{\text{words}}$ $\overline{\text{are}}$ $\overline{\text{@OverWorded}}$.

Manuscript Form: @MathDisplay

```
@Sr(The least upper limit )@OverWord(lim)@Sr( can be expressed as @LDots )
```

Document Form: @MathDisplay

The least upper limit $\overline{lim}$ can be expressed as . . .

The Universal Form

The Universal Form @VSym was designed to create an unlimited variety of scaled and positioned symbols, characters, and text using a three parameter Form. At present, the @VSym Form is separate from the mathematics facility and is resident in its own @LibraryFile — it can easily be included in a mathematics file if desired. The complete LibraryFile (**UNIVER.LIB**) is included here as an example.

@VSym Form Definition and Usage

```
@Marker(Library, Universal, PostScript, ScaleableLaser)

@Comment[Copyright 10/88, Alan D. Guisewite]

@Form[VSym     (@Univ<Script @ParmQuote 'Script, Default (0)',
                      Size @ParmQuote 'Size, Default (10)',
                      Text @ParmQuote 'Text, Default ()'>)]

@Form[Univ     "@Begin(MathemArea,
                      Centered,
                      Script @Parm(Script),
                      Size @Parm(Size))
                      @Parm(Text)
                      @End(MathemArea)"]
```

and is used in the following manner:

Manuscript Form:

```
This is a @VSym(Script "-16pt", Size "+24", Text "*").
```

Document Form:

This is a $*$.

Multi-Line Forms

The following two multi-line forms have been created:

1. The @Int form has been modified to create a loop or closed integral form.
2. A nested sum form [2] for concatenation of series sum expressions.[14]

Both forms exist in the @Math and @MathDisplay environments.

The Loop Integral Form

A loop integral can be created using in either of two ways

1. A CommandString form (used by the Mathematics library for its single-line forms) which is prefixed to the @Int form
2. A modification to the @Int form creating a new @LoopInt form

The @LInt CommandString

The easier to implement method is to create a Scribe overprint "loop" TextForm and then prefix it to the standard @Int form. This, however, requires two different TextForms, one for each of the @Math and @MathDisplay environments. Also, since there is no reasonably-sized circle readily available from the Mathematics LibraryFile and the Helvetica lower-case "O" is not a circle, part of the TextForm uses a lower-case "O" from the AvantGardeBook font. Therefore, it is necessary to include a @SpecialFont command at the top of the manuscript to call the AvantGarde font.[15]

This procedure creates a loop overprint and then combines it with Scribe's @Int form in a CommandString. The resultant loop integral is then used in the same way as the @Int form.

---

14. An example (the basic form of the power series):

$$\sum_{k=0}^{n} \overline{\beta}\, a_k x^k = a_0 \pm a_1 x + a_2 x^2 \pm a_3 x^3 + \cdots + \overline{\alpha}\, a_n x^n$$

which reduces to

$$\sum_{k=0}^{n} \overline{\beta}\, a_k x^k = a_0 (1 \pm b_1 x (1 \pm b_2 x (1 \pm b_3 x (1 \pm \cdots \pm b_n x))))$$

which then equals

$$a_0 \bigwedge_{k=1}^{n} [1 \pm b_k x] \text{ where } b_k = \frac{a_k}{a_{k-1}}$$

15. The Scribe @Circ command could be used although it would require not only the use of the @Hsp and @Ovp commands but the **Size** attribute as well.

For the @Math Environment

```
@CommandString[LIntM "@Begin(Transparent, Script -1.5pt)@~
                @Ovp(@F3(o))@Hsp(+0.5pt)@End(Transparent)"]
```

For the @MathDisplay Environment

```
@CommandString[LIntD "@Begin(Transparent)@Ovp(@F3(o))@End(Transparent)"]
```

and are used so:

Manuscript Form: @Math

```
@LIntm(From "-@Infty", To "+@Infty")
```

Manuscript Form: @MathDisplay

```
@LIntd(From "-@Infty", To "+@Infty")
```

to give:

Document Form: @Math

$$\oint_{-\infty}^{+\infty}$$

Document Form: @MathDisplay

$$\oint_{-\infty}^{+\infty}$$

The @LoopInt Form

The second approach embeds the standard @Int form inside a new @LoopInt form. This is somewhat more involved but has the advantage of being a form in itself and thus easier to use. It does, however, entail the creation of a separate Mathematics Library which then has to be invoked at the beginning of the manuscript as well as Scribe's Mathematics10 or 12. The file containing these new Scribe Forms is listed in Appendix C and can be copied as is (with an @Marker name change for protection reasons). Both environments must then be invoked when forms from both are needed which causes indentation problems unless two different @MathDisplay forms are created, one for use with Scribe's Mathematics10 Library and the other for standalone requirements.

The foregoing problems are the primary reasons for copying Scribe's Mathematics10 Library (from **/../../USR/MISC/.SCRIBE/LIB/DATABASE.V6/MATHEM.LIB**) to your own directory, renaming it, and consequently using it as your @LibraryFile mathematics database.

The @LoopInt form is used in the same way as the @Int:

Note

- The loop in the @Math version has been dropped slighly below the text baseline in order to be symmetric about the integral sign.

Manuscript Form:

```
@LoopInt(From "@C(S)", To "") (@G(a)@Nabla@G(b)) : d@B(@R(S))@ = @Int(From
"@C(V)", To "")@Nabla:(@G(a)@Nabla@G(b)) dV
```

Document Form: @Math

$\oint_S (\alpha\nabla\beta) \cdot d\mathbf{S} = \int_V \nabla \cdot (\alpha\nabla\beta)\, dV$

Document Form: @MathDisplay

$$\oint_S (\alpha\nabla\beta) \cdot d\mathbf{S} = \int_V \nabla \cdot (\alpha\nabla\beta)\, dV$$

The Nested Sum Form

The nested sum form, @NestSum, has been created in the same way as the @LoopInt Form and its definition is also in Appendix C. The form appears as follows in the manuscript.

Manuscript Form:

```
@NestSum(From "n=1", To "m") n
```

Document Form: @Math

$\Lambda_{n=1}^{m} n$

Document Form: @MathDisplay

$$\bigwedge_{n=1}^{m} n$$

The Evaluate Environment

The @Evaluate environment extends the Scribe @VBar command for use with multi-line forms. An auxiliary definition is included to place the evaluant at the correct position.

@Eval Definition

```
  @DefineBox(EvalBox, Right StandardLineType)
@Define(Eval Box, BoxType EvalBox, BoxRM 5pt, AfterExit "@ ")[16]
@Define(DDown Down, Script -1line)
```

@Eval Examples

Since the @Evaluate environment is a modified @Box, the usage is the same:

Manuscript Form:

```
@Eval[@Int(From "-@G(p)", To "+@G(p)")@ @Cos@G(w)]@DDown(@G(p)/4)@Quad =
```

Document Form: @Math

$$\int_{-\pi}^{+\pi} \cos\omega \Big|_{\pi/4} =$$

Document Form: @MathDisplay

$$\int_{-\pi}^{+\pi} \cos\omega \Big|_{\pi/4} =$$

Introduction to the @Determ and @Matrix Environments

The following two environments are, like @Eval, based on modified @Box and @DoubleBox environments. Many different methods can be used to create determinants and matrices — the simplest ones (in terms of Scribe definitions and procedures) are also, unfortunately, the most cumbersome to use and therefore will not be covered here.[17] The following sections will describe determinants and matrices that use two different approaches depending upon the number and order of these environments that are needed

small number	Scribe user-defined table and box definitions entered during manuscript generation
large number	modified Scribe standard table which is then called with an @LibraryFile command

The symbol for the matrix environment is a standard one [3] although not the usual large square brackets.

---

16. The **AfterExit** attribute simply means that Scribe places the quoted characters after the environment (in this case, the line) and before the following text.

17. The intuitively easiest methods require megadoses of Scribe's @_ to maintain proper internal spacing and generally require multiple compilations to achieve correct results.

Note

- These two environments are currently defined only as mathematical inserts and therefore appear in the manuscript the same way as Scribe's other multi-line expressions.
- As can be seen from the definitions and examples the inserts are created with two extra columns — one before and one after the insert itself. These extra columns allow for the inclusion of mathematical expressions or additional inserts.
- In both of the following methods the **BoxLM** and **BoxRM** box margin attributes have been increased from the standard 4 point distance to 10 point to improve readability. Also, the spread has been increased to 1.1 lines to match the remaining TechReport text.

Determinant and Matrix Environment Local Definitions

For a small number of these environments in a given document the following definition gives a 3×3 determinant:

```
@DefineLineType(Determ,         Weight 72)
@Define(Element         Math,   Centered)
```

Note

- The above two commands only need to be defined once per manuscript and the **Element** definition is good for both the determinant and matrix environments.

```
@DefineRowFormat(Determ3,
    Columns (Column Element, LineType Determ,
        Column Element, Column Element, Column Element,
        LineType Determ, Column Element))
@Begin(MathDisplay,     TableColumns Determ3,
                        BoxLM 10pt, BoxRM 10pt, Spread 1.1)
```

Note

- These two statements (and an **@End(MathDisplay)** closing) are necessary for each determinant or matrix — the **LineType** command separates the pre- and post-text expressions from the insert itself. To change the size of the insert merely add or subtract **Column Element** definitions to the group between the **LineType** commands.

---

and a 3×3 matrix:

```
@DefineLineType(Matrix,         Weight 400, Groove 200)
```

Note

- **The above command creates the matrix double line separator and only needs to be defined once per manuscript.**

```
@Define(Element        Math,   Centered)
```

Note

- **See comment above.**

```
@DefineRowFormat(Matrix3,
    Columns (Column Element, LineType Matrix,
        Column Element, Column Element, Column Element,
        LineType Matrix, Column Element))
@Begin(MathDisplay,    TableColumns Matrix3,
                       BoxLM 10pt, BoxRM 10pt, Spread 1.1)
```

Note

- **Again, the above two statements and an @End(MathDisplay) are needed for each matrix.**

Local Determinant Example

The following example shows the creation of a 3×3 determinant using the local definitions stated earlier along with Scribe's standard table commands.

Manuscript Form: @MathDisplay

```
@DefineRowFormat(Determ3,
    Columns (Column Element, LineType Determ,
        Column Element, Column Element, Column Element,
        LineType Determ, Column Element))
@Begin(MathDisplay,    TableColumns Determ3,
                       BoxLM 10pt, BoxRM 10pt, Spread 1.1)
@TableID(LocalDet)
@\2@\15@\1@\

A@ =@\10@\4@\3@\@ =

@\3@\12@\9@\
@End(MathDisplay)
```

which appears as

Document Form: @MathDisplay

$$A = \begin{vmatrix} 2 & 15 & 1 \\ 10 & 4 & 3 \\ 3 & 12 & 9 \end{vmatrix} =$$

and

Local Matrix Example

The example below describes the formation of a 3×3 matrix using the definitions shown above.

Manuscript Form: @MathDisplay

```
@DefineRowFormat(Matrix3,
    Columns (Column Element, LineType Matrix,
        Column Element, Column Element, Column Element,
        LineType Matrix, Column Element))
@Begin(MathDisplay,    TableColumns Matrix3,
                       BoxLM 10pt, BoxRM 10pt, Spread 1.1)
@TableID(LocalMat)
@\2@\15@\1@\

B@ =@\10@\4@\3@\@ =

@\3@\12@\9@\
@End(MathDisplay)
```

Document Form: @MathDisplay

$$B = \begin{Vmatrix} 2 & 15 & 1 \\ 10 & 4 & 3 \\ 3 & 12 & 9 \end{Vmatrix} =$$

in the manuscript.[18]Athough the number of columns can be changed by adding or subtracting **Column Element** attributes in the **@DefineRowFormat** command and renaming it (e.g. from **Determ3** to **Determ5**), for a document with a large number of different sized determinants or matrices it is easier to use a modified Scribe standard table.

Determinant and Matrix Environment LibraryFile Definitions

This approach uses Scribe's standard tables (modified and renamed as separate LibraryFiles) to obtain the inserts. These are presently available as part of the **NTABLE.LIB** libraryfile (see Appendix D), and are very easy to use.

18. The **MathDisplay** environment can be changed, of course, as needed.

A 3 × 3 Determinant Example Using DETERM.LIB LibraryFile

Manuscript Form: @MathDisplay

```
@DetermTable(Name LibDet,      Columns 5)
@Begin(LibDet)
@TableID(LibDet)

@\3@\7@\1@\

A@-(1) =@\1@\1@\2@\ =
```
[19]
```
@\5@\2@\8@\

@End(LibDet)
```

which appears as

Document Form: @MathDisplay

$$A_1 = \begin{vmatrix} 3 & 7 & 1 \\ 1 & 1 & 2 \\ 5 & 2 & 8 \end{vmatrix} =$$

A 3 × 3 Matrix Example Using DETERM.LIB LibraryFile

Manuscript Form: @MathDisplay

```
@MatrixTable(Name LibMat,      Columns 5)
@Begin(LibMat)
@TableID(LibMat)

@\2@\6@\2@\

A@-(2) =@\1@\1@\2@\ =

@\5@\2@\1@\
@End(LibMat)
```

which appears as

Document Form: @MathDisplay

$$A_2 = \begin{Vmatrix} 2 & 6 & 2 \\ 1 & 1 & 2 \\ 5 & 2 & 1 \end{Vmatrix} =$$

19. The first and last columns are defined (with the attribute **Spaces Kept**) so that they can be written as normal text.

At present the library file will only allow up to 8-column determinants and matrices (10-column without pre- and post-text expressions).

The CHEMIS.LIB Structural Chemical Formula Database File

The Senslab Chemistry library file allows the insertion (using a modified Scribe **format** environment) of linear or branched structural formulas. The database also includes a few relevant commandstrings as well.[20] Since a structural formula is displayed in text the same way as a mathematical one, the insert was designed as a @ChemDisplay environment (similar to the @MathDisplay but with fixed-width characters and **kept** spaces).

CHEMIS.LIB Definition

The complete **CHEMIS.LIB** database file is shown here:

```
@Marker(Library, Chemistry, PostScript, ScaleableLaser)

@Define(ChemDisplay,
                Use Format,
                AfterEntry "@TabDivide(20)",
                ScriptPush off,
                Above 0.5,
                Below 0.5,
                Centered,
                Spacing 1.1)

@CommandString[

        Vb      "@Hsp(+3pt)@Begin(JSym, Script +1pt, Size +2)|@End(JSym)",
        Sbond   "@Begin(Y, Script +1pt)M@End(Y)",
        Dbond   "@Y(M)@Hsp(-10pt)@Begin(Y, Script +2pt)M@End(Y)",
        Tbond   "@Begin(Y, Script -1pt)M@End(Y)@~
                 @Hsp(-10pt)@Begin(Y, Script +1pt)M@End(Y)@~
                 @Hsp(-10pt)@Begin(Y, Script +3pt)M@End(Y)"]

@TextForm[

        Fwd     "@Ovp<@Begin(JSym, Script +1.5pt)b@Hsp(-2pt)L@End(JSym)>@~
                 @Begin(F2, Script +10pt, Size -2)@~
                 @Parm(text)@~
                 @End(F2)",

        Rev     "@Ovp<@Begin(JSym, Script +1.5pt)b@Hsp(-2pt)J@End(JSym)>@~
                 @Begin(R, Script +10pt, Size -2)@~
                 @Parm(text)@~
                 @End(R)"]
```

@ChemDisplay Example

For example, the following (1-chloropropene) formula would be entered as:

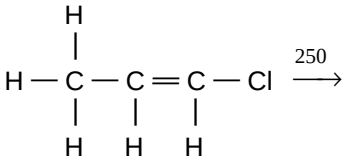
```
@ChemDisplay[

         @\H
```

20. This file is one of the most recently developed ones and is by no means complete.

```
          @\@Vb
 @\H @SBond C @SBond C @DBond C @SBond Cl   @Fwd{250}
          @\@Vb      @\@Vb      @\@Vb
         @\H      @\H      @\H
```

]

and would appear as:



in the document.

Modified Scribe Environmental Database Files

The libraryfiles mentioned earlier (**NEWACC.LIB**, **NEWSYM.TYP**, and **MATHSL.LIB**) were redesiged to handle the full PostScript range of accessible characters, symbols, and fonts as well as to add to Scribe's mathematics facility. Some of the other Scribe database (environmental) files have been copied and modified as part of the @TechReport document design procedure and other database files have been created to improve document production. They are

- a modified standardtable libraryfile to augment the Scribe version
- a modified figures libraryfile which improves captioning
- a modified titlepage libraryfile to match CMU's Technical Report style

The NTABLE.LIB Table Database File

The original Scribe **STABLE.LIB** libraryfile is limited to only 10 columns. While modifying that file to increase the number of columns (to **24**) an additional parameter in the **@StandardTable** command was added, a **LastColumn** name which is used so:

```
@StandardTable(Name nap2,
               Columns 4,
               FirstColumnWidth .75",
               OtherColumnWidth 1.35",
               LastColumnWidth .75",
               AllColumns Center,
               Flushtop,
               HeadingBoxed)
```

This makes it easier to produce tables since there is now control over both the first and last column parameters.[21] The determinant and matrix tables are also included in this file. A listing of **NTABLE.LIB** is in Appendix D.

The TFIGUR.LIB Figure Environment Database File

Scribe's **FIGURE.LIB** database file contains the figure and table environment[22] parameters and has been modified to improve labeling, captioning, and overall appearance. The modified file is named **TFIGUR.LIB** for TechFigure is listed in Appendix E.

---

21. For a two-column table the **OtherColumns** and **OtherColumnWidth** parameters no longer apply.

22. Table environment here and in this subsection refers to the text surrounding the figure and table inserts — **not** the table itself, which is controlled by **NTABLE.LIB** or **STABLE.LIB**.

Improved Features of the TechFigure Library File

The improvements to the TechFigure database file closely follow those suggestions concerning figures and tables in the *Chicago Manual of Style*.

The prelim[23] page headings for figures and tables are now labeled **Illustrations** and **Tables**, respectively, with a subheading under Illustrations called **Figures**.[24] The page heading for the table of contents has also been relabeled to **Contents**.

Figures — Originally, there was only an @Caption command which appeared under the figure in the text and also in the List of Figures in the prelims. The TechFigure database file splits the caption into two new pieces: an @Caption command which sends an abbreviated title to the Illustrations page and an @Legend command for the figure caption itself.

- Also, previously, if the figure caption was longer than one line the remaining line(s) would not be centered. (The remaining line centering procedure would ignore the label "Figure 1:" and center only the text within the caption.} That problem has now been fixed so that every line is centered.
- The captions are displayed in the text as small capitals and both captions and legends are one point size smaller while still arriving in their usual typefaces and sizes on the Illustrations page.

Tables — Since tables only have titles the original @Caption command is again split into a new @Caption and an @TLegend command. The caption again sends a shortened title to the Tables page while the table title in the text is in the @TLegend environment.

- The full title appears **above** the table with the label "Table" and table number centered above it.
- A standard series of footnote symbols is available to avoid the conflict of numerical superscripts with traditional footnote labeling. In order of usage they are ∗, †, ‡, §, ‖, and # with doubling and tripling to create an additional twelve symbols.
- As described above for figures, the title line(s) are now fully centered.
- Also, the titles are one point size smaller than the running text.

Both figures and tables can have longer explanations as well — these are handled by the @LongLegend command, a filled, right-justified doubly-indented environment which is also one point size smaller. There is also an @Credit environment (four point sizes smaller) for acknowledgements. The following is an example of a figure explanantion using the @LongLegend and @Credit environments.

> Total internal reflection leads to an exponentially decaying field which penetrates into the lower index medium. Although the diagram schematically shows discrete reflection points along the surface of an optical fiber, the actual evanescent field along an optical fiber is continuous.
>
> EXPLANATION COURTESY OF AMERICAN BIOTECHNOLOGY LABORATORY, MARCH 1989

23. Prelims are the front matter of a book, before the text itself.

24. Other subheadings would include **Plates** and **Maps**, if needed.

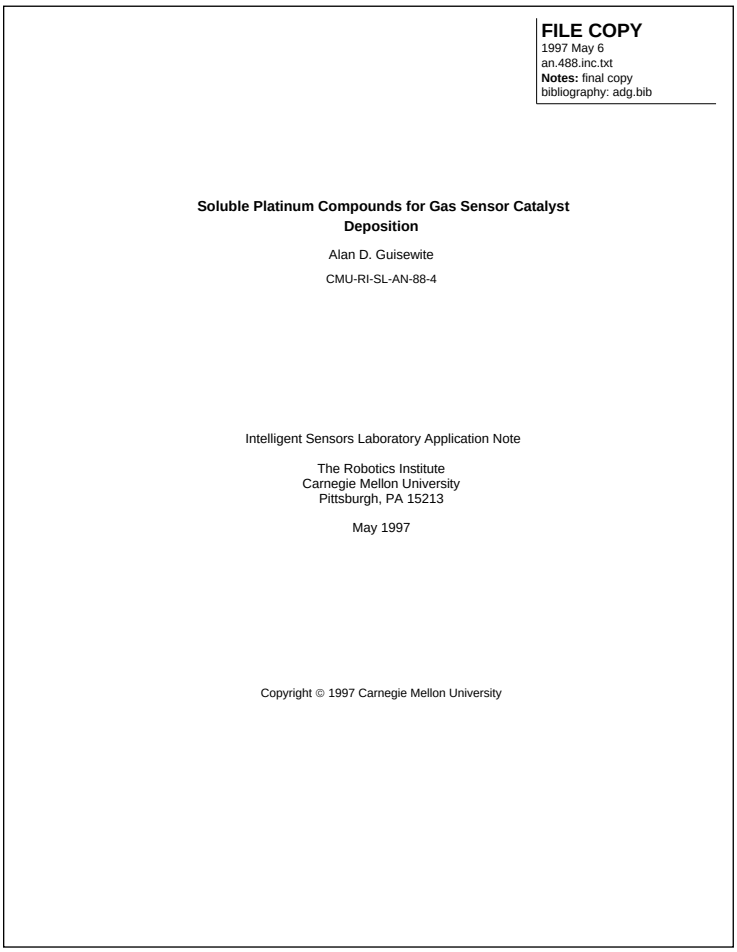
The TECHTI.LIB Title Page Environment Database File

The TechTitle library file has been designed to meet the title page requirements of the Robotics Institute for their Technical Reports. When a document with a title page is produced an index file **FILENAME.INDEX** is generated which contains (as a one-word string) the Technical Report title and number. Since the **TECHTI.LIB** library file is called from within **TECHRE.MAK** however, even if an untitled document is created there will be a message from Scribe stating that an index file has been generated.

Title Page Examples

```
@Make(TechReport)
@Begin(TechTitle)
@Archive(final copy)
@Title(Soluble Platinum Compounds for Gas Sensor Catalyst
 Deposition)
@Author(Alan D. Guisewite)
@Report(ID "SL-AN", # "88-4", Type "Application Note")
@CopyrightNotice(Carnegie Mellon University)
@End(TechTitle)
```

This gives a title page as follows:



FILE COPY
1997 May 6
an.488.inc.txt
Notes: final copy
bibliography: adg.bib

Soluble Platinum Compounds for Gas Sensor Catalyst Deposition

Alan D. Guisewite

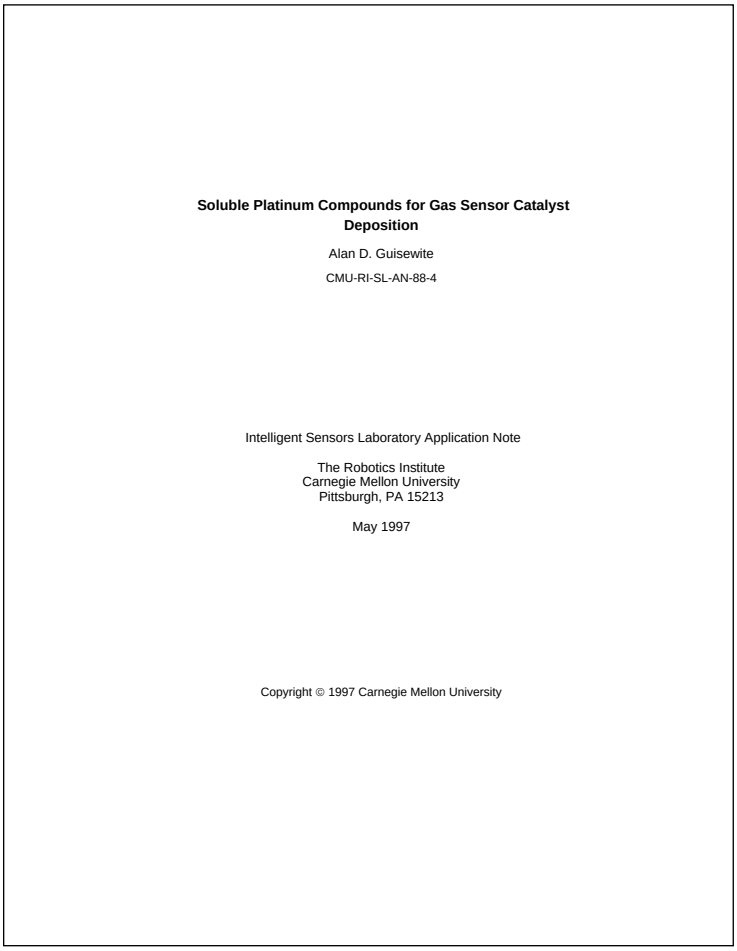
CMU-RI-SL-AN-88-4

Intelligent Sensors Laboratory Application Note

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

May 1997

When making a draft copy the title page becomes:



Soluble Platinum Compounds for Gas Sensor Catalyst Deposition

Alan D. Guisewite

CMU-RI-SL-AN-88-4

Intelligent Sensors Laboratory Application Note

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

May 1997

The @Archive command puts an archival information label in the upper right corner comprising the following;

- the words "FILE COPY"
- the date of the manuscript
- the Scribe **`FullManuscript`** filename
- an area named "**Notes:** for comments which is defined with a **`LongLines Wrap`** attribute

Using the @Archive command with empty brackets leaves the **Notes:** area blank for future comments and leaving the @Archive command out of the manuscript produces a normal title page.

The @Title command is also defined with the **`LongLines Wrap`** attribute so that long titles will still fit within the Technical Report cover window.

In the example above, the @Report command is shown with all of its specifiers — for ordinary Technical Reports only the # parameter is necessary.

Also, even though they are not shown in the example, the **`Copyright`** and **`ResearchCredit`** environments are available.

Miscellaneous New Environments

There are other environments which have been created for use within running text and are resident within **TECHRE.MAK**. These are described (with some examples) in Appendix A.

References

1. beyer, william h., *mathematical symbols and abbreviations,* crc press, inc., west palm beach, fl, 1975, pp. 946.

2. tuma, jan j., *sequences and series,* mcgraw-hill book company, 1979, pp. 102-110, ch. 8.

3. beyer, william h., *mathematical symbols and abbreviations,* crc press, inc., west palm beach, fl, 1975, pp. 949.

Appendix A: The Senslab Techreport Document Definition File

A.1 TECHRE.MAK Summary

This section will describe the salient features of the **TECHRE.MAK** document definition file. The TechReport document thus produced is easy to work with in manuscript as most of the textual partitioning is imbedded in the database. In addition, there exist many enhancements to the usual group of Scribe environments and improved figure and table insertion procedures. Finally, all of the PostScript characters and fonts accessible on a LaserWriter Plus™ printer are made available with single-letter facecodes.

Features of the @Make (TechReport) Document Definition File

The TechReport document type is designed to create a CMU Technical Report (conforming closely to the *Chicago Manual of Style*, as requested) for the Sensors Lab (using **newscribe**[25]) with these characteristics:

1. The manuscript's beginning is uncluttered, requiring only the

   ```
   @Make(TechReport)
   @Use(Database "/usr/senslab/lib")
   ```

   and whatever @Use(Bibliography or other Database) commands are necessary.
   Note: The @Use commands can be inserted in the **TECHRE.MAK** file instead — the only drawback is that Scribe displays all the bibliography files available even if not all of them are used. It may be expedient to concatenate all our **.BIB** files.
2. The document is divided into the following parts.

   a. Title Page
   b. Table of Contents (retitled Contents according to standard practice)
   c. List of Figures (retitled Illustrations according to standard practice)
   d. List of Tables (retitled Tables according to standard practice)
   e. Abstract
   f. MajorPart
   g. Chapter
   h. NewSection
   i. Section
   j. SubSection
   k. SubSubSection or Paragraph
   l. References
   m. Appendix
   n. AppendixSection
   o. Index

3. Parts (g), (h), (k), (l), and (m) are redefined with the Scribe attribute **announced** so that parenthetical pairs will be displayed on the CRT along with the page numbers for diagnostic purposes.
4. Sections, subsections, and subsubsections are also available as UnSection, UnSubsection, and UnSubsubsection for manuscript portions whose titles are **not** sent to the table of contents but are otherwise identical in manuscript appearance.
5. The left margin is indented 1.25" to allow for binding and the line width is correspondingly shortened to 6.25" for a standard 1" right margin.
6. All the document divisions listed above are unnumbered with the exception of the appendices and appendix sections which are lettered A, B, C etc. and A.1, A.2, A.3 etc. respectively.

---

25. Newscribe here refers to Scribe, Version 6 — TechReport will be made compatible with Version 7 when available.

7. The normal usage thus far has been to produce documents only at the Section level or lower (MajorPart and Chapter being reserved primarily for books).
8. By processing the manuscript file with the command

   `newscribe filename.xxx -draft:yes`

   a group of running headers and footers are generated on each page with draft-related information as well as two lines of text under the title box on the title page (one line giving the draft date and time of printing and the second line specifying the complete filename). Draft copies are also double-spaced to facilitate annotation. Because of the increased pagination making a draft version does **not** print out the Contents, Illustrations, or Tables pages. Running newscribe without the command line switch makes a final copy.
9. By processing the manuscript file with the command

   `newscribe filename.xxx -nochangebars`

   any changebars (revision bars) present will not be printed.
10. Three new sectioning commands have been created to eliminate some of the @NewPage commands

    a. @Abstract — places an abstract on a separate page
    b. @NewSection — starts a section of text on a new page
    c. @References — starts the References section on a new page

    Even though the @Abstract and @NewSection have the same effect (their subsequent text sections are on a separate page) they are defined individually so that an Abstract environment can be designed, if needed, with the @Abstract counter intact.
11. The bibliography format is currently a modified (via consensus) CACM in that the citation style is numerical ([n]) and the reference listing is by order of citation. This format is subject to revision and we should consider using the author/date style favored by the *Chicago Manual of Style.* The bibliography reference filename is **NEWCAC.REF**.
12. An index is now available for lengthy TechReports and is in a standard 2-column 8-point (for a 10-point main text) style.
13. The various sectioning commands have been defined with blank lines to eliminate a lot of @BlankSpace separations.
14. The title page has been redesigned (using **TECHTI.LIB**) and an example of its use is shown here:

```
@Make(TechReport)
@Begin(TechTitle)
@Archive(notes)
@ReportTitle(Remote High Speed Measurement[26]
of Small Diameters by
Optical Diffractometry and Interferometry)
@Author(M. W. Siegel)
@Report(# "88-6")
@CopyrightNotice(Carnegie Mellon University)
@End(TechTitle)
@Abstract(Abstract)
@NewSection(Introduction)
```

15. Using the **`TechTitle`** environment automatically generates an index file **FILENAME.INDEX** which lists (as one-word strings)[27] the title and report number. **Note:** If the **`TechTitle`** is not used in the manuscript (e.g., this document), Scribe will still claim the index file is generated — it is a null file and therefore does not exist.

---

26. The command has recently been changed from **`@Title`** to **`@ReportTitle`** because of a possible conflict with Scribe's **`@Title`** command which prints a counter title in the text. For a time there will be a reminder message.

27. A one-word string is the price we pay for doing it automatically.

16. The @Archive command is used to label the upper right hand corner of the title page with archival information comprising

    - the words "FILE COPY"
    - the date
    - the "FullManuscript" filename
    - a **Notes:** area (with automatic line wrap around) for comments

    Using the @Archive command with empty brackets leaves the "notes" area blank — leaving the @Archive command out of the manuscript creates a normal title page. It is suggested that the **FILENAME.BIB** filename be included here until such time as we have a centralized bibliography.
17. The @ReportTitle command automatically wraps long title lines around to a second line to stay within the Technical Report cover window. The above example also shows the use of the @NewSection command to place the Abstract section on a separate page.
18. Note that the @Report line has been designed so that @Make(TechReport) can be used to make other style-related documents such as application notes, etc. The usage for other documents is then

    ```
    @Report(ID "SL", # "88-xx", type "Application Note")
    ```

    This changes the identifying line under the title and adds the argument of the type parameter to the publishing information on the title page.
19. The procedure for labeling figures has been improved.

    - The former @Caption command now comprises an @Caption **and** an @Legend command — the Illustrations list in the prelim should only have brief titles (captions) and not the longer descriptions (legends) of the text figures.
    - Both captions and legends are now truly centered horizontally whereas before captions of more than one line were offset by the length of the word "Figure".
    - The captions of the text figures are in small capitals and both captions and legends are one point size smaller than the running text.
    - The captions arrive in the usual typefaces in the Illustrations list.
    - **Note:** Introductory tags (location descriptors and starting points) should always

        - precede their respective phrases
        - be italicized
        - be followed by a comma unless the phrase is a list in which case a colon

20. Similarly, the labeling procedure for tables has been improved.

    - Tables only have titles and shortened titles appear via the @Caption command in the Tables list.
    - Full titles appear in the text **above** the table with the **@TLegend** command (T for table) with the label "Table " and number centered above it.
    - Only the table numbers appear in the prelim list Tables whereas the labels "Figure " and numbers appear in the Illustrations list since that list also can include plates and maps, etc.
    - The titles are now truly centered horizontally whereas before titles of more than one line were offset by the length of the word "Table".
    - Titles are one point size smaller than the running text.

21. A standard series of footnote symbols is available for tables so as to avoid the conflict of exponential superscripts with traditional footnote labeling. They are (in order of usage) ∗, †, ‡, §, ‖, and # with doubled and tripled symbols used for twelve additional footnotes.
22. There also exists a LongLegend environment for lengthier explanations of both figures and tables — a filled, right-justified doubly-indented text insert. It also is one point size smaller.
23. A Credit environment is available for acknowledging figures if required.
24. The following LibraryFiles are resident within **TECHRE.MAK** and therefore do **not** need to be part of the manuscript file.

    - SansSerif (**SANSSE.LIB**) — along with a BodyStyle attribute defines Helvetica 10 bodyfont
    - MathSL10 (**MATHSL.LIB**) — the Sensor Lab's mathematics facility (includes Scribe's Mathematics10 facility)
    - Abbr1 (**ABBR1.LIB**) — a CMU CSD string file for use with @Value
    - AbbrSL (**ABBRSL.LIB**) — the Sensor Lab's expansion of the above CSD file
    - STable (**STABLE.LIB**) — Scribe's standard table facility
    - NTable (**NTABLE.LIB**) — the Sensor Lab's expansion of the above Scribe database file to include up to 24 columns and will contain **DETERM.LIB** and **MATRIX.LIB** for mathematical determinants and matrices[28]
    - TFigures (**TFIGUR.LIB**) — a replacement for Scribe's **FIGURE.LIB** which renames the prelim's titles (parts b, c, and d in item 2, above} and modifies the figure and table labeling in accordance with items 13 and 14, above
    - TechTitle (**TECHTI.LIB**) — a replacement for Scribe's **TITLEP.LIB** which eases the production of title pages
    - Universal (**UNIVER.LIB**) — the Sensor Lab's universal positioning and scaling command
    - Chemistry (**CHEMIS.LIB**) — the Sensor Lab's database for creating structural chemical formulas in text
    - NewAccents (**NEWACC.LIB**) — the Sensor Lab's database for obtaining the PostScript set of accented characters.

28. When using the **NTABLE.LIB** database file for standard tables, note that an additional parameter has been created, **`LastColumn`**, which must be included in the **`@StandardTable`** command.

25. Two additional LibraryFiles are available for use at the individual manuscript level.

    a. A LibraryFile is available for creating pages (especially wide tables) that are rotated 90° and has to be invoked at the beginning of the manuscript with the command **@LibraryFile(Landscape)** (filename **LANDSC.LIB**). The table at the end was created and Scribed as a separate file with the landscape library file and then inserted in this text with an @Graphic command. It takes a few iterations to position everything correctly (including negative blankspaces in the separate file) and Scribe incorrectly issues two error messages (which can safely be ignored).
    b. The other LibraryFile is for printing on legal-sized paper and is also invoked at the beginning of the manuscript. During compilation, **LEGAL.LIB** displays a reminder to exchange paper trays before and after printing.

26. The database files of our own design will be amended as necessary and comments and new inclusions are requested.
27. A new environment named List has been created and produces a balanced two-column indented list (a list with an odd number of items is printed so that the first column is longer). The command @ListTitle gives the list a centered and unnumbered title. As an example:

```
@ListTitle(Gases and Vapors used in Sensor Testing)
@Begin(List)
1,1,1-trichloroethane
1,1,2-trichloroethene
carbon tetrachloride
ethanol
trichloromethane
sulfur hexafluoride
sodium hypochlorite
methanol
@End(List)
```

gives

Gases and Vapors used in Sensor Testing

1,1,1-trichloroethane
1,1,2-trichloroethene
carbon tetrachloride
ethanol
trichloromethane
sulfur hexafluoride
sodium hypochlorite
methanol

28. Another environment named TNote has been created as an itemized (using Newitem) subsubsection set off by horizontal bars as follows:

```
@TNote(This is the first line of text.
This is the second line.
Finally, this is the third line.)
```

gives

Note

- This is the first line of text.
- This is the second line.
- Finally, this is the third line.

---

29. A similar environment called VCom is designed to insert comments in Verbatim-displayed programs:

```
This is the last program line before the comment.
@VCom(This is the comment.)
This is the first program line after the comment.
```

gives

```
This is the last program line before the comment.
```

```
Comment:        This is the comment.
```

```
This is the first program line after the comment.
```

30. The Scribe Enumerate and Itemize environments have been modified slightly (the spacing and spread attributes are changed and the enumerate is now a four-level list) and nine **additional** environments based on them have been designed.

    - Enumerate# – Scribe's Enumerate with an unlimited set of numbered levels (1, 1.1, 1.1.1, 1.1.1.1, etc.)
    - NewEnum — an Enumerate with the Spread attribute at 0 (used here)
    - NewEnum# — an Enumerate with the Spread attribute at 0 giving an unlimited set of numbered levels (1, 1.1, 1.1.1, 1.1.1.1, etc.)
    - NewItem — an Itemize with the Spread attribute at 0 (used here)
    - ProgEnum — an Enumerate with an F FaceCode for program listing
    - ProgSecEnum — same as ProgEnum but numbered from 100
    - NewDef — an unnumbered Itemize with an F FaceCode for definitions (used here)
    - ChicagoEnum — an Enumerate approved by the *University of Chicago Press*
    - XeroxEnum — an Enumerate conforming to the *Xerox Corporation* style

31. Enumeration according to the *Chicago Manual of Style* is available with the command

    `@Begin(ChicagoEnum)`

    with the result shown below.

    I. Text text text text text text text.
    II. Text text text text text text text.

        A. Text text text text text text text.
        B. Text text text text text text text.

            1. Text text text text text text text.
            2. Text text text text text text text.

                a) Text text text text text text text.
                b) Text text text text text text text.

                    (1) Text text text text text text text.
                    (2) Text text text text text text text.

                        (*a*) Text text text text text text text.
                        (*b*) Text text text text text text text.

                            i) Text text text text text text text.
                            ii) Text text text text text text text.

32. Enumeration according to *Xerox Corporation* documentation is available with the command

    `@Begin(XeroxEnum)`

    with the result shown below.

    I. Text text text text text text text.
    II. Text text text text text text text.

        A. Text text text text text text text.
        B. Text text text text text text text.

            1. Text text text text text text text.
            2. Text text text text text text text.

                a. Text text text text text text text.
                b. Text text text text text text text.

                    • Text text text text text text text.
                    • Text text text text text text text.

                        — Text text text text text text text.
                        — Text text text text text text text.

                            – Text text text text text text text.
                            – Text text text text text text text.

    These are the only listing environments with seven uniquely labeled levels — Scribe's Enumerate and Itemize repeat their labels after 3 and 2 levels respectively.

33. A complete set of facecodes has been defined for running text use — Scribe's original set only included the @B, @C, @G, @I, @P, @R, @T, and @Y environments. The new set also includes

    - @A — gives TimesItalic facecode corresponding to the @Math environment
    - @E — selects the upper 95-character ZapfDingbat symbol assortment using the 95-character ASCII printing set (the lower 95-character assortment is available using the special font environment F8)
    - @F — gives CourierBold facecode
    - @J — same usage as @JSYM
    - @K — same usage as @KSYM
    - @N — selects 51 of the remaining 55 available PostScript characters (the 4 unselected characters are available elsewhere) along with 3 extra Greek characters (final sigma and mathematical lower case phi and theta) and a few characters borrowed from other font families
    - @X — one point size smaller **bold** capital facecode
    - @Z — gives TimesItalic facecode

    A complete table of @E, @J, @K, @N, and @F8 character translations is on the last page.
34. There are presently 9 Special Fonts available using the @SpecialFont command described in the *Scribe User's Manual*.

    - @F1 HelveticaNarrow
    - @F2 TimesRoman
    - @F3 AvantGardeBook
    - @F4 BookmanLight
    - @F5 NewCenturySchlbkRoman
    - @F6 PalatinoRoman
    - @F7 ZapfChanceryMediumItalic
    - @F8 ZapfDingbats
    - @F9 TimesItalic

    These, however, are **not** available in their complete range of facecodes as the names of the font families above refer only to one facecode. See the next item for information on obtaining the complete set of font families available on the Apple LaserWriter printer.

35. In order to access the full facecode range of the above font families a command based on the following definition should be used

```
@Begin(Environment, Font FontName)
```

with the results (using the NewItem environment)

- This sentence is written in HelveticaNarrow.
- *This sentence is written in HelveticaNarrowOblique.*
- **This sentence is written in HelveticaNarrowBold.**
- ***This sentence is written in HelveticaNarrowBoldOblique.***

- This sentence is written in TimesRoman.
- *This sentence is written in TimesItalic.*
- **This sentence is written in TimesBold.**
- ***This sentence is written in TimesBoldItalic.***

- This sentence is written in AvantGardeBook.
- *This sentence is written in AvantGardeBookOblique.*
- **This sentence is written in AvantGardeDemi.**
- ***This sentence is written in AvantGardeDemiOblique.***

- This sentence is written in BookmanLight.
- *This sentence is written in BookmanLightItalic.*
- **This sentence is written in BookmanDemi.**
- ***This sentence is written in BookmanDemiItalic.***

- This sentence is written in NewCenturySchlbkRoman.
- *This sentence is written in NewCenturySchlbkItalic.*
- **This sentence is written in NewCenturySchlbkBold.**
- ***This sentence is written in NewCenturySchlbkBoldItalic.***

- This sentence is written in PalatinoRoman.
- *This sentence is written in PalatinoItalic.*
- **This sentence is written in PalatinoBold.**
- ***This sentence is written in PalatinoBoldItalic.***

- This sentence is written in Courier.
- *This sentence is written in CourierOblique.*
- **This sentence is written in CourierBold.**
- ***This sentence is written in CourierBoldOblique.***

COMPLETE POSTSCRIPT CHARACTER TO SYMBOL TRANSLATION

UPPER CASE

Character	@E	@J	@K	@N	@F8
A	②	≤	·	Æ	✡
B	③	⁄	¬		✣
C	④	∞	∧	*f*	✤
D	⑤	*f*	∨	‡	✥
E	⑥	♣	⇔	Œ	✦
F	⑦	♦	⇐	fi	✧
G	⑧	♥	⇑	„	✩
H	⑨	♠	⇒		★
I	⑩	↔	⇓	ϖ	✪
J	❶	←	◊		✫
K	❷	↑	〈	°	✬
L	❸	→	®	Ł	✭
M	❹	↓	©	‹	✮
N	❺	°	™	«	✯
O	❻	±	∑	Ø	✰
P	❼	″	⎛	¶	✱
Q	❽	≥	⎜	”	✶
R	❾	×	⎝		✲
S	❿	∝	⎡	§	✳
T	➔	∂	⎜		✴
U	→	•	⎣	ϒ	✵
V	↔	÷	⎛		✶
W	↕	≠	⎨	ˇ	✷
X	➘	≡	⎣	º	✸
Y	➔	≈	⎜	¥	✹
Z	➘	…	〉	ˇ	✺

LOWER CASE

Character	@E	@J	@K	@N	@F8
a	➡	⏐	⌠	æ	❁
b	➢	—	⌡		❂
c	➢	↵	⎮	¢	❃
d	➤	ℵ	⌡	†	❄
e	➥	ℑ	⎞	œ	❅
f	➦	ℜ	⎮	fl	❆
g	➧	℘	⌡	“	❇
h	➨	⊗	⎤		❈
i	➩	⊕	⎮	ı	❉
j	➪	∅	⌋		❊
k	➫	∩	⎤	¸	❋
l	➬	∪	⎬	ł	●
m	➭	⊃	⌋	›	❍
n	➮	⊇	(	»	■
o	➯	⊄	)	ø	❑
p	N/A	⊂	[	φ	❒
q	➱	⊆	]	'	❑
r	➲	∈	{		❒
s	➳	∉	}	ß	▲
t	➴	∠	∃	ϑ	▼
u	➵	∇	υ		◆
v	➶	®	ϖ		❖
w	➷	©	ω	¨	◗
x	➸	™	ξ	ª	❘
y	➹	∏	ψ	ς	❙
z	➺	√	ζ	¸	❚

NUMBER AND SHIFTED NUMBER

Character	@E	@J	@K	@N	@F8
1	⑥	1	1		✎
2	⑦	2	2		✏
3	⑧	3	3		✓
4	⑨	4	4		✔
5	⑩	5	5		✕
6	❶	6	6		✖
7	❷	7	7		✗
8	❸	8	8		✘
9	❹	9	9		✙
0	⑤	0	0		✐
!	❡	!	!	¡	✚
@	①	≅	≅		✛
#	❣	#	#		✜
$	❤	∃	∃	£	✝
%	❥	%	%	‰	✞
^	➞	⊥	⊥	ˆ	✟
&	❦	&	&	&	✠
*	♥	*	*	¤	✡
(	♣	(	(	(	✍
)	✦	)	)	)	✉

REMAINING PUNCTUATION

Character	@E	@J	@K	@N	@F
-	②	–	–	¯	✍
_	➟	_	_		✎
=	❽	=	=		†
+	♠	+	+		✐
‘	➠	¯	¯	`	✑
~	➢	~	~	˜	”
[	→	[	[	[	*
]	→	]	]	]	*
\	➔	∴	∴		*
|	➨	|	|		’
;	❻	;	;	·	✢
:	❺	:	:	·	✚
’	❧	϶	϶	´	✣
"	❣	∀	∀	″	✁
{	➔	«	{		‘
}	➡	»	}		“
,	①	,	,	,	✌
<	❼	<	<		✤
.	③	.	.	·	✏
>	❾	>	>		✞
/	④	/	/	/	✁
?	❿	?	?	¿	✝
SP					

Comment: **TECHRE.MAK** is the definitive version —— attribute pair lines here have been concatenated to save space. Also, there may be additions and modifications that will not appear here.

```
@Marker{Make, TechReport, PostScript, ScaleableLaser}
@Comment[Copyright 8/88, Alan D. Guisewite]
```

Comment: selects Helvetica font family

```
@LibraryFile{SansSerif}
```

Comment: bibliography database selection

```
@Use{Bibliography "/../sl/usr/mws/bib/microm.bib",
     Bibliography "/../sl/usr/adg/appnotes/adg.bib"}
```

Comment: defines the various font families and typecase selections

---

```
@DefineFont[BodyFont,
        A {TypeCase     "Mathematics10"},
        B {RawFont      "HelveticaBold"},
        D {RawFont      "ZapfDingbats"},
        E {TypeCase     "ZapfDingbats"},
        F {RawFont      "CourierBold"},
        G {TypeCase     "PSGreek"},
        H {RawFont      "ExHelvetica"},
        I {RawFont      "HelveticaOblique"},
        J {TypeCase     "PsSymbol1"},
        K {TypeCase     "PsSymbol2"},
        N {TypeCase     "NewSymbol"},
        O {RawFont      "OHelveticaBold"},
        P {RawFont      "HelveticaBoldOblique"},
        R {RawFont      "Helvetica"},
        T {RawFont      "Courier"},
        Y {TypeCase     "PsPubsSansSerif"},
        Z {RawFont      "TimesItalic"},
        1 {RawFont      "NewAccentHelvetica"},
        2 {RawFont      "NewAccentHelveticaBold"},
        3 {RawFont      "NewAccentHelveticaOblique"},
        4 {RawFont      "NewAccentHelveticaBoldOblique"},
]
@Define{BodyStyle,      Font BodyFont, Spread 1, Size 10, Indent 0}
@Define{NoteStyle,      Font BodyFont, FaceCode R, Spacing 1,
                        Size 8, Indent 0}
```

Comment: defines the page headings for the table of contents and endnotes pages —— the actual headings are the strings in the @PrefaceSection commands (the page headings for the tables and figures are defined in Senslab's **TFIGUR.LIB** or in Scribe's **FIGURE.LIB**

```
@Generate{Notes,Outline,Contents}
@Send{Notes    "@PrefaceSection{Notes}"}
@Send{Contents "@PrefaceSection{Contents}"}
@Send{Contents "@Set{Page=1}@Style{PageNumber <@i>}"}
@Send{Contents "@define{foot,invisible}"}
```

Comment: defines single-character facecodes (in addition to the ones defined in **POSTSC.DEV**)

```
@Define{A, FaceCode A, TabExport}
@Define{E, FaceCode E, TabExport}
@Define{F, FaceCode F, TabExport}
@Define{H, FaceCode H, TabExport}
@Define{J, FaceCode J, TabExport}
@Define{K, FaceCode K, TabExport}
@Define{N, FaceCode N, TabExport}
@Define{O, FaceCode O, TabExport}
@Define{X, FaceCode B, Size -1, Capitalized, TabExport}
@Define{Z, FaceCode Z, TabExport}
@Define{1, FaceCode 1, TabExport}
@Define{2, FaceCode 2, TabExport}
@Define{3, FaceCode 3, TabExport}
@Define{4, FaceCode 4, TabExport}
@Define{jsym, facecode j}
@Define{ksym, facecode k}
@Define{zsym, facecode z}
```

Comment: defines various headings (for Section, SubSection, etc.)

```
@Define{HDX,   Hyphenation off, LeftMargin 0, Indent 0, Fill,
               Spaces compact, Above 0, Below 0, Break, Need 4,
               Justification Off, Font BodyFont, FaceCode B}
@Define{Hd0,   Use HdX, Above 2.0, Below 1.2, Size +5, PageBreak off}

@Define{Hd1,   Use HdX, Above 2.2, Below 1.4, Size +3, PageBreak off}

@Define{Hd1A   Hd1, Centered, Tabexport False, Afterentry "@tabclear{}"}

@Define{Hd2,   Use HdX, Above 2.4, Below 1.5, Size +2}

@Define{Hd2A   Hd2, Centered}

@Define{Hd2B   Hd2, PageBreak before}

@Define{Hd3,   Use HdX, Above 2.6, Below 1.6, Size +1}

@Define{Hd4,   Use HdX, Above 2.8, Below 1.7, Size +0}

@Define{Hd5,   Use HdX, Above 1, Below 1, Size -1}
```

Comment: defines various Contents text

```
@Define{TcX,   LeftMargin 5, Indent -5, RightMargin 5, Fill,
               Spaces compact, Above 0, Below 0, Spacing 1,
               Spread 0.6, Break, FaceCode R, Font BodyFont}

@Define{Tc0    TcX, Size +3, Below 2}
```

```
@Define{Tc1    TcX, Size +2}

@Define{Tc2    TcX, LeftMargin 8, Spacing 1.5, Size +1}

@Define{Tc3    TcX, LeftMargin 12, Size +0}

@Define{Tc4    TcX, LeftMargin 16, Size -1}
```

Comment: defines various unnumbered text divisions

```
@Counter{UnSection, TitleEnv HD2}

@Counter{UnSubSection, TitleEnv HD3}

@Counter{UnSubSubSection, TitleEnv HD4}

@Counter{Abstract,     TitleEnv HD2B, ContentsEnv tc2,
                       Numbered [@@hsp{-3pt}]}

@Counter{MajorPart,    TitleEnv HD0, ContentsEnv tc0,
                       Numbered [@@hsp{-3.75pt}], IncrementedBy Use}

@Counter{Chapter,      TitleEnv HD1, ContentsEnv tc1,
                       Numbered [@@hsp{-3.25pt}],
                       IncrementedBy Use, Referenced []}

@Counter{Appendix,     TitleEnv HD2B, ContentsEnv tc2,
                       Numbered [Appendix @A: ], IncrementedBy,
                       Referenced [@A], Announced}

@Counter{UnNumbered,   TitleEnv HD1, ContentsEnv tc1}

@Counter{NewSection,   TitleEnv HD2B, ContentsEnv tc2,
                       Numbered [@@hsp{-3pt}], Referenced [],
                       IncrementedBy Use, Announced}

@Counter{Section,      Within Chapter, TitleEnv HD2, ContentsEnv tc2,
                       Numbered [@@hsp{-3pt}], Referenced [],
                       IncrementedBy Use, Announced}

@Counter{AppendixSection,      Within Appendix, TitleEnv HD3,
                               ContentsEnv tc3, Numbered [@#@:.@1],
                               Referenced [@#@:.@1],
                               IncrementedBy Use, Announced}

@Counter{References,   TitleEnv HD2B, ContentsEnv tc2,
                       Numbered [@@hsp{-3pt}], Announced}

@Counter{SubSection,   Within Section, TitleEnv HD3, ContentsEnv tc3,
                       Numbered [@@hsp{-2.75pt}], Referenced [],
                       IncrementedBy Use}

@Counter{Paragraph,    Within SubSection, TitleEnv HD4,
                       ContentsEnv tc4, Numbered [@@hsp{-2.5pt}],
                       Referenced [], IncrementedBy Use}

@Counter{PrefaceSection,       TitleEnv HD1A, Alias Chapter}
```

Comment: defines various Figure and Table text environments

```
@Define{Legend,         Use Center, Spacing 1.1, Above 1,
                        Longlines wrap, Font BodyFont,
                        FaceCode R, Size -1}

@Define{TLegend         Legend, AfterExit "@BlankSpace{1.1}"}

@Define{LongLegend,     Indent 0, LeftMargin +1", RightMargin +1",
                        Size -1, Spacing 1, Fill, Justification On}

@define{credit,         use center, size -4}

@LibraryFile{Abbr1}
@LibraryFile{AbbrSL}
@LibraryFile{NTable}
@LibraryFile{TechTitle}
@LibraryFile{Universal}
@LibraryFile{Chemistry}

@Equate{        Sec Section, Subsec SubSection, Para Paragraph,
                SubSubSec Paragraph, SubSubSection Paragraph,
                AppendixSec AppendixSection}

@Begin{Text,    LeftMargin 1.25", TopMargin 1", BottomMargin 1",
                LineWidth 6.25", Use BodyStyle, FaceCode R}

@LibraryFile{MathSL10}

@modify{enumerate,      spacing 1.1, spread 0.6, above 1, below 1}

@modify{itemize,        spacing 1.1, spread 0.6, above 1, below 1,
                        numbered [@@jsym{U} @,@@~
                 @@begin{jsym, size -2, script +1.0pt}U@@end{jsym} @,@@~
                 @@begin{jsym, size -4, script +2.0pt}U@@end{jsym} @,@@~
                 @@begin{jsym, size -6, script +2.4pt}U@@end{jsym} ]}

@modify{mathdisplay,    above 0.5, below 0.5, scriptpush yes}
@modify{majorheading,   above 2, below 1, font bodyfont}
@modify{heading,        above 2, below 1, font bodyfont}
@modify{subheading,     above 2, below 1, font bodyfont}
@modify{center,         above 1, below 1}
```

Comment: defines font families and typecase selections for fonts other than Helvetica

---

```
@DefineFont     {HelveticaNarrow,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "HelveticaNarrowBold">,
        i <RawFont "HelveticaNarrowOblique">,
        p <RawFont "HelveticaNarrowBoldOblique">,
        r <RawFont "HelveticaNarrow">}
@DefineFont     {AvantGardeBook,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
```

```
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "AvantGardeDemi">,
        i <RawFont "AvantGardeBookOblique">,
        p <RawFont "AvantGardeDemiOblique">,
        r <RawFont "AvantGardeBook">}
@DefineFont    {BookmanLight,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "BookmanDemi">,
        i <RawFont "BookmanLightItalic">,
        p <RawFont "BookmanDemiItalic">,
        r <RawFont "BookmanLight">}
@DefineFont    {NewCenturySchlbkRoman,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "NewCenturySchlbkBold">,
        i <RawFont "NewCenturySchlbkItalic">,
        p <RawFont "NewCenturySchlbkBoldItalic">,
        r <RawFont "NewCenturySchlbkRoman">}
@DefineFont    {PalatinoRoman,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "PalatinoBold">,
        i <RawFont "PalatinoItalic">,
        p <RawFont "PalatinoBoldItalic">,
        r <RawFont "PalatinoRoman">}
@DefineFont    {TimesRoman,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        f <RawFont "CourierBold">,
        t <RawFont "Courier">,
        b <RawFont "TimesBold">,
        i <RawFont "TimesItalic">,
        p <RawFont "TimesBoldItalic">,
        r <RawFont "TimesRoman">}
@DefineFont    {Courier,
        g <TypeCase "PsGreek">,
        y <TypeCase "PsPubsSansSerif">,
        j <TypeCase "PsSymbol1">,
        k <TypeCase "PsSymbol2">,
        b <RawFont "CourierBold">,
        i <RawFont "CourierOblique">,
        p <RawFont "CourierBoldOblique">,
        r <RawFont "Courier">}

@SpecialFont{F1 HelveticaNarrow}
```

```
@SpecialFont{F2 TimesRoman}
@SpecialFont{F3 AvantGardeBook}
@SpecialFont{F4 BookmanLight}
@SpecialFont{F5 NewCenturySchlbkRoman}
@SpecialFont{F6 PalatinoRoman}
@SpecialFont{F7 ZapfChanceryMediumItalic}
@SpecialFont{F8 ZapfDingbats}
@SpecialFont{F9 TimesItalic}

@Style{         StringMax 16000, DoubleSided off, ChangeBarOffset 36pt,
                References newcacm, ScriptPush yes, Justification on,
                WidowAction force, Date {1952 Mar 8 (Sat)},
                Time {16:30 o'clock}}
```

Comment: selects different manuscript styles for Draft or Final copies

---

```
@case[ draft,yes
                (@pageheading{ immediate,
left "@value{username}", center "CMU/RI/SensorLab", right "DRAFT"}
                @pagefooting{  immediate,
left "@value{date}, @value{time}", center "@value{page}", right "DRAFT"}),
        else
                (@pageheading{ immediate,
                               left "", center "", right ""}
                @pagefooting{  immediate,
                               left "", center "@value{page}", right ""})]
@Case[ draft,yes
                (@disable{contents, illustrations, tables}),
        else""]
@Case[ draft,yes
                (@style{       Spacing 2.4}),
        else
                (@style{       Spacing 1.1})]
@Set{  Page 0}
```

Comment: redefines FootNote environment to eliminate prevailing margin dependencies

---

```
@define{fnenv,          use text, above 0, foot, use r, tabexport false,
                        leftmargin 0, linewidth 6.25", size -2,
                        crspace, unnumbered, indent 2, spacing 1,
                        spread 1, break off, counter footnotecounter}

@LibraryFile{TFigures}
```

Comment: defines new manuscript environments

```
@index{NewEnum environment definition, TechReport}
@define{newenum         enumerate, leftmargin +36pt, indent -36pt,
                        rightmargin 25pt, above 1, below 1,
                        spacing 1.1, spread 0}

@define{newitem         itemize, leftmargin +36pt, indent -36pt,
                        rightmargin 25pt, above 1, below 1,
                        spacing 1.1, spread 0}
```

```
@define{progenum       enumerate, leftmargin +12,
                       numbered <@1 @@ @@ @@ >,
                       above 1, below 1, spread 0,
                       facecode f, blanklines kept}

@define{progsecenum    progenum, numbered <@100 @@ @@ @@ >}

@define{newdef         newitem, indent 0, unnumbered,
                       leadingspaces kept, blanklines kept,
                       spaces kept, facecode f}

@define{chicagoenum    newenum, leftmargin +20pt, indent -20pt,
                       rightmargin 25pt,
                       numbered <@I. @, @A. @, @1. @, @@f9{@a}) @,
                       (@1) @, @@f9{@a}) @, @i) >}

@define{enumerate#     enumerate,      numbered (@#@:.@1.@@ @@]),
                                       referenced [@#@:.@1]}

@define{newenum#       newenum,        numbered (@#@:.@1.@@ @@]),
                                       referenced [@#@:.@1]}

@textform[tnote                ’@unsubsubsection{Note}@~
                        @bar{}@newitem<@parm{text}>@bar{}’]

@textform[listtitle    ’@blankspace{1.1}
                @center<@begin{r, size -1}@parm{text}@end{r}>’]

@define{list,  columns 2, size -1, above 1, below 2, break,
               crbreak, continue allowed, blanklines break,
               boxed, columnmargin 0.5", columnbalance on,
               footnoteposition page, indent 36pt, spread 0}
```

Appendix B: The NEWACC.LIB Senslab NewAccents Library File

```
@Marker(Library,NewAccents,Postscript,Scaleablelaser)
```

Comment: This provides an initial comment on the procedure to obtain accented characters.

```
@Comment[
Copyright 8/88, Alan D. Guisewite,
                renaming and updating for Scribe Version 6.

Library file to get to PostScript's accented characters.
This includes the DeviceInitialization string, which can be either
here or in the POSTSC.DEV file.  Putting it in POSTSC.DEV file will
cause all Scribe-generated .PS files to be 2K larger.
This is what was done:

        Generated a new encoding vector that maps the accented
        characters into the ASCII alphabetic slots.  (all but two,
        since there's 28 accented characters and only 26 letters)
        This bit of PostScript gets appended to the Scribe
        DeviceInitialization string.

        Defined forms for the new characters.

        Defined .raw font files for the accented fonts.

        Added the facecodes "1, 2, 3, and 4" to get to the accented fonts.

The new header is about 2000 bytes longer than the original 870-byte
header.  The new fonts use up about 7K of PostScript VM, about 4% of
the available VM on a LaserWriter.

This will allow accents in regular text and headings.

Modified version of Glenn Trewitt's (Stanford) original creation.

        Modified 8/88 to run with NewScribe (Version 6) and renamed.
]
```

Comment: This section is the PostScript Device Initialization string —— whenever a library file containing a PostScript program is included in a Scribe manuscript it must be preceded by this string.

```
@style(stringmax 8000)

@style(DeviceInitialization=
<%%EndComments
% PostScript Prelude for Scribe.
/BS {/SV save def 0.0 792.0 translate .01 -.01 scale} bind def
/ES {showpage SV restore} bind def
/SC {setrgbcolor} bind def
/FMTX matrix def
/RDF {WFT SLT 0.0 eq
  {SSZ 0.0 0.0 SSZ neg 0.0 0.0 FMTX astore}
  {SSZ 0.0 SLT sin SLT cos div SSZ mul SSZ neg 0.0 0.0 FMTX astore}
```

```
  ifelse makefont setfont} bind def
/SLT 0.0 def
/SI { /SLT exch cvr def RDF} bind def
/WFT /Courier findfont def
/SF { /WFT exch findfont def RDF} bind def
/SSZ 1000.0 def
/SS { /SSZ exch 100.0 mul def RDF} bind def
/AF { /WFT exch findfont def /SSZ exch 100.0 mul def RDF} bind def
/MT /moveto load def
/XM {currentpoint exch pop moveto} bind def
/UL {gsave newpath moveto dup 2.0 div 0.0 exch rmoveto
   setlinewidth 0.0 rlineto stroke grestore} bind def
/LH {gsave newpath moveto setlinewidth
   0.0 rlineto
   gsave stroke grestore} bind def
/LV {gsave newpath moveto setlinewidth
   0.0 exch rlineto
   gsave stroke grestore} bind def
/BX {gsave newpath moveto setlinewidth
   exch
   dup 0.0 rlineto
   exch 0.0 exch neg rlineto
   neg 0.0 rlineto
   closepath
   gsave stroke grestore} bind def
/BX1 {grestore} bind def
/BX2 {setlinewidth 1 setgray stroke grestore} bind def
/PB {/PV save def newpath translate
   100.0 -100.0 scale pop /showpage {} def} bind def
/PE {PV restore} bind def
/GB {/PV save def newpath translate rotate
   div dup scale 100.0 -100.0 scale /showpage {} def} bind def
/GE {PV restore} bind def
/FB {dict dup /FontMapDict exch def begin} bind def
/FM {cvn exch cvn exch def} bind def
/FE {end /original-findfont /findfont load def  /findfont
   {dup FontMapDict exch known{FontMapDict exch get} if
   original-findfont} def} bind def
/BC {gsave moveto dup 0 exch rlineto exch 0 rlineto neg 0 exch rlineto
closepath clip} bind def
/EC /grestore load def
/SH /show load def
/MX {exch show 0.0 rmoveto} bind def
/W {0 32 4 -1 roll widthshow} bind def
/WX {0 32 5 -1 roll widthshow 0.0 rmoveto} bind def
```

Comment: This is the actual reencoding to obtain PostScript's accented character set.

```
% Generate encodings for accent fonts.  (from the Postscript "Cookbook")
/reencodedict 5 dict def
/ReEncode
  { reencodedict begin
    /newencoding exch def
    /newfontname exch def
    /basefontname exch def
    /basefontdict basefontname findfont def
    /newfont basefontdict maxlength dict def
    basefontdict
     {exch dup /FID ne dup /Encoding ne and
```

```
        { exch newfont 3 1 roll put }
        { pop pop }
        ifelse
      } forall
     newfont /FontName newfontname put
     newfont /Encoding newencoding put
     newfontname newfont definefont pop
     end
    } def

/Stuff { exch 1 add dup Accent 4 2 roll exch put } def
/Accent 512 array def
0 1 511 { Accent exch /.notdef put } for
8#100
  [ /Aacute /Acircumflex /Adieresis /Agrave /Aring /Atilde
    /Ccedilla
    /Eacute /Ecircumflex /Edieresis /Egrave
    /Iacute /Icircumflex /Idieresis /Igrave
    /Ntilde
    /Oacute /Ocircumflex /Odieresis /Ograve /Otilde
    /Scaron
    /Uacute /Ucircumflex /Udieresis /Ugrave
    /Ydieresis
    /Zcaron ]
  { Stuff } forall
pop
8#140
  [ /aacute /acircumflex /adieresis /agrave /aring /atilde
    /ccedilla
    /eacute /ecircumflex /edieresis /egrave
    /iacute /icircumflex /idieresis /igrave
    /ntilde
    /oacute /ocircumflex /odieresis /ograve /otilde
    /scaron
    /uacute /ucircumflex /udieresis /ugrave
    /ydieresis
    /zcaron ]
  { Stuff } forall
pop
8#177
  [ /a /b /c /d /e /f /g /h /i /j /k /l /m
    /n /o /p /q /r /s /t /u /v /w /x /y /z
    /A /B /C /D /E /F /G /H /I /J /K /L /M
    /N /O /P /Q /R /S /T /U /V /W /X /Y /Z
    /dotlessi /grave /acute /circumflex /tilde
    /macron /breve /dotaccent /dieresis
    /ring /cedilla /hungarumlaut /ogonek /caron]
  { Stuff } forall
pop
```

Comment: This listing of the reencoded fonts describes the now available accented fonts — the list can be expanded to include other font families.

```
/Times-Roman /Times-Accent Accent ReEncode
/Times-Bold /Times-Bold-Accent Accent ReEncode
/Times-Italic /Times-Italic-Accent Accent ReEncode
/Times-BoldItalic /Times-BoldItalic-Accent Accent ReEncode
/Helvetica /Helvetica-NewAccent Accent ReEncode
/Helvetica-Bold /Helvetica-Bold-NewAccent Accent ReEncode
```

```
/Helvetica-Oblique /Helvetica-Oblique-NewAccent Accent ReEncode
/Helvetica-BoldOblique /Helvetica-BoldOblique-NewAccent Accent ReEncode
/Courier /Courier-Accent Accent ReEncode
/Courier-Bold /Courier-Bold-Accent Accent ReEncode
/Courier-Oblique /Courier-Oblique-Accent Accent ReEncode
/Courier-BoldOblique /Courier-BoldOblique-Accent Accent ReEncode

%%EndProlog
>)
```

Comment: This is the facecode to environment definition. These facecodes must also be defined in a @DefineFont command (see **TECHRE.MAK**)

```
@define(act, facecode 1, TabExport)
@define(bact, facecode 2, TabExport)
@define(iact, facecode 3, TabExport)
@define(pact, facecode 4, TabExport)
```

Comment: These are the form definitions to select the accented characters.

```
@comment[Macros for Roman accents.]

@form<aac="@act(a)">
@form<acr="@act(b)">
@form<aum="@act(c)">
@form<agr="@act(d)">
@form<arg="@act(e)">
@form<atl="@act(f)">
@form<cca="@act(1)">
@form<ccd="@act(g)">
@form<eac="@act(h)">
@form<ecr="@act(i)">
@form<eum="@act(j)">
@form<egr="@act(k)">
@form<iac="@act(l)">
@form<icr="@act(m)">
@form<ium="@act(n)">
@form<igr="@act(o)">
@form<ntl="@act(p)">
@form<oac="@act(q)">
@form<ocr="@act(r)">
@form<oum="@act(s)">
@form<ogr="@act(t)">
@form<otl="@act(u)">
@form<sca="@act(v)">
@form<uac="@act(w)">
@form<ucr="@act(x)">
@form<uum="@act(y)">
@form<ugr="@act(z)">
@form<yac="@act(2)">
@form<yum="@act({)">
@form<zca="@act(|)">

@form<uaac="@act(A)">
@form<uacr="@act(B)">
@form<uaum="@act(C)">
```

```
@form<uagr="@act(D)">
@form<uarg="@act(E)">
@form<uatl="@act(F)">
@form<ucca="@act(!)">
@form<uccd="@act(G)">
@form<ueac="@act(H)">
@form<uecr="@act(I)">
@form<ueum="@act(J)">
@form<uegr="@act(K)">
@form<uiac="@act(L)">
@form<uicr="@act(M)">
@form<uium="@act(N)">
@form<uigr="@act(O)">
@form<untl="@act(P)">
@form<uoac="@act(Q)">
@form<uocr="@act(R)">
@form<uoum="@act(S)">
@form<uogr="@act(T)">
@form<uotl="@act(U)">
@form<usca="@act(V)">
@form<uuac="@act(W)">
@form<uucr="@act(X)">
@form<uuum="@act(Y)">
@form<uugr="@act(Z)">
@form<uyac="@act(#)">
@form<uyum="@act([)">
@form<uzca="@act(\)">

@Comment[Macros for Bold accents.]

@form<baac="@bact(a)">
@form<bacr="@bact(b)">
@form<baum="@bact(c)">
@form<bagr="@bact(d)">
@form<barg="@bact(e)">
@form<batl="@bact(f)">
@form<bccd="@bact(g)">
@form<beac="@bact(h)">
@form<becr="@bact(i)">
@form<beum="@bact(j)">
@form<begr="@bact(k)">
@form<biac="@bact(l)">
@form<bicr="@bact(m)">
@form<bium="@bact(n)">
@form<bigr="@bact(o)">
@form<bntl="@bact(p)">
@form<boac="@bact(q)">
@form<bocr="@bact(r)">
@form<boum="@bact(s)">
@form<bogr="@bact(t)">
@form<botl="@bact(u)">
@form<bsca="@bact(v)">
@form<buac="@bact(w)">
@form<bucr="@bact(x)">
@form<buum="@bact(y)">
@form<bugr="@bact(z)">
@form<byum="@bact({)">
@form<bzca="@bact(|)">

@form<buaac="@bact(A)">
@form<buacr="@bact(B)">
@form<buaum="@bact(C)">
```

```
@form<buagr="@bact(D)">
@form<buarg="@bact(E)">
@form<buatl="@bact(F)">
@form<buccd="@bact(G)">
@form<bueac="@bact(H)">
@form<buecr="@bact(I)">
@form<bueum="@bact(J)">
@form<buegr="@bact(K)">
@form<buiac="@bact(L)">
@form<buicr="@bact(M)">
@form<buium="@bact(N)">
@form<buigr="@bact(O)">
@form<buntl="@bact(P)">
@form<buoac="@bact(Q)">
@form<buocr="@bact(R)">
@form<buoum="@bact(S)">
@form<buogr="@bact(T)">
@form<buotl="@bact(U)">
@form<busca="@bact(V)">
@form<buuac="@bact(W)">
@form<buucr="@bact(X)">
@form<buuum="@bact(Y)">
@form<buugr="@bact(Z)">
@form<buyum="@bact([)">
@form<buzca="@bact(\)">

@Comment[Macros for Italic accents.]

@form<iaac="@iact(a)">
@form<iacr="@iact(b)">
@form<iaum="@iact(c)">
@form<iagr="@iact(d)">
@form<iarg="@iact(e)">
@form<iatl="@iact(f)">
@form<iccd="@iact(g)">
@form<ieac="@iact(h)">
@form<iecr="@iact(i)">
@form<ieum="@iact(j)">
@form<iegr="@iact(k)">
@form<iiac="@iact(l)">
@form<iicr="@iact(m)">
@form<iium="@iact(n)">
@form<iigr="@iact(o)">
@form<intl="@iact(p)">
@form<ioac="@iact(q)">
@form<iocr="@iact(r)">
@form<ioum="@iact(s)">
@form<iogr="@iact(t)">
@form<iotl="@iact(u)">
@form<isca="@iact(v)">
@form<iuac="@iact(w)">
@form<iucr="@iact(x)">
@form<iuum="@iact(y)">
@form<iugr="@iact(z)">
@form<iyum="@iact({)">
@form<izca="@iact(|)">

@form<iuaac="@iact(A)">
@form<iuacr="@iact(B)">
@form<iuaum="@iact(C)">
@form<iuagr="@iact(D)">
@form<iuarg="@iact(E)">
```

```
@form<iuatl="@iact(F)">
@form<iuccd="@iact(G)">
@form<iueac="@iact(H)">
@form<iuecr="@iact(I)">
@form<iueum="@iact(J)">
@form<iuegr="@iact(K)">
@form<iuiac="@iact(L)">
@form<iuicr="@iact(M)">
@form<iuium="@iact(N)">
@form<iuigr="@iact(O)">
@form<iuntl="@iact(P)">
@form<iuoac="@iact(Q)">
@form<iuocr="@iact(R)">
@form<iuoum="@iact(S)">
@form<iuogr="@iact(T)">
@form<iuotl="@iact(U)">
@form<iusca="@iact(V)">
@form<iuuac="@iact(W)">
@form<iuucr="@iact(X)">
@form<iuuum="@iact(Y)">
@form<iuugr="@iact(Z)">
@form<iuyum="@iact([)">
@form<iuzca="@iact(\)">

@Comment[Macros for BoldItalic accents.]

@form<paac="@pact(a)">
@form<pacr="@pact(b)">
@form<paum="@pact(c)">
@form<pagr="@pact(d)">
@form<parg="@pact(e)">
@form<patl="@pact(f)">
@form<pccd="@pact(g)">
@form<peac="@pact(h)">
@form<pecr="@pact(i)">
@form<peum="@pact(j)">
@form<pegr="@pact(k)">
@form<piac="@pact(l)">
@form<picr="@pact(m)">
@form<pium="@pact(n)">
@form<pigr="@pact(o)">
@form<pntl="@pact(p)">
@form<poac="@pact(q)">
@form<pocr="@pact(r)">
@form<poum="@pact(s)">
@form<pogr="@pact(t)">
@form<potl="@pact(u)">
@form<psca="@pact(v)">
@form<puac="@pact(w)">
@form<pucr="@pact(x)">
@form<puum="@pact(y)">
@form<pugr="@pact(z)">
@form<pyum="@pact({)">
@form<pzca="@pact(|)">

@form<puaac="@pact(A)">
@form<puacr="@pact(B)">
@form<puaum="@pact(C)">
@form<puagr="@pact(D)">
@form<puarg="@pact(E)">
@form<puatl="@pact(F)">
@form<puccd="@pact(G)">
```

```
@form<pueac="@pact(H)">
@form<puecr="@pact(I)">
@form<pueum="@pact(J)">
@form<puegr="@pact(K)">
@form<puiac="@pact(L)">
@form<puicr="@pact(M)">
@form<puium="@pact(N)">
@form<puigr="@pact(O)">
@form<puntl="@pact(P)">
@form<puoac="@pact(Q)">
@form<puocr="@pact(R)">
@form<puoum="@pact(S)">
@form<puogr="@pact(T)">
@form<puotl="@pact(U)">
@form<pusca="@pact(V)">
@form<puuac="@pact(W)">
@form<puucr="@pact(X)">
@form<puuum="@pact(Y)">
@form<puugr="@pact(Z)">
@form<puyum="@pact([)">
@form<puzca="@pact(\)">
```

Appendix C: The MATHSL.LIB Senslab Mathemathics 10 Library File

```
@Marker(Library,Mathsl10,Postscript,Scaleablelaser)

@LibraryFile{Mathematics10}

@comment[Internal definitions]

@define(mathemarea,matharea,spacing 0inch,scriptpush)

@define(mathemsmall,font smallbodyfont,size -2)
```

Comment: Any new multi-line forms must be included in the following string commands.

```
@string(  mathemtextsetup
          "@form(sum  {@sumtext<from=@parmquote ‘from, default=[]’,
                                to=@parmquote ‘to, default=[]’>},
           prod {@prodtext<from=@parmquote ‘from, default=[]’,
                                 to=@parmquote ‘to, default=[]’>},
           int  {@inttext<from=@parmquote ‘from, default=[]’,
                                to=@parmquote ‘to, default=[]’>},
           over {@overtext<num=@parmquote ‘num, default=[]’,
                                 denom=@parmquote ‘denom, default=[]’>},
           choose {@choosetext<chosen=@parmquote ‘chosen, default=[]’,
                                   from=@parmquote ‘from, default=[]’>},
           loopint  {@loopinttext<from=@parmquote ’from, default=[]’,
                                      to=@parmquote ’to, default=[]’>},
           nestsum  {@nestsumtext<from=@parmquote ’from, default=[]’,
                                      to=@parmquote ’to, default=[]’>})",

          mathemdisplaysetup
          "@form(sum  {@sumdisplay<from=@parmquote ‘from, default=[]’,
                                  to=@parmquote ‘to, default=[]’>},
           prod {@proddisplay<from=@parmquote ‘from, default=[]’,
                                   to=@parmquote ‘to, default=[]’>},
           int  {@intdisplay<from=@parmquote ‘from, default=[]’,
                                  to=@parmquote ‘to, default=[]’>},
           over {@overdisplay<num=@parmquote ‘num, default=[]’,
                                   denom=@parmquote ‘denom, default=[]’>},
           choose {@choosedisplay<chosen=@parmquote ‘chosen, default=[]’,
                                     from=@parmquote ‘from, default=[]’>},
           loopint  {@loopintdisplay<from=@parmquote ’from, default=[]’,
                                      to=@parmquote ’to, default=[]’>},
           nestsum  {@nestsumdisplay<from=@parmquote ’from, default=[]’,
                                      to=@parmquote ’to, default=[]’>})")

@comment[Environments]

@define(math w, facecode a, break off, spaces null, tabexport off,
        afterentry "@value(mathemtextsetup)")
@define(mathdisplay w, use insert, facecode a, spaces null, group,
        tabexport off, indent 0, scriptpush on, nofill,
        afterentry "@value(mathemdisplaysetup)")
@define(sr, facecode r, spaces kept, beforeentry "#", afterexit "#")
@define(down,font smallbodyfont, size 8, facecode a, script -0.24lines)
@define(up down,script +0.24 lines)

@define(upp down,size 6,script +0.44 lines)
@define{ddown down, script -1line}
```

```
@definelinetype{matrixdoublelinetype,
                        weight 300,
                        groove 150}
@definebox{    matrixbox,
                vertical matrixdoublelinetype}
@define{matrix         box,
                boxtype matrixbox,
                spaces kept}

@define{determ box,
                boxtype standardcolumnlines,
                spaces kept,
                beforeentry "@ ",
                afterexit "@ "}

@define{labs   box,
                boxtype standardcolumnlines,
                beforeentry "@ ",
                afterexit "@ "}

@definebox{    evalbox,
                right standardlinetype}
@define{eval   box,
                boxtype evalbox,
                boxrm 5pt,
                afterexit "@ "}

@definelinetype{overstandardlinetype,  weight 25}
@definebox{overbox,    top overstandardlinetype}
@define{overword,      boxtype overbox, boxtm 1pt}

@definebox{qedbox,     all standardlinetype}
@define{qedboxmargins, boxtm 8pt, boxbm 0pt, boxlm 7pt, boxrm 7pt}
@define{qedsym,        boxtype qedbox, use qedboxmargins}

@definebox{sqibox,     left standardlinetype,
                        bottom standardlinetype,
                        right standardlinetype}
@define{sqiboxmargins, boxtm 8pt, boxbm -0.5pt, boxlm 7pt, boxrm 7pt}
@define{sqint,         boxtype sqibox, use sqiboxmargins}

@definebox{squbox,     left standardlinetype,
                        top standardlinetype,
                        right standardlinetype}
@define{squboxmargins, boxtm 8pt, boxbm 0pt, boxlm 7pt, boxrm 7pt}
@define{squnt,         boxtype squbox, use squboxmargins}

@comment[Special Characters]

@define(jsym,facecode j)
@define(ksym,facecode k)
@define(zsym,facecode z)
```

Comment: The following commandstrings define the single-line forms.

```
@CommandString[

biglp "@begin(transparent,size 16,script -.087lines)(@end(transparent)",
bigrp "@begin(transparent,size 16,script -.087lines))@end(transparent)",
bigglp "@begin(transparent,size 26,script -.17lines)(@end(transparent)",
```

```
biggrp "@begin(transparent,size 26,script -.17lines))@end(transparent)",
biggglp "@begin(transparent,size 36,script -.26lines)(@end(transparent)",
bigggrp "@begin(transparent,size 36,script -.26lines))@end(transparent)"

]

@CommandString[ add         "#@jsym(+)#",
                aleph       "@jsym(d)",
                and         "@ksym(C)",
                angle       "@jsym(t)",
                approx      "@jsym(Y)",
                ast         "@jsym(*)",
                bot         "@jsym(^)",
                bullet      "@jsym(U)",
        circ        "#@begin{r, script -3pt, size +2}@jsym{N}@end{r}#",
        cset        "@begin{t, size +2}C@end{t}@hsp{-5.5pt}@~
                     @begin{r, script +2pt, size -4}@vbar@end{r}",
                degr        "@jsym(N)",
                delta       "@g(D)",
                div         "@jsym(/)",
                downarrow   "@jsym(M)",
                dvbar       "@jsym(|)@jsym(|)",
                emptyset    "@jsym(j)",
                eq          "#@jsym(=)#",
                eqv         "#@jsym(X)#",
                exists      "@ksym(t)",
                forall      ‘@jsym(")’,
                gt          "#@r(>)#",
                gte         "#@jsym(Q)#",
gtlt  "#@begin{r, script -1pt}@begin{r, script +4pt}>@end{r}@~
       @hsp{-6pt}@r{<}@end{r}#",
hbar  "h@begin{r, script +5pt, slant -75, size 1}@hsp{-4pt}|@end{r}#",
                in          "#@jsym(r)#",
                infty       "@jsym(C)",
                inter       "@jsym(k)",
                langle      "@ksym(K)",
                leftarrow   "#@jsym(J)#",
                li          "@ovp{@f3{o}}@int",
                lt          "#@r(<)#",
                lte         "#@jsym(A)#",
ltgt  "#@begin{r, script -1pt}@begin{r, script +4pt}<@end{r}@~
       @hsp{-6pt}>@end{r}#",
mdot  "@begin{jsym, script +2pt, size -4}U@end{jsym}",
       mp    "@add@begin{r, script +3.5pt}@hsp{-7pt}@jsym{-}@end{r}#",
                mult        "#@jsym(R)#",
                nabla       "@jsym(u)",
                neq         "#@jsym(W)#",
                neqv        "#@ovp(@hsp[-300rasters]@eqv)@~
                             @begin(transparent,script -.034lines)@div@~
                             @end(transparent)#",
                not         "#@ksym(B)#",
                notin       "#@jsym(s)#",
nset    "@begin{t, size +2}N@end{t}@hsp{-7.5pt}@~
         @begin{r, script +1.4pt, size -2}@vbar@end{r}",
odiv    "#@ovp[@begin{r, size -1}@f3{O}@end{r}]@hsp{+3.5pt}@~
         @begin{r, script +1pt, size -2}@jsym{B}@end{r}###",
odot    "#@ovp[@begin{r, size -1}@f3{O}@end{r}]@hsp{+3.3pt}@~
         @begin{r, script +2.5pt, size 3}@bullet@end{r}###",
ominus  "#@begin{r, size -1}@f3{O}@end{r}@hsp{-8pt}@~
         @begin{transparent, script +0.75pt}@sub@end{transparent}#",
                op          "@b(OP)",
                oplus       "#@jsym(i)#",
```

```
                or          "#@ksym(D)#",
                otimes      "#@jsym(h)#",
                partial     "@jsym(T)",
                pm          "#@jsym(O)#",
                prsubset    "#@jsym(p)#",
                prsupset    "#@jsym(m)#",
                qed         "#@qedsym()##",
qset    "@begin{t, size +2}Q@end{t}@hsp{-5.5pt}@~
        @begin{r, script +2pt, size -4}@vbar@end{r}",
                rangle      "@ksym(Z)",
                rightarrow  "#@jsym(L)#",
rset    "@begin{t, size +2}R@end{t}@hsp{-7.5pt}@~
        @begin{r, script +1.4pt, size -2}@vbar@end{r}",
                simeq       "#@jsym(Y)#",
                similar     "#@jsym(~)#",
                simpar      "#@ovp{@jsym{~}}@hsp{+1pt}@r{:}##",
                sqinter     "#@sqint{}##",
                squnion     "#@squnt{}##",
std     "#@begin{transparent, script +8pt, size -1}@ovp{@f3{o}}@~
        @begin{r, script +0.5pt, size -2}@hsp{-0.5pt}@y{M}@end{r}@~
        @end{transparent}##",
                sub         "#@jsym(-)#",
                subset      "#@jsym(q)#",
                supset      "#@jsym(n)#",
top     "@begin{transparent, script +4.5pt}@begin{y, size -3}M@~
        @end{y}@end{transparent}@begin{r, script +1.25pt, size -2}@~
        @hsp{-4.25pt}@vbar@end{r}##",
                union       "@jsym(l)",
                uparrow     "@jsym(K)",
uplus   "#@f3{U}@hsp{-6pt}@begin{transparent, script +1pt, size -5}@~
        @add@end{transparent}##",
                vbar        "@jsym(|)",
                zset        "@f3{Z}@hsp{-3pt}@f3{Z}",
                0           "@ovp{0}@hsp{+1pt}/#"]

@comment[Other special notation]

@commandstring[ arctg      "@r(arctg)#",
                atan       "@r(atan)#",
                cos        "@r(cos)#",
                cot        "@r(cot)#",
                csc        "@r(csc)#",
                deg        "@r(deg)#",
                det        "@r(det)#",
                inf        "@r(inf)#",
                lg         "@r(lg)#",
                lim        "@r(lim)#",
                liminf     "@r(liminf)#",
                limsup     "@r(limsup)#",
                ln         "@r(ln)#",
                log        "@r(log)#",
                log2       "@r(log)@down(2)#",
                max        "@r(max)#",
                min        "@r(min)#",
                mod        "@r(mod)#",
                sin        "@r(sin)#",
                sup        "@r(sup)#",
                tan        "@r(tan)#",
                tg         "@r(tg)#",
                trace      "@r(trace)#"]

@commandstring[ quad    "@hsp(1 quad)",
```

```
                get     "#@r(:)=#",
                ldots   "@math{@r[@ .@ .@ .@ ]}",
        cdots   "@begin[math,script -.028lines]@ :@ :@ :@ @end(math)",
                st      "@r(@+(st))",
                nd      "@r(@+(nd))",
                rd      "@r(@+(rd))",
                th      "@r(@+(th))"]

@comment[Simple macros]
```

Comment: When adding more TextForms remember that a TextForm can only have at most 10 names.

```
@textform[      abs         "@vbar@;#@parm(text)#@vbar",
                bigo        "@r(O)#(@parm(text))",
                ceiling     "@ksym(S)@parm(text)@ksym(h)",
                detrm       "#@r(det)(@parm(text))",
                exp         "#@r(exp)(@parm(text))",
                floor       "@ksym(U)@parm(text)@ksym(j)",
                gcd         "#@r(gcd)(@parm(text))",
                norm        "@vbar@vbar@;@parm(text)@vbar@vbar",
                omega       "@g(W)#(@parm(text))",
                overline    "@ovp(@parm(text))@~
                             @begin(transparent,script +.17lines)@~
                             -@end(transparent)"]

@textform[     nsqrt       "@begin{jsym, slant -3,}z@end{jsym}@~
                            @begin{mathemarea}
                            @mathbar
                            @parm{text}
                            @end{mathemarea}",

                msqrt       "@begin{jsym, slant -3, size +2}z@end{jsym}@~
                            @begin{mathemarea}
                            @mathbar
                            @parm{text}
                            @end{mathemarea}",

                tsqrt       "@begin{jsym, slant -3, size +3}z@end{jsym}@~
                            @begin{mathemarea}
                            @mathbar
                            @parm{text}
                            @end{mathemarea}",

ttsqrt          "@begin{transparent, script -0.3line, slant -3, size +17}@~
                 @jsym{z}@end{transparent}@~
                 @begin{mathemarea}
                 @mathbar
                 @parm{text}
                 @end{mathemarea}"]

@textform[      overlinecap     "@ovp<@parm(text)>@~
                                 @begin(transparent,script 1line)@~
                                 @ux<@quad>@end(transparent)",

                overwordcap     "@begin{mathemarea}
                                 @mathbar
                                 @parm{text}
                                 @end{mathemarea}",
```

```
                scr     "@begin{f7, size +2}@parm{text}@end{f7}",

                sqrt    "@begin(jsym,slant 3)z@end(jsym)@~
                         @begin(mathemarea)
                         @mathbar
                         @parm(text)
                         @end(mathemarea)",

                cbrt    "@upp(3)@hsp(-.17lines)@begin(jsym,slant 3)z@~
                         @end(jsym)@~
                         @begin(mathemarea)
                         @mathbar
                         @parm(text)
                         @end(mathemarea)",

                theta   "@g(Q)#(@parm(text))",

                vec     "@ovp(@parm(text))@~
                         @begin(jsym,size 6,script +.27lines)L@end(jsym)"]

@comment[Multi-line Forms]

@form(  sumdisplay

        "@begin(mathemarea,centered,script -.087lines)
         @begin(mathemsmall)@parm(to, default=[])@end(mathemsmall)
         @begin(ksym,size 16)O@end(ksym)@markbaseline()
         @begin(mathemsmall,script -.6lines)@parm(from, default=[])@~
         @end(mathemsmall)
         @end(mathemarea)#")

@form(   proddisplay

        "@begin(mathemarea,centered,script -.087lines)
         @begin(mathemsmall)@parm(to, default=[])@end(mathemsmall)
         @begin(jsym,size 16)y@end(jsym)@markbaseline()
         @begin(mathemsmall,script -.6lines)@parm(from, default=[])@~
         @end(mathemsmall)
         @end(mathemarea)#")

@form(  intdisplay

        "@begin(ksym,size 20,script -.22lines)a@end(ksym)@~
         @begin(mathemarea,nofill,use mathemsmall,script 1line,
                spacing 2.25lines)
         @hsp(+2pts)@parm(to, default=[])@markbaseline()
         @parm(from, default=[])
         @end(mathemarea)")

@form(  overdisplay

        "@begin(mathemarea,centered,script +.16lines)
         @parm(num)
         @mathbar@markbaseline()
         @parm(denom)
         @end(mathemarea)")

@form(  choosedisplay

        "@begin(ksym,size 26,script -.087lines)n@end(ksym)@~
         @begin(mathemarea,centered,script +.26lines,spacing 1line)
         @parm(from)@markbaseline()
```

```
         @parm(chosen)
         @end(mathemarea)@~
         @begin(ksym,size 26,script -.087lines)o@end(ksym)")

@form(  ss

        "@begin(mathemarea,nofill,use mathemsmall,script +0.5 lines,
                spacing 1 line,makescript)
         @parm(super)@markbaseline()
         @parm(sub)
         @end(mathemarea)")

@form(  loopintdisplay

        "@ovp{@begin(ksym,size 20,script -.22lines)a@end(ksym)}@f3(o)@~
         @begin(mathemarea,nofill,use mathemsmall,script 1line,
                spacing 2.25lines)
         @hsp(+2pts)@parm(to, default=[])@markbaseline()
         @parm(from, default=[])
         @end(mathemarea)")

@form( nestsumdisplay

       "@begin{mathemarea, centered}
        @begin{mathemsmall}@parm{to, default=[]}@end{mathemsmall}
        @begin{g, size +10, script -4pt}L@end{g}@markbaseline{}
        @begin{mathemsmall}@parm{from, default=[]}@end{mathemsmall}
        @end{mathemarea}#")

@form(  sumtext

        "@begin(ksym,size 15,script -.07)O@end(ksym)@~
         @ss(sub [@parm(from,default=<>)],
             super [@parm(to,default=<>)])#")

@form(  prodtext

        "@begin(jsym,size 15,script -.07)y@end(jsym)@~
         @ss(sub [@parm(from,default=<>)],
             super [@parm(to,default=<>)])#")

@form(  inttext

        "@begin(ksym,size 16,script -.18lines)a@end(ksym)@~
         @begin(mathemarea,nofill,use mathemsmall,script +0.5 lines,
                spacing 1.6 lines,makescript)
         @hsp(2pts)@parm(to,default=[])@markbaseline()
         @parm(from,default=[])#
         @end(mathemarea)")

@form(  overtext

        "@begin(mathemarea,use mathemsmall,centered,script +.21lines)
         @parm(num)
         @mathbar@markbaseline()
         @parm(denom)
         @end(mathemarea)")

@form(  choosetext

        "@begin(transparent,size 16,script -.2em)(@end(transparent)@~
         @begin(mathemarea,centered,use mathemsmall,script +.17line)
```

```
         @parm(from)@markbaseline()
         @parm(chosen)
         @end(mathemarea)@~
         @begin(transparent,size 16,script -.2em))@end(transparent)")

@form( loopinttext

"@begin{transparent, script -1pt}@ovp{@f3(o)}@hsp{+0.5pt}@end{transparent}
        @inttext(      from [@parm{from, default=[]}],
                        to [@parm{to, default=[]}])")

@form( nestsumtext

       "@begin{g, size +9, script -3pt}L@end{g}#@~
        @ss(   sub [@parm{from, default=[]}],
                super [@parm{to, default=[]}])#")

@form(  brace

        "@begin(ksym,size 24,script -.1line)r@end(ksym)@~
         @begin(mathemarea,nofill,script +.13lines)
         @parm(top)@markbaseline()
         @parm(bot)
         @end(mathemarea)")

@form(  limit

        "@begin(mathemarea,centered)
         @r(lim)@markbaseline()
         @begin(mathemsmall,script +.04lines)@parm(as)@end(mathemsmall)
         @end(mathemarea)#")

@form(  smallfraction

        "@begin(mathemarea,use mathemsmall,centered)
         @parm(num)
         @mathbar
         @parm(denom)
         @end(mathemarea)")
```

Appendix D: The NTABLE.LIB Senslab Tabular Generation File

```
@Marker(library,ntable)
@comment[Copyright 12/88, Alan D. Guisewite]
@style{stringmax 10000}
@form(YKKQdfr / "@~
@definerowformat[@parm(name),
                 @imbed(boxtype,def "boxtype @parm(boxtype),")
                 @imbed(flushtop,def "flushtop,")
                 @imbed(flushbottom,def "flushbottom,")
                 columns (
@imbed(YKKQ2,def "column @parm(first),column @parm(last)")
@imbed(YKKQ3,def "column @parm(first),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ4,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ5,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ6,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ7,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ8,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ9,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ10,def "column @parm(first),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(last)")
@imbed(YKKQ11,def "column @parm(first),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(last)")
@imbed(YKKQ12,def "column @parm(first),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(last)")
@imbed(YKKQ13,def "column @parm(first),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(last)")
@imbed(YKKQ14,def "column @parm(first),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
```

```
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ15,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ16,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ17,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ18,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ19,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(last)")
@imbed(YKKQ20,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(last)")
@imbed(YKKQ21,def "column @parm(first),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
```

```
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(last)")
@imbed(YKKQ22,def "column @parm(first),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(last)")
@imbed(YKKQ23,def "column @parm(first),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(last)")
@imbed(YKKQ24,def "column @parm(first),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(last)"))]")

@form(YKKQdfrs / "@~
@YKKQdfr[name=YKQ@parm(name),
         @imbed(boxed,def "boxtype standardbox,",
         undef / "
         @imbed(rowlined,def "boxtype standardrowlines,",
         undef / "
         @imbed(columnlined,def "boxtype standardcolumnlines,")
         ")")
         @imbed(flushtop,def "flushtop,")
         @imbed(flushbottom,def "flushbottom,")
         YKKQ@parm(columns)=!,
first=@parm(name)first,other=@parm(name)other,last @parm{name}last]
@YKKQdfr[name=@parm(name)columnheadings,
         @imbed(boxed,def "boxtype standardbox,",
         undef / "
         @imbed(headingboxed,def "boxtype standardbox,")
         ")
         @imbed(flushtop,def "flushtop,")
         @imbed(flushbottom,def "flushbottom,")
```

```
         YKKQ@parm(columns)=!,
first=@parm(name)colhead,other=@parm(name)colhead,last @parm{name}colhead]
@YKKQdfr[name=@parm(name)topheading,
         @imbed(boxed,def "boxtype standardbox,",
         undef / "
         @imbed(headingboxed,def "boxtype standardbox,")
         ")
         YKKQ@parm(columns)=!,
         first=@parm(name)tophead,other=combined,last combined]")

@form(standardtable / "@~
@define[@parm(name)first=
   @imbed(allcolumns,def "@parm(allcolumns)",
   undef / "
   @imbed(firstcolumn,def "@parm(firstcolumn)",
   undef / "
   flushleft
   ")")
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(firstcolumnwidth,def
      ",linewidth @parm(firstcolumnwidth)
       ,maxwidth @parm(firstcolumnwidth)
       ,minwidth @parm(firstcolumnwidth)"
   )")]
@define[@parm(name)other=
   @imbed(allcolumns,def "@parm(allcolumns)",
   undef / "
   @imbed(othercolumns,def "@parm(othercolumns)",
   undef / "
   flushleft
   ")")
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(othercolumnwidth,def
      ",linewidth @parm(othercolumnwidth)
       ,maxwidth @parm(othercolumnwidth)
       ,minwidth @parm(othercolumnwidth)"
   )")]
@define[@parm(name)last=
   @imbed(allcolumns,def "@parm(allcolumns)",
   undef / "
   @imbed(othercolumns,def "@parm(othercolumns)",
   undef / "
   flushleft
   ")")
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(lastcolumnwidth,def
      ",linewidth @parm(lastcolumnwidth)
       ,maxwidth @parm(lastcolumnwidth)
       ,minwidth @parm(lastcolumnwidth)"
   )")]
```

```
@define[@parm(name)colhead=
   @imbed(columnheadings,def "@parm(columnheadings)",
   undef / "
   center
   ")]
@define[@parm(name)tophead=
   @imbed(topheading,def "@parm(topheading)",
   undef / "
   center
   ")]
@YKKQdfrs[name=@parm(name),
          @imbed(boxed,def "boxed,")
          @imbed(headingboxed,def "headingboxed,")
          @imbed(rowlined,def "rowlined,")
          @imbed(columnlined,def "columnlined,")
          @imbed(flushtop,def "flushtop,")
          @imbed(flushbottom,def "flushbottom,")
          columns=@parm(columns)]
@define(@parm(name)=center,
        tablecolumns=YKQ@parm(name),
        @imbed(boxed,def "use box,",
        undef / "
        @imbed(headingboxed,def "use box,",
        undef / "
        @imbed(rowlined,def "use box,",
        undef / "
        @imbed(columnlined,def "use box,")
        ")")")
        use standardboxmargins)")

@comment[Determinant libraryfile]
@comment[Copyright February, 1989, Alan D. Guisewite]

@style{stringmax 16000}
@definelinetype{matrixdoublelinetype, weight 400, groove 200}
@definelinetype{determinantlinetype, weight 72}
@define{mathboxmargins standardboxmargins, boxlm 10pt, boxrm 10pt}

@form(YAKKQdfr / "@~
@definerowformat[@parm(name),
                 columns (
@imbed(YAKKQ2,def "column @parm(first),column @parm(last)")
@imbed(YAKKQ3,def "column @parm(first),
                   linetype determinantlinetype,
                   column @parm(other),
                   linetype determinantlinetype,
                   column @parm(last)")
@imbed(YAKKQ4,def "column @parm(first),
                   linetype determinantlinetype,
                   column @parm(other),column @parm(other),
                   linetype determinantlinetype,
                   column @parm(last)")
@imbed(YAKKQ5,def "column @parm(first),
                   linetype determinantlinetype,
           column @parm(other),column @parm(other),column @parm(other),
                   linetype determinantlinetype,
                   column @parm(last)")
@imbed(YAKKQ6,def "column @parm(first),
                   linetype determinantlinetype,
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   linetype determinantlinetype,
```

```
                  column @parm(last)")
@imbed(YAKKQ7,def "column @parm(first),
                  linetype determinantlinetype,
          column @parm(other),column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  linetype determinantlinetype,
                  column @parm(last)")
@imbed(YAKKQ8,def "column @parm(first),
                  linetype determinantlinetype,
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  linetype determinantlinetype,
                  column @parm(last)")
@imbed(YAKKQ9,def "column @parm(first),
                  linetype determinantlinetype,
                  column @parm(other),column @parm(other),
          column @parm(other),column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  linetype determinantlinetype,
                  column @parm(last)")
@imbed(YAKKQ10,def "column @parm(first),
                   linetype determinantlinetype,
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   linetype determinantlinetype,
                   column @parm(last)"))]")

@form(YAKKQdfrs / "@~
@YAKKQdfr[name YAKQ@parm(name),
         YAKKQ@parm(columns) !,
         first @parm(name)first,
         other @parm(name)other,
         last @parm{name}last]")

@form(determtable / "@~
@define[@parm(name)first center, spaces kept
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(firstcolumnwidth,def
      ",linewidth @parm(firstcolumnwidth)
       ,maxwidth @parm(firstcolumnwidth)
       ,minwidth @parm(firstcolumnwidth)"
   )")]
@define[@parm(name)other center
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(othercolumnwidth,def
      ",linewidth @parm(othercolumnwidth)
       ,maxwidth @parm(othercolumnwidth)
       ,minwidth @parm(othercolumnwidth)"
   )")]
@define[@parm(name)last center, spaces kept
   @imbed(columnwidth,def
```

```
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(lastcolumnwidth,def
      ",linewidth @parm(lastcolumnwidth)
       ,maxwidth @parm(lastcolumnwidth)
       ,minwidth @parm(lastcolumnwidth)"
   )")]
@YAKKQdfrs[name @parm(name), columns @parm(columns)]
@define(@parm(name) mathdisplay,
        tablecolumns YAKQ@parm(name),
        beforeentry '@blankspace{1.1}',        spread 1.1line,
        use mathboxmargins)")

@comment[Matrix libraryfile]
@comment[Copyright February, 1989, Alan D. Guisewite]

@style{stringmax 16000}
@definelinetype{matrixdoublelinetype, weight 400, groove 200}
@definelinetype{determinantlinetype, weight 50}
@define{mathboxmargins standardboxmargins, boxlm 10pt, boxrm 10pt}

@form(YBKKQdfr / "@~
@definerowformat[@parm(name),
                 columns (
@imbed(YBKKQ2,def "column @parm(first),column @parm(last)")
@imbed(YBKKQ3,def "column @parm(first),
                   linetype matrixdoublelinetype,
                   column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ4,def "column @parm(first),
                   linetype matrixdoublelinetype,
                   column @parm(other),column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ5,def "column @parm(first),
                   linetype matrixdoublelinetype,
          column @parm(other),column @parm(other),column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ6,def "column @parm(first),
                   linetype matrixdoublelinetype,
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ7,def "column @parm(first),
                   linetype matrixdoublelinetype,
          column @parm(other),column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ8,def "column @parm(first),
                   linetype matrixdoublelinetype,
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   column @parm(other),column @parm(other),
                   linetype matrixdoublelinetype,
                   column @parm(last)")
@imbed(YBKKQ9,def "column @parm(first),
```

```
                  linetype matrixdoublelinetype,
                  column @parm(other),column @parm(other),
           column @parm(other),column @parm(other),column @parm(other),
                  column @parm(other),column @parm(other),
                  linetype matrixdoublelinetype,
                  column @parm(last)")
@imbed(YBKKQ10,def "column @parm(first),
                    linetype matrixdoublelinetype,
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    column @parm(other),column @parm(other),
                    linetype matrixdoublelinetype,
                    column @parm(last)"))]")

@form(YBKKQdfrs / "@~
@YBKKQdfr[name YBKQ@parm(name),
          YBKKQ@parm(columns) !,
          first @parm(name)first,
          other @parm(name)other,
          last @parm{name}last]")

@form(matrixtable / "@~
@define[@parm(name)first center, spaces kept
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(firstcolumnwidth,def
      ",linewidth @parm(firstcolumnwidth)
       ,maxwidth @parm(firstcolumnwidth)
       ,minwidth @parm(firstcolumnwidth)"
   )")]
@define[@parm(name)other center
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(othercolumnwidth,def
      ",linewidth @parm(othercolumnwidth)
       ,maxwidth @parm(othercolumnwidth)
       ,minwidth @parm(othercolumnwidth)"
   )")]
@define[@parm(name)last center, spaces kept
   @imbed(columnwidth,def
      ",linewidth @parm(columnwidth)
       ,maxwidth @parm(columnwidth)
       ,minwidth @parm(columnwidth)",
   undef / "
   @imbed(lastcolumnwidth,def
      ",linewidth @parm(lastcolumnwidth)
       ,maxwidth @parm(lastcolumnwidth)
       ,minwidth @parm(lastcolumnwidth)"
   )")]
@YBKKQdfrs[name @parm(name), columns @parm(columns)]
@define(@parm(name) mathdisplay,
        tablecolumns YBKQ@parm(name),
        beforeentry '@blankspace{1.1}',        spread 1.1line,
        use mathboxmargins)")
```

Appendix E: The TFIGUR.LIB Senslab Figure/Table Environment File

```
@Marker(Library,TFigures)
```

Comment: The @Generate and @Send commands create the Illustrations and Tables pages and send headings and pagination information to them.

```
@Generate(Illustrations,Tables)
@Send(Illustrations "@Style(PageNumber <@i>)")
@Send(Illustrations "@PrefaceSection(Illustrations@blankspace{2}@begin{b,
flushleft, leftmargin 3, size -2}Figures@end{b})")
@Send(Illustrations "@define(foot,invisible)")
@Send(Tables "@define(foot,invisible)")
@Send(Tables "@Style(PageNumber <@i>)")
@Send(Tables "@PrefaceSection(Tables)")
```

Comment: The FigureCounter counter numbers each figure and sends caption and textual information to their proper places.

```
@Counter(FigureCounter,        Table "Illustrations",
                                ContentsEnv tc2,
                ContentsForm   "@begin@parmquote(ContentsEnv)
                                @rfstr{@parm{page}}
                        @imbed{numbered, @def [@parm{numbered}@|@$]}
                                @parm{referenced}@hsp{10pt}@parm(title)
                                @end@parmquote(ContentsEnv)",
                TitleForm      "@begin(CaptionEnv)
                                @center[@parm{numbered}@c{@parm{title}}]
                                @end(CaptionEnv)",
                                Within Chapter,
Numbered [@@begin{b, size -1}Figure@@ @#@:-@1.@@ @@ @@end{b}@@$],
                                Referenced [@#@:-@1],
                                IncrementedBy Use,
                                Init 0)
```

Comment: This definition sets the environment for the figures.

```
@Define(Figure,         Nofill,
                         Spaces Kept,
                         Use R,
                         use center,
                         leftmargin 0",
                         rightmargin 0",
                         BlankLines kept,
                         Float,
                         Above 1,
                         Below 2,
                         Counter FigureCounter,
                         NumberLocation LFL)
```

Comment: Similarly to the FigureCounter counter, the TableCounter counter numbers each table and sends caption and textual information to their proper places.

```
@Counter(TableCounter,         Table "Tables",
```

```
                        ContentsEnv tc2,
                        TitleEnv Legend,
            ContentsForm   "@begin(TCaptionEnv)
                        @rfstr{@parm{page}}
                  @imbed{numbered, @def [@parm{numbered}@|@$]}
                        @parm{referenced}@hsp{10pt}@parm(title)
                        @end(TCaptionEnv)",
            TitleForm     "@center{@parm{numbered}}
                        @$",
                        Within Chapter,
      Numbered (@@begin{b, size -1}Table@@ @#@:-@1@@end{b}@@$),
                        Referenced [@#@:-@1],
                        IncrementedBy Use,
                        Init 0)
```

Comment: Again, as above, the table environment controls the overall appearance of the table. Here also is where the change in footnote symbols occurs.

```
@Define(Table,          Nofill, Use R,
                         LongLines wrap,
                         blanklines hinge,
                         scriptpush no,
                         indent -1.25",
        afterentry      '@counter{footnotecounter,
                                numbered <@@+[@$"@~
                         @jsym{*}@,@n{d}@,@n{D}@,@~
                         @n{S}@,@a{@dvbar}@,@jsym{#}@,@~
                         @jsym{**}@,@n{dd}@,@n{DD}@,@~
                         @n{SS}@,@a{@dvbar @dvbar}@,@jsym{##}@,@~
                         @jsym{***}@,@n{ddd}@,@n{DDD}@,@~
                         @n{SSS}@,@a{@dvbar @dvbar @dvbar}@,@jsym{###}"]>,
                                referenced <@@+[@$"@~
                         @jsym{*}@,@n{d}@,@n{D}@,@~
                         @n{S}@,@a{@dvbar}@,@jsym{#}@,@~
                         @jsym{**}@,@n{dd}@,@n{DD}@,@~
                         @n{SS}@,@a{@dvbar @dvbar}@,@jsym{##}@,@~
                         @jsym{***}@,@n{ddd}@,@n{DDD}@,@~
                         @n{SSS}@,@a{@dvbar @dvbar @dvbar}@,@jsym{###}"]>}',
                         Spread 0.8, Above 2, Below 2,
                         Counter TableCounter, NumberLocation LFL)

@Define(FullPageFigure,         Use Figure,
                                 FloatPage)

@Define(FullPageTable,          Use Table,
                                 FloatPage)
```

Comment: These two commands set the environments for the figure and table caption appearance on the Contents page. The textual captions (legends) are defined in the **TECHRE.MAK** file.

```
@Define(CaptionEnv      Center,
                         Use R,
                         Spacing 1.1,
                         Font BodyFont)

@Define(TCaptionEnv     tc2)
```

Appendix F: The TECHTI.LIB Senslab Title Page Environment File

```
@marker{library, techtitle, PostScript, scaleablelaser}
```

Comment: This definition sets the environment for the @Title command.

```
@define[techtitle,     centered,
                        pagebreak around,
                        break,
                        longlines wrap,
                        leftmargin 16mm,
                        rightmargin 16mm,
                        spacing 1.1,
                        blanklines kept,
                        sink 60mm]
```

Comment: This generates an index file only if the TechTitle Libray File is called.

```
@generate{filedex ".index"}
```

Comment: These next two definitions set up the two-sided box for the @Archive textform.

```
@definebox{    cornerbox,
                left standardlinetype,
                bottom standardlinetype}

@define{       corner,
                boxtype cornerbox,
                use standardboxmargins}
```

Comment: The @Archive textform controls the appearance of the upper right corner of the title page with both environmental and textual commands.

```
@textform[archive      '@begin{corner,
                               fixed 0.25",
                               break,
                               longlines wrap,
                               blanklines ignore,
                               leftmargin 125mm,
                               rightmargin -35mm,
                               flushleft}
                        @begin{b, size +6}FILE COPY@end{b}
                        @value{date}
                        @value{fullmanuscript}
                        @b{Notes:} @parm{text}
                        @end{corner}']
```

Comment: The next three forms control the appearance of the @Title, @Author, and @Report commands including an @Case change for draft copies.

```
@textform[title                ’@begin{b, size +2, spacing 1.3}
                        @parm{text}
                        @send{filedex @parm{text}}
                        @end{b}’]

@textform[author       ’@blankspace{-0.8line}
                        @begin{center, size +1}
                        @parm{text}
                        @end{center}’]

@form[report           ’CMU-RI-@parm{ID, default "TR"}-@parm{#}
                @send{filedex CMU-RI-@parm{ID, default "TR"}-@parm{#}}
                        @blankspace{0.5"}
@case[ draft,yes
                (DRAFT: @value{date}, @value{time}@*
                        @value{fullmanuscript}),
        else "@blankspace{2.5line}"]
                        @blankspace{0.8"}
                        @begin{r, size +1}
                Intelligent Sensors Laboratory @parm{type, default ""}

                        The Robotics Institute
                        Carnegie Mellon University
                        Pittsburgh, PA 15213

                        @value{month} @value{year}
                        @end{r}
```

Comment: The Message command displays its argument on the CRT during Scribe compilation.

```
@message
{
        [Report Type "@parm{type, default "Technical Report"}"]

}’]
```

Comment: These three definitions control the appearance of the @CopyrightNotice, @Copyright, and @ResearchCredit commands.

```
@define{copyrightnotice,       fixed 8",
                        afterentry "Copyright @y{C} @value{year}@: "}

@define{copyright,     fixed 8",
                        afterentry "Copyright @y{C} @: "}

@define{researchcredit,                fixed 8.5",
                                fill,
                                centered,
                                leftmargin 0,
                                rightmargin 0,
                                justification,
                                spacing 1.1}
```

Comment: As above, another CRT message notifying the creation of the index file.

```
@message
{
        [Index file "manuscript_filename.index" created.]
}
```

Index

Contents

Illustrations

Figures

Tables