```
% LATEX VERSION 2.09 <25 March 1992>
% Copyright (C) 1992 by Leslie Lamport

\everyjob{\typeout{LaTeX Version 2.09 <25 March 1992>}}
\immediate\write10{LaTeX Version 2.09 <25 March 1992>}

%                 TABLE OF CONTENTS
% COMMAND LIST ..................................... 2
% GENERAL CONVENTIONS .............................. 6
% COUNTERS, ETC. ................................... 7
% USEFUL HACKS ..................................... 8
% ERROR HANDLING ................................. 12
% \par AND \everypar .............................. 15
% SPACING / LINE AND PAGE BREAKING ................ 17
% PROGRAM CONTROL STRUCTURE MACROS ................ 21
% FILE HANDLING ................................... 24
% ENVIRONMENT COUNTER MACROS ...................... 27
% PAGE NUMBERING .................................. 30
% CROSS REFERENCING MACROS  ....................... 31
% ENVIRONMENTS .................................... 33
% MATH ENVIRONMENTS ............................... 36
% CENTER, FLUSHRIGHT, FLUSHLEFT, ETC. ............. 39
% VERBATIM ......................................... 40
% THE LIST ENVIRONMENT ............................ 41
% ITEMIZE AND ENUMERATE ........................... 49
% BOXES ............................................ 51
% THE TABBING ENVIRONMENT ......................... 57
% ARRAY AND TABULAR ENVIRONMENTS .................. 63
% THE PICTURE ENVIRONMENT ......................... 72
% THEOREM ENVIRONMENTS ............................ 86
% LENGTHS .......................................... 88
% THE TITLE .......................................89
% SECTIONING ....................................... 90
% TABLE OF CONTENTS, ETC. ......................... 94
% INDEX COMMANDS ................................... 97
% BIBLIOGRAPHY ..................................... 98
% FLOATS ........................................... 100
% FOOTNOTES ........................................ 106
% INITIAL DECLARATION COMMANDS ..................... 110
% OUTPUT ........................................... 113
% DEBUGGING AND TEST INITIALIZATIONS  ............... 137


\catcode`\~=13 \def~{\penalty\@M \ }


%     ****************************************
%     *           COMMAND LIST               *
%     ****************************************
%
% DECLARATIONS:
%  PREAMBLE:     \nofiles \documentstyle \includeonly
%                \makeindex \makeglossary
%  IN DOCUMENT :
%    FONT SELECTION:
%         SIZE: \normalsize \small \footnotesize \scriptsize \tiny
%               \large \Large \LARGE \huge \Huge
%         STYLE: \bf \it \rm \sl \ss \tt \mit[math mode only]
%    STYLE:
%         PAGE: [all global] \pagestyle \thispagestyle \pagenumbering \head
%         MISC: \raggedright \thicklines \thinlines
```

```
%    PARAMETER: \setlength \settowidth \addtolength \setcounter \addtocounter
%    NEW:       \newlength \newtheorem \newcommand
%    MISC:      \savebox \sbox \obeycr \restorecr
%
% ENVIRONMENTS:
%    ?   -> PAR:  document
%   PAR  -> PAR:  list enumerate itemize description
%                 center flushright flushleft
%                 verbatim picture float
%   PAR  -> BOX:  tabular tabbing
%   PAR  -> MATH: math displaymath equation
%   MATH -> MATH: array
%   ANY  -> PAR:  minipage
%   ANY  -> BOX:  stack
%
% TEXT-PRODUCING:
%   WITH TEXT ARGUMENT:
%      ANY -> BOX: \makebox \mbox \framebox \fbox \dashbox
%                  \shortstack \footnotemark \cite[] \raisebox
%      ANY -> PAR: \parbox[inner]
%      PAR -> PAR: \chapter \section ... \footnote \footnotetext
%                  \topnewpage \verb
%      MATH:       \sqrt \underline \overline
%      PICTURE:    \put \multiput
%      LIST:       \item
%   WITHOUT TEXT ARGUMENT:
%     ANY MODE:
%          SYMBOLS: \$ \{ \} \_ \@ \& \#
%          ACCENTS: See TeXbook
%          OTHER:   \rule \ref \pageref \today \usebox \typein \input \cite
%     MATH:     \over
%     PAR MODE: \include \bibliography \tableofcontents \listoffigures ...
%     LIST:     \item \arabic \roman \Roman \alph \Alph
%     PICTURE:  \line \vector \circle \oval
%     ARRAY & TABULAR: \hline \vline
%
% SPACING & BREAKING:
%    ANY       : \hfill \hspace
%    PAR       : \newpage \newpage \vspace \noindent
%    PAR & INNER MATH
%              : \newpage \clearpage \cleardoublepage
%              : \pagebreak \nopagebreak \linebreak \nolinebreak \newline
%    MATH      : \over \; \, \!
%    MULTILINE : \\
%    TABBING   : \pushtab \poptab \> \< \+ \- \kill ...
%    ARRAY & TABULAR
%              : \multicolumn \noalign
%
% NO DIRECT CHANGES TO DOCUMENT:
%    \index \glossary \typeout \label \tableentry \stop \protect
%
% PARAMETERS:
%
% \columnsep         \skip\footin        \intextsep
% \columnseprule                         \oddsidemargin
% \columnwidth                           \textfloatsep
% \evensidemargin    \footsep            \textheight
% \floatsep          \headheight         \textwidth
% \headsep            \topmargin
```

```
%   ALPHABETIZED LIST:
%
% ORDINARY COMMANDS:
%
% \Alph                 \include            \parbox
% \Roman                \index              \put
% \\                    \item               \raisebox
% \alph                 \label              \ref
% \appendix             \line               \roman
% \arabic               \linebreak          \rule
% \bibliography         \listoffigures      \section
% \chapter              \listoftables       \shortstack
% \circle               \makebox            \stop
% \cite                 \mbox               \subsection
% \cite                 \multicolumn        \subsubsection
% \cleardoublepage      \multiput           \tableentry
% \clearpage                                \tableofcontents
% \dashbox              \newline            \today
% \fbox                 \newpage            \typein
% \footnotemark         \noindent           \typeout
% \footnotetext         \nolinebreak        \usebox
% \framebox             \nopagebreak        \vector
% \glossary             \oval               \vline
% \hline                \pagebreak          \vspace
% \hspace               \pageref            \protect
%
%
% ENVIRONMENTS & DECLARATIONS:
%
% For each of these commands, the same command name prefixed by 'end'
% is also reserved--e.g., \enddocument.
%
% \BIG                \footnotesize       \pagestyle
% \Big                \head               \picture
% \addtocounter       \includeonly        \raggedright
% \addtolength        \itemize            \restorecr
% \array              \list               \savebox
% \big                \makeglossary       \sbox
% \center             \makeindex          \scriptscriptsize
% \description        \math               \scriptsize
% \displaymath        \minipage           \setcounter
% \document           \newcommand         \setlength
% \documentstyle      \newlength          \settowidth
% \enumerate          \newtheorem         \small
% \equation           \nofiles            \shortstack\tabbing
%                     \normalsize             \tabular
% \float              \obeycr             \thicklines
% \flushleft          \pagelayout         \thinlines
% \flushright         \pagenumbering      \thispagestyle
%                                         \verb, \verbatim
%
% PARAMETERS :
%
% \columnsep          \footinsertskip     \intextsep
% \columnseprule                          \oddsidemargin
% \columnwidth                            \textfloatsep
% \evensidemargin     \footsep            \textheight
% \floatsep           \headheight         \textwidth
% \headsep             \topmargin
%
%
```

```
% TABBING COMMANDS:
%
% These commannds are defined only within a tabbing environment.
%
% \kill      \>    \-
% \pushtab   \<    \=
% \poptab    \+


% COMPLETE LIST :
% Below is a complete list of every command starting with `\' that
% appears in LATEX.TEX.

% \
% \!
% \#
% \$
% \&
% \'
% \(
% \)
% \+
% \,
% \-
% \.
% \:
% \;
% \<
% \=
% \>
% \@
% \@@
% \@@end
% \@@endpbox
% \@@eqncr
% \@@hyph
% \@@input
% \@@par
% \@@sqrt
% \@@startpbox
% \@@underline
% \@@warning
% \@acci
% \@accii
% \@acciii
% \@acol
% \@acolampacol
% \@addamp
% \@addfield
% \@addmarginpar
% \@addtobot
% \@addtocurcol
% \@addtodblcol
% \@addtonextcol
% \@addtopreamble
% \@addtoreset
% \@addtotoporbot
% \@afterheading
% \@afterindentfalse
% \@afterindenttrue
% \@Alph
```

```
% \@alph
% \@ampacol
% \@arabic
% \@argarraycr
% \@argdef
% \@argrsbox
% \@argtabularcr
% \@array
% \@arrayacol
% \@arrayclassiv
% \@arrayclassv
% \@arrayclassz
% \@arraycr
% \@arrayparboxrestore
% \@arrayrule
% \@arstrut
% \@arstrutbox
% \@auxout
% \@badcrerr
% \@badend
% \@badlinearg
% \@badmath
% \@badpoptabs
% \@badtab
% \@beginparpenalty
% \@begintheorem
% \@bibitem
% \@biblabel
% \@bitor
% \@botlist
% \@botnum
% \@botroom
% \@bsphack
% \@caption
% \@captype
% \@car
% \@carcube
% \@cclv
% \@cdr
% \@centercr
% \@centering
% \@cfla
% \@cflb
% \@charlb
% \@charrb
% \@chclass
% \@checkend
% \@chnum
% \@circ
% \@circle
% \@circlefnt
% \@cite
% \@citea
% \@citeb
% \@citex
% \@cla         % counter used in \cline
% \@classi
% \@classii
% \@classiii
% \@classiv
% \@classv
```

```
% \@classz
% \@clb         % counter used in \cline
% \@cline
% \@clnht
% \@clnwd
% \@clubpenalty
% \@colht
% \@colnum
% \@colroom
% \@combinedblfloats
% \@combinefloats
% \@comdblflelt
% \@comflelt
% \@cons
% \@contfield
% \@ctrerr
% \@curfield
% \@curline
% \@currbox
% \@currentlabel
% \@currentreference
% \@currenvir
% \@currlist
% \@currtype
% \@curtab
% \@curtabmar
% \@dascnt
% \@dashbox
% \@dashcnt
% \@dashdim
% \@dblarg
% \@dbldeferlist
% \@dblfloat
% \@dblfloatplacement
% \@dblfloatsep
% \@dblfpbot
% \@dblfpsep
% \@dblfptop
% \@dblmaxsep
% \@dbltextfloatsep
% \@dbltoplist
% \@dbltopnum
% \@dbltoproom
% \@deferlist
% \@definecounter
% \@defpar
% \@depth
% \@dischyph
% \@doclearpage
% \@documentstyle
% \@doendpe
% \@donoparitem
% \@dot
% \@dotsep
% \@dottedtocline
% \@downline
% \@downvector
% \@eha
% \@ehb
% \@ehc
% \@ehd
```

```
% \@elt
% \@empty
% \@endparenv
% \@endparpenalty
% \@endpbox
% \@endpefalse
% \@endpetrue
% \@endtabbing
% \@endtheorem
% \@enumctr
% \@enumdepth
% \@enumspacing
% \@eqncr
% \@eqnnum
% \@eqnsel
% \@eqnswtrue
% \@esphack
% \@Esphack
% \@evenfoot
% \@evenhead
% \@expast
% \@failedlist
% \@fcolmadefalse
% \@filesw
% \@fileswfalse
% \@fileswtrue
% \@firstampfalse
% \@firstamptrue
% \@firstcolumntrue
% \@firsttab
% \@flfail
% \@float
% \@floatpenalty
% \@floatplacement
% \@floatsep
% \@flsucceed
% \@fltovf
% \@flushglue
% \@fnsymbol
% \@footnotemark
% \@footnotetext
% \@for
% \@forloop
% \@fornoop
% \@fpbot
% \@fpmin
% \@fpsep
% \@fptop
% \@framebox
% \@framepicbox
% \@freelist
% \@getcirc
% \@getlarrow
% \@getlinechar
% \@getpen
% \@getrarrow
% \@glossaryfile
% \@gobble
% \@gobblecr
% \@gobbletwo
% \@gtempa
```

```
% \@halfwidth
% \@halignto
% \@hangfrom
% \@height
% \@highpenalty
% \@hightab
% \@hline
% \@holdpg
% \@hspace
% \@hspacer
% \@hvector
% \@icentercr
% \@iden
% \@ifatmargin
% \@ifdefinable
% \@ifnch
% \@ifnextchar
% \@iforloop
% \@iframebox
% \@iframepicbox
% \@ifstar
% \@ifundefined
% \@iinput            % used in \input
% \@iirsbox
% \@imakebox
% \@imakepicbox
% \@iminipage
% \@index
% \@indexfile
% \@inlabelfalse
% \@input
% \@inputcheck
% \@insertfalse
% \@inserttrue
% \@iparbox
% \@irsbox
% \@isavebox
% \@isavepicbox
% \@ishortstack
% \@istackcr
% \@itabcr
% \@item
% \@itemdepth
% \@itemfudge
% \@itemitem
% \@itemlabel
% \@itempenalty
% \@itemspacing
% \@iwhiledim
% \@iwhilenum
% \@iwhilesw
% \@ixstackcr
% \@killglue
% \@labels
% \@lastchclass
% \@latexbug
% \@latexerr
% \@lbibitem
% \@leftcolumn
% \@leftmarginskip
% \@leftmark
```

```
% \@lhead
% \@linechar
% \@linefnt
% \@linelen
% \@list
% \@listctr
% \@listdepth
% \@listi
% \@listii
% \@listvi
% \@lnbk
% \@lowpenalty
% \@lquote
% \@ltab
% \@M
% \@m
% \@mainaux
% \@mainout
% \@makebox
% \@makecaption
% \@makecol
% \@makefcolumn
% \@makefnmark
% \@makefntext
% \@makeonecolumn
% \@makeother
% \@makepicbox
% \@maketwocolumn
% \@marbox
% \@markright
% \@maxdepth
% \@maxsep
% \@maxtab
% \@medpenalty
% \@Mi
% \@midlist
% \@Mii
% \@Miii
% \@minipagefalse
% \@minipagerestore
% \@Miv
% \@mkboth
% \@mklab
% \@mkpream
% \@MM
% \@mparbottom
% \@mparswitchfalse
% \@mpfn
% \@mpfnnumber
% \@mpfootins
% \@mpfootnotetext
% \@mplistdepth
% \@multicnt
% \@namedef
% \@nameuse
% \@nbitem
% \@ne
% \@negargfalse
% \@negargtrue
% \@newctr
% \@newenv
```

```
% \@newline
% \@newlist
% \@newlistfalse
% \@next
% \@nextchar
% \@nextwhile
% \@nil
% \@nmbrlistfalse
% \@nmbrlisttrue
% \@nnil
% \@nobreakfalse
% \@nocnterr
% \@nodocument
% \@nofonterror
% \@noitemargfalse
% \@noitemargtrue
% \@noitemerr
% \@noligs
% \@nolnbk
% \@nolnerr
% \@noparitemfalse
% \@noparitemtrue
% \@noparlistfalse
% \@noparlisttrue
% \@nopgbk
% \@normalcr
% \@normalsize
% \@noskipsecfalse
% \@notdefinable
% \@notprerr
% \@nthm
% \@nxttabmar
% \@oddfoot
% \@oddhead
% \@opargbegintheorem
% \@opcol
% \@optionfiles
% \@optionlist
% \@options
% \@othm
% \@outerparskip
% \@outputbox
% \@outputdblcol
% \@outputpage
% \@oval
% \@ovbtrue
% \@ovdx
% \@ovdy
% \@ovhorz
% \@ovltrue
% \@ovri
% \@ovro
% \@ovrtrue
% \@ovttrue
% \@ovvert
% \@ovxx
% \@ovyy
% \@pagedp
% \@pageht
% \@par
% \@parboxrestore
```

```
% \@parmoderr
% \@partaux
% \@partlist
% \@partout
% \@partsw
% \@partswfalse
% \@partswtrue
% \@pboxswfalse
% \@pboxswtrue
% \@pgbk
% \@picbox
% \@picht
% \@picture
% \@pnumwidth
% \@preamble
% \@preamblecmds
% \@preamerr
% \@put
% \@qend
% \@qrelax
% \@reargdef
% \@renewenv
% \@restorepar
% \@reversemarginfalse
% \@reversemargintrue
% \@rhead
% \@rightmark
% \@rightskip
% \@Roman
% \@roman
% \@rsbox
% \@rtab
% \@rule
% \@sanitize
% \@savebox
% \@savemarbox
% \@savepicbox
% \@savsf
% \@savsk
% \@scolelt
% \@sdblcolelt
% \@secpenalty
% \@sect
% \@setpar
% \@settab
% \@sharp
% \@shortstack
% \@sline
% \@spaces
% \@specialoutput
% \@specialpagefalse
% \@specialstyle
% \@sptoken
% \@sqrt
% \@ssect
% \@startcolumn
% \@startdblcolumn
% \@startfield
% \@startline
% \@startpbox
% \@startsection
```

```
% \@starttoc
% \@stopfield
% \@stopline
% \@stpelt
% \@svector
% \@sverb
% \@svsec
% \@svsechd
% \@tabacol
% \@tabarray
% \@tabclassiv
% \@tabclassz
% \@tabcr
% \@tablab
% \@tabminus
% \@tabplus
% \@tabpush
% \@tabrj
% \@tabular
% \@tabularcr
% \@temp
% \@tempa
% \@tempb
% \@tempbox
% \@tempboxa
% \@tempc
% \@tempcnta
% \@tempcntb
% \@tempd
% \@tempdima
% \@tempdimb
% \@tempe
% \@tempskipa
% \@tempskipb
% \@tempswa
% \@tempswafalse
% \@tempswatrue
% \@temptokena
% \@testdef
% \@testfp
% \@testpach
% \@textbottom
% \@textfloatsep
% \@textmin
% \@texttop
% \@tfor
% \@tforloop
% \@thanks
% \@thefnmark
% \@thefoot
% \@thehead
% \@themargin
% \@themark
% \@thm
% \@thmcounter
% \@thmcountersep
% \@tocrmarg
% \@toodeep
% \@toplist
% \@topnewpage
% \@topnum
```

```
% \@toproom
% \@topsep
% \@topsepadd
% \@totalleftmargin
% \@trivlist
% \@tryfcolumn
% \@trylist
% \@twocolumnfalse
% \@twoside
% \@twosidefalse
% \@typein
% \@upline
% \@upordown
% \@upvector
% \@verb
% \@verbatim
% \@vline
% \@vobeyspaces
% \@vspace
% \@vspacer
% \@vtryfc
% \@vvector
% \@warning
% \@wckptelt
% \@whiledim
% \@whilenoop
% \@whilenum
% \@whilesw
% \@whileswnoop
% \@wholewidth
% \@width
% \@wrindex
% \@writeckpt
% \@writefile
% \@wtryfc
% \@x@sf
% \@xarg
% \@xargarraycr
% \@xarraycr
% \@xbitor
% \@xcentercr
% \@xdblarg
% \@xdblfloat
% \@xdim
% \@xeqncr
% \@xexnoop
% \@xexpast
% \@xfloat
% \@xfootnote
% \@xfootnotemark
% \@xfootnotenext
% \@xhead
% \@xifnch
% \@xmpar
% \@xnewline
% \@xnthm
% \@xobeysp
% \@xsect
% \@xstartcol
% \@xtabcr
% \@xtabularcr
```

```
% \@xthm
% \@xtryfc
% \@xtypein
% \@xverbatim
% \@xxxii
% \@xympar
% \@yarg
% \@yargarraycr
% \@ydim
% \@yeqncr
% \@yhead
% \@ympar
% \@ynthm
% \@ythm
% \@ytryfc
% \@yyarg
% \@ztryfc
% \a
% \active
% \addcontentsline
% \addpenalty
% \addtocontents
% \addtocounter
% \addtolength
% \addvspace
% \advance
% \alloc@
% \allocationnumber
% \Alph
% \alph
% \and
% \appendix
% \arabic
% \array
% \arraycolsep
% \arrayrulewidth
% \arraystretch
% \author
% \bar
% \baselineskip
% \begin
% \begingroup
% \bf
% \bgroup
% \bibcite
% \bibdata
% \bibitem
% \bibliography
% \bibliographystyle
% \bibstyle
% \BIG
% \Big
% \big
% \bigskip
% \botfigrule
% \botmark
% \botnum
% \bottomfraction
% \box
% \boxmaxdepth
% \buildrel
```

```
% \bullet
% \c@bottomnumber
% \c@chapter
% \c@dbltopnumber
% \c@equation
% \c@eval
% \c@footnote
% \c@mpfootnote
% \c@page
% \c@secnumdepth
% \c@section
% \c@tocdepth
% \c@topnumber
% \c@totalnumber
% \caption
% \catcode
% \catcoded
% \center
% \centering
% \chapter
% \chaptermark
% \char
% \chardef
% \circle
% \cite
% \cl@@ckpt
% \cleardoublepage
% \clearpage
% \cline
% \closeout
% \clubpenalty
% \columnsep
% \columnseprule
% \columnwidth
% \contentsline
% \copy
% \count
% \countdef
% \cr
% \crcr
% \csname
% \dag
% \dagger
% \dashbox
% \date
% \dblfigrule
% \dblfloatpagefraction
% \dblfloatsep
% \dbltexfloatsep
% \dbltextfloatsep
% \dbltopfraction
% \ddagger
% \deadcycles
% \def
% \description
% \dimen
% \dimen@
% \discretionary
% \displaymath
% \displaystyle
% \displaywidth
```

```
% \divide
% \do
% \document
% \documentstyle
% \dospecials
% \doublerulesep
% \dp
% \edef
% \egroup
% \else
% \end
% \end@dblfloat
% \end@float
% \endarray
% \endcsname
% \enddocument
% \endenumerate
% \endequation
% \endfigure
% \endgroup
% \enditemize
% \endlist
% \endpicture
% \endsloppypar
% \endtabbing
% \endtabular
% \endthebibliography
% \endtrivlist
% \enumerate
% \eqnarray
% \eqno
% \equation
% \errmessage
% \errorstopmode
% \eval
% \evensidemargin
% \everyjob
% \everypar
% \expandafter
% \extracolsep
% \fbox
% \fboxrule
% \fboxsep
% \fi
% \figure
% \fill
% \firstmark
% \float
% \floatingpenalty
% \floatpagefraction
% \floatsep
% \flushbottom
% \flushleft
% \flushright
% \fnsymbol
% \footins
% \footinsertskip
% \footnote
% \footnotemark
% \footnoterule
% \footnotesep
```

```
% \footnotesize
% \footnotetext
% \footsep
% \footskip
% \frac
% \frame
% \framebox
% \frenchspacing
% \fussy
% \futurelet
% \gdef
% \global
% \glossary
% \halfwidth
% \halign
% \hangindent
% \hbox
% \head
% \headheight
% \headsep
% \hfil
% \hfill
% \hfuzz
% \hline
% \hrule
% \hsize
% \hskip
% \hspace
% \hss
% \ht
% \Huge
% \huge
% \hyphenchar
% \if
% \if@afterindent
% \if@eqnsw
% \if@endpe
% \if@fcolmade
% \if@filesw
% \if@firstamp
% \if@firstcolumn
% \if@ignore
% \if@inlabel
% \if@insert
% \if@minipage
% \if@mparswitch
% \if@negarg
% \if@newlist
% \if@nmbrlist
% \if@nobreak
% \if@noitemarg
% \if@noparitem
% \if@noparlist
% \if@noskipsec
% \if@ovb
% \if@ovl
% \if@ovr
% \if@ovt
% \if@pboxsw
% \if@reversemargin
% \if@rjfield
```

```
% \if@specialpage
% \if@tempswa
% \if@test
% \if@twocolumn
% \if@twoside
% \ifcase
% \ifdim
% \ifeof
% \ifhmode
% \ifinner
% \ifmmode
% \ifnum
% \ifodd
% \ifvmode
% \ifvoid
% \ifx
% \ignorespaces
% \immediate
% \include
% \includeonly
% \indent
% \index
% \indexentry
% \input
% \insc@unt
% \insert
% \interdisplaylinepenalty
% \interfootnotelinepenalty
% \interlinepenalty
% \intextsep
% \it
% \item
% \itemindent
% \itemize
% \itemsep
% \jobname
% \kern
% \kill
% \label
% \labelenumi
% \labelenumiv
% \labelitemi
% \labelitemii
% \labelitemiii
% \labelitemiv
% \labelsep
% \labelwidth
% \LARGE
% \Large
% \large
% \lastbox
% \lastskip
% \LaTeX
% \lbrace
% \leaders
% \leavevmode
% \lefteqn
% \leftmargin
% \leftmargini
% \leftmarginvi
% \leftmark
```

```
% \leftskip
% \let
% \limits
% \line
% \linebreak
% \lineskip
% \linethickness
% \linewidth
% \list
% \listoffigures
% \listoftables
% \listparindent
% \llap
% \long
% \lower
% \m@ne
% \m@th
% \makeatletter
% \makeatother
% \makebox
% \makeglossary
% \makeindex
% \makelabel
% \maketitle
% \marginpar
% \marginparpush
% \marginparsep
% \marginparwidth
% \mark
% \markboth
% \markright
% \math
% \mathchar
% \mathchardef
% \mathop
% \mathrel
% \maxdeadcycles
% \maxdepth
% \maxdimen
% \mb@b
% \mb@eval
% \mb@l
% \mb@r
% \mb@t
% \mbox
% \medskip
% \message
% \minipage
% \mit
% \mkern
% \moveright
% \mskip
% \multicolumn
% \multiply
% \multiput
% \multispan
% \newbox
% \newcommand
% \newcount
% \newcounter
% \newdimen
```

```
% \newenvironment
% \newif
% \newinsert
% \newlabel
% \newlength
% \newline
% \newlinechar
% \newpage
% \newsavebox
% \newskip
% \newswitch
% \newtheorem
% \newtoks
% \newwrite
% \noalign
% \nobreak
% \nocite
% \noexpand
% \nofiles
% \noindent
% \nointerlineskip
% \nolinebreak
% \nonumber
% \nopagebreak
% \normalbaselineskip
% \normallineskip
% \normalmarginpar
% \normalsize
% \nullfont
% \number
% \numberline
% \obeycr
% \obeylines
% \obeyspaces
% \oddsidemargin
% \of
% \on@line
% \onecolumn
% \openin
% \or
% \outer
% \output
% \outputpenalty
% \oval
% \over
% \overfullrule
% \overline
% \p@
% \pagebreak
% \pagelayout
% \pagenumbering
% \pageref
% \pagestyle
% \par
% \paragraph
% \parbox
% \parfillskip
% \parindent
% \parsep
% \parshape
% \parskip
```

```
% \partopsep
% \partsw
% \penalty
% \picture
% \poptab
% \poptabs
% \postdisplaypenalty
% \prevdepth
% \protect
% \ps@empty
% \ps@plain
% \pushtab
% \pushtabs
% \put
% \quotation
% \raggedbottom
% \raggedleft
% \raggedright
% \raise
% \raisebox
% \rbrace
% \read
% \ref
% \refstepcounter
% \relax
% \renewcommand
% \renewenvironment
% \reset@font
% \restorecr
% \reversemarginpar
% \right
% \rightmargin
% \rightmark
% \rightskip
% \rlap
% \rm
% \Roman
% \roman
% \romannumeral
% \root
% \rule
% \samepage
% \savebox
% \sbox
% \sc
% \scriptscriptsize
% \scriptsize
% \secdef
% \section
% \sectionmark
% \setbox
% \setcounter
% \setlength
% \settowidth
% \shipout
% \shortstack
% \showboxbreadth
% \showboxdepth
% \sixt@@n
% \skip
% \sl
```

```
% \SLiTeX
% \sloppy
% \sloppypar
% \small
% \smallskip
% \space
% \spacefactor
% \splitmaxdepth
% \splittopskip
% \sqrt
% \ss
% \stackrel
% \stepcounter
% \stop
% \stretch
% \string
% \strut
% \subsection
% \subsubsection
% \tabalign
% \tabbing
% \tabbingsep
% \tabcolsep
% \tableentry
% \tableofcontents
% \tabskip
% \tabular
% \tencirc
% \tencircw
% \tenln
% \tenlnw
% \textfloatsep
% \textfraction
% \textheight
% \textwidth
% \thanks
% \the
% \thebibliography
% \theenumi
% \theenumii
% \theequation
% \thefigure
% \thefootnote
% \thempfn
% \thempfootnote
% \thepage
% \thesection
% \thicklines
% \thinlines
% \thinspace
% \thispagestyle
% \tiny
% \title
% \today
% \tolerance
% \topfigrule
% \topfraction
% \topmargin
% \topnewpage
% \topnum
% \topsep
```

```
% \topskip
% \tracingonline
% \tracingoutput
% \tracingstats
% \trivlist
% \tt
% \tw@
% \twocolumn
% \typein
% \typeout
% \unbox
% \underline
% \unhbox
% \unitlength
% \unskip
% \unvbox
% \usebox
% \usecounter
% \vadjust
% \value
% \vbox
% \vcenter
% \vector
% \verb
% \verbatim
% \vfil
% \vfuzz
% \vline
% \vrule
% \vsize
% \vskip
% \vspace
% \vsplit
% \vss
% \vtop
% \wd
% \write
% \writes
% \xdef
% \z@
% \[
% \\
% \]
% \^
% \_
% \‘
% \{
% \|
% \}
% \~


%      ****************************************
%      *         GENERAL CONVENTIONS          *
%      ****************************************
%
% THE \LaTeX AND \SLiTeX LOGOS ARE DEFINED HERE.
%
%% RmS 91/09/29: \reset@font added to \LaTeX logo.
```

```
\def\p@LaTeX{{\reset@font\rm L\kern-.36em\raise.3ex\hbox{\sc a}\kern-.15em%
    T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}}

%% RmS 91/09/29: \SLiTeX logo added.
\def\p@SLiTeX{{\reset@font\rm S\kern-.06em{\sc l\kern-.035emi}\kern-.06emT\kern
   -.1667em\lower.7ex\hbox{E}\kern-.125emX}}

%% RmS 91/10/17: \protect'ed the logos
\def\LaTeX{\protect\p@LaTeX}
\def\SLiTeX{\protect\p@SLiTeX}


% SAVED VERSIONS OF TeX PRIMITIVES:
%
%  The TeX primitive \foo is saved as \@@foo .  The following primitives
%  are handled in this way:

\let\@@par=\par
%\let\@@relax=\relax  % This was needed at one time, but seems to be obsolete.
\let\@@input=\input
\let\@@end=\end

% The following was added 19 April 1986:
% The \- command is redefined to allow it to work in the \tt type style,
% where automatic hyphenation is suppressed by setting \hyphenchar to -1.
% The original definition is saved as \@@hyph just in case anyone needs it.

\let\@@hyph=\-        % Original defin
\def\-{\discretionary{-}{}{}}

% SAVED VERSIONS OF TeX PARAMETERS
%
%  \normalbaselineskip and \normallineskip hold the
%  normal values of \baselineskip and \lineskip

% Any font-changing commands that change the normal value of \lineskip
% and \baselineskip should change their saved values.

% The following definitions save token space.  E.g., using \@height
% instead of height saves 5 tokens at the cost in time of one macro
% expansion.

\def\@height{height}
\def\@depth{depth}
\def\@width{width}

% The following implements the LaTeX \{ and \} commands.
% Changed 21 Apr 87 to make them robust.

\def\{{\protect\@lb}
\def\@lb{\relax\ifmmode\lbrace\else$\m@th\lbrace$\fi}
\def\}{\protect\@rb}
\def\@rb{\relax\ifmmode\rbrace\else$\m@th\rbrace$\fi}

 \message{counters,}
%      ****************************************
%      *          COUNTERS, ETC.              *
%      ****************************************
%
% THE FOLLOWING ARE FROM PLAIN:
% \z@         : A zero dimen or number.  It's more efficient to write
```

```
%               \parindent\z@ than \parindent 0pt.
% \@ne         : The number 1.
% \m@ne        : The number -1.
% \tw@         : The number 2.
% \sixt@@n     : The number 16.
% \@m          : The number 1000.
% \@xxxii      : The number 32
% \@M          : The number 10000.
% \@Mi         : The number 10001.
% \@Mii        : The number 10002.
% \@Miii       : The number 10003.
% \@Miv        : The number 10004.
% \@MM         : The number 20000.
%
% \@flushglue : Glue used for \right- & \leftskip to = 0pt plus 1fil

\chardef\@xxxii=32
\mathchardef\@Mi=10001
\mathchardef\@Mii=10002
\mathchardef\@Miii=10003
\mathchardef\@Miv=10004

% Redefine PLAIN.TEX macros not to be \outer

\def\newcount{\alloc@0\count\countdef\insc@unt}
\def\newdimen{\alloc@1\dimen\dimendef\insc@unt}
\def\newskip{\alloc@2\skip\skipdef\insc@unt}
\def\newbox{\alloc@4\box\chardef\insc@unt}
\def\newwrite{\alloc@7\write\chardef\sixt@@n}

\newwrite\@unused
\newcount\@tempcnta
\newcount\@tempcntb
\newif\if@tempswa\@tempswatrue

\newdimen\@tempdima
\newdimen\@tempdimb

\newbox\@tempboxa

\newskip\@flushglue \@flushglue = 0pt plus 1fil
\newskip\@tempskipa
\newskip\@tempskipb
\newtoks\@temptokena

 \message{hacks,}
%      ****************************************
%      *            USEFUL HACKS              *
%      ****************************************
%
%  \@namedef{NAME}   : Expands to \def\NAME , except name can contain any
%                      characters.
%  \@nameuse{NAME}   : Expands to \NAME .
%
%  \@ifnextchar X{YES}{NO}
%                    : Expands to YES if next character is an 'X',
%                      and to NO otherwise. (Uses temps a-c.)
%                      NOTE: GOBBLES ANY SPACE FOLLOWING IT.
%
%  \@ifstar{YES}{NO} : Gobbles following spaces and then tests if next the
%                      character is a '*'.  If it is, then it gobbles the
```

```
%                       '*' and expands to YES, otherwise it expands to NO.
%
%  \@dblarg{CMD}{ARG}  : \@dblarg{CMD}{ARG} expands to CMD[ARG]{ARG}.  Use
%                        \@dblarg\CS when \CS takes arguments [ARG1]{ARG2},
%                        where default is ARG1 = ARG2.
%
%  \@ifundefined{NAME}{YES}{NO}
%                     : If \NAME is undefined then it executes YES,
%                       otherwise it executes NO.  More precisely,
%                       true if \NAME either undefined or = \relax.
%  \@ifdefinable \NAME {YES}
%                     : Executes YES if the user is allowed to define \NAME,
%                       otherwise it gives an error.  The user can define \NAME
%                       if \@ifundefined{NAME} is true, 'NAME' /= 'relax'
%                       and the first three letters of 'NAME' are not
%                       'end'.
%  \newcommand{\FOO}[i]{TEXT}
%                     : User command to define \FOO to be a macro with
%                       i arguments (i = 0 if missing) having the definition
%                       TEXT.  Produces an error if \FOO already defined.
%
%  \renewcommand{\FOO}[i]{TEXT} : Same as \newcommand, except it
%                       checks if \FOO already defined.
%
%  \newenvironment{FOO}[i]{DEF1}{DEF2}
%         equivalent to
%         \newcommand{\FOO}[i]{DEF1} \def{\endFOO}{DEF2}
%
%  \renewenvironment : obvious companion to \newenvironment
%
%  \@cons : See description of \output routine.
%
%  \@car T1 T2 ... Tn\@nil == T1  (unexpanded)
%
%  \@cdr T1 T2 ... Tn\@nil == T2 ... Tn     (unexpanded)
%
%  \typeout{message} : produces a warning message on the terminal
%
%  \@warning{message}: prints 'LaTeX Warning: message.'
%                       With TeX 3.x, it also prints line number.
%                       (Changed 24 Jun 91 RmS)
%  \@@warning{message}: like \@warning, except that it never prints
%                       the line number (added 24 Jun 91 RmS).
%
%  \typein{message}  : Types message, asks the user to type in a command, then
%                       executes it
%
%  \typein[\CS]{MSG} : Same as above, except defines \CS to be the input
%                       instead of executing it.

%% RmS 92/03/18: changed input channel from 0 to \@inputcheck to avoid conflicts
%%               with other channels allocated by \newread
\newread\@inputcheck
\def\typein{\let\@typein\relax\@ifnextchar[{\@xtypein}{\@xtypein[\@typein]}}
\def\@xtypein[#1]#2{\typeout{#2}\read\@inputcheck to#1\ifx #1\@defpar \def#1{}\else
   \@iden{\expandafter\@strip\expandafter
   #1#1\@gobble\@gobble} \@gobble\fi\@typein}
\def\@strip#1#2 \@gobble{\def #1{#2}}
\def\@defpar{\par}
\def\@iden#1{#1}

```

```
\ifx\inputlineno\undefined
  \let\on@line\empty
\else
  \ifnum\inputlineno=\m@ne
    \let\on@line\empty
  \else
    \def\on@line{ on input line \the\inputlineno}
  \fi
\fi

\def\typeout#1{{\let\protect\string\immediate\write\@unused{#1}}}
\def\@@warning#1{\typeout{LaTeX Warning: #1.}}
\def\@warning#1{\@@warning{#1\on@line}}
\def\@namedef#1{\expandafter\def\csname #1\endcsname}
\def\@nameuse#1{\csname #1\endcsname}

\def\@cons#1#2{\begingroup\let\@elt\relax\xdef#1{#1\@elt #2}\endgroup}

\def\@car#1#2\@nil{#1}
\def\@cdr#1#2\@nil{#2}

% \@carcube T1 ... Tn\@nil = T1 T2 T3 , n > 3
\def\@carcube#1#2#3#4\@nil{#1#2#3}

\def\newcommand#1{\@ifnextchar [{\@argdef#1}{\@argdef#1[0]}}

\def\renewcommand#1{\edef\@tempa{\expandafter\@cdr\string
  #1\@nil}\@ifundefined{\@tempa}{\@latexerr{\string#1\space undefined}\@ehc
    }{}\@ifnextchar [{\@reargdef#1}{\@reargdef#1[0]}}

\def\newenvironment#1{\@ifnextchar
     [{\@newenv{#1}}{\@newenv{#1}[0]}}

\long\def\@newenv#1[#2]#3#4{\expandafter\newcommand
     \csname #1\endcsname[#2]{#3}\long
     \expandafter\def\csname end#1\endcsname{#4}}

\def\renewenvironment#1{\@ifnextchar
     [{\@renewenv{#1}}{\@renewenv{#1}[0]}}

\long\def\@renewenv#1[#2]#3#4{\expandafter\renewcommand
     \csname #1\endcsname[#2]{#3}\long
     \expandafter\def\csname end#1\endcsname{#4}}

\long\def\@argdef#1[#2]#3{\@ifdefinable #1{\@reargdef#1[#2]{#3}}}

% Absolutely untypable control sequence \@?@?  substituted for \@tempb in
% definition of \@reargdef because it (and therefore \newcommand and
% \renewcommand) leaves the control sequence dangerously \let to #.
% (Change made 23 November 87.)
%
\catcode`\?=11\relax
\long\def\@reargdef#1[#2]#3{\@tempcnta#2\relax\let#1\relax
\edef\@tempa{\long\def#1}\@tempcntb \@ne
\let\@?@?\relax\@whilenum\@tempcnta>\z@
\do{\edef\@tempa{\@tempa\@?@?\the\@tempcntb}\advance\@tempcntb \@ne \advance
\@tempcnta \m@ne}\let\@?@?##\@tempa{#3}}
\catcode`\?=12\relax


% 9 Jan 90 : Missing % added to following definition.
```

```
\long\def\@ifdefinable #1#2{\edef\@tempa{\expandafter\@cdr\string #1\@nil}%
\@ifundefined{\@tempa}{\edef\@tempb{\expandafter\@carcube \@tempa xxxx\@nil}%
\ifx \@tempb\@qend \@notdefinable\else
\ifx \@tempa\@qrelax \@notdefinable\else  #2\fi\fi}{\@notdefinable}}

\long\def\@ifundefined#1#2#3{\expandafter\ifx\csname
  #1\endcsname\relax#2\else#3\fi}


% The following define \@qend and \@qrelax to be the strings 'end' and
% 'relax' with the characters \catcoded 12.

\edef\@qend{\expandafter\@cdr\string\end\@nil}
\edef\@qrelax{\expandafter\@cdr\string\relax\@nil}

% \@ifnextchar X{YES}{NO}
%  BEGIN
%    \@tempe := X  % uses \let
%    \@tempa := YES
%    \@tempb := NO
%    \futurelet\@tempc
%    \@ifnch
%  END
%
% \@ifnch ==
%   BEGIN
%     if  \@tempc = blank space
%       then  \@tempd := def(\@xifnch)
%       else  if  \@tempc = \@tempe
%                 then  \@tempd := def(\@tempa)
%                 else  \@tempd := def(\@tempb)
%             fi
%     fi
%     \@tempd
%   END
%
% \@xifnch ==
%  BEGIN
%    gobble blanks
%    \futurelet\@tempc
%    \@ifnch
%  END
%
\def\@ifnextchar#1#2#3{\let\@tempe #1\def\@tempa{#2}\def\@tempb{#3}\futurelet
    \@tempc\@ifnch}
\def\@ifnch{\ifx \@tempc \@sptoken \let\@tempd\@xifnch
      \else \ifx \@tempc \@tempe\let\@tempd\@tempa\else\let\@tempd\@tempb\fi
      \fi \@tempd}

% NOTE: the following hacking must precede the definition of \:
%  as math medium space.

\def\:{\let\@sptoken= } \:  % this makes \@sptoken a space token

\def\:{\@xifnch} \expandafter\def\: {\futurelet\@tempc\@ifnch}

\def\@ifstar#1#2{\@ifnextchar *{\def\@tempa*{#1}\@tempa}{#2}}

\long\def\@dblarg#1{\@ifnextchar[{#1}{\@xdblarg{#1}}}
\long\def\@xdblarg#1#2{#1[{#2}]{#2}}

```

```
% The command \@sanitize changes the catcode of all special characters
% except for braces to 'other'.  It can be used for commands like
% \index that want to write their arguments verbatim.  Needless to
% say, this command should only be executed within a group, or chaos
% will ensue.

\def\@sanitize{\@makeother\ \@makeother\\\@makeother\$\@makeother\&%
\@makeother\#\@makeother\^\@makeother\_\@makeother\%\@makeother\~}


 \message{errors,}
%      ***************************************
%      *           ERROR HANDLING            *
%      ***************************************
%
%  \@latexerr{MSG}{HLP}: Types a LaTeX error message MSG and gives an error
%                          halt with error help message HLP.
%
\newlinechar`\^^J

% 19 Jun 86, took out the grouping. re: John Hobby
\def\@latexerr#1#2{%
\edef\@tempc{#2}\errhelp\expandafter{\@tempc}%
\typeout{LaTeX error. \space See LaTeX manual for explanation.^^J
 \space\@spaces\@spaces\@spaces Type \space H <return> \space for
 immediate help.}\errmessage{#1}}

\def\@spaces{\space\space\space\space}

%% error help message pieces.
\def\@eha{Your command was ignored.
^^JType \space I <command> <return> \space to replace it
  with another command,^^Jor \space <return> \space to continue without it.}
\def\@ehb{You've lost some text. \space \@ehc}
\def\@ehc{Try typing \space <return>
  \space to proceed.^^JIf that doesn't work, type \space X <return> \space to
  quit.}
\def\@ehd{You're in trouble here.  \space\@ehc}

% Here are all the error message-generating commands of LaTeX.
%
% \@notdefinable : Error message generated in \@ifdefinable from calls
%                  by \newcommand, \newlength, \newtheorem specifying an
%                  already-defined command name.
%
% \@nolnerr      : Generated by \newline and \\ when called in vertical mode.
%
% '\... undefined' : Generated in \renewcommand.
%
% \@nocnterr     : Generated by \setcounter, \addtocounter or \newcounter
%                  for undefined counter.
%
% \@ctrerr       : Called when trying to print the value of a counter
%                  numbered by letters that's greater than 26.
%
% 'Environment --- undefined' : Issued by \begin for undefined environment.
%
% \@badend       : Called by \end that doesn't match its \begin.
%
% \@badmath      : Called by \[, \], \( or \) when used in wrong mode.
%
```

```
% \@toodeep      : Called by a list environment nested more than six levels
%                   deep, or an enumerate or itemize nested more than four
%                   levels.
%
% \@badpoptabs   : Called by \endtabbing when not enough \poptabs have
%                   occurred, or by \poptabs when too many have occurred.
%
% \@badtab : Called by \>, \+ , \- or \< when stepping to an undefined tab.
%
% 'tab overflow' : Occurs in \= when maximum number of tabs exceeded.
%
% '\< in mid line' : Occurs in \< when it appears in middle of line.
%
% \@preamerr : Occurs in array or tabular environment, or in \multicolumn
%               command, when error in argument detected.
%
% \@badlinearg : Occurs in \line and \vector command when a bad slope
%                 argument is encountered.
%
% \@parmoderr  : Occurs in a float environment or a \marginpar when
%                 encountered in inner vertical mode.
%
% \@fltovf     : Occurs in float environment or \marginpar when there
%                 are no more free boxes for storing floats.
%
% \@latexbug   :  Occurs in output routine.  This is bad news.
%
% 'Float(s) lost' : In output routine, caused by a float environment or
%                   \marginpar occurring in inner vertical mode.
%
% \@nofonterror   : Typeface not available.  %%% OBSOLETE; DELETED.
%
% \@badcrerr      : A \\ used where it shouldn't be in a centering or flushing
%                   environment.
%
% \@noitemerr     : \addvspace or \addpenalty was called when not in vmode.
%                   Probably caused by a missing \item.
%
% \@notprerr      : A command that can be used only in the preamble
%                   appears after the \begin{document} command.

\def\@notdefinable{\@latexerr{Command name '\@tempa' already used}\@eha}

\def\@nolnerr{\@latexerr{There's no line here to end}\@eha}

\def\@nocnterr{\@latexerr{No such counter}\@eha}

\def\@ctrerr{\@latexerr{Counter too large}\@ehb}

\def\@nodocument{\@latexerr{Missing \string\begin{document}}\@ehd}

\def\@badend#1{\@latexerr{\string\begin{\@currenvir} ended by
      \string\end{#1}}\@eha}

\def\@badmath{\@latexerr{Bad math environment delimiter}\@eha}

\def\@toodeep{\@latexerr{Too deeply nested}\@ehd}

\def\@badpoptabs{\@latexerr{\string\pushtabs \space and \string\poptabs
    \space don't match}\@ehd}

```

```
\def\@badtab{\@latexerr{Undefined tab position}\@ehd}

\def\@preamerr#1{\@latexerr{\ifcase #1 Illegal character\or
     Missing @-exp\or Missing p-arg\fi\space
     in array arg}\@ehd}

\def\@badlinearg{\@latexerr{Bad \string\line\space or \string\vector
   \space argument}\@ehb}

\def\@parmoderr{\@latexerr{Not in outer par mode}\@ehb}

\def\@fltovf{\@latexerr{Too many unprocessed floats}\@ehb}

\def\@latexbug{\@latexerr{This may be a LaTeX bug}{Call for help}}

% \def\@nofonterror{\@latexerr{Typeface not available}\@eha}

\def\@badcrerr {\@latexerr{Bad use of \string\\}\@ehc}

\def\@noitemerr{\@latexerr{Something's wrong--perhaps a missing
\string\item}\@ehc}

\def\@notprerr {\@latexerr{Can be used only in preamble}\@eha}

 \message{par,}
%       ****************************************
%       *          \par AND \everypar          *
%       ****************************************
%
% There are two situations in which \par may be changed:
%
%   - Long-term changes, in which the new value is to remain in effect
%     until the current environment is left.  The environments that
%     change \par in this way are the following:
%
%         * All list environments (itemize, quote, etc.)
%         * Environments that turn \par into a noop:
%              tabbing, array and tabular.
%
%   - Temporary changes, in which \par is restored to its previous value the
%     next time it is executed. The following are all such uses.
%         * \end [when preceded by \@endparenv, which is called by
%                 \endtrivlist]
%         * The mechanism for avoiding page breaks and getting the
%           spacing right after section heads.
%
% To permit the proper interaction of these two situations, long-term
% changes are made by the following command:
%     \@setpar{VAL}      : To set \par. It \def's \par and \@par to VAL.
% Short-term changes are made by the usual \def\par commands.
% The original values are restored after a short-term change
% by the \@restorepar commands.
%
% NOTE: \@@par always is defined to be the original TeX \par.
%
% \everypar is changed only for the short term.  Whenever \everypar
% is set non-null, it should restore itself to null when executed.
% The following commands change \everypar in this way:
%         * \item
%         * \end [when preceded by \@endparenv, which is called by
%                 \endtrivlist]
```

```
% 	* \minipage
%
% WARNING: Commands that make short-term changes to \par and \everypar
% must take account of the possibility that the new commands and the
% ones that do the restoration may be executed inside a group.  In
% particular, \everypar is executed inside a group whenever a new paragraph
% begins with a left brace.  The \everypar command that restores its
% definition should be local to the current group (in case the command
% is inside a minipage used inside someplace where \everypar has been
% redefined).  Thus, if \everypar is redefined to do an \everypar{}
% it could take several executions of \everypar before
% the restoration 'holds'.  This usually causes no problem.  However, to
% prevent the extra executions from doing harm, use a global switch
% to keep anything harmful in the new \everypar from being done twice.
%
% WARNING: Commands that change \everypar should remember that \everypar
% might be supposed to set the following switches false:
%              @nobreak
%              @minipage
% they should do the setting if necessary.

\def\@par{\let\par\@@par\par}

\def\@setpar#1{\def\par{#1}\def\@par{#1}}
\def\@restorepar{\def\par{\@par}}

 \message{spacing,}
%      ********************************************
%      *     SPACING / LINE AND PAGE BREAKING       *
%      ********************************************
%
% USER COMMANDS:
% \nopagebreak[i] : i = 0,...,4.  Default argument = 4.  Puts a penalty
%                  into the vertical list output as follows:
%                    0 : penalty = 0
%                    1 : penalty = \@lowpenalty
%                    2 : penalty = \@medpenalty
%                    3 : penalty = \@highpenalty
%                    4 : penalty = 10000
% \pagebreak[i]   : same as \nopagebreak except negatives of its penalty
% \linebreak[i], \nolinebreak[i] : analogs of the above
% \samepage : inhibits page breaking most places by setting the following
%             penalties to 10000
%                    \interlinepenalty
%                    \postdisplaypenalty
%                    \interdisplaylinepenalty
%                    \@beginparpenalty
%                    \@endparpenalty
%                    \@itempenalty
%                    \@secpenalty
%                    \interfootnotelinepenalty
%
% \obeycr    : defines <CR> == \\.
% \restorecr : restores <CR> to its usual meaning.
%
% \\         : initially defined to be \newline
% \\[LENGTH] : initially defined to be \vspace{LENGTH}\newline
%              Note: \\* adds a \vadjust{\penalty 10000}

\def\nopagebreak{\@ifnextchar[{\@nopgbk}{\@nopgbk[4]}}
\def\@nopgbk[#1]{\ifvmode \penalty \@getpen{#1}\else
```

```
\@bsphack\vadjust{\penalty \@getpen{#1}}\@esphack\fi}

\def\pagebreak{\@ifnextchar[{\@pgbk}{\@pgbk[4]}}
\def\@pgbk[#1]{\ifvmode \penalty -\@getpen{#1}\else
\@bsphack\vadjust{\penalty -\@getpen{#1}}\@esphack\fi}

\def\nolinebreak{\@ifnextchar[{\@nolnbk}{\@nolnbk[4]}}
\def\@nolnbk[#1]{\ifvmode \@nolnerr\else \@tempskipa\lastskip
     \unskip \penalty \@getpen{#1}\ifdim \@tempskipa >\z@
     \hskip\@tempskipa\ignorespaces\fi\fi}

\def\linebreak{\@ifnextchar[{\@lnbk}{\@lnbk[4]}}
\def\@lnbk[#1]{\ifvmode \@nolnerr\else
     \unskip\penalty -\@getpen{#1}\fi}

\def\samepage{\interlinepenalty\@M
   \postdisplaypenalty\@M
   \interdisplaylinepenalty\@M
   \@beginparpenalty\@M
   \@endparpenalty\@M
   \@itempenalty\@M
   \@secpenalty\@M
   \interfootnotelinepenalty\@M}

% \nobreak added to \newline to prevent null lines when \newline
% ends an overfull line.  Change made 24 May 89 as suggested by
% Frank Mittelbach and Rainer Sch\"opf
%
\def\newline{\ifvmode \@nolnerr \else \unskip\nobreak\hfil
  \penalty -\@M\fi}


\def\@normalcr{\@ifstar{\vadjust{\penalty\@M}\@xnewline}{\@xnewline}}

\def\@xnewline{\@ifnextchar[{\@newline}{\newline}}

\def\@newline[#1]{\ifhmode\unskip\fi\vspace{#1}\newline}

\let\\=\@normalcr

\def\@getpen#1{\ifcase #1 0 \or \@lowpenalty\or
         \@medpenalty \or \@highpenalty
         \else \@M \fi}

% @nobreak : Switch used to avoid page breaks caused by \label after a section
%            heading, etc. It should be GLOBALLY set true after the \nobreak
%            and GLOBALLY set false by the next invocation of \everypar.
%            Commands that reset \everypar should globally set it false
%            if appropriate.
%
\newif\if@nobreak \@nobreakfalse

% \@bsphack ... \@esphack
%     used by macros such as \index and \begin{@float} ... \end{@float}
%     that want to be invisible -- i.e.,
%     not leave any extra space when used in the middle of text.  Such
%     a macro should begin with \@bsphack and end with \@esphack
%     The macro in question should not create any text, nor change the
%     mode.
%
% \@Esphack is a variant of \@esphack that sets the @ignore switch to true
```

```
%     (as \@esphack used to do previously). This is currently used only
%     for float and similar environments.
%
% \@bsphack ==
%  BEGIN
%   if not mmode then                %% Test for math mode added 18 Dec 89
%         \dimen\@savsk := \lastskip
%       if  hmode  then  \@savsf := \spacefactor  fi
%   fi
%  END
%
% \@esphack ==
%  BEGIN
%   if not mmode then                %% Test for math mode added 18 Dec 89
%    if  hmode
%      then  \spacefactor := \@savsf
%            if \dimen\@savsk > 0pt  then  \ignorespaces fi
%    fi
%   fi
%  END
%
% \@Esphack ==
%  BEGIN
%   if not mmode then
%    if  hmode
%      then  \spacefactor := \@savsf
%            if \dimen\@savsk > 0pt  then  \ignorespaces
%                                          \global\@ignoretrue   fi
%    fi
%   fi
%  END
%

\newdimen\@savsk
\newcount\@savsf

\def\@bsphack{\relax\ifmmode\else\@savsk\lastskip
    \ifhmode\@savsf\spacefactor\fi\fi}

\def\@esphack{\relax\ifmmode\else\ifhmode\spacefactor\@savsf
     {}\ifdim \@savsk >\z@ \ignorespaces
  \fi \fi\fi}

\def\@Esphack{\relax\ifmmode\else\ifhmode\spacefactor\@savsf
     {}\ifdim \@savsk >\z@ \global\@ignoretrue \ignorespaces
  \fi \fi\fi}

% VERTICAL SPACING:
%
% LaTeX supports the PLAIN TeX commands \smallskip, \medskip and \bigskip.
% However, it redefines them using \vspace instead of \vskip.
%
% Extra vertical space is added by the command command \addvspace{SKIP},
% which adds a vertical skip of SKIP to the document.  The sequence
%         \addvspace{S1} \addvspace{S2}
% is equivalent to
%         \addvspace{maximum of S1, S2}.
% \addvspace should be used only in vertical mode, and gives an error if it's
% not.  The \addvspace command does NOT add vertical space if
% @minipage = T. The minipage environment uses this to inhibit
% the addition of extra vertical space at the beginning.
```

```
%
% Penalties are put into the vertical list with the \addpenalty{PENALTY}
% command.  It works properly when \addpenalty and \addvspace commands
% are mixed.
%
% The @nobreak switch is set true used when in vertical mode and no page
% break should occur.  (Right now, it is used only by the section heading
% commands to inhibit page breaking after a heading.)
%
%
% \addvspace{SKIP} ==
%  BEGIN
%   if vmode
%     then if @minipage
%            else if \lastskip =0
%                    then  \vskip SKIP
%                    else  if \lastskip < SKIP
%                             then  \vskip -\lastskip
%                                   \vskip SKIP
%                             else if SKIP < 0 and \lastskip >= 0
%                                    then \vskip -\lastskip
%                                         \vskip \lastskip + SKIP
%          fi      fi       fi      fi
%     else 'missing \item' error.
%   fi
%  END

\def\addvspace#1{\ifvmode
     \if@minipage\else
          \ifdim \lastskip =\z@ \vskip #1\relax
             \else \@tempskipb#1\relax\@xaddvskip
     \fi\fi
  \else\@noitemerr\fi}

\def\@xaddvskip{\ifdim \lastskip <\@tempskipb\vskip-\lastskip\vskip
              \@tempskipb
        \else  \ifdim \@tempskipb<\z@
                  \ifdim \lastskip <\z@
                     \else \advance\@tempskipb\lastskip
                           \vskip -\lastskip \vskip \@tempskipb
      \fi\fi\fi}

\def\addpenalty#1{\ifvmode
   \if@minipage\else\if@nobreak\else
      \ifdim\lastskip=\z@ \penalty#1\relax
         \else \@tempskipb\lastskip
               \vskip -\lastskip \penalty#1\vskip\@tempskipb
      \fi\fi\fi
   \else\@noitemerr\fi}

\def\vspace{\@ifstar{\@vspacer}{\@vspace}}
\def\@vspace#1{\ifvmode
    \dimen@\prevdepth \vskip #1\vskip\z@ \prevdepth\dimen@
       \else
        \@bsphack\vadjust{\dimen@\prevdepth
            \vskip #1\vskip\z@ \prevdepth\dimen@}\@esphack\fi}
\def\@vspacer#1{\ifvmode \dimen@\prevdepth
         \hrule \@height\z@ \nobreak \vskip #1\vskip\z@
           \prevdepth\dimen@
        \else
         \@bsphack\vadjust{\dimen@\prevdepth \hrule \@height\z@ \nobreak
```

```
            \vskip #1\vskip\z@ \prevdepth\dimen@}\@esphack\fi}

\def\smallskip{\vspace\smallskipamount}
\def\medskip{\vspace\medskipamount}
\def\bigskip{\vspace\bigskipamount}


% See list environment for explanation of the following macros.

\def\endtrivlist{\if@newlist\@noitemerr\fi
   \if@inlabel\indent\fi
   \ifhmode\unskip \par\fi
   \if@noparlist \else
      \ifdim\lastskip >\z@ \@tempskipa\lastskip \vskip -\lastskip
         \advance\@tempskipa\parskip \advance\@tempskipa -\@outerparskip
         \vskip\@tempskipa
   \fi\@endparenv\fi}

% CHANGES TO \@endparenv:
% Changed  \hskip -\parindent  to  \setbox0=\lastbox  so a \noindent
% becomes a no-op when used before a line immediately following a
% list environment.  (Changed 23 Oct 86)
%
% To suppress the paragraph indentation in text immediately following
% a paragraph-making environment, \everypar is changed to remove the
% space, and \par is redefined to restore \everypar.  Instead of redefining
% \par and \everpar, \@endparenv was changed to set the @endpe switch,
% letting \end redefine \par and \everypar.  This allows paragraph-
% making environments work right when called by other environments.
% (Changed 27 Oct 86)

\def\@endparenv{\addpenalty\@endparpenalty\addvspace\@topsepadd\@endpetrue}

\def\@doendpe{\@endpetrue
     \def\par{\@restorepar\everypar{}\par\@endpefalse}\everypar
               {\setbox\z@\lastbox\everypar{}\@endpefalse}}

\newif\if@endpe
\@endpefalse

% HORIZONTAL SPACE
%
% \, : used in paragraph mode produces a \thinspace.  It has the ordinary
%      definition in math mode.  Useful for quotes inside quotes, as in
%      ``\,`Foo', he said.''
%
% \@ : placed before a '.', makes it a sentence-ending period.  Does the
%     right thing for other punctuation marks as well.  Does this by
%     setting spacefactor to 1000.

\def\,{\protect\pcomma}
\def\pcomma{\relax\ifmmode\mskip\thinmuskip\else\thinspace\fi}


\def\@{\spacefactor\@m}

\def\hspace{\protect\phspace}
\def\phspace{\@ifstar{\@hspacer}{\@hspace}}
\def\@hspace#1{\leavevmode\hskip #1\relax}

\def\@hspacer#1{\leavevmode\vrule \@width\z@\nobreak
```

```
                \hskip #1\hskip \z@skip}
                      %% extra \hskip 0pt added 12/17/85 to guard
                      %% against a following \unskip
                      %% \relax added 13 Oct 88 for usual TeX lossage
                      %% replaced both changes by \hskip\z@skip 27 Nov 91

% define \fill to = 0pt plus 1fill
\newskip\fill \fill = 0pt plus 1fill

% \stretch{N} == 0pt plus N fill
\def\stretch#1{\z@ plus #1fill\relax}

{\catcode`\^^M=13 \gdef\obeycr{\catcode`\^^M=13 \def^^M{\\}\@gobblecr}%
\gdef\restorecr{\catcode`\^^M=5 }} %} BRACE MATCHING


 \message{control,}
%      ********************************************
%      *     PROGRAM CONTROL STRUCTURE MACROS       *
%      ********************************************
%
% \@whilenum TEST \do {BODY}
% \@whiledim TEST \do {BODY}  : These implement the loop
%           while  TEST  do  BODY  od
%     where  TEST  is a TeX \ifnum or \ifdim test, respectively.
%     They are optimized for the normal case of TEST initially false.
%
% \@whilesw SWITCH \fi {BODY} : Implements the loop
%               while SWITCH do BODY od
%     where SWITCH is a command defined by \newswitch.
%     Optimized for normal case of SWITCH initially false.
%
% \@for NAME := LIST \do {BODY} : Assumes that LIST expands to A1,A2, ... ,An .
%      Executes  BODY  n  times, with  NAME = Ai  on the i-th iteration.
%      Optimized for the normal case of n = 1.  Works for n=0.
%
% \@tfor NAME := LIST \do {BODY}
%      if, before expansion, LIST = T1 ... Tn  where each Ti is a
%      token or {...}, then executes  BODY  n  times, with  NAME = Ti
%      on the i-th iteration.  Works for n=0.
%
%  NOTES: 1. These macros use no \@temp sequences.
%         2. These macros do not work if the body contains anything that looks
%            syntactically to TeX like an improperly balanced \if \else \fi.
%
% \@whilenum TEST \do {BODY} ==
%  BEGIN
%    if  TEST
%      then  BODY
%            \@iwhilenum{TEST \relax BODY}
%  END
%
% \@iwhilenum {TEST BODY} ==
%  BEGIN
%    if  TEST
%      then  BODY
%            \@nextwhile = def(\@iwhilenum)
%      else  \@nextwhile = def(\@whilenoop)
%    fi
%    \@nextwhile {TEST BODY}
%  END
```

```
%
% \@whilesw SWITCH \fi {BODY} ==
%  BEGIN
%    if SWITCH
%      then BODY
%           \@iwhilesw {SWITCH BODY}\fi
%    fi
%  END
%
% \@iwhilesw {SWITCH BODY} \fi ==
%  BEGIN
%    if SWITCH
%      then BODY
%           \@nextwhile = def(\@iwhilesw)
%      else \@nextwhile = def(\@whileswnoop)
%    fi
%    \@nextwhile {SWITCH BODY} \fi
%  END

\def\@whilenoop#1{}
\def\@whilenum#1\do #2{\ifnum #1\relax #2\relax\@iwhilenum{#1\relax
     #2\relax}\fi}
\def\@iwhilenum#1{\ifnum #1\let\@nextwhile\@iwhilenum
         \else\let\@nextwhile\@whilenoop\fi\@nextwhile{#1}}

\def\@whiledim#1\do #2{\ifdim #1\relax#2\@iwhiledim{#1\relax#2}\fi}
\def\@iwhiledim#1{\ifdim #1\let\@nextwhile\@iwhiledim
        \else\let\@nextwhile\@whilenoop\fi\@nextwhile{#1}}

\long\def\@whileswnoop#1\fi{}
\long\def\@whilesw#1\fi#2{#1#2\@iwhilesw{#1#2}\fi\fi}
\long\def\@iwhilesw#1\fi{#1\let\@nextwhile\@iwhilesw
         \else\let\@nextwhile\@whileswnoop\fi\@nextwhile{#1}\fi}

% \@for NAME := LIST \do {BODY} ==
%    BEGIN \@forloop expand(LIST),\@nil,\@nil \@@ NAME {BODY} END
%
% \@forloop CAR, CARCDR, CDRCDR \@@ NAME {BODY} ==
%   BEGIN
%     NAME = CAR
%     if def(NAME) = def(\@nnil)
%       else BODY;
%            NAME = CARCDR
%            if def(NAME) = def(\@nnil)
%              else BODY
%                   \@iforloop CDRCDR \@@ NAME \do {BODY}
%            fi
%     fi
%   END
%
% \@iforloop CAR, CDR \@@ NAME {BODY} =
%     NAME = CAR
%     if def(NAME) = def(\@nnil)
%        then  \@nextwhile = def(\@fornoop)
%        else  BODY ;
%              \@nextwhile = def(\@iforloop)
%     fi
%     \@nextwhile name cdr {body}
%
% \@tfor NAME := LIST \do {BODY}
%    =  \@tforloop LIST \@nil \@@ NAME {BODY}
```

```
%
% \@tforloop car cdr \@@ name {body} =
%     name = car
%     if def(name) = def(\@nnil)
%        then  \@nextwhile == \@fornoop
%        else  body ;
%              \@nextwhile == \@forloop
%     fi
%     \@nextwhile name cdr {body}
%

\def\@nnil{\@nil}
\def\@empty{}
\def\@fornoop#1\@@#2#3{}

\def\@for#1:=#2\do#3{\edef\@fortmp{#2}\ifx\@fortmp\@empty \else
    \expandafter\@forloop#2,\@nil,\@nil\@@#1{#3}\fi}

\def\@forloop#1,#2,#3\@@#4#5{\def#4{#1}\ifx #4\@nnil \else
       #5\def#4{#2}\ifx #4\@nnil \else#5\@iforloop #3\@@#4{#5}\fi\fi}

\def\@iforloop#1,#2\@@#3#4{\def#3{#1}\ifx #3\@nnil
       \let\@nextwhile\@fornoop \else
      #4\relax\let\@nextwhile\@iforloop\fi\@nextwhile#2\@@#3{#4}}

%%RmS 91/10/17: Corrected bug in \@tfor: \xdef replaced by \def
%% (See FMi's array.doc)
\def\@tfor#1:=#2\do#3{\def\@fortmp{#2}\ifx\@fortmp\@empty \else
    \@tforloop#2\@nil\@nil\@@#1{#3}\fi}
\def\@tforloop#1#2\@@#3#4{\def#3{#1}\ifx #3\@nnil
       \let\@nextwhile\@fornoop \else
      #4\relax\let\@nextwhile\@tforloop\fi\@nextwhile#2\@@#3{#4}}


 \message{files,}
%     ****************************************
%     *           FILE HANDLING              *
%     ****************************************
%
% THE FOLLOWING USER COMMANDS ARE DEFINED IN THIS PART:
%  \document            : Reads in the .AUX files and \catcode's @ to 12.
%  \nofiles             : Suppresses all file output by setting \@filesw false.
%  \includeonly{NAME1, ... ,NAMEn}
%                       : Causes only parts NAME1, ... ,NAMEn to be read by
%                         their \include commands.  Works by setting \@partsw true
%                         and setting \@partlist to NAME1, ... ,NAMEn.
%  \include{NAME}    : Does an \input NAME unless \partsw is true and
%                         NAME is not in \@partlist.  If \@filesw is true, then
%                         it directs .AUX output to NAME.AUX, including a
%                         checkpoint at the end.
% \input{NAME}       : The same as TeX's \input, except it allows optional
%                         braces around the file name.
%
%  VARIABLES, SWITCHES AND INTERNAL COMMANDS:
%    \@mainaux    : Output file number for main .AUX file.
%    \@partaux    : Output file number for current part's .AUX file.
%    \@auxout     : Either \@mainout or \@partout, depending on which .AUX
%                     file output goes to.
%    \@input{foo} : If file foo exists, then \input's it, otherwise types
%                     a warning message.
%    @filesw       : Switch -- set false if no .AUX, .TOC, .IDX etc files are
```

```
%                   to be
%    @partsw      : Set true by a \includeonly command.
%    \@partlist   : Set to the argument of the \includeonly command.
%
%    \cp@FOO      : The checkpoint for \include'd file FOO.TEX, written
%                   by \@writeckpt at the end of file FOO.AUX
%
% \document ==
%   BEGIN
%     \endgroup   % cancels \begingroup generated by \begin command
%     \@colht := \@colroom := \vsize := \textheight
%     \columnwidth := \textwidth
%     \@clubpenalty := \clubpenalty          % \@clubpenalty saves value.
%     IF @twocolumn = T
%       THEN \columnwidth := (\columnwidth - \columnsep)/2
%            @firstcolumn := T
%     FI
%     \hsize  := \linewidth := \columnwidth
%     \begingroup
%        \@floatplacement \@dblfloatplacement
%        \@input{\jobname.aux}
%     \endgroup
%     IF \@filesw = T
%       THEN  open file \@mainaux for writing
%             write ``\relax''on file \@mainaux
%     FI
%     \do{COMMAND} == BEGIN \let COMMAND = \@notprerr END
%     \@preamblecmds
%     \do == \noexpand
%     \@normalsize
%     \everypar{}
%     @noskipsec := F
%   END
%
% \includeonly{FILELIST} ==
%  BEGIN
%   \@partsw   := T
%   \@partlist := FILELIST
%  END
%
% \include{FILE} ==
%  BEGIN
%   \clearpage
%   if \@filesw = T
%     then  \immediate\write\@mainaux{\string\@input{FILE.AUX}}
%   fi
%   if  \@partsw = T
%     then \@tempswa := F
%          \@tempb == FILE
%          for \@tempa := \@partlist
%              do if eval(\@tempa) = eval(\@tempb)
%                   then \@tempswa := T          fi
%              od
%   fi
%
%   if \@tempswa = T
%      then \@auxout := \@partaux
%           if \@filesw = T
%             then  \immediate\openout\@partaux{FILE.AUX}
%                   \immediate\write\@partaux{\relax}
%           fi
```

```
%           \@input{FILE.TEX}
%           \clearpage
%           \@writeckpt{FILE}
%           if @filesw then \closeout \@partaux fi
%           \@auxout := \@mainaux
%      else \cp@FILE
%   fi
%  END
%
% \@writeckpt{FILE} ==
%  BEGIN
%    if \@filesw = T
%        \immediate\write on file \@partaux:
%                  \gdef\cp@FILE{                  %% }
%        for \@tempa := \cl@@ckpt
%           do  \immediate\write on file \@partaux:
%                   \global\string\setcounter
%                       {eval(\@tempa)}{eval(\c@eval(\@tempa))}
%           od                                     %% {
%        \immediate\write on file \@partaux:  }
%    fi
%  END
%
%  INITIALIZATION
%    \@tempswa := T

\newif\if@filesw \@fileswtrue
\newif\if@partsw \@partswfalse
\newwrite\@mainaux
\newwrite\@partaux

\newcount\@clubpenalty

%% FMi & RmS 91/08/26 set @noskipsec switch to true in the preamble
%% and to false by \begin{document} to catch lists in the preamble,
%% i.e., to produce a ``nodocument'' error when things like
%% \maketitle appear before \begin{document}.
%
% \@noskipsectrue %% set below where switch is defined

% 91/03/26 FMi: |\process@table| added to support NFSS.
% This will also work with old lfonts if no other style defines
% |\process@table|.
%
\def\document{\endgroup
  \@colht\textheight  \@colroom\textheight \vsize\textheight
   \columnwidth\textwidth \@clubpenalty\clubpenalty
   \if@twocolumn \advance\columnwidth -\columnsep
      \divide\columnwidth\tw@ \hsize\columnwidth \@firstcolumntrue
   \fi
  \hsize\columnwidth \linewidth\hsize
  \begingroup\@floatplacement\@dblfloatplacement
   \makeatletter\let\@writefile\@gobbletwo
   \@input{\jobname.aux}\endgroup
  \if@filesw \immediate\openout\@mainaux=\jobname.aux
    \immediate\write\@mainaux{\relax}\fi
  \csname process@table\endcsname
  \let\glb@currsize\@empty  %% Force \baselineskip initialisation.
  \def\do##1{\let ##1\@notprerr}%
  \@preamblecmds
  \let\do\noexpand
```

```
  \@normalsize\everypar{}\@noskipsecfalse}

\def\@gobbletwo#1#2{}

\def\nofiles{\@fileswfalse \typeout
   {No auxiliary output files.}\typeout{}}

%% RmS 92/03/18: changed input channel from 1 to \@inputcheck to avoid
%%               conflicts with other channels allocated by \newread
\def\@input#1{\openin\@inputcheck #1 \ifeof\@inputcheck \typeout
  {No file #1.}\else\closein\@inputcheck \relax\@@input #1 \fi}
\let\@auxout=\@mainaux

\def\includeonly#1{\@partswtrue\edef\@partlist{#1}}

% In the definition of \include, \def\@tempb changed to \edef\@tempb to
% be consistent with the \edef in \includeonly.  (Suggested by Rainer
% Sch\"opf & Frank Mittelbach.  Change made 20 Jul 88.)
%
% Changed definition of \include to allow space at end of file name--
% otherwise, typing \include{foo } would cause LaTeX to overwrite
% foo.tex.  Change made 24 May 89, suggested by Rainer Sch\"opf  and
% Frank Mittelbach

\def\include#1{\@include#1 }
\def\@include#1 {\clearpage
\if@filesw \immediate\write\@mainaux{\string\@input{#1.aux}}\fi
\@tempswatrue\if@partsw \@tempswafalse\edef\@tempb{#1}\@for
\@tempa:=\@partlist\do{\ifx\@tempa\@tempb\@tempswatrue\fi}\fi
\if@tempswa \if@filesw \let\@auxout\@partaux
\immediate\openout\@partaux #1.aux
\immediate\write\@partaux{\relax}\fi\@input{#1.tex}\clearpage
\@writeckpt{#1}\if@filesw \immediate\closeout\@partaux \fi
\let\@auxout\@mainaux\else\@nameuse{cp@#1}\fi}

\def\@writeckpt#1{\if@filesw
\immediate\write\@partaux{\string\global\string\@namedef{cp@#1}\@charlb}%
{\let\@elt\@wckptelt \cl@@ckpt}\immediate\write\@partaux{\@charrb}\fi}

\def\@wckptelt#1{\immediate\write\@partaux
{\string\setcounter{#1}{\the\@nameuse{c@#1}}}}

\def\input{\@ifnextchar \bgroup{\@iinput}{\@@input }}
\def\@iinput#1{\@@input #1 }

% The following defines \@charlb and \@charrb to be { and }, respectively
% with \catcode 11.
{\catcode`[=1 \catcode`]=2
\catcode`{=11 \catcode`}=11
\gdef\@charlb[{]
\gdef\@charrb[}]
]% }brace matching


 \message{env. counters,}
%          ***************************************
%          *     ENVIRONMENT COUNTER MACROS       *
%          ***************************************
%
%  An environment  foo  has an associated counter defined by the
%  following control sequences:
```

```
%    \c@foo  :  Contains the counter's numerical value.  It is defined by
%                    \newcount\foocounter.
%    \thefoo : Macro that expands to the printed value of \foocounter.
%              For example, if sections are numbered within chapters,
%              and section headings look like
%                  Section II-3.  The Nature of Counters
%              then \thesection might be defined by:
%                 \def\thesection{\@Roman{\c@chapter}-\@arabic{\c@section}}
%
%    \p@foo  : Macro that expands to a printed 'reference prefix' of
%              counter foo.  Any \ref to a value created by counter
%              foo will produce the expansion of \p@foo\thefoo  when the
%              the \label command is executed.
%
% NOTE: \thefoo and \p@foo MUST BE DEFINED IN SUCH A WAY THAT
% \edef\bar{\thefoo} OR \edef\bar{\p@foo}
% DEFINES \bar SO THAT IT WILL EVALUATE TO THE COUNTER VALUE AT THE TIME
% OF THE \edef, EVEN AFTER \foocounter AND ANY OTHER COUNTERS HAVE BEEN
% CHANGED.  THIS WILL HAPPEN IF YOU USE THE STANDARD COMMANDS \@arabic,
% \@Roman, ETC.
%
%    \cl@foo : List of counters to be reset when foo stepped.  Has format
%              \@elt{countera}\@elt{counterb}\@elt{counterc}.
%
%  The following commands are used to define and modify counters.
%    \setcounter{FOO}{VAL}  : Globally sets \foocounter equal to VAL.
%    \addtocounter{FOO}{VAL}: Globally increments \foocounter by VAL.
%    \newcounter{NEWCTR}[OLDCTR] : Defines NEWCTR to be a counter, which is
%                               reset when counter OLDCTR is stepped.  If
%                               NEWCTR already defined produces 'c@NEWCTR
%                               already defined' error.
%    \value{CTR}           : produces the value of counter CTR, for use with
%                            a \setcounter or \addtocounter command.
%    \stepcounter{FOO}     : Globally increments counter \c@FOO
%                               and resets all subsidiary counters.
%    \refstepcounter{FOO}  : Same a \stepcounter, but it also defines
%                               \@currentreference so that a subsequent
%                               \label{bar} command causes \ref{bar} to
%                               generate the current value of counter foo.
%    \@definecounter{FOO}   : Initializes counter FOO (with empty reset list),
%                               defines \p@FOO and \theFOO to be null.
%                               Also adds FOO to \cl@@ckpt -- the reset
%                               list of a dummy counter @ckpt used for
%                               taking checkpoints.
%    \@addtoreset{FOO}{BAR} : Adds counter FOO to the list of counters
%                               \cl@BAR to be reset when counter bar is stepped.
%
%   NUMBERING MACROS:
%     \arabic{COUNTER} : Representation of COUNTER as arabic numerals.
%                        Changed 29 Apr 86 to make it print the obvious thing
%                        it COUNTER not positive.
%
%     \roman{COUNTER}  : Representation of COUNTER as lower-case
%                           Roman numerals.
%     \Roman{COUNTER}  : Representation of COUNTER as upper-case
%                           Roman numerals.
%     \alph{COUNTER}   : Representation of COUNTER as a lower-case
%                           letter: 1 = a, 2 = b, etc.
%     \Alph{COUNTER}   : Representation of COUNTER as an upper-case
%                           letter: 1 = A, 2 = B, etc.
%     \fnsymbol{COUNTER} : Representation of COUNTER as a footnote
```

```
%                           symbol: 1 = *, 2 = \dagger, etc.  Can be
%                           used only in math mode.
%
%  THE ABOVE ARE IMPLEMENTED IN TERMS OF THE FOLLOWING:
%     \@arabic\FOOcounter : Representation of \FOOcounter as arabic numerals.
%     \@roman\FOOcounter  : Representation of \FOOcounter as lower-case
%                           Roman numerals.
%     \@Roman\FOOcounter  : Representation of \FOOcounter as upper-case
%                           Roman numerals.
%     \@alph\FOOcounter   : Representation of \FOOcounter as a lower-case
%                           letter: 1 = a, 2 = b, etc.
%     \@Alph\FOOcounter   : Representation of \FOOcounter as an upper-case
%                           letter: 1 = A, 2 = B, etc.
%     \@fnsymbol\FOOcounter : Representation of \FOOcounter as a footnote
%                             symbol.  Can be used only in math mode.

\def\setcounter#1#2{\@ifundefined{c@#1}{\@nocnterr}%
{\global\csname c@#1\endcsname#2\relax}}

\def\addtocounter#1#2{\@ifundefined{c@#1}{\@nocnterr}%
{\global\advance\csname c@#1\endcsname #2\relax}}

\def\newcounter#1{\expandafter\@ifdefinable \csname c@#1\endcsname
    {\@definecounter{#1}}\@ifnextchar[{\@newctr{#1}}{}}

\def\value#1{\csname c@#1\endcsname}

\def\@newctr#1[#2]{\@ifundefined{c@#2}{\@nocnterr}{\@addtoreset{#1}{#2}}}

\def\stepcounter#1{\global\advance\csname c@#1\endcsname \@ne
    {\let\@elt\@stpelt \csname cl@#1\endcsname}}

\def\@stpelt#1{\global\csname c@#1\endcsname \z@}

\def\cl@@ckpt{\@elt{page}}

\def\@definecounter#1{\expandafter\newcount\csname c@#1\endcsname
     \setcounter{#1}0 \expandafter\gdef\csname cl@#1\endcsname{}\@addtoreset
     {#1}{@ckpt}\expandafter\gdef\csname p@#1\endcsname{}\expandafter
     \gdef\csname the#1\endcsname{\arabic{#1}}}

\def\@addtoreset#1#2{\expandafter\@cons\csname cl@#2\endcsname {{#1}}}

% Numbering commands for definitions of \theCOUNTER and \list arguments.
% \fnsymbol produces the standard footnoting symbols: asterisk, dagger, etc.
% They can be used only in math mode.

\def\arabic#1{\@arabic{\@nameuse{c@#1}}}
\def\roman#1{\@roman{\@nameuse{c@#1}}}
\def\Roman#1{\@Roman{\@nameuse{c@#1}}}
\def\alph#1{\@alph{\@nameuse{c@#1}}}
\def\Alph#1{\@Alph{\@nameuse{c@#1}}}
\def\fnsymbol#1{\@fnsymbol{\@nameuse{c@#1}}}

\def\@arabic#1{\number #1}  %% changed 29 Apr 86
\def\@roman#1{\romannumeral #1}
\def\@Roman#1{\expandafter\uppercase\expandafter{\romannumeral #1}}
\def\@alph#1{\ifcase#1\or a\or b\or c\or d\else\@ialph{#1}\fi}
\def\@ialph#1{\ifcase#1\or \or \or \or \or e\or f\or g\or h\or i\or j\or
   k\or l\or m\or n\or o\or p\or q\or r\or s\or t\or u\or v\or w\or x\or y\or
   z\else\@ctrerr\fi}
```

```
\def\@Alph#1{\ifcase#1\or A\or B\or C\or D\else\@Ialph{#1}\fi}
\def\@Ialph#1{\ifcase#1\or \or \or \or \or E\or F\or G\or H\or I\or J\or
   K\or L\or M\or N\or O\or P\or Q\or R\or S\or T\or U\or V\or W\or X\or Y\or
   Z\else\@ctrerr\fi}
\def\@fnsymbol#1{\ifcase#1\or *\or \dagger\or \ddagger\or
   \mathchar "278\or \mathchar "27B\or \|\or **\or \dagger\dagger
   \or \ddagger\ddagger \else\@ctrerr\fi\relax}



 \message{page nos.,}
%          ****************************************
%          *          PAGE NUMBERING              *
%          ****************************************
%
% Page numbers are produced by a page counter, used just like any other
% counter.  The only difference is that \c@page contains the number of
% the next page to be output (the one currently being produced), rather
% than one minus it.  Thus, it is normally initialized to 1 rather than
% 0.  \c@page is defined to be \count0, rather than a count assigned by
% \newcount.
%
% The user sets the pagenumber style with the \pagenumbering{FOO}
% command, which sets the page counter to 1 and defines \thepage to be
% \FOO.  For example, \pagenumbering{roman} causes pages to be numbered
% i, ii, etc.


\countdef\c@page=0 \c@page=1
\def\cl@page{}
\def\pagenumbering#1{\global\c@page \@ne \gdef\thepage{\csname @#1\endcsname
   \c@page}}


 \message{x-ref,}
%          ****************************************
%          *      CROSS REFERENCING MACROS        *
%          ****************************************
%
%  The user writes  \label{foo}  to define the following cross-references:
%     \ref{foo}     : value of most recently incremented referencable counter.
%                     in the current environment. (Chapter, section, theorem
%                     and enumeration counters counters are referencable,
%                     footnote counters are not.)
%     \pageref{foo} : page number at which \label{foo} command appeared.
%  where  foo  can be any string of characters not containing '\', '{' or '}'.
%
%  Note: The scope of the \label command is delimited by environments, so
%          \begin{theorem} \label{foo} ... \end{theorem} \label{bar}
%  defines \ref{foo} to be the theorem number and \ref{bar} to be
%  the current section number.
%
%  Note: \label does the right thing in terms of spacing -- i.e.,
%  leaving a space on both sides of it is equivalent to leaving
%  a space on either side.
%
%  This is implemented as follows.  A referencable counter  CNT  is
%  incremented by the command  \refstepcounter{CNT} , which sets
%  \@currentlabel == {CNT}{eval(\p@cnt\theCNT)}.   The command
%  \label{FOO} then writes the following on file \@auxout :
```

```
%        \newlabel{FOO}{{eval(\@currentlabel)}{eval(\thepage)}}
%
%  \ref{FOO} ==
%    BEGIN
%      if \r@foo undefined
%        then  ??
%              Warning: 'reference foo on page ... undefined'
%        else  \@car \eval(\r@FOO)\@nil
%      fi
%    END
%
%  \pageref{foo} =
%    BEGIN
%      if \r@foo undefined
%        then  ??
%              Warning: 'reference foo on page ... undefined'
%        else  \@cdr \eval(\r@FOO)\@nil
%      fi
%    END
%

%% RmS 91/10/25: added a few extra \reset@font,
%%               as suggested by Bernd Raichle
\def\ref#1{\@ifundefined{r@#1}{{\reset@font\bf ??}\@warning
   {Reference `#1' on page \thepage \space
    undefined}}{\edef\@tempa{\@nameuse{r@#1}}\expandafter
    \@car\@tempa \@nil\null}}

\def\pageref#1{\@ifundefined{r@#1}{{\reset@font\bf ??}\@warning
   {Reference `#1' on page \thepage \space
    undefined}}{\edef\@tempa{\@nameuse{r@#1}}\expandafter
    \@cdr\@tempa\@nil\null}}

\def\newlabel#1#2{\@ifundefined{r@#1}{}{\@warning{Label `#1' multiply
   defined}}\global\@namedef{r@#1}{#2}}

% \label and \refstepcounter changed to allow \protect'ed commands to
% work properly.  For example,
%   \def\thechapter{\protect\foo{\arabic{chapter}.\roman{section}}}
% will cause a \label{bar} command to define \ref{bar} to expand to
% something like \foo{4.d}.  Change made 20 Jul 88.

\def\label#1{\@bsphack\if@filesw {\let\thepage\relax
   \def\protect{\noexpand\noexpand\noexpand}%
   \edef\@tempa{\write\@auxout{\string
      \newlabel{#1}{{\@currentlabel}{\thepage}}}}%
   \expandafter}\@tempa
   \if@nobreak \ifvmode\nobreak\fi\fi\@esphack}

\def\refstepcounter#1{\stepcounter{#1}\let\@tempa\protect
\def\protect{\noexpand\protect\noexpand}%
\edef\@currentlabel{\csname p@#1\endcsname\csname the#1\endcsname}%
\let\protect\@tempa}

\def\@currentlabel{} % For \label commands that come before any environment

 \message{environments,}
%          ****************************************
%          *            ENVIRONMENTS              *
%          ****************************************
%
```

```
%  \begin{foo} and \end{foo} are used to delimit environment foo.
%  \begin{foo} starts a group and calls \foo if it is defined, otherwise
%  it does nothing.  \end{foo} checks to see that it matches the
%  corresponding \begin and if so, it calls \endfoo and does an
%  \endgroup.  Otherwise, \end{foo} does nothing.
%
%  If \end{foo} needs to ignore blanks after it, then \endfoo should
%  globally set the @ignore switch true with \global\@ignoretrue.
%
%  \@currenvir : the name of the current environment.  Initialized to
%                'document' to make \end{document} work right.
%
%  \@preamblecmds : a list of commands that can be used only in the
%                   preamble (before the \begin{document}), in the
%                   form  \do \CMDA \do \CMDB ... .   These commands
%                   are redefined to \@notprerr by \begin{document}
%                   to save space.  They include the following:
%                       \document \documentstyle \@documentstyle
%                       \@options \@preamblecmds \@optionlist
%                       \@optionfiles \nofiles \includeonly \makeindex
%                       \makeglossary
%                   The document style can add any other commands to
%                   this list by
%                      \def\do{\noexpand\do\noexpand}
%                      \edef\@preamblecmds{\@preamblecmds \do ...}
%
%  NOTE: \@@end is defined to be the \end command of TeX82.
%
%  \enddocument is the user's command for ending the manuscript file.
%
%  \stop is a panic button -- to end TeX in the middle.
%
% \enddocument ==
%   BEGIN
%    \@checkend{document}   %% checks for unmatched \begin
%    \clearpage
%    \begingroup
%      if @filesw = true
%        then  close file @mainaux
%              \global \@namedef {ARG1}{ARG2} == null
%              \newlabel{LABEL}{VAL} ==
%                  BEGIN
%                    \@tempa == VAL
%                    if def(\@tempa) = def(\r@LABEL)
%                      else @tempswa := true          fi
%                  END
%              \bibcite{LABEL}{VAL} == null
%                  BEGIN
%                    \@tempa == VAL
%                    if def(\@tempa) = def(\g@LABEL)
%                      else @tempswa := true          fi
%                  END
%              @tempswa := false
%              make @ a letter
%              \input \jobname.AUX
%              if @tempswa = true
%                then LaTeX Warning: 'Label may have changed.
%                                    Rerun to get cross-references right.'
%       fi     fi
%    \endgroup
%    finish up
```

```
%   END
%
%  \@writefile{EXT}{ENTRY} ==
%      if tf@EXT undefined
%        else \write\tf@EXT{ENTRY}
%      fi
%
\def\@currenvir{document}

\def\@preamblecmds{\do\document \do\documentstyle \do\@documentstyle
   \do\@options \do\@preamblecmds \do\@optionlist \do\@optionfiles
   \do\nofiles \do\includeonly \do\makeindex \do\makeglossary}

\newif\if@ignore

\def\enddocument{\@checkend{document}\clearpage\begingroup
\if@filesw \immediate\closeout\@mainaux
\def\global\@namedef##1##2{}\def\newlabel{\@testdef r}%
\def\bibcite{\@testdef b}\@tempswafalse \makeatletter\input \jobname.aux
\if@tempswa \@@warning{Label(s) may have changed.  Rerun to get
cross-references right}\fi\fi\endgroup\deadcycles\z@\@@end}

\def\@testdef #1#2#3{\def\@tempa{#3}\expandafter \ifx \csname #1@#2\endcsname
 \@tempa  \else \@tempswatrue \fi}

\long\def\@writefile#1#2{\@ifundefined{tf@#1}{}{%
   \immediate\write\csname tf@#1\endcsname{#2}}}
 % \long added 8 Feb 90, as suggested by Chris Rowley

\def\stop{\clearpage\deadcycles\z@\let\par\@@par\@@end}

\everypar{\@nodocument} %% To get an error if text appears before the
\nullfont               %% \begin{document}

% \begin, \end, and \@checkend changed so \end{document} will catch
% an unmatched \begin.  Changed 24 May 89 as suggested by
% Frank Mittelbach and Rainer Sch\"opf.


% \begin{NAME} ==
%  BEGIN
%    IF \NAME undefined  THEN  \@tempa == BEGIN report error END
%                        ELSE  \@tempa ==  (\@currenvir :=L NAME) \NAME
%    FI
%    @ignore :=G F      %% Added 30 Nov 88
%    \begingroup
%    \@currenvir :=L NAME
%    \NAME
%  END

% \end{NAME} ==
%  BEGIN
%   \endNAME
%   \@checkend{NAME}
%   IF @endpe = T                 %% @endpe set True by \@endparenv
%     THEN \@gtempa :=G \@doendpe  %% \@doendpe redefines \par and \everypar
%     ELSE \@gtempa :=G \relax     %% to suppress paragraph indentation in
%   FI                             %% immediately following text
%   \endgroup
%   \@gtempa
%   IF @ignore = T
```

```
%     THEN @ignore :=G F
%          \ignorespaces
%   FI
%  END

% \@checkend{NAME} ==
%  BEGIN
%   IF \@currenvir = NAME
%     ELSE \@badend{NAME}
%   FI
%  END

%% RmS 92/03/18: changed \@ignoretrue to \@ignorefalse (as documented)
\def\begin#1{\@ifundefined{#1}{\def\@tempa{\@latexerr{Environment #1
  undefined}\@eha}}{\def\@tempa{\def\@currenvir{#1}%
  \csname #1\endcsname}}\global\@ignorefalse %% \global... added 2 May 90
     \begingroup\@endpefalse\@tempa}

\def\end#1{\csname end#1\endcsname\@checkend{#1}%
  \expandafter\endgroup \if@endpe \@doendpe \fi
  \if@ignore \global\@ignorefalse \ignorespaces\fi}

\def\@checkend#1{\def\@tempa{#1}\ifx
      \@tempa\@currenvir \else\@badend{#1}\fi}


 \message{math,}
%      ********************************************
%      *               MATH ENVIRONMENTS          *
%      ********************************************
%
% \( ==  BEGIN  if math mode
%                 then error: '\( in math mode'
%                 else $
%               fi
%        END
%
% \) ==  BEGIN  if math mode
%                 then if inner mode
%                        then $
%                        else error ``\[ closed with \)''
%                 else error 'unmatched \)'
%               fi
%        END
%
% \[ ==  BEGIN  if math mode
%                 then error: '\[ in math mode'
%                 else $$
%               fi
%        END
%
% \] ==  BEGIN  if math mode
%                 then if inner mode
%                        then error '\( closed with \]'
%                        else $$
%                 else error 'unmatched \]'
%               fi
%        END
%
% \equation      ==  BEGIN \refstepcounter{equation} $$ END
%
```

```
% \endequation   ==  BEGIN \eqno (\theequation) $$\ignorespaces END
%
% NOTE: The document style must define \theequation etc., and do
% the appropriate \@addtoreset.  It should also redefine \@eqnnum
% if another format for the equation number is desired other than the
% standard (...), or to move the equation numbers to the flushleft.
% (See comment on the \def of \@eqnnum.)
%
% \stackrel{TOP}{BOT} == PLAIN TeX's \buildrel {TOP} \over {BOT}
%
% \frac{TOP}{BOT} == {TOP \over BOT}
%
% \sqrt[N]{EXP} produces an Nth root of EXP formula.
%
%  \: == \>  (medium space)

\def\({\relax\ifmmode\@badmath\else$%%$BRACE MATCH HACK
\fi}

\def\){\relax\ifmmode\ifinner$\else\@badmath%%$ BRACE MATCH HACK
\fi\else \@badmath\fi}

\def\[{\relax\ifmmode\@badmath\else
    \ifvmode \nointerlineskip \makebox[.6\linewidth]\fi$$%%$$ BRACE MATCH HACK
\fi}

\def\]{\relax\ifmmode\ifinner\@badmath\else$$\fi%%$$ BRACE MATCH HACK
        \else \@badmath \fi\ignorespaces}

\let\math=\(
\let\endmath=\)
\def\displaymath{\[}
\def\enddisplaymath{\]\global\@ignoretrue}

\@definecounter{equation}
\def\equation{$$ % $$ BRACE MATCHING HACK
\refstepcounter{equation}}

%% RmS 92/01/10: put \hbox around \@eqnnum to typeset the equation
%%               number in text mode (as in the eqnarray env.).
\def\endequation{\eqno \hbox{\@eqnnum}% $$ BRACE MATCHING HACK
$$\global\@ignoretrue}

%  \@eqnnum: Produces the equation number for equation and
%     eqnarray environments.  The following definition is for
%     flushright numbers; for flushleft numbers, see leqno.doc.
%     The {\rm ... } puts the equation number in roman type even if
%     an eqnarray environment appears in an italic environment.
%
%% RmS 91/09/29: \reset@font added.
\def\@eqnnum{{\reset@font\rm (\theequation)}}


\def\stackrel#1#2{\mathrel{\mathop{#2}\limits^{#1}}}
\def\frac#1#2{{#1\over #2}}

\let\@@sqrt=\sqrt
\def\sqrt{\@ifnextchar[{\@sqrt}{\@@sqrt}}
\def\@sqrt[#1]{\root #1\of}

\let\:=\>
```

```
3051
3052 % Here's the eqnarray environment:
3053 %  Default is for left-hand side of equations to be flushleft.
3054 %  To make them flushright, \let\@eqnsel = \hfil
3055
3056 \newcount\@eqcnt
3057 \newcount\@eqpen
3058 \newif\if@eqnsw\@eqnswtrue
3059
3060 \@centering = 0pt plus 1000pt % Changed 11/4/85 to produce warning message
3061                               % if line extends into margin.  Doesn't warn
3062                               % about formula overprinting equation number.
3063
3064 \def\eqnarray{\stepcounter{equation}\let\@currentlabel\theequation
3065 \global\@eqnswtrue\m@th
3066 \global\@eqcnt\z@\tabskip\@centering\let\\\@eqncr
3067 $$\halign to\displaywidth\bgroup\@eqnsel\hskip\@centering
3068   $\displaystyle\tabskip\z@{##}$&\global\@eqcnt\@ne
3069   \hskip 2\arraycolsep \hfil${##}$\hfil
3070   &\global\@eqcnt\tw@ \hskip 2\arraycolsep $\displaystyle\tabskip\z@{##}$\hfil
3071    \tabskip\@centering&\llap{##}\tabskip\z@\cr}
3072
3073 \def\endeqnarray{\@@eqncr\egroup
3074       \global\advance\c@equation\m@ne$$\global\@ignoretrue}
3075
3076 \let\@eqnsel=\relax
3077
3078 \def\nonumber{\global\@eqnswfalse}
3079
3080 \def\@eqncr{{\ifnum0=`}\fi\@ifstar{\global\@eqpen\@M
3081     \@yeqncr}{\global\@eqpen\interdisplaylinepenalty \@yeqncr}}
3082
3083 \def\@yeqncr{\@ifnextchar [{\@xeqncr}{\@xeqncr[\z@]}}
3084
3085 \def\@xeqncr[#1]{\ifnum0=`{\fi}\@@eqncr
3086    \noalign{\penalty\@eqpen\vskip\jot\vskip #1\relax}}
3087
3088 \def\@@eqncr{\let\@tempa\relax
3089     \ifcase\@eqcnt \def\@tempa{& & &}\or \def\@tempa{& &}%
3090       \else \def\@tempa{&}\fi
3091      \@tempa \if@eqnsw\@eqnnum\stepcounter{equation}\fi
3092      \global\@eqnswtrue\global\@eqcnt\z@\cr}
3093
3094 % Here's the eqnarray* environment:
3095
3096 \let\@seqncr=\@eqncr
3097 \@namedef{eqnarray*}{\def\@eqncr{\nonumber\@seqncr}\eqnarray}
3098 \@namedef{endeqnarray*}{\nonumber\endeqnarray}
3099
3100 % \lefteqn{FORMULA} typesets FORMULA in display math style
3101 %  flushleft in a box of width zero.
3102 %
3103
3104 \def\lefteqn#1{\hbox to\z@{$\displaystyle #1$\hss}}
3105
3106
3107  \message{center,}
3108 %      ************************************************
3109 %      *      CENTER, FLUSHRIGHT, FLUSHLEFT, ETC.     *
3110 %      ************************************************
3111 %
```

```
%
% \center, \flushright and \flushleft set
%   \rightskip = 0pt or \@flushglue (as appropriate)
%   \leftskip  = 0pt or \@flushglue (as appropriate)
%   \parindent = 0pt
%   \parfillskip   = 0pt. (except \flushleft)
%   \\         == \par \vskip -\parskip
%   \\[LENGTH] == \\ \vskip LENGTH
%   \\*        == \par \penalty 10000 \vskip -\parskip
%   \\*[LEN]   == \\* \vskip LENGTH
%
% They invoke the trivlist environment to handle vertical spacing before
% and after them.
%
% \centering, \raggedright and \raggedleft are the declaration analogs
% of the above.
%
% \raggedright has a more universal effect, however.  It sets
% \@rightskip := flushglue.  Every environment, like the list environments,
% that set \rightskip to its 'normal' value set it to \@rightskip

\def\@centercr{\ifhmode \unskip\else \@badcrerr\fi
       \par\@ifstar{\penalty \@M\@xcentercr}{\@xcentercr}}

\def\@xcentercr{\addvspace{-\parskip}\@ifnextchar
    [{\@icentercr}{\ignorespaces}}

\def\@icentercr[#1]{\vskip #1\ignorespaces}

\def\center{\trivlist \centering\item[]}
\def\centering{\let\\=\@centercr\rightskip\@flushglue\leftskip\@flushglue
\parindent\z@\parfillskip\z@}
\let\endcenter=\endtrivlist

\newskip\@rightskip \@rightskip \z@

\def\flushleft{\trivlist \raggedright\item[]}
\def\raggedright{\let\\=\@centercr\@rightskip\@flushglue \rightskip\@rightskip
  \leftskip\z@
  \parindent\z@}
\let\endflushleft=\endtrivlist

\def\flushright{\trivlist \raggedleft\item[]}
\def\raggedleft{\let\\=\@centercr\rightskip\z@\leftskip\@flushglue
  \parindent\z@\parfillskip\z@}
\let\endflushright=\endtrivlist

 \message{verbatim,}
%       ****************************************
%       *              VERBATIM                *
%       ****************************************
%
%  The verbatim environment uses the fixed-width \tt font, turns blanks into
%  spaces, starts a new line for each carrige return (or sequence of
%  consecutive carriage returns), and interprets EVERY character literally.
%  I.e., all special characters \, {, $, etc. are \catcode'd to 'other'.
%
%  The command \verb produces in-line verbatim text, where the argument
%  is delimited by any pair of characters.  E.g., \verb #...# takes
%  '...' as its argument, and sets it verbatim in \tt font.
%
```

```
%  The *-variants of these commands is the same, except that spaces
%  print as the TeXbook's space character instead of as blank spaces.

{\catcode`\^^M=13 \gdef\@gobblecr{\@ifnextchar
{\@gobble}{\ignorespaces}}}

{\catcode`\ =\active\gdef\@vobeyspaces{\catcode`\ \active\let \@xobeysp}}

% Definition of \@xobeysp chaned on 19 Nov 86 from
% \def\@xobeysp{\leavevmode{} }
% to prevent line breaks at spaces.  Change suggested by
% Nelson Beebe
%
\def\@xobeysp{\leavevmode\penalty10000\ }



\begingroup \catcode `|=0 \catcode `[= 1
\catcode`]=2 \catcode `\{=12 \catcode `\}=12
\catcode`\\=12 |gdef|@xverbatim#1\end{verbatim}[#1|end[verbatim]]
|gdef|@sxverbatim#1\end{verbatim*}[#1|end[verbatim*]]
|endgroup

% \@sverbatim obsolete -- removed 24 May 89, as suggested by
% Rainer Sch\"opf and Frank Mittelbach
% \def\@sverbatim{\obeyspaces\@verbatim}

\def\@gobble#1{}

% 91/07/24 RmS: added \penalty\interlinepenalty to definition
%               of \par so that \samepage works.

\def\@verbatim{\trivlist \item[]\if@minipage\else\vskip\parskip\fi
\leftskip\@totalleftmargin\rightskip\z@
\parindent\z@\parfillskip\@flushglue\parskip\z@
%%RmS 91/08/26 Added \@@par to clear possible \parshape definition
%%from a surrounding list (the verbatim guru says)
\@@par
\@tempswafalse \def\par{\if@tempswa\hbox{}\fi\@tempswatrue\@@par
\penalty\interlinepenalty}%
\obeylines \tt \catcode``=13 \@noligs \let\do\@makeother \dospecials}

\def\verbatim{\@verbatim \frenchspacing\@vobeyspaces \@xverbatim}
\let\endverbatim=\endtrivlist

\@namedef{verbatim*}{\@verbatim\@sxverbatim}
\expandafter\let\csname endverbatim*\endcsname =\endtrivlist


\def\@makeother#1{\catcode`#1=12\relax}

\def\verb{\begingroup \catcode``=13 \@noligs
\tt \let\do\@makeother \dospecials
\@ifstar{\@sverb}{\@verb}}

% Definitions of \@sverb and \@verb changed so \verb+ foo+  does not lose
% leading blanks when it comes at the beginning of a line.
% Change made 24 May 89. Suggested by Frank Mittelbach and Rainer Sch\"opf.
%
\def\@sverb#1{\def\@tempa ##1#1{\leavevmode\null##1\endgroup}\@tempa}

```

```
\def\@verb{\@vobeyspaces \frenchspacing \@sverb}


%% \@noligs prevents ?` and !` from being treated as ligatures
%% added 19 April 86

\begingroup
\catcode``=13
\gdef\@noligs{\let`\@lquote}
\endgroup

%% RmS 91/06/21: added \leavevmode to definition of \@lquote
%%  to avoid the \kern being processed in vertical mode

\def\@lquote{\leavevmode{\kern\z@}`}
 \message{list,}
%       ****************************************
%       *         THE LIST ENVIRONMENT         *
%       ****************************************
%
% The generic commands for creating an indented environment -- enumerate,
% itemize, quote, etc -- are
%        \list{LABEL}{COMMANDS} ... \endlist
% which can be invoked by the user as the list environment.  The LABEL
% argument specifies item labeling.  COMMANDS contains commands for
% changing the horizontal and vertical spacing parameters.
%
% Each item of the environment is begun by the command \item[ITEMLABEL]
% which produces an item labeled by ITEMLABEL.  If the argument is
% missing, then the LABEL argument of the \list command is used as the
% item label.
%
% The label is formed by putting \makelabel{ITEMLABEL} in an hbox whose
% width is either its natural width or else \labelwidth, whichever is
% larger.  The \list command defines \makelabel to have the default
% definition
%     \makelabel{ARG} == BEGIN \hfil ARG END
% which, for a label of width less than \labelwidth, puts the label
% flushright, \labelsep to the left of the item's text.  However,
% \makelabel can be \let to another command by the \list's COMMANDS
% argument.
%
% A \usecounter{foo} command in the second argument causes the counter
% foo to be initialized to zero, and stepped by every \item command
% without an argument.  (\label commands within the list refer to this
% counter.)
%
% When you leave a list environment, returning either to an enclosing
% list or normal text mode, LaTeX begins a new paragraph if and only if
% you leave a blank line after the \end command.  This is accomplished
% by the \@endparenv command.
%
% Blank lines are ignored every other reasonable place--i.e.:
%     - Between the \begin{list} and the first \item,
%     - Between the \item and the text of that item.
%     - Between the end of the last item and the \end{list}.
%
% For an environment like quotation, in which items are not labeled,
% the entire environment is a single item.  It is defined by
% letting \quotation == \list{}{...}\item[].  (Note the [], there in
% case the first character in the environment is a '['.)  The spacing
```

```
% parameters provide a great deal of flexability in designing the
% format, including the ability to let the indentation of the first
% paragraph be different from that of the subsequent ones.
%
% The trivlist environment is equivalent to a list environment
% whose second argument sets the following parameter values:
%    \leftmargin = 0 : causes no indentation of left margin
%    \labelwidth = 0 : see below for precise effect this has.
%    \itemindent = 0 : with a null label, makes first paragraph
%        have no indentation.  Succeeding paragraphs have \parindent
%        indentation.  To give first paragraph same indentation, set
%        \itemindent = \parindent before the \item[].
% Every \item in a trivlist environment must have an argument---in many
% cases, this will be the null argument (\item[]).  The trivlist
% environment is mainly used for paragraphing environments, like
% verbatim, in which there is no margin change.  It provides the same
% vertical spacing as the list environment, and works reasonably well
% when it occurs immediately after an \item command in an enclosing list.
%
% The following variables are used inside a list environment:
%   \@totalleftmargin : The distance that the prevailing left margin is
%                       indented from the outermost left margin,
%   \linewidth        : The width of the current line.  Must be
%                       initialized to \hsize.
%   \@listdepth       : A count for holding current list nesting depth.
%   \makelabel        : A macro with a single argument, used to generate
%                       the label from the argument (given or implied) of the
%                       \item command.  Initialized to \@mklab by the \list
%                       command.  This command must produce some stretch--i.e.,
%                       an \hfil.
%   @inlabel          : A switch that is false except between the time an
%                       \item is encountered and the time that TeX actually
%                       enters horizontal mode.  Should be tested by
%                       commands that can be messed up by the list
%                       environment's use of \everypar.
%   \box\@labels      : When @inlabel = true, it holds the labels
%                       to be put out by \everypar.
%   @noparitem        : A switch set by \list when @inlabel = true.
%                       Handles the case of a \list being the first thing
%                       in an item.
%   @noparlist        : A switch set true for a list that begins an
%                       item.  No \topsep space is added before or after
%                       such a list.
%   @newlist          : Set true by \list, set false by the first \item's
%                       text (by \everypar).
%   @noitemarg        : Set true when executing an \item with no explicit
%                       argument.  Used to save space.  To save time,
%                       make two separate \@item commands.
%   @nmbrlist         : Set true by \usecounter command, causes list to
%                       be numbered.
%   \@listctr         : \def'ed by \usecounter to name of counter.
%   @noskipsec        : A switch set true by a sectioning command when it is
%                       creating an in-text heading with \everypar.
%
% Throughout a list environment, \hsize is the width of the current
% line, measured from the outermost left margin to the outermost right
% margin.  Environments like tabbing should use \linewidth instead of
% \hsize.
%
% Here are the parameters of a list that can be set by commands in
% the \list's COMMANDS argument.  These parameters are all TeX
```

```
% skips or dimensions (defined by \newskip or \newdimen), so the usual
% TeX or LaTeX commands can be used to set them.  The commands will
% be executed in vmode if and only if the \list was preceded by a
% \par (or something like an \end{list}), so the spacing parameters
% can be set according to whether the list is inside a paragraph
% or is its own paragraph.
%
%   VERTICAL SPACING (skips):
%
%      \topsep   : Space between first item and preceding paragraph.
%      \partopsep : Extra space added to \topsep when environment starts
%                   a new paragraph (is called in vmode).
%      \itemsep : Space between successive items.
%      \parsep   : Space between paragraphs within an item -- the \parskip
%                  for this environment.
%
%   PENALTIES
%     \@beginparpenalty : put at the beginning of a list
%     \@endparpenalty   : put at end of list
%     \@itempenalty     : put between items.
%
%   HORIZONTAL SPACING (dimens)
%      \leftmargin    : space between left margin of enclosing environment
%                       (or of page if top level list) and left margin of
%                       this list.  Must be nonnegative.
%      \rightmargin   : analogous.
%      \listparindent : extra indentation at beginning of every paragraph
%                       of a list except the one started by the \item
%                       command.  May be negative!  Usually, labeled lists
%                       have \listparindent equal to zero.
%      \itemindent    : extra indentation added right BEFORE an item label.
%      \labelwidth    : nominal width of box that contains the label.
%                       If the natural width of the label < = \labelwidth,
%                       then the label is flushed right inside a box
%                       of width \labelwidth (with an \hfil).  Otherwise,
%                       a box of the natural width is employed, which causes
%                       an indentation of the text on that line.
%      \labelsep      : space between end of label box and text of
%                       first item.
%
%   DEFAULT VALUES:
%      Defaults for the list environment are set as follows.
%      First, \rightmargin, \listparindent and \itemindent are set
%      to 0pt.  Then, one of the commands \@listi, \@listii, ... , \@listvi
%      is called, depending upon the current level of the list.
%      The \@list... commands should be defined by the document
%      style.  A convention that the document style should follow is
%      to set \leftmargin to \leftmargini, ... , \leftmarginvi for
%      the appropriate level.  Items that aren't changed may be left
%      alone, but everything that could possibly be changed must be
%      reset.
%
%  \list{LABEL}{COMMANDS} ==
%   BEGIN
%     if \@listdepth > 5
%       then LaTeX error: 'Too deeply nested'
%       else \@listdepth :=G \@listdepth + 1
%     fi
%     \rightmargin     := 0pt
%     \listparindent   := 0pt
%     \itemindent      := 0pt
```

```
%     \eval(@list \romannumeral\the\@listdepth)  %% Set default values:
%     \@itemlabel      :=L LABEL
%     \makelabel       == \@mklab
%     @nmbrlist        :=L false
%     COMMANDS
%
%     \@trivlist               % commands common to \list and \trivlist
%
%     \parskip          :=L \parsep
%     \parindent        :=L \listparindent
%     \linewidth        :=L \linewidth - \rightmargin -\leftmargin
%     \@totalleftmargin :=L \@totalleftmargin + \leftmargin
%     \parshape 1 \@totalleftmargin \linewidth
%     \ignorespaces                    % gobble space up to \item
%    END
%
% \endlist == BEGIN \@listdepth :=G \@listdepth -1
%                   \endtrivlist
%             END
%
% \@trivlist ==
%  BEGIN
%     if @newlist = T then \@noitemerr fi  %% This command removed for some
%                                          %% forgotten reason.
%     \@topsepadd :=L \topsep
%     if @noskipsec then leave vertical mode fi  %% Added 11 Jun 85
%     if vertical mode
%       then \@topsepadd :=L \@topsepadd + \partopsep
%       else \unskip \par                % remove glue from end of last line
%     fi
%     if @inlabel = true
%        then @noparitem :=L true
%             @noparlist :=L true
%        else @noparlist :=L false
%             \@topsep   :=L \@topsepadd
%     fi
%     \@topsep         :=L \@topsep + \parskip  %% Change 4 Sep 85
%     \leftskip        :=L 0pt           % Restore paragraphing parameters
%     \rightskip       :=L \@rightskip
%     \parfillskip     :=L 0pt + 1fil
%
%   NOTE: \@setpar called on every \list in case \par has been temporarily
%         munged before the \list command.
%     \@setpar{if @newlist = false then {\@@par} fi}
%     \@newlist         :=G T
%     \@outerparskip    :=L \parskip
% END
%
% \trivlist  ==
% BEGIN
%  \parsep     := \parskip
%  \@trivlist
%  \labelwidth := 0
%  \leftmargin := 0
%  \itemindent := \parindent
%  \makelabel{LABEL} == LABEL
% END
%
% \endtrivlist ==
%   BEGIN
%     if @inlabel = T then \indent fi
```

```
%     if horizontal mode then \unskip \par fi
%     if @noparlist = true
%       else if \lastskip > 0
%               then \@tempskipa := \lastskip
%                    \vskip - \lastskip
%                    \vskip \@tempskipa -\@outerparskip + \parskip
%            fi
%            \@endparenv
%     fi
%   END
%
% \@endparenv ==
%   BEGIN
%    \addpenalty{@endparpenalty}
%    \addvspace{\@topsepadd}
%    \endgroup    %% ends the \begin command's \begingroup
%    \par  ==  BEGIN
%                \@restorepar
%                \everypar{}
%                \par
%              END
%    \everypar == BEGIN remove \lastbox \everypar{} END
%    \begingroup  %% to match the \end commands \endgroup
%   END
%
% \item == BEGIN if  next char = [
%                  then  \@item
%                  else  @noitemarg := true
%                        \@item[@itemlabel]
%          END
%
% \@item[LAB] ==
%    BEGIN
%     if @noparitem = true
%       then @noparitem := false                        % NOTE: then clause
%                                                       % hardly every taken,
%            \box\@labels :=G \hbox{\hskip -\leftmargin % so made a macro
%                                 \box\@labels        % \@donoparitem
%                                 \hskip \leftmargin }
%            if @minipage = false then
%               \@tempskipa := \lastskip
%               \vskip -\lastskip
%               \vskip \@tempskipa + \@outerparskip - \parskip
%            fi
%       else if @inlabel = true
%              then \indent \par   % previous item empty.
%            fi
%            if hmode then 2 \unskip's  % To remove any space at end of prev.
%                     \par              % paragraph that could cause a blank
%            fi                         % line.
%            if @newlist = T
%               then if @nobreak = T   % Kludge if list follows \section
%                      then \addvspace{\@outerparskip - \parskip}
%                      else \addpenalty{\@beginparpenalty}
%                           \addvspace{\@topsep}
%                           \addvspace{-\parskip}   %% added 4 Sep 85
%                    fi
%               else \addpenalty{\@itempenalty}
%                    \addvspace{\itemsep}
%            fi
%            @inlabel :=G true
```

```
%     fi
%
%     \everypar{ @minipage :=G F
%                @newlist :=G F
%                if  @inlabel = true
%                  then @inlabel :=G false
%                       \hskip -\parindent
%                       \box\@labels
%                       \penalty 0       %% 3 Oct 85  -- allow line break here
%                       \box\@labels :=G null
%                fi
%                \everypar{} }
%     @nobreak :=G false
%     if  @noitemarg = true
%       then @noitemarg := false
%            if @nmbrlist
%              then \refstepcounter{\@listctr}
%     fi     fi
%     \@tempboxa   :=L \hbox{\makelabel{LAB}}
%     \box\@labels :=G \@labels \hskip \itemindent
%                       \hskip - (\labelwidth + \labelsep)
%                       if \wd \@tempboxa > \labelwidth
%                          then \box\@tempboxa
%                          else \hbox to \labelwidth {\makelabel{LAB}}
%                       fi
%                       \hskip\labelsep
%     \ignorespaces                        %gobble space up to text
%   END
%
%   \usecounter{CTR} == BEGIN  @nmbrlist :=L true
%                              \@listctr == CTR
%                              \setcounter{CTR}{0}
%                       END
%
% DEFINE \dimen's and \count
\newskip\topsep
\newskip\partopsep
\newskip\itemsep
\newskip\parsep
\newskip\@topsep
\newskip\@topsepadd
\newskip\@outerparskip

\newdimen\leftmargin
\newdimen\rightmargin
\newdimen\listparindent
\newdimen\itemindent
\newdimen\labelwidth
\newdimen\labelsep
\newdimen\linewidth
\newdimen\@totalleftmargin \@totalleftmargin=\z@
\newdimen\leftmargini
\newdimen\leftmarginii
\newdimen\leftmariniii
\newdimen\leftmariniv
\newdimen\leftmarinv
\newdimen\leftmarinvi

\newcount\@listdepth \@listdepth=0
\newcount\@itempenalty
\newcount\@beginparpenalty
```

```
\newcount\@endparpenalty

\newbox\@labels

\newif\if@inlabel \@inlabelfalse
\newif\if@newlist   \@newlistfalse
\newif\if@noparitem \@noparitemfalse
\newif\if@noparlist \@noparlistfalse
\newif\if@noitemarg \@noitemargfalse
\newif\if@nmbrlist  \@nmbrlistfalse

\def\list#1#2{\ifnum \@listdepth >5\relax \@toodeep
     \else \global\advance\@listdepth\@ne \fi
  \rightmargin \z@ \listparindent\z@ \itemindent\z@
  \csname @list\romannumeral\the\@listdepth\endcsname
  \def\@itemlabel{#1}\let\makelabel\@mklab \@nmbrlistfalse #2\relax
  \@trivlist
  \parskip\parsep \parindent\listparindent
  \advance\linewidth -\rightmargin \advance\linewidth -\leftmargin
  \advance\@totalleftmargin \leftmargin
  \parshape \@ne \@totalleftmargin \linewidth
  \ignorespaces}

\def\@trivlist{\@topsepadd\topsep
  \if@noskipsec \leavevmode \fi
  \ifvmode \advance\@topsepadd\partopsep \else \unskip\par\fi
  \if@inlabel \@noparitemtrue \@noparlisttrue
    \else \@noparlistfalse \@topsep\@topsepadd \fi
    \advance\@topsep \parskip
  \leftskip\z@\rightskip\@rightskip \parfillskip\@flushglue
  \@setpar{\if@newlist\else{\@@par}\fi}%
  \global\@newlisttrue \@outerparskip\parskip}

%% RmS 92/03/18 added \@nmbrlistfalse
\def\trivlist{\parsep\parskip\@nmbrlistfalse
  \@trivlist \labelwidth\z@ \leftmargin\z@
  \itemindent\z@ \def\makelabel##1{##1}}

\def\endlist{\global\advance\@listdepth\m@ne
    \endtrivlist}

% Definition of \endtrivlist moved earlier in file so other commands
% can be \let = to it.

\def\@mklab#1{\hfil #1}

\def\item{\@ifnextchar [{\@item}{\@noitemargtrue \@item[\@itemlabel]}}

\def\@donoparitem{\@noparitemfalse
   \global\setbox\@labels\hbox{\hskip -\leftmargin
                               \unhbox\@labels
                                \hskip \leftmargin}\if@minipage\else
  \@tempskipa\lastskip
  \vskip -\lastskip \advance\@tempskipa\@outerparskip
  \advance\@tempskipa -\parskip \vskip\@tempskipa\fi}

\def\@item[#1]{\if@noparitem \@donoparitem
  \else \if@inlabel \indent \par \fi
        \ifhmode \unskip\unskip \par \fi
        \if@newlist \if@nobreak \@nbitem \else
                        \addpenalty\@beginparpenalty
```

```
                        \addvspace\@topsep \addvspace{-\parskip}\fi
           \else \addpenalty\@itempenalty \addvspace\itemsep
          \fi
    \global\@inlabeltrue
\fi
\everypar{\global\@minipagefalse\global\@newlistfalse
          \if@inlabel\global\@inlabelfalse \hskip -\parindent \box\@labels
             \penalty\z@ \fi
          \everypar{}}\global\@nobreakfalse
\if@noitemarg \@noitemargfalse \if@nmbrlist \refstepcounter{\@listctr}\fi \fi
\setbox\@tempboxa\hbox{\makelabel{#1}}%
\global\setbox\@labels
 \hbox{\unhbox\@labels \hskip \itemindent
       \hskip -\labelwidth \hskip -\labelsep
       \ifdim \wd\@tempboxa >\labelwidth
                \box\@tempboxa
%% RmS 91/11/22: Changed second call to \makelabel to \unhbox\@tempboxa.
%%               Avoids problems with side effects in \makelabel and is
%%               more efficient.
%          \else \hbox to\labelwidth {\makelabel{#1}}\fi
          \else \hbox to\labelwidth {\unhbox\@tempboxa}\fi
       \hskip \labelsep}\ignorespaces}

%% RmS 91/11/04: added default definition for \makelabel,
%%               to produce an error message.
\def\makelabel#1{\@latexerr{Lonely \string\item--perhaps a missing
        list environment}\@ehc}

\def\@nbitem{\@tempskipa\@outerparskip \advance\@tempskipa -\parskip
              \addvspace{\@tempskipa}}

\def\usecounter#1{\@nmbrlisttrue\def\@listctr{#1}\setcounter{#1}\z@}

 \message{itemize,}
%      ****************************************
%      *        ITEMIZE AND ENUMERATE         *
%      ****************************************
%
%  Enumeration is done with four counters: enumi, enumii, enumiii
%  and enumiv, where enumN controls the numbering of the Nth level
%  enumeration.  The label is generated by the commands
%  \labelenumi ... \labelenumiv, which should be defined by the
%  document style.  Note that \p@enumN\theenumN defines the output
%  of a \ref command.  A typical definition might be:
%     \def\theenumii{\alph{enumii}}
%     \def\p@enumii{\theenumi\theenumii}
%     \def\labelenumii{(\theenumii)}
% which will print the labels as '(a)', '(b)', ... and print a \ref as
% '3a'.
%
% The item numbers are moved to the right of the label box, so they are
% always a distance of \labelsep from the item.
%
% \@enumdepth holds the current enumeration nesting depth.
%
% Itemization is controlled by four commands: \labelitemi, \labelitemii,
% \labelitemiii, and \labelitemiv.  To cause the second-level list to be
% bulleted, you just define \labelitemii to be $\bullet$.  \@itemspacing
% and \@itemdepth are the analogs of \@enumspacing and \@enumdepth.
%
% \enumerate ==
```

```
%   BEGIN
%     if \@enumdepth > 3
%       then errormessage: ``Too deeply nested''.
%       else \@enumdepth :=L \@enumdepth + 1
%            \@enumctr :=L eval(enum@\romannumeral\the\@enumdepth)
%            \list{\label(\@enumctr)}
%                 {\usecounter{\@enumctr}
%                  \makelabel{LABEL} ==  \hss \llap{LABEL}}
%     fi
%   END
%
% \endenumerate == \endlist
%
\newcount\@enumdepth \@enumdepth = 0

\@definecounter{enumi}
\@definecounter{enumii}
\@definecounter{enumiii}
\@definecounter{enumiv}

\def\enumerate{\ifnum \@enumdepth >3 \@toodeep\else
      \advance\@enumdepth \@ne
      \edef\@enumctr{enum\romannumeral\the\@enumdepth}\list
      {\csname label\@enumctr\endcsname}{\usecounter
        {\@enumctr}\def\makelabel##1{\hss\llap{##1}}}\fi}

\let\endenumerate =\endlist


%  \itemize ==
%    BEGIN
%      if \@itemdepth > 3
%        then  errormessage: 'Too deeply nested'.
%        else  \@itemdepth :=L \@itemdepth + 1
%              \@itemitem  == eval(labelitem\romannumeral\the\@itemdepth)
%              \list{\@nameuse{\@itemitem}}
%                   {\makelabel{LABEL} ==  \hss \llap{LABEL}}
%      fi
%    END
%
%  \enditemize ==  \endlist
%
\newcount\@itemdepth \@itemdepth = 0

\def\itemize{\ifnum \@itemdepth >3 \@toodeep\else \advance\@itemdepth \@ne
\edef\@itemitem{labelitem\romannumeral\the\@itemdepth}%
\list{\csname\@itemitem\endcsname}{\def\makelabel##1{\hss\llap{##1}}}\fi}

\let\enditemize =\endlist

 \message{boxes,}
%       ********************************************
%       *                 BOXES                    *
%       ********************************************
%
%  USER COMMANDS:
%
%  \makebox [WID][POS]{OBJ}
%          : puts OBJ in an \hbox of width WID, positioned by POS.
%            POS = l -> flushleft, POS = r -> flushright.
%            Default is centered.
```

```
%           If WID is missing, then POS is also missing and OBJ
%           is put in an \hbox of its natural width.
%
%  \mbox{OBJ} == \makebox{OBJ}, and is more efficient.
%
%  \makebox (X,Y)[POS]{OBJ}
%         : puts OBJ in an \hbox of width X * \unitlength
%           and height Y * \unitlength.  POS arguments are
%           l or r for flushleft, flushright and  t or b
%           for top, bottom -- or combinations like  tr or rb.
%           Default for horizontal and vertical are centered.
%
%  \newsavebox{\CMD} : If \CMD is undefined, then defines it
%          to be a TeX box register.
%
%  \savebox {\CMD} ... : \CMD is defined to be a TeX box register,
%           and the '...' are any \makebox arguments.  It is
%           like \makebox, except it doesn't produce text but
%           saves the value in \box \CMD.
%           \sbox N{OBJ} is an efficient abbreviation for
%           \savebox N{OBJ}.
%
%  \framebox ...  : like \makebox, except it puts a 'frame' around
%           the box.  The frame is made of lines of thickness
%           \fboxrule, separated by space \fboxsep from the
%           text -- except for \framebox(X,Y) ... , where the
%           thickness of the lines is as for the picture environment,
%           and there is no separation added.
%           \fbox{OBJ} is an efficient abbreviation for \framebox{OBJ}
%
%  \parbox[POS]{WIDTH}{TEXT} : Makes a box with \hsize TEXT, positioned
%        by POS as follows:
%             c : \vcenter (placed in $...$ if not in math mode)
%             b : \vbox
%             t : \vtop
%        default value is c.
%    Sets \hsize := WIDTH and calls \@parboxrestore, which does
%    the following:
%        Restores the original definitions of:
%             \par
%             \\
%             \- \' \` \=
%        Resets the following parameters:
%             \parindent        = 0pt
%             \parskip          = 0pt           %% added 20 Jan 87
%             \linewidth        = \hsize
%             \@totalleftmargin = 0pt
%             \leftskip         = 0pt
%             \rightskip        = 0pt
%             \@rightskip       = 0pt
%             \parfillskip      = 0pt plus 1fil
%             \lineskip         = \normallineskip
%             \baselineskip     = \normalbaselineskip
%        Calls \sloppy
%
%  Note: \@arrayparboxrestore same as \@parboxrestore
%        but it doesn't restore \\.
%
% \minipage  :  Similar to parbox, except it also
%       makes this look like a page by setting
%             \textwidth == \columnwidth == box width
```

```
%       changes footnotes by redefining:
%             \@mpfn         == mpfootnote
%             \thempfn       == \thempfootnote
%             \@footnotetext == \@mpfootnotetext
%       resets the following list environment parameters
%             \@listdepth    == \@mplistdepth
%       where  \@mplistdepth is initialized to zero,
%       and executes \@minipagerestore to allow the document
%       style to reset any other parameters it desires.
%       It sets @minipage := T, and resets \everypar to set
%       it false.  This switch keeps \addvspace from putting space
%       at the top of a minipage.
%
%       Change added 24 May 89: \minipage sets @minipage globally;
%       \endminipage resets it false.
%
%
% \rule [RAISED]{WIDTH}{HEIGHT} : Makes a WIDTH X HEIGHT rule, raised
%       RAISED.
%
% \underline {TEXT} : Makes an underlined hbox with TEXT in it.
%
% \raisebox{DISTANCE}[HEIGHT][DEPTH]{BOX} : Raises BOX up by DISTANCE
%       length (down if DISTANCE negative).  Makes TeX think that
%       the new box extends HEIGHT above the line and DEPTH below, for
%       a total vertical length of HEIGHT+DEPTH.  Default values of
%       HEIGHT & DEPTH = actual height and depth of box in new position.
%
% \makebox ==
%  BEGIN
%    if next char = (
%      then  \@makepicbox
%      else  if  next char = [
%              then  \@makebox
%              else  \mbox     fi
%    fi
%  END
%
% \@makebox[LEN] ==
% BEGIN
%   leave vertical mode
%   if next char '[' then \@imakebox[LEN]
%                    else \@imakebox[LEN][x]  fi
% END
%
% \@imakebox[LEN][POS]{OBJ} ==
%  BEGIN
%    \hbox to LEN
%      { \mb@l :=L \mb@r :=L \hss
%        \let\mb@POS = \relax
%       \mb@l OBJ \mb@r }
%  END
%
% \@makepicbox(X,Y) ==
%  BEGIN
%    leave vertical mode
%    if next char = [  then  \@imakepicbox(X,Y)
%                      else  \@imakepicbox(X,Y)[]  fi
%  END
%
% \@imakepicbox(X,Y)[POS]{OBJ} ==
```

```
%  BEGIN
%    \vbox to Y * \unitlength
%       { \mb@l :=L \mb@r :=L \hss
%         \mb@t :=L \mb@b :=L \hss
%         tfor \@tempa := POS    % one iteration for each token in POS
%           do  \mb@eval(\@tempa) :=L null od
%         \mb@t
%         \hbox to X * \unitlength
%           {\mb@l OBJ \mb@r }
%        \mb@b}
%  END
%

\def\makebox{\@ifnextchar ({\@makepicbox}{\@ifnextchar
     [{\@makebox}{\mbox}}}

\def\mbox#1{\leavevmode\hbox{#1}}

\def\@makebox[#1]{\leavevmode\@ifnextchar [{\@imakebox[#1]}{\@imakebox[#1][x]}}

\long\def\@imakebox[#1][#2]#3{\hbox to#1{\let\mb@l\hss
\let\mb@r\hss \expandafter\let\csname mb@#2\endcsname\relax
\mb@l #3\mb@r}}

\def\@makepicbox(#1,#2){\leavevmode\@ifnextchar
   [{\@imakepicbox(#1,#2)}{\@imakepicbox(#1,#2)[]}}

\long\def\@imakepicbox(#1,#2)[#3]#4{\vbox to#2\unitlength
   {\let\mb@b\vss \let\mb@l\hss\let\mb@r\hss
    \let\mb@t\vss
    \@tfor\@tempa :=#3\do{\expandafter\let
        \csname mb@\@tempa\endcsname\relax}%
\mb@t\hbox to #1\unitlength{\mb@l #4\mb@r}\mb@b}}

\def\newsavebox#1{\@ifdefinable#1{\newbox#1}}

\def\savebox#1{\@ifnextchar ({\@savepicbox#1}{\@ifnextchar
     [{\@savebox#1}{\sbox#1}}}

\def\sbox#1#2{\setbox#1\hbox{#2}}

\def\@savebox#1[#2]{\@ifnextchar [{\@isavebox#1[#2]}{\@isavebox#1[#2][x]}}

\long\def\@isavebox#1[#2][#3]#4{\setbox#1 \hbox{\@imakebox[#2][#3]{#4}}}

\def\@savepicbox#1(#2,#3){\@ifnextchar
   [{\@isavepicbox#1(#2,#3)}{\@isavepicbox#1(#2,#3)[]}}

\long\def\@isavepicbox#1(#2,#3)[#4]#5{\setbox#1 \hbox{\@imakepicbox
     (#2,#3)[#4]{#5}}}

\def\usebox#1{\leavevmode\copy #1\relax}

%% The following definition of \frame was written by Pavel Curtis
%% (Extra space removed 14 Jan 88)
\long\def\frame#1{\leavevmode
    \hbox{\hskip-\@wholewidth
     \vbox{\vskip-\@wholewidth
           \hrule \@height\@wholewidth
          \hbox{\vrule \@width\@wholewidth #1\vrule \@width\@wholewidth}\hrule
           \@height \@wholewidth\vskip -\@halfwidth}\hskip-\@wholewidth}}
```

```
\newdimen\fboxrule
\newdimen\fboxsep

%% (Extra space removed 21 Jun 1991)
\long\def\fbox#1{\leavevmode\setbox\@tempboxa\hbox{#1}\@tempdima\fboxrule
    \advance\@tempdima \fboxsep \advance\@tempdima \dp\@tempboxa
   \hbox{\lower \@tempdima\hbox
  {\vbox{\hrule \@height \fboxrule
          \hbox{\vrule \@width \fboxrule \hskip\fboxsep
          \vbox{\vskip\fboxsep \box\@tempboxa\vskip\fboxsep}\hskip
                 \fboxsep\vrule \@width \fboxrule}%
                 \hrule \@height \fboxrule}}}}

\def\framebox{\@ifnextchar ({\@framepicbox}{\@ifnextchar
     [{\@framebox}{\fbox}}}

\def\@framebox[#1]{\@ifnextchar [{\@iframebox[#1]}{\@iframebox[#1][x]}}

%% (Extra space removed 21 Jun 1991)
\long\def\@iframebox[#1][#2]#3{\leavevmode
  \savebox\@tempboxa[#1][#2]{\kern\fboxsep #3\kern\fboxsep}\@tempdima\fboxrule
    \advance\@tempdima \fboxsep \advance\@tempdima \dp\@tempboxa
   \hbox{\lower \@tempdima\hbox
  {\vbox{\hrule \@height \fboxrule
          \hbox{\vrule \@width \fboxrule \hskip-\fboxrule
              \vbox{\vskip\fboxsep \box\@tempboxa\vskip\fboxsep}\hskip
                  -\fboxrule\vrule \@width \fboxrule}%
                  \hrule \@height \fboxrule}}}}

\def\@framepicbox(#1,#2){\@ifnextchar
   [{\@iframepicbox(#1,#2)}{\@iframepicbox(#1,#2)[]}}

\long\def\@iframepicbox(#1,#2)[#3]#4{\frame{\@imakepicbox(#1,#2)[#3]{#4}}}

\def\parbox{\@ifnextchar [{\@iparbox}{\@iparbox[c]}}

\long\def\@iparbox[#1]#2#3{\leavevmode \@pboxswfalse
   \if #1b\vbox
     \else \if #1t\vtop
              \else \ifmmode \vcenter
                        \else \@pboxswtrue $\vcenter
                     \fi
           \fi
%% RmS 91/11/04 added \m@th
    \fi{\hsize #2\@parboxrestore #3}\if@pboxsw \m@th$\fi}

\let\@dischyph=\-
\let\@acci=\'
\let\@accii=\`
\let\@acciii=\=


\def\@arrayparboxrestore{\let\par\@@par
    \let\-\@dischyph
    \let\'\@acci \let\`\@accii \let\=\@acciii
    \parindent\z@ \parskip\z@
    \everypar{}\linewidth\hsize
    \@totalleftmargin\z@ \leftskip\z@ \rightskip\z@ \@rightskip\z@
    \parfillskip\@flushglue \lineskip\normallineskip
    \baselineskip\normalbaselineskip\sloppy}
```

```
4027
4028 \def\@parboxrestore{\@arrayparboxrestore\let\\=\@normalcr}
4029
4030 \newif\if@minipage \@minipagefalse
4031
4032 \def\minipage{\@ifnextchar [{\@iminipage}{\@iminipage[c]}}
4033
4034 \def\@iminipage[#1]#2{\leavevmode \@pboxswfalse
4035    \if #1b\vbox
4036      \else \if #1t\vtop
4037               \else \ifmmode \vcenter
4038                        \else \@pboxswtrue $\vcenter
4039                     \fi
4040            \fi
4041     \fi\bgroup
4042     \hsize #2\textwidth\hsize \columnwidth\hsize
4043     \@parboxrestore
4044     \def\@mpfn{mpfootnote}\def\thempfn{\thempfootnote}\c@mpfootnote\z@
4045     \let\@footnotetext\@mpfootnotetext
4046     \let\@listdepth\@mplistdepth \@mplistdepth\z@
4047     \@minipagerestore\global\@minipagetrue %% \global added 24 May 89
4048     \everypar{\global\@minipagefalse\everypar{}}}
4049
4050
4051 \let\@minipagerestore=\relax
4052
4053 \def\endminipage{\par\vskip-\lastskip
4054 \ifvoid\@mpfootins\else
4055 \vskip\skip\@mpfootins\footnoterule\unvbox\@mpfootins\fi
4056 \global\@minipagefalse   %% added 24 May 89
4057 \egroup\if@pboxsw \m@th$\fi} %% RmS 91/11/04 added \m@th
4058
4059 \newcount\@mplistdepth
4060 \newinsert\@mpfootins
4061
4062 %% RmS 91/09/29: added \reset@font
4063 \long\def\@mpfootnotetext#1{\global\setbox\@mpfootins
4064     \vbox{\unvbox\@mpfootins
4065     \reset@font\footnotesize
4066     \hsize\columnwidth \@parboxrestore
4067    \edef\@currentlabel{\csname p@mpfootnote\endcsname\@thefnmark}\@makefntext
4068      {\rule{\z@}{\footnotesep}\ignorespaces #1\strut}}}
4069      % \strut added 27 Mar 89, on suggestion by Don Hosek
4070
4071 \newif\if@pboxsw
4072
4073 \def\rule{\@ifnextchar[{\@rule}{\@rule[\z@]}}
4074
4075 \def\@rule[#1]#2#3{\@tempdima#3\advance\@tempdima #1\leavevmode\hbox{\vrule
4076   \@width#2 \@height\@tempdima \@depth-#1}}
4077
4078 \let\@@underline\underline
4079 \def\underline#1{\relax\ifmmode
4080   \@@underline{#1}\else $\@@underline{\hbox{#1}}\m@th$\relax\fi}
4081
4082 \def\raisebox#1{\@ifnextchar[{\@argrsbox{#1}}{\@rsbox{#1}}}
4083
4084 \def\@argrsbox#1[#2]{%
4085 \@ifnextchar[{\@iirsbox{#1}[#2]}{\@irsbox{#1}[#2]}}
4086
4087 \long\def\@rsbox#1#2{\leavevmode\hbox{\raise #1\hbox{#2}}}
```

```
4088
4089 \long\def\@irsbox#1[#2]#3{\setbox\@tempboxa \hbox
4090    {\raise #1\hbox{#3}}\ht\@tempboxa#2\leavevmode\box\@tempboxa}
4091
4092 \long\def\@iirsbox#1[#2][#3]#4{\setbox\@tempboxa \hbox
4093    {\raise #1\hbox{#4}}\ht\@tempboxa#2\dp\@tempboxa#3\leavevmode\box\@tempboxa}
4094
4095
4096  \message{tabbing,}
4097 %       ****************************************
4098 %       *       THE TABBING ENVIRONMENT        *
4099 %       ****************************************
4100 %
4101 %  \dimen(\@firsttab + i) = distance of tab stop i from left margin
4102 %         0 <= i <= 15 (?).
4103 %
4104 %  \dimen\@firsttab is initialized to \@totalleftmargin, so it starts
4105 %      at the prevailing left margin.
4106 %
4107 %  \@maxtab          = number of highest defined tab register
4108 %                        probably = \@firsttab + 12
4109 %  \@nxttabmar = tab stop number of next line's left margin
4110 %  \@curtabmar = tab stop number of current line's left margin
4111 %  \@curtab    = number of the current tab. At start of line,
4112 %                      it equals \@curtabmar
4113 %  \@hightab   = largest tab number currently defined.
4114 %  \@tabpush   = depth of \pushtab's
4115 %
4116 %  \box\@curline     = contents of current line, excluding left margin skip,
4117 %                      and excluding contents of current field
4118 %  \box\@curfield    = contents of current field
4119 %
4120 %  @rjfield          = switch: T iff the last field of the line should be
4121 %                      right-justified at the right margin.
4122 %
4123 %  \tabbingsep          = distance left by the \' command between the current
4124 %                      position and the field that is ``left-shifted''.
4125 %
4126 %  UTILITY MACROS
4127 %   \@stopfield  : closes the current field
4128 %   \@addfield   : adds the current field to the current line.
4129 %   \@contfield  : continues the current field
4130 %   \@startfield : begins the next field
4131 %   \@stopline   : closes the current line and outputs it
4132 %   \@startline  : starts the next line
4133 %   \@ifatmargin : an \if that is true iff the current line.
4134 %                  has width zero
4135 %
4136 % \@startline ==
4137 %  BEGIN
4138 %   \@curtabmar :=G \@nxttabmar
4139 %   \@curtab :=G \@curtabmar
4140 %   \box\@curline :=G null
4141 %   \@startfield
4142 %   \strut
4143 %  END
4144 %
4145 % \@stopline ==
4146 %  BEGIN
4147 %   \unskip
4148 %   \@stopfield
```

```
%   if @rjfield = T
%     then  @rjfield :=G F
%           \@tempdima := \@totalleftmargin + \linewidth
%           \hbox to \@tempdima{\@itemfudge
%                               \hskip \dimen\@curtabmar
%                               \box\@curline
%                               \hfil
%                               \box\@curfield}
%     else \@addfield
%          \hbox {\@itemfudge
%                 \hskip \dimen\@curtabmar
%                 \box\@curline}
%   fi
%  END
%
% \@startfield ==
%  BEGIN
%    \box\@curfield :=G \hbox {
%  END
%
% \@stopfield ==
%  BEGIN
%     }
%  END
%
% \@contfield ==
%  BEGIN
%   \box\@curfield :=G \hbox { \unhbox\@currfield  %%} brace matching
%  END
% \@addfield ==
%  BEGIN
%   \box\@curline :=G \unbox\@curline * \unbox\@curfield
%  END
%
% \@ifatmargin ==
%  BEGIN
%   if  dim of box\@curline = 0pt  then
%  END
%
%
% \tabbing ==
%  BEGIN
%   \lineskip :=L 0pt
%   \> == \@rtab
%   \< == \@ltab
%   \= == \@settab
%   \+ == \@tabplus
%   \- == \@tabminus
%   \‘ == \@tabrj
%   \’ == \@tablab
%   \\ == BEGIN \@stopline \@startline END
%   \\[DIST] == BEGIN \@stopline \vskip DIST \@startline\ignorespaces END
%   \\* == BEGIN \@stopline \penalty 10000 \@startline END
%   \\*[DIST] == BEGIN \@stopline \penalty 10000 \vskip DIST
%                      \@startline\ignorespaces              END
%   \@hightab :=G \@nxttabmar :=G \@firsttab
%   \@tabpush :=G 0
%   \dimen\@firsttab := \@totalleftmargin
%   @rjfield :=G F
%   \trivlist  \item[]
%   if @minipage = F then \vskip \parskip fi
```

```
%   \box\@tabfbox = \rlap{\indent\the\everypar}  % note: \the\everypar sets
%   \@itemfudge == BEGIN \box\@tabfbox END       %       @inlabel :=G F
%   \@startline
%   \ignorespaces
%  END
%
% \@endtabbing ==
%  BEGIN
%   \@stopline
%   if \@tabpush > 0 then error message: ''unmatched \poptabs'' fi
%   \endtrivlist
%  END
%
% \@rtab ==
%  BEGIN
%   \@stopfield
%   \@addfield
%   if \@curtab < \@hightab
%     then \@curtab :=G \@curtab + 1
%     else error message ``Undefined Tab''   fi
%   \@tempdima := \dimen\@curtab - \dimen\@curtabmar
%                         - width of box \@curline
%   \box\@curline :=G \hbox{\unhbox\@curline + \hskip\@tempdima}
%   \@startfield
%  END
%
% \@settab ==
%  BEGIN
%   \@stopfield
%   \@addfield
%   if \@curtab < \@maxtab
%     then \@curtab :=G \@curtab+1
%     else error message: ``Too many tabs''    fi
%   if \@curtab > \@hightab
%     then \@hightab :=L \@curtab    fi
%   \dimen\@curtab :=L \dimen\@curtabmar + width of \box\@curline
%   \@startfield
%  END
%
% \@ltab ==
%  BEGIN
%   \@ifatmargin
%     then if \@curtabmar > \@firsttab
%            then \@curtab :=G \@curtab - 1
%                 \@curtabmar :=G \@curtabmar - 1
%            else error message ``Too many untabs''      fi
%     else error message ``Left tab in middle of line''
%    fi
%  END
%
% \@tabplus ==
%  BEGIN
%        if  \@nxttabmar < \@hightab
%           then \@nxttabmar :=G \@nxttabmar+1
%           else error message ``Undefined tab''
%        fi
%  END
%
% \@tabminus ==
%  BEGIN
%        if \@nxttabmar > \@firsttab
```

```
%           then \@nxttabmar :=G \@nxttabmar-1
%           else error message ``Too many untabs''
%        fi
%  END
%
% \@tabrj ==
%  BEGIN \@stopfield
%        \@addfield
%        @rjfield :=G T
%        \@startfield
%  END
%
% \@tablab ==
%  BEGIN \@stopfield
%        \box\@curline G:= \hbox{ \box\@curline %% `G' added 17 Jun 86
%                                 \hskip - width of \box\@curfield
%                                 \hskip -\tabbingsep
%                                 \box\@curfield
%                                 \hskip \tabbingsep }
%        \@startfield
%  END
%
% \pushtabs ==
%   BEGIN
%     \@stopfield
%     \@tabpush :=G \@tabpush + 1
%     \begingroup
%     \@contfield
%   END
%
% \poptabs ==
%  BEGIN
%    \@stopfield
%    if \@tabpush > 0
%      then \endgroup
%           \@tabpush :=G \@tabpush - 1
%      else error message: ``Too many \poptabs''
%    fi
%    \@contfield
%  END
%
% The accents \` , \' , and \= that have been redefined inside a tabbing
% environment can be called by typing \a` , \a' , and \a=.
%

\expandafter \let \csname a`\endcsname = \`
\expandafter \let \csname a'\endcsname = \'
\expandafter \let \csname a=\endcsname = \=
\def\a#1{\csname a#1\endcsname}

\newif\if@rjfield
\newcount\@firsttab
\newcount\@maxtab
\newdimen\@gtempa \@firsttab=\allocationnumber
\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa
\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa
\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa\newdimen\@gtempa
\newdimen\@gtempa \@maxtab=\allocationnumber
\dimen\@firsttab=0pt
\newcount\@nxttabmar
\newcount\@curtabmar
```

```
\newcount\@curtab
\newcount\@hightab
\newcount\@tabpush
\newbox\@curline
\newbox\@curfield
\newbox\@tabfbox

\def\@startline{\global\@curtabmar\@nxttabmar
   \global\@curtab\@curtabmar\global\setbox\@curline\hbox % missing \global
    {}\@startfield\strut}                                 % added 17 Jun 86

\def\@stopline{\unskip\@stopfield\if@rjfield \global\@rjfieldfalse
   \@tempdima\@totalleftmargin \advance\@tempdima\linewidth
\hbox to\@tempdima{\@itemfudge\hskip\dimen\@curtabmar
   \box\@curline\hfil\box\@curfield}\else\@addfield
   \hbox{\@itemfudge\hskip\dimen\@curtabmar\box\@curline}\fi}

\def\@startfield{\global\setbox\@curfield\hbox\bgroup}%{ BRACE MATCH HACK
\let\@stopfield=}
\def\@contfield{\global\setbox\@curfield\hbox\bgroup\unhbox\@curfield}
\def\@addfield{\global\setbox\@curline\hbox{\unhbox
     \@curline\unhbox\@curfield}}
\def\@ifatmargin{\ifdim \wd\@curline =\z@}

\def\@tabcr{\@stopline \@ifstar{\penalty \@M \@xtabcr}{\@xtabcr}}

\def\@xtabcr{\@ifnextchar[{\@itabcr}{\@startline\ignorespaces}}

\def\@itabcr[#1]{\vskip #1\@startline\ignorespaces}

\def\kill{\@stopfield\@startline\ignorespaces}

% REMOVE \outer FROM PLAIN'S DEF OF \+

\def\+{\tabalign}


\def\tabbing{\lineskip \z@\let\>\@rtab\let\<\@ltab\let\=\@settab
     \let\+\@tabplus\let\-\@tabminus\let\`\@tabrj\let\'\@tablab
     \let\\=\@tabcr
     \global\@hightab\@firsttab
     \global\@nxttabmar\@firsttab
     \dimen\@firsttab\@totalleftmargin
     \global\@tabpush\z@ \global\@rjfieldfalse
     \trivlist \item[]\if@minipage\else\vskip\parskip\fi
     \setbox\@tabfbox\hbox{\rlap{\indent\hskip\@totalleftmargin
       \the\everypar}}\def\@itemfudge{\box\@tabfbox}\@startline\ignorespaces}

\def\endtabbing{\@stopline\ifnum\@tabpush >\z@ \@badpoptabs \fi\endtrivlist}

\def\@rtab{\@stopfield\@addfield\ifnum \@curtab<\@hightab
      \global\advance\@curtab \@ne \else\@badtab\fi
      \@tempdima\dimen\@curtab
      \advance\@tempdima -\dimen\@curtabmar
      \advance\@tempdima -\wd\@curline
      \global\setbox\@curline\hbox{\unhbox\@curline\hskip\@tempdima}%
      \@startfield\ignorespaces}
% Omitted \global added to \@rtab 17 Jun 86

\def\@settab{\@stopfield\@addfield\ifnum \@curtab <\@maxtab
      \global\advance\@curtab \@ne \else\@latexerr{Tab overflow}\@ehd\fi
```

```
      \ifnum\@curtab >\@hightab
      \@hightab\@curtab\fi
      \dimen\@curtab\dimen\@curtabmar
      \advance\dimen\@curtab \wd\@curline\@startfield\ignorespaces}
\def\@ltab{\@ifatmargin\ifnum\@curtabmar >\@firsttab
      \global\advance\@curtab \m@ne \global\advance\@curtabmar \m@ne \else
      \@badtab\fi\else
      \@latexerr{\string\<\space in mid line}\@ehd\fi\ignorespaces}
\def\@tabplus {\ifnum \@nxttabmar <\@hightab
      \global\advance\@nxttabmar \@ne \else
      \@badtab\fi\ignorespaces}
\def\@tabminus{\ifnum\@nxttabmar >\@firsttab
      \global\advance\@nxttabmar \m@ne \else
      \@badtab\fi\ignorespaces}
\def\@tabrj{\@stopfield\@addfield\global\@rjfieldtrue\@startfield\ignorespaces}

\def\@tablab{\@stopfield\global\setbox\@curline\hbox{\box\@curline
     \hskip -\wd\@curfield \hskip -\tabbingsep \box\@curfield
      \hskip \tabbingsep}\@startfield\ignorespaces}
% \setbox\@curline made \global in \@tablab. 17 Jun 86

\def\pushtabs{\@stopfield\@addfield\global\advance\@tabpush \@ne \begingroup
       \@contfield}
\def\poptabs{\@stopfield\@addfield\ifnum\@tabpush >\z@ \endgroup
     \global\advance\@tabpush \m@ne \else
     \@badpoptabs\fi\@contfield}

\newdimen\tabbingsep

 \message{array,}
%      ****************************************
%      *    ARRAY AND TABULAR ENVIRONMENTS    *
%      ****************************************
%
% ARRAY PARMETERS:
%  \arraycolsep    : half the width separating columns in an array environment
%  \tabcolsep      : half the width separating columns in a tabular environment
%  \arrayrulewidth : width of rules
%  \doublerulesep  : space between adjacent rules in array or tabular
%  \arraystretch   : line spacing in array and tabular environments is done by
%                    placing a strut in every row of height and depth
%                    \arraystretch times the height and depth of the strut
%                    produced by an ordinary \strut commmand.
%
% PREAMBLE:
%  The PREAMBLE argument of an array or tabular environment can contain
%  the following:
%    l,r,c  : indicate where entry is to be placed.
%    |      : for vertical rule
%    @{EXP} : inserts the text EXP in every column.  \arraycolsep or \tabcolsep
%             spacing is suppressed.
%    *{N}{PRE} : equivalent to writing N copies of PRE in the preamble.  PRE
%                may contain *{N'}{EXP'} expressions.
%    p{LEN} : makes entry in parbox of width LEN.
%
% SPECIAL ARRAY COMMANDS:
%   \multicolumn{N}{FORMAT}{ITEM} : replaces the next N column items by
%       ITEM, formatted according to FORMAT.  FORMAT should contain at most
%       one l,r or c.  If it contains none, then ITEM is ignored.
%
%   \vline : draws a vertical line the height of the current row.  May
```

```
%           appear in an array element entry.
%  \hline : draws a horizontal line between rows.  Must appear either
%           before the first entry (to appear above the first row) or right
%           after a \\ command.  If followed by another \hline, then adds
%           a \vskip of \doublerulesep.
%
%  \cline[i-j] : draws horizontal lines between rows covering columns
%                i through j, inclusive.  Multiple commands may follow
%                one another to provide lines covering several disjoint
%                columns
%  \extracolsep{WIDTH} : for use inside an @ in the preamble.  Causes a WIDTH
%                space to be added between columns for the rest of the
%                columns.  This is in addition to the ordinary intercolumn
%                space.
%
% \array ==
%   BEGIN
%     \@acol    == \@arrayacol
%     \@classz  == \@arrayclassz
%     \@classiv == \@arrayclassiv
%     \\        == \@arraycr
%     \@halignto == NULL
%     \@tabarray
%   END
%
% \endarray{NAME} ==  BEGIN  \crcr }}  END
%
% \tabular  ==
%   BEGIN
%     \@halignto == NULL
%     \@tabular
%   END
%
% \tabular*{WIDTH} ==
%   BEGIN
%     \@halignto == to WIDTH
%     \@tabular
%   END
%
% \@tabular ==
%   BEGIN
%     \leavevmode
%     \hbox { $
%        \@acol    == \@tabacol
%        \@classz  == \@tabclassz
%        \@classiv == \@tabclassiv
%        \\        == \@tabularcr
%        \@tabarray
%   END
%
% \endtabular == BEGIN \crcr}} $} END
%
% \@tabarray == if next char = [ then \@array else \@array[c] fi
%
% \@array[POS]{PREAMBLE} ==
%   BEGIN
%     define \@arstrutbox to make \@arstrut produce strut of height
%       and depth \arraystretch times the height and
%       depth of a normal strut.
%     \@mkpream{PREAMBLE}
%     \@preamble == \halign \@halignto {\tabskip=0pt\@arstrut
```

```
%                              eval{\@preamble}\tabskip = 0pt\cr  %% }
%      \@startpbox == \@@startpbox
%      \@endpbox == \@@endpbox
%      if POS = t then \vtop
%                 else if POS = b then \vbox
%                                 else \vcenter
%      fi              fi
%     {
%      \par          ==L \relax
%      \@sharp       == #
%      \protect      == \relax
%      \lineskip     :=L 0pt
%      \baselineskip :=L 0pt
%      \@preamble
%    END
%
%
%  \@arraycr ==
%   BEGIN
%     $                    %% Prevents extra space at end of row's last entry.
%     if next char = [
%      then  \@argarraycr
%      else  $ \cr          %% Needed to balance $
%   END
%
%  \@argarraycr[LENGTH] ==
%   BEGIN
%     $                    %% Needed to balance $ of \@arraycr
%     if LENGTH > 0
%       then  \@tempdima := depth of \@arstrutbox + LENGTH
%             \vrule height 0pt width 0pt depth \@tempdima
%             \cr
%       else  \cr \noalign{\vskip LENGTH}
%   END
%
%  \@tabularcr and \@argtabularcr  same as \@arraycr and \@argarraycr
%  except without the extra $'s.

\def\extracolsep#1{\tabskip #1\relax}

\def\array{\let\@acol\@arrayacol \let\@classz\@arrayclassz
 \let\@classiv\@arrayclassiv \let\\\@arraycr\let\@halignto\@empty\@tabarray}

\def\endarray{\crcr\egroup\egroup}
\def\endtabular{\crcr\egroup\egroup $\egroup}
\expandafter \let \csname endtabular*\endcsname = \endtabular

\def\tabular{\let\@halignto\@empty\@tabular}

\expandafter \def\csname tabular*\endcsname #1{\def\@halignto{to#1}\@tabular}

\def\@tabular{\leavevmode \hbox \bgroup $\let\@acol\@tabacol
   \let\@classz\@tabclassz
   \let\@classiv\@tabclassiv \let\\\@tabularcr\@tabarray}

%% RmS 91/11/04 added \m@th
\def\@tabarray{\m@th\@ifnextchar[{\@array}{\@array[c]}}

\def\@array[#1]#2{\setbox\@arstrutbox\hbox{\vrule
     \@height\arraystretch \ht\strutbox
     \@depth\arraystretch \dp\strutbox
```

```
     \@width\z@}\@mkpream{#2}\edef\@preamble{\halign \noexpand\@halignto
\bgroup \tabskip\z@ \@arstrut \@preamble \tabskip\z@ \cr}%
\let\@startpbox\@@startpbox \let\@endpbox\@@endpbox
  \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi\fi
  \bgroup \let\par\relax
  \let\@sharp##\let\protect\relax \lineskip\z@\baselineskip\z@\@preamble}

\def\@arraycr{${\ifnum0=`}\fi\@ifstar{\@xarraycr}{\@xarraycr}}
\def\@xarraycr{\@ifnextchar[{\@argarraycr}{\ifnum0=`{\fi}${}\cr}}

\def\@argarraycr[#1]{\ifnum0=`{\fi}${}\ifdim #1>\z@ \@xargarraycr{#1}\else
   \@yargarraycr{#1}\fi}

\def\@tabularcr{{\ifnum0=`}\fi\@ifstar{\@xtabularcr}{\@xtabularcr}}
\def\@xtabularcr{\@ifnextchar[{\@argtabularcr}{\ifnum0=`{\fi}\cr}}

\def\@argtabularcr[#1]{\ifnum0=`{\fi}\ifdim #1>\z@
   \unskip\@xargarraycr{#1}\else \@yargarraycr{#1}\fi}

\def\@xargarraycr#1{\@tempdima #1\advance\@tempdima \dp \@arstrutbox
   \vrule \@height\z@ \@depth\@tempdima \@width\z@ \cr}

\def\@yargarraycr#1{\cr\noalign{\vskip #1}}


% \multicolumn{NUMBER}{FORMAT}{ITEM} ==
%  BEGIN
%  \multispan{NUMBER}
%  \begingroup
%  \@addamp == null
%  \@mkpream{FORMAT}
%  \@sharp == ITEM
%  \protect == \relax
%  \@startpbox == \@@startpbox
%  \@endpbox == \@@endpbox
%  \@arstrut
%  \@preamble
%  \endgroup
%  END

% The command \def\@addamp{} was removed from \multicolumn on 6 Dec 86
% because it caused embedded array environments not to work.  I think
% that it was included originally to prevent an error message if
% the 2nd argument to the \multicolumn command had two column specifiers.
%
% 8 Feb 89 - \hbox{} added after \@preamble to correct bug that
%            occurred if \multicolumn preceded \\[D] with D > 0,
%            caused by \\[] command doing an \unskip, which removed
%            \tabcolsep glue inserted by \multicolumn

\def\multicolumn#1#2#3{\multispan{#1}\begingroup
\@mkpream{#2}%
\def\@sharp{#3}\let\protect\relax
  \let\@startpbox\@@startpbox\let\@endpbox\@@endpbox
  \@arstrut \@preamble\hbox{}\endgroup\ignorespaces}


% Codes for classes and character numbers of array, tabular and
% multicolumn arguments.
%
%    Character     Class       Number
```

```
%    ---------     -----       ------
%        c           0           0
%        l           0           1
%        r           0           2
%
%        |           1           -
%        @           2           -
%        p           3           -
%      {@-exp}       4           -
%      {p-arg}       5           -
%
% \@testpach \foo : expands \foo, which should be an array parameter token,
%                   and sets \@chclass and \@chnum to its class and number.
%                   Uses \@lastchclass to distinguish 4 and 5
%
% Preamble error codes
%    0: 'illegal character'
%    1: 'Missing @-exp'
%    2: 'Missing p-arg'
%
% \@addamp ==
%   BEGIN if @firstamp = true then @firstamp := false
%                              else &                     fi
%   END
%
% \@mkpream TOKENLIST ==
%   BEGIN
%    @firstamp     := T
%    \@lastchclass := 6
%    \@preamble    == null
%    \@sharp       == \relax
%    \protect      == BEGIN \noexpand\protect\noexpand END
%    \@startpbox   == \relax
%    \@endpbox     == \relax
%    \@expast{TOKENLIST}
%    for \@nextchar := expand(\@tempa)
%      do  \@testpach{\@nextchar}
%          case of \@chclass
%            0 -> \@classz
%            1 -> \@classi
%              ...
%            5 -> \@classv
%          end case
%          \@lastchclass := \@chclass
%      od
%      case of \@lastchclass
%         0 -> \hskip \arraycolsep             % lrc
%         1 ->                                  % |
%         2 -> \@preamerr1 % 'Missing @-exp'    % @
%         3 -> \@preamerr2 % 'Missing p-arg'    % p
%         4 ->                                  % @-exp
%         5 -> \hskip \arraycolsep             % p-exp
%      end case
%   END
%
%  \@arrayclassz ==
%    BEGIN
%      \@preamble := \@preamble *
%                    case of \@lastchclass
%                       0 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
%                       1 -> \@addamp \hskip \arraycolsep
```

```
%                       2 ->  % impossible
%                       3 ->  % impossible
%                       4 -> \@addamp
%                       5 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
%                       6 -> \@addamp \hskip \arraycolsep
%                     end case
%                   * case of \@chnum
%                       0 -> \hfil$\relax\@sharp$\hfil
%                       1 -> $\relax\@sharp$\hfil
%                       2 -> \hfil$\relax\@sharp$
%                     end case
%    END
%
% \@tabclassz == similar to \@arrayclassz
%
% \@classi ==
%  BEGIN
%    \@preamble := \@preamble *
%                  case of \@lastchclass
%                     0 -> \hskip \arraycolsep \@arrayrule
%                     1 -> \hskip \doublerulesep \@arrayrule
%                     2 -> % impossible
%                     3 -> % impossible
%                     4 -> \@arrayrule
%                     5 -> \hskip \arraycolsep \@arrayrule
%                     6 -> \@arrayrule
%                  end case
%  END
%
% \@classii ==
%  BEGIN
%    \@preamble := \@preamble *
%                  case of \@lastchclass
%                     0    ->
%                     1    -> \hskip .5\arrayrulewidth
%                     2    -> % impossible
%                     else ->
%                  end case
%  END
%
% \@classiii ==
%  BEGIN
%    \@preamble := \@preamble *
%                  case of \@lastchclass
%                     0 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
%                     1 -> \@addamp \hskip \arraycolsep
%                     2 -> % impossible
%                     3 -> % impossible
%                     4 -> \@addamp
%                     5 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
%                     6 -> \@addamp \hskip \arraycolsep
%                  end case
%  END
%
% \@arrayclassiv  ==  BEGIN  \@preamble := \@preamble * $ \@nextchar$  END
%
% \@tabclassiv   == same as \@arrayclassv except without the $ ... $
%
% \@classv ==
%   BEGIN
%    \@preamble := \@preamble * \@startpbox{\@nextchar}\ignorespaces\@sharp
```

```
%                                \@endpbox
%   END
%
% \@expast{S}: Sets \@tempa := S with all instances of *{N}{STRING}
%              replaced by N copies of STRING, where N > 0.  An *
%              appearing inside braces is ignored, but *-expressions
%              inside STRING are expanded, so nested *-expressions are
%              handled properly.
%
% \@expast{S} == BEGIN  \@xexpast S *0x\@@  END
%
% \@xexpast S1 *{N}{S2} S3 \@@ ==
%  BEGIN
%    \@tempa   := S1
%    \@tempcnta := N
%    if \@tempcnta > 0
%      then  while \@tempcnta > 0 do \@tempa   := \@tempa S2
%                                    \@tempcnta := \@tempcnta - 1 od
%            \@tempb == \@xexpast
%      else  \@tempb == \@xexnoop
%    fi
%    \expandafter \@tempb \@tempa S3 \@@
%  END
%

\def\@xexnoop #1\@@{}

\def\@expast#1{\@xexpast #1*0x\@@}

\def\@xexpast#1*#2#3#4\@@{\edef\@tempa{#1}\@tempcnta#2\relax
    \ifnum\@tempcnta >\z@ \@whilenum\@tempcnta >\z@\do
       {\edef\@tempa{\@tempa#3}\advance\@tempcnta \m@ne}\let\@tempb\@xexpast
      \else \let\@tempb\@xexnoop\fi
    \expandafter\@tempb \@tempa #4\@@}


\newif\if@firstamp
\def\@addamp{\if@firstamp \@firstampfalse \else
    \edef\@preamble{\@preamble &}\fi}
\def\@arrayacol{\edef\@preamble{\@preamble \hskip \arraycolsep}}
\def\@tabacol{\edef\@preamble{\@preamble \hskip \tabcolsep}}
\def\@ampacol{\@addamp \@acol}
\def\@acolampacol{\@acol\@addamp\@acol}

\def\@mkpream#1{\@firstamptrue\@lastchclass6
\let\@preamble\@empty\def\protect{\noexpand\protect\noexpand}\let\@sharp\relax
\let\@startpbox\relax\let\@endpbox\relax
\@expast{#1}\expandafter\@tfor \expandafter
  \@nextchar \expandafter:\expandafter=\@tempa\do{\@testpach\@nextchar
  \ifcase \@chclass \@classz \or \@classi \or \@classii \or \@classiii
    \or \@classiv \or\@classv \fi\@lastchclass\@chclass}%
\ifcase \@lastchclass \@acol
    \or \or \@preamerr \@ne\or \@preamerr \tw@\or \or \@acol \fi}

\def\@arrayclassz{\ifcase \@lastchclass \@acolampacol \or \@ampacol \or
   \or \or \@addamp \or
   \@acolampacol \or \@firstampfalse \@acol \fi
\edef\@preamble{\@preamble
  \ifcase \@chnum
     \hfil$\relax\@sharp$\hfil \or $\relax\@sharp$\hfil
    \or \hfil$\relax\@sharp$\fi}}
```

```
%% RmS 91/08/14 inserted extra braces around entry for NFSS
\def\@tabclassz{\ifcase \@lastchclass \@acolampacol \or \@ampacol \or
   \or \or \@addamp \or
   \@acolampacol \or \@firstampfalse \@acol \fi
\edef\@preamble{\@preamble{%
  \ifcase \@chnum
     \hfil\ignorespaces\@sharp\unskip\hfil
     \or \ignorespaces\@sharp\unskip\hfil
     \or \hfil\hskip\z@ \ignorespaces\@sharp\unskip\fi}}}

\def\@classi{\ifcase \@lastchclass \@acol \@arrayrule \or
   \@addtopreamble{\hskip \doublerulesep}\@arrayrule\or
   \or \or \@arrayrule \or
   \@acol \@arrayrule \or \@arrayrule \fi}


\def\@classii{\ifcase \@lastchclass \or
   \@addtopreamble{\hskip .5\arrayrulewidth}\fi}

\def\@classiii{\ifcase \@lastchclass \@acolampacol \or
   \@addamp\@acol \or
   \or \or \@addamp \or
   \@acolampacol \or \@ampacol \fi}

\def\@tabclassiv{\@addtopreamble\@nextchar}

\def\@arrayclassiv{\@addtopreamble{$\@nextchar$}}

\def\@classv{\@addtopreamble{\@startpbox{\@nextchar}\ignorespaces
\@sharp\@endpbox}}

\def\@addtopreamble#1{\edef\@preamble{\@preamble #1}}

\newcount\@chclass
\newcount\@lastchclass
\newcount\@chnum

\newdimen\arraycolsep
\newdimen\tabcolsep
\newdimen\arrayrulewidth
\newdimen\doublerulesep

\def\arraystretch{1}    % Default value.

\newbox\@arstrutbox
\def\@arstrut{\relax\ifmmode\copy\@arstrutbox\else\unhcopy\@arstrutbox\fi}


\def\@arrayrule{\@addtopreamble{\hskip -.5\arrayrulewidth
   \vrule \@width \arrayrulewidth\hskip -.5\arrayrulewidth}}

\def\@testpach#1{\@chclass \ifnum \@lastchclass=\tw@ 4 \else
    \ifnum \@lastchclass=3 5 \else
     \z@ \if #1c\@chnum \z@ \else
                              \if #1l\@chnum \@ne \else
                              \if #1r\@chnum \tw@ \else
          \@chclass \if #1|\@ne \else
                    \if #1@\tw@ \else
                    \if #1p3 \else \z@ \@preamerr 0\fi
  \fi  \fi  \fi  \fi  \fi  \fi
```

```
\fi}

\def\hline{\noalign{\ifnum0=`}\fi\hrule \@height \arrayrulewidth \futurelet
   \@tempa\@xhline}

\def\@xhline{\ifx\@tempa\hline\vskip \doublerulesep\fi
      \ifnum0=`{\fi}}

\def\vline{\vrule \@width \arrayrulewidth}

\newcount\@cla
\newcount\@clb

\def\cline#1{\@cline[#1]}
\def\@cline[#1-#2]{\noalign{\global\@cla#1\relax
\global\advance\@cla\m@ne
\ifnum\@cla>\z@\global\let\@gtempa\@clinea\else
  \global\let\@gtempa\@clineb\fi
\global\@clb#2\relax
\global\advance\@clb-\@cla}\@gtempa
\noalign{\vskip-\arrayrulewidth}}

\def\@clinea{\multispan\@cla&\multispan\@clb
\unskip\leaders\hrule \@height \arrayrulewidth \hfill
\cr}

\def\@clineb{\multispan\@clb
\unskip\leaders\hrule \@height \arrayrulewidth \hfill
\cr}

% \@startpbox{WIDTH} TEXT \egroup == \parbox{WIDTH}{TEXT}
% \@endpbox == \unskip \strut \par \egroup\hfil (Changed 14 Jan 89)
%

\def\@startpbox#1{\vtop\bgroup \hsize #1\@arrayparboxrestore}
\def\@endpbox{\unskip\strut\par\egroup\hfil}

% 14 Jan 89: Def of \@endpbox changed from
%    \def\@endpbox{\par\vskip\dp\@arstrutbox\egroup\hfil}
% so vertical spacing works out right if the last line of a `p' entry
% has a descender.

\let\@@startpbox=\@startpbox
\let\@@endpbox=\@endpbox

 \message{picture,}
%      ****************************************
%      *       THE PICTURE ENVIRONMENT        *
%      ****************************************
%
%  \unitlength     = value of dimension argument
%  \@wholewidth    = current line width
%  \@halfwidth     = half of current line width
%  \@linefnt       = font for drawing lines
%  \@circlefnt     = font for drawing circles
%
% \linethickness{DIM} : Sets the width of horizontal and vertical lines
%     in a picture to DIM.  Does not change width of slanted lines
%     or circles.   Width of all lines reset by \thinlines and
%     \thicklines
%
```

```
% \picture(XSIZE,YSIZE)(XORG,YORG)
%   BEGIN
%     \@picht :=L YSIZE * \unitlength
%     box \@picbox :=
%          \hbox to XSIZE * \unitlength
%            {\hskip -XORG * \unitlength
%             \lower YORG * \unitlength
%             \hbox{
%             \ignorespaces    %% added 13 June 89
%   END
%
% \endpicture ==
%   BEGIN
%                   } \hss }
%                   heigth of \@picbox := \@picht
%                   depth  of \@picbox := 0
%                   \mbox{\box\@picbox}   %% change 26 Aug 91
%   END
%
% \put(X, Y){OBJ} ==
%   BEGIN
%     \@killglue
%     \raise Y * \unitlength  \hbox to 0pt { \hskip X * \unitlength
%                                            OBJ \hss             }
%     \ignorespaces
%   END
%
% \multiput(X,Y)(DELX,DELY){N}{OBJ} ==
%   BEGIN
%    \@killglue
%    \@multicnt := N
%    \@xdim  := X * \unitlength
%    \@ydim  := Y * \unitlength
%    while \@multicnt > 0
%      do \raise \@ydim \hbox to 0pt { \hskip \@xdim
%                                      OBJ \hss   }
%         \@multicnt := \@multicnt - 1
%         \@xdim     := \@xdim + DELX * \unitlength
%         \@ydim     := \@ydim + DELY * \unitlength
%      od
%    \ignorespaces
%   END
%
%  \shortstack[POS]{TEXT} : Makes a \vbox containing TEXT stacked as
%      a one-column array, positioned l, r or c as indicated by POS.

\newdimen\@wholewidth
\newdimen\@halfwidth
\newdimen\unitlength \unitlength =1pt
\newbox\@picbox
\newdimen\@picht

\def\picture(#1,#2){\@ifnextchar({\@picture(#1,#2)}{\@picture(#1,#2)(0,0)}}

\def\@picture(#1,#2)(#3,#4){\@picht #2\unitlength
\setbox\@picbox\hbox to#1\unitlength\bgroup
\hskip -#3\unitlength \lower #4\unitlength \hbox\bgroup\ignorespaces}

%% 91/08/26 RmS & FMi: extra boxing level around \@picbox
%%     to guard against unboxing in math mode
%%     (proposed by John Hobby)
```

```
5003
5004 \def\endpicture{\egroup\hss\egroup\ht\@picbox\@picht
5005 \dp\@picbox\z@\mbox{\box\@picbox}}
5006
5007 % In the definitions of \put and \multiput, \hskip was replaced by \kern
5008 % just in case arg #3 = ``plus''.  (Bug detected by Don Knuth.
5009 % changed 20 Jul 87).
5010 %
5011 \long\def\put(#1,#2)#3{\@killglue\raise#2\unitlength\hbox to\z@{\kern
5012 #1\unitlength #3\hss}\ignorespaces}
5013
5014 \long\def\multiput(#1,#2)(#3,#4)#5#6{\@killglue\@multicnt #5\relax
5015 \@xdim #1\unitlength
5016 \@ydim #2\unitlength
5017 \@whilenum \@multicnt >\z@\do
5018 {\raise\@ydim\hbox to\z@{\kern
5019 \@xdim #6\hss}\advance\@multicnt \m@ne\advance\@xdim
5020 #3\unitlength\advance\@ydim #4\unitlength}\ignorespaces}
5021
5022 \def\@killglue{\unskip\@whiledim \lastskip >\z@\do{\unskip}}
5023
5024 \def\thinlines{\let\@linefnt\tenln \let\@circlefnt\tencirc
5025   \@wholewidth\fontdimen8\tenln \@halfwidth .5\@wholewidth}
5026 \def\thicklines{\let\@linefnt\tenlnw \let\@circlefnt\tencircw
5027   \@wholewidth\fontdimen8\tenlnw \@halfwidth .5\@wholewidth}
5028
5029 \def\linethickness#1{\@wholewidth #1\relax \@halfwidth .5\@wholewidth}
5030
5031 \def\shortstack{\@ifnextchar[{\@shortstack}{\@shortstack[c]}}
5032
5033 \def\@shortstack[#1]{\leavevmode
5034 \vbox\bgroup\baselineskip-\p@\lineskip 3\p@\let\mb@l\hss
5035 \let\mb@r\hss \expandafter\let\csname mb@#1\endcsname\relax
5036 \let\\\@stackcr\@ishortstack}
5037
5038 %% RmS 91/08/14 inserted extra braces around entry for NFSS
5039 \def\@ishortstack#1{\halign{\mb@l {##}\unskip\mb@r\cr #1\crcr}\egroup}
5040
5041
5042 \def\@stackcr{\@ifstar{\@ixstackcr}{\@ixstackcr}}
5043 \def\@ixstackcr{\@ifnextchar[{\@istackcr}{\cr\ignorespaces}}
5044
5045 \def\@istackcr[#1]{\cr\noalign{\vskip #1}\ignorespaces}
5046
5047
5048 % \line(X,Y){LEN} ==
5049 % BEGIN
5050 %  \@xarg    := X
5051 %  \@yarg    := Y
5052 %  \@linelen := LEN * \unitlength
5053 %  if \@xarg = 0
5054 %     then \@vline
5055 %     else if \@yarg = 0
5056 %            then \@hline
5057 %            else \@sline
5058 %          if
5059 %  if
5060 % END
5061 %
5062 % \@sline ==
5063 %  BEGIN
```

```
%    if \@xarg < 0
%      then @negarg := T
%           \@xarg  := -\@xarg
%           \@yyarg := -\@yarg
%      else @negarg := F
%           \@yyarg := \@yarg
%    fi
%    \@tempcnta := |\@yyarg|
%    if \@tempcnta > 6
%      then error: 'LATEX ERROR: Illegal \line or \vector argument.'
%           \@tempcnta := 0
%    fi
%    \box\@linechar := \hbox{\@linefnt \@getlinechar(\@xarg,\@yyarg) }
%     if \@yarg > 0 then \@upordown = \raise
%                           \@clnht := 0
%                    else \@upordown = \lower
%                         \@clnht := height of \box\@linechar
%     fi
%     \@clnwd  := width of \box\@linechar
%     if @negarg
%       then \hskip - width of \box\@linechar
%            \@tempa == \hskip - 2* width of box \@linechar
%       else \@tempa == \relax
%     fi
%  %% Put out integral number of line segments
%     while \@clnwd <  \@linelen
%       do  \@upordown \@clnht \copy\@linechar
%           \@tempa
%           \@clnht := \@clnht + ht of \box\@linechar
%           \@clnwd := \@clnwd + width of \box\@linechar
%       od
%
%  %% Put out last segment
%     \@clnht := \@clnht - height of \box\@linechar
%     \@clnwd := \@clnwd - width of \box\@linechar
%     \@tempdima   := \@linelen - \@clnwd
%     \@tempdimb   := \@tempdima - width of \box\@linechar
%     if @negarg  then \hskip -\@tempdimb
%                 else \hskip  \@tempdimb
%     fi
%     \@tempdima   := 1000 * \@tempdima
%     \@tempcnta   := \@tempdima / width of \box\@linechar
%     \@tempdima   := (\@tempcnta * ht of \box\@linechar)/1000
%     \@clnht := \@clnht + \@tempdima
%     if \@linelen < width of box\@linechar
%         then \hskip width of box\@linechar
%         else \hbox{\@upordown \@clnht \copy\@linechar}
%     fi
% END
%
% \@hline ==
%   BEGIN
%     if \@xarg < 0 then  \hskip -\@linelen \fi
%     \vrule height \@halfwidth depth \@halfwidth width \@linelen
%     if \@xarg < 0 then  \hskip -\@linelen \fi
%  END
%
% \@vline == if \@yarg < 0 \@downline else \@upline  fi
%
%
% \@getlinechar(X,Y) ==
```

```
%   BEGIN
%     \@tempcnta := 8*X - 9
%     if Y > 0
%       then \@tempcnta := \@tempcnta + Y
%       else \@tempcnta := \@tempcnta - Y + 64
%     fi
%     \char\@tempcnta
%   END
%
% \vector(X,Y){LEN} ==
% BEGIN
%  \@xarg    := X
%  \@yarg    := Y
%  \@linelen := LEN * \unitlength
%  if \@xarg = 0
%     then \@vvector
%     else if \@yarg = 0
%            then \@hvector
%            else \@svector
%          if
%  if
% END
%
% \@hvector ==
%   BEGIN
%     \@hline
%     {\@linefnt if \@xarg < 0 then  \@getlarrow(1,0)
%                              else  \@getrarrow(1,0)
%                 fi}
%   END
%
% \@vvector == if \@yarg < 0 \@downvector else \@upvector  fi
%
% \@svector ==
%  BEGIN
%   \@sline
%   \@tempcnta := |\@yarg|
%     if  \@tempcnta < 5
%        then  \hskip - width of \box\@linechar
%              \@upordown \@clnht \hbox
%                       {\@linefnt
%                        if @negarg then \@getlarrow(\@xarg,\@yyarg)
%                                   else \@getrarrow(\@xarg,\@yyarg)
%                        fi }
%        else  error: 'LATEX ERROR: Illegal \line or \vector argument.'
%     fi
%  END
%
% \@getlarrow(X,Y) ==
%  BEGIN
%   if Y = 0
%     then \@tempcnta := '33
%     else \@tempcnta := 16 * X  -  9
%          \@tempcntb := 2 * Y
%          if \@tempcntb > 0
%            then  \@tempcnta := \@tempcnta  +  \@tempcntb
%            else  \@tempcnta := \@tempcnta  -  \@tempcntb +  64
%          fi
%   fi
%   \char\@tempcnta
%  END
```

```
%
% \@getrarrow(X,Y) ==
%  BEGIN
%   \@tempcntb := |Y|
%   case of \@tempcntb
%     0 : \@tempcnta := '55
%     1 : if X < 3
%           then \@tempcnta :=  24*X - 6
%           else if X = 3
%                  then \@tempcnta := 49
%                  else \@tempcnta := 58  fi
%         fi
%     2 : if X < 3
%           then \@tempcnta :=  24*X - 3
%           else \@tempcnta := 51     % X must = 3
%         fi
%     3 : \@tempcnta := 16*X - 2
%     4 : \@tempcnta := 16*X + 7
%   endcase
%   if Y < 0
%     then \@tempcnta := \@tempcnta + 64
%   fi
%   \char\@tempcnta
%  END

\newif\if@negarg

\def\line(#1,#2)#3{\@xarg #1\relax \@yarg #2\relax
\@linelen #3\unitlength
\ifnum\@xarg =\z@ \@vline
  \else \ifnum\@yarg =\z@ \@hline \else \@sline\fi
\fi}

\def\@sline{\ifnum\@xarg<\z@ \@negargtrue \@xarg -\@xarg \@yyarg -\@yarg
  \else \@negargfalse \@yyarg \@yarg \fi
\ifnum \@yyarg >\z@ \@tempcnta\@yyarg \else \@tempcnta -\@yyarg \fi
\ifnum\@tempcnta>6 \@badlinearg\@tempcnta\z@ \fi
\ifnum\@xarg>6 \@badlinearg\@xarg \@ne \fi
\setbox\@linechar\hbox{\@linefnt\@getlinechar(\@xarg,\@yyarg)}%
\ifnum \@yarg >\z@ \let\@upordown\raise \@clnht\z@
   \else\let\@upordown\lower \@clnht \ht\@linechar\fi
\@clnwd \wd\@linechar
\if@negarg \hskip -\wd\@linechar \def\@tempa{\hskip -2\wd\@linechar}\else
     \let\@tempa\relax \fi
\@whiledim \@clnwd <\@linelen \do
  {\@upordown\@clnht\copy\@linechar
   \@tempa
   \advance\@clnht \ht\@linechar
   \advance\@clnwd \wd\@linechar}%
\advance\@clnht -\ht\@linechar
\advance\@clnwd -\wd\@linechar
\@tempdima\@linelen\advance\@tempdima -\@clnwd
\@tempdimb\@tempdima\advance\@tempdimb -\wd\@linechar
\if@negarg \hskip -\@tempdimb \else \hskip \@tempdimb \fi
\multiply\@tempdima \@m
\@tempcnta \@tempdima \@tempdima \wd\@linechar \divide\@tempcnta \@tempdima
\@tempdima \ht\@linechar \multiply\@tempdima \@tempcnta
\divide\@tempdima \@m
\advance\@clnht \@tempdima
\ifdim \@linelen <\wd\@linechar
   \hskip \wd\@linechar
```

```
  \else\@upordown\@clnht\copy\@linechar\fi}

\def\@hline{\ifnum \@xarg <\z@ \hskip -\@linelen \fi
\vrule \@height \@halfwidth \@depth \@halfwidth \@width \@linelen
\ifnum \@xarg <\z@ \hskip -\@linelen \fi}

 \def\@getlinechar(#1,#2){\@tempcnta#1\relax\multiply\@tempcnta 8%
\advance\@tempcnta -9\ifnum #2>\z@ \advance\@tempcnta #2\relax\else
\advance\@tempcnta -#2\relax\advance\@tempcnta 64 \fi
\char\@tempcnta}

\def\vector(#1,#2)#3{\@xarg #1\relax \@yarg #2\relax
\@tempcnta \ifnum\@xarg<\z@ -\@xarg\else\@xarg\fi
\ifnum\@tempcnta<5\relax
\@linelen #3\unitlength
\ifnum\@xarg =\z@ \@vvector
  \else \ifnum\@yarg =\z@ \@hvector \else \@svector\fi
\fi
\else\@badlinearg\fi}

\def\@hvector{\@hline\hbox to\z@{\@linefnt
\ifnum \@xarg <\z@ \@getlarrow(1,0)\hss\else
    \hss\@getrarrow(1,0)\fi}}

\def\@vvector{\ifnum \@yarg <\z@ \@downvector \else \@upvector \fi}

\def\@svector{\@sline
\@tempcnta\@yarg \ifnum\@tempcnta <\z@ \@tempcnta -\@tempcnta\fi
\ifnum\@tempcnta <5%
  \hskip -\wd\@linechar
  \@upordown\@clnht \hbox{\@linefnt  \if@negarg
  \@getlarrow(\@xarg,\@yyarg)\else \@getrarrow(\@xarg,\@yyarg)\fi}%
\else\@badlinearg\fi}

\def\@getlarrow(#1,#2){\ifnum #2=\z@ \@tempcnta'33 \else
\@tempcnta #1\relax\multiply\@tempcnta \sixt@@n \advance\@tempcnta
-9 \@tempcntb #2\relax\multiply\@tempcntb \tw@
\ifnum \@tempcntb >\z@ \advance\@tempcnta \@tempcntb
\else\advance\@tempcnta -\@tempcntb\advance\@tempcnta 64
\fi\fi\char\@tempcnta}

\def\@getrarrow(#1,#2){\@tempcntb #2\relax
\ifnum\@tempcntb <\z@ \@tempcntb -\@tempcntb\relax\fi
\ifcase \@tempcntb\relax \@tempcnta'55 \or
\ifnum #1<\thr@@ \@tempcnta #1\relax\multiply\@tempcnta
24\advance\@tempcnta -6 \else \ifnum #1=\thr@@ \@tempcnta 49
\else\@tempcnta 58 \fi\fi\or
\ifnum #1<\thr@@ \@tempcnta=#1\relax\multiply\@tempcnta
24\advance\@tempcnta -\thr@@ \else \@tempcnta 51 \fi\or
\@tempcnta #1\relax\multiply\@tempcnta
\sixt@@n \advance\@tempcnta -\tw@ \else
\@tempcnta #1\relax\multiply\@tempcnta
\sixt@@n \advance\@tempcnta 7 \fi\ifnum #2<\z@ \advance\@tempcnta 64 \fi
\char\@tempcnta}



\def\@vline{\ifnum \@yarg <\z@ \@downline \else \@upline\fi}

\def\@upline{\hbox to \z@{\hskip -\@halfwidth \vrule \@width \@wholewidth
   \@height \@linelen \@depth \z@\hss}}
```

```
5308
5309 \def\@downline{\hbox to \z@{\hskip -\@halfwidth \vrule \@width \@wholewidth
5310    \@height \z@ \@depth \@linelen \hss}}
5311
5312 \def\@upvector{\@upline\setbox\@tempboxa\hbox{\@linefnt\char'66}\raise
5313      \@linelen \hbox to\z@{\lower \ht\@tempboxa\box\@tempboxa\hss}}
5314
5315 \def\@downvector{\@downline\lower \@linelen
5316       \hbox to \z@{\@linefnt\char'77\hss}}
5317
5318 % \dashbox{D}(X,Y) ==
5319 %  BEGIN
5320 %  leave vertical mode
5321 %  \hbox to 0pt {
5322 %       \baselineskip := 0pt
5323 %       \lineskip     := 0pt
5324 %  %% HORIZONTAL DASHES
5325 %       \@dashdim := X * \unitlength
5326 %       \@dashcnt := \@dashdim + 200 % to prevent roundoff error
5327 %       \@dashdim := D * \unitlength
5328 %       \@dashcnt := \@dashcnt / \@dashdim
5329 %       if \@dashcnt is odd
5330 %         then \@dashdim := 0pt
5331 %              \@dashcnt := (\@dashcnt + 1) / 2
5332 %         else \@dashdim := \@dashdim / 2
5333 %              \@dashcnt := \@dashcnt / 2 - 1
5334 %              \box\@dashbox   := \hbox{\vrule height \@halfwidth
5335 %                                       depth \@halfwidth width \@dashdim}
5336 %              \put(0,0){\copy\@dashbox}
5337 %              \put(0,Y){\copy\@dashbox}
5338 %              \put(X,0){\hskip -\@dashdim\copy\@dashbox}
5339 %              \put(X,Y){\hskip -\@dashdim\box\@dashbox}
5340 %              \@dashdim := 3 * \@dashdim
5341 %       fi
5342 %       \box\@dashbox := \hbox{\vrule height \@halfwidth
5343 %                                 depth \@halfwidth width D * \unitlength
5344 %                               \hskip D * \unitlength}
5345 %       \@tempcnta := 0
5346 %       \put(0,0){\hskip \@dashdim
5347 %                 while \@tempcnta < \@dascnt
5348 %                   do \copy\@dashbox
5349 %                      \@tempcnta := \@tempcnta + 1
5350 %                   od
5351 %                }
5352 %       \@tempcnta := 0
5353 %       put(0,Y){\hskip \@dashdim
5354 %                 while \@tempcnta < \@dascnt
5355 %                   do \copy\@dashbox
5356 %                      \@tempcnta := \@tempcnta + 1
5357 %                   od
5358 %                }
5359 %
5360 % %% vertical dashes
5361 %       \@dashdim := Y * \unitlength
5362 %       \@dashcnt := \@dashdim + 200 % to prevent roundoff error
5363 %       \@dashdim := D * \unitlength
5364 %       \@dashcnt := \@dashcnt / \@dashdim
5365 %       if \@dashcnt is odd
5366 %         then \@dashdim := 0pt
5367 %              \@dashcnt := (\@dashcnt + 1) / 2
5368 %         else \@dashdim := \@dashdim / 2
```

```
% \@dashcnt := \@dashcnt / 2 - 1
% \box\@dashbox   := \hbox{\hskip -\@halfwidth
%                                 \vrule width \@wholewidth
%                                         height \@dashdim  }
% \put(0,0){\copy\@dashbox}
% \put(X,0){\copy\@dashbox}
% \put(0,Y){\lower\@dashdim\copy\@dashbox}
% \put(X,Y){\lower\@dashdim\copy\@dashbox}
% \@dashdim := 3 * \@dashdim
%       fi
%       \box\@dashbox := \hbox{\vrule width \@wholewidth
%                                 height D * \unitlength       }
%       \@tempcnta := 0
%       put(0,0){\hskip -\halfwidth
%                \vbox{while \@tempcnta < \@dashcnt
%                       do \vskip D*\unitlength
%                          \copy\@dashbox
%                          \@tempcnta := \@tempcnta + 1
%                       od
%                      \vskip \@dashdim
%                     } }
%       \@tempcnta := 0
%       put(X,0){\hskip -\halfwidth
%                \vbox{while \@tempcnta < \@dashcnt
%                       do \vskip D*\unitlength
%                          \copy\@dashbox
%                          \@tempcnta := \@tempcnta + 1
%                       od
%                      \vskip \@dashdim
%                     }
%                }
%    }     % END DASHES
%
%  \@imakepicbox(X,Y)
% END

\def\dashbox#1(#2,#3){\leavevmode\hbox to\z@{\baselineskip \z@
\lineskip \z@
\@dashdim #2\unitlength
\@dashcnt \@dashdim \advance\@dashcnt 200
\@dashdim #1\unitlength\divide\@dashcnt \@dashdim
\ifodd\@dashcnt\@dashdim \z@
\advance\@dashcnt \@ne \divide\@dashcnt \tw@
\else \divide\@dashdim \tw@ \divide\@dashcnt \tw@
\advance\@dashcnt \m@ne
\setbox\@dashbox \hbox{\vrule \@height \@halfwidth \@depth \@halfwidth
\@width \@dashdim}\put(0,0){\copy\@dashbox}%
\put(0,#3){\copy\@dashbox}%
\put(#2,0){\hskip-\@dashdim\copy\@dashbox}%
\put(#2,#3){\hskip-\@dashdim\box\@dashbox}%
\multiply\@dashdim \thr@@
\fi
\setbox\@dashbox \hbox{\vrule \@height \@halfwidth \@depth \@halfwidth
\@width #1\unitlength\hskip #1\unitlength}\@tempcnta\z@
\put(0,0){\hskip\@dashdim \@whilenum \@tempcnta <\@dashcnt
\do{\copy\@dashbox\advance\@tempcnta \@ne }}\@tempcnta\z@
\put(0,#3){\hskip\@dashdim \@whilenum \@tempcnta <\@dashcnt
\do{\copy\@dashbox\advance\@tempcnta \@ne }}%
\@dashdim #3\unitlength
\@dashcnt \@dashdim \advance\@dashcnt 200
\@dashdim #1\unitlength\divide\@dashcnt \@dashdim
```

```
\ifodd\@dashcnt \@dashdim \z@
\advance\@dashcnt \@ne \divide\@dashcnt \tw@
\else
\divide\@dashdim \tw@ \divide\@dashcnt \tw@
\advance\@dashcnt \m@ne
\setbox\@dashbox\hbox{\hskip -\@halfwidth
\vrule \@width \@wholewidth
\@height \@dashdim}\put(0,0){\copy\@dashbox}%
\put(#2,0){\copy\@dashbox}%
\put(0,#3){\lower\@dashdim\copy\@dashbox}%
\put(#2,#3){\lower\@dashdim\copy\@dashbox}%
\multiply\@dashdim \thr@@
\fi
\setbox\@dashbox\hbox{\vrule \@width \@wholewidth
\@height #1\unitlength}\@tempcnta\z@
\put(0,0){\hskip -\@halfwidth \vbox{\@whilenum \@tempcnta <\@dashcnt
\do{\vskip #1\unitlength\copy\@dashbox\advance\@tempcnta \@ne }%
\vskip\@dashdim}}\@tempcnta\z@
\put(#2,0){\hskip -\@halfwidth \vbox{\@whilenum \@tempcnta<\@dashcnt
\do{\vskip #1\unitlength\copy\@dashbox\advance\@tempcnta \@ne }%
\vskip\@dashdim}}}\@makepicbox(#2,#3)}

% CIRCLES AND OVALS
%
%  USER COMMANDS:
%
%  \circle{D} : Produces the circle with the diameter as close as
%               possible to D * \unitlength.  \put(X,Y){\circle{D}}
%               puts the circle with its center at (X,Y).
%
%  \oval(X,Y) : Makes an oval as round as possible that fits in the
%               rectangle of width X * \unitlength and height
%               Y * \unitlength. The reference point is the center.
%
% \oval(X,Y)[POS] : Save as \oval(X,Y) except it draws only the
%                   half or quadrant of the oval indicated by POS.
%                   E.G., \oval(X,Y)[t] draws just the top half
%                   and \oval(X,Y)[br] draws just the bottom right
%                   quadrant.  In all cases, the reference point is
%                   the same as the unqualified \oval(X,Y) command.
%
% \@ovvert {DELTA1} {DELTA2} : Makes a vbox containing either the left side
%        or the right side of the oval being constructed.  The baseline
%        will coincide with the outside bottom edge of the oval; the left
%        side of the box will coincide with the left edge of the vertical
%        rule.  The width of the box will be \@tempdima.
%        DELTA1 and DELTA2 are added to the character number in \@tempcnta
%        to get the characters for the top and bottom quarter circle pieces.
%
% \@ovhorz : Makes an hbox containing the straight rule for either the
%         top or the bottom of the oval being constructed.  The baseline
%         will coincide with bottom edge of the rule; the left side of
%         the box will coincide with the left side of the oval.
%         The width of the box will be \@ovxx.
%
% \@getcirc {DIAM} : Sets \@tempcnta to the character number
%                    of the top-right quarter circle with the largest
%                    diameter less than or equal to DIAM.
%                    Sets \@tempboxa to an hbox containing that character.
%                    Sets \@tempdima to \wd \@tempboxa, which is the distance
%                    from the circle's left outside edge to its right
```

```
%                   inside edge.
%                   (These characters are like those described in the
%                   TeXbook, pp. 389-90.)
%
% \@getcirc {DIAM} ==
%   BEGIN
%     \@tempcnta       := integer coercion of (DIAM + 2pt)     %% + 2pt added
%     \@tempcnta       := \@tempcnta / integer coercion of 4pt %%    1 Nov 88
%     if \@tempcnta > 10
%       then \@tempcnta := 10 fi
%     if \@tempcnta > 0
%       then \@tempcnta := \@tempcnta-1
%       else LaTeX Warning: Oval too small.
%     fi
%     \@tempcnta       := 4 * \@tempcnta
%     \@tempboxa       := \hbox{\@circlefnt \char \@tempcnta}
%     \@tempdima       := \wd \@tempboxa
%   END
%
% \@put{X}{Y}{OBJ} ==
%   BEGIN
%     \raise Y \hbox to 0pt{\hskip X OBJ \hss}
%   END
%
% \@oval(X,Y)[POS] ==
%   BEGIN
%     \begingroup
%       \boxmaxdepth := \maxdimen
%       @ovt := @ovb := @ovl := @ovr := true
%       for all E in POS
%         do  @ovE := false od
%       \@ovxx      := X * \unitlength
%       \@ovyy      := Y * \unitlength
%       \@tempdimb := min(\@ovxx,\@ovyy)
%       \@getcirc{\@tempdimb-2pt}   %% "-2pt" added 7 Dec 89
%       \@ovro     := \ht \@tempboxa
%       \@ovri     := \dp \@tempboxa
%       \@ovdx     := \@ovxx - \@tempdima
%       \@ovdx     := \@ovdx/2
%       \@ovdy     := \@ovyy - \@tempdima
%       \@ovdy     := \@ovyy/2
%       \@circlefnt
%       \@tempboxa :=
%           \hbox{
%                 if @ovr
%                   then \@ovvert{3}{2} \kern -\@tempdima
%                 fi
%                 if @ovl
%                   then \kern \@ovxx \@ovvert{0}{1} \kern -\@tempdima
%                        \kern -\@ovxx
%                 fi
%                 if @ovt
%                   then \@ovhorz \kern -\@ovxx
%                 fi
%                 if @ovb
%                   then \raise \@ovyy \@ovhorz
%                 fi
%                }
%       \@ovdx    := \@ovdx + \@ovro
%       \@ovdy    := \@ovdy + \@ovro
%      \ht\@tempboxa := \dp\@tempboxa := 0
```

```
%       \@put{-\@ovdx}{-\@ovdy}{\box\@tempboxa}
%    \endgroup
%   END
%
% \@ovvert {DELTA1} {DELTA2} ==
%   BEGIN
%      \vbox to \@ovyy {
%                       if @ovb
%                         then \@tempcntb := \@tempcnta + DELTA1
%                              \kern -\@ovro
%                              \hbox { \char \@tempcntb }
%                              \nointerlineskip
%                         else \kern \@ovri \kern \@ovdy
%                       fi
%                       \leaders \vrule width \@wholewidth \vfil
%                       \nointerlineskip
%                       if @ovt
%                         then \@tempcntb := \@tempcnta + DELTA2
%                              \hbox { \char \@tempcntb }
%                         else \kern \@ovdy \kern \@ovro
%                       fi
%                      }
%   END
%
% \@ovhorz ==
%   BEGIN
%    \hbox to \@ovxx{
%                    \kern \@ovro
%                    if @ovr
%                      then
%                      else \kern \@ovdx
%                    fi
%                    \leaders \hrule height \@wholewidth \hfil
%                    if @ovl
%                      then
%                      else \kern \@ovdx
%                    fi
%                    \kern \@ovri
%                   }
%   END
%
% \circle{DIAM} ==
%   BEGIN
%    \begingroup
%    \boxmaxdepth := maxdimen
%    \@tempdimb := DIAM *\unitlength
%    if \@tempdimb > 15.5pt
%      then \@getcirc{\@tempdimb}
%           \@ovro := \ht \@tempboxa
%           \@tempboxa := \hbox{
%                   \@circlefnt
%                   \@tempcnta := \@tempcnta + 2
%                   \char \@tempcnta
%                   \@tempcnta := \@tempcnta - 1
%                   \char \@tempcnta
%                   \kern -2\@tempdima
%                   \@tempcnta := \@tempcnta + 2
%                   \raise \@tempdima \hbox { \char \@tempcnta }
%                   \raise \@tempdima \box\@tempboxa
%                  }
%           \ht\@tempboxa := \dp\@tempboxa := 0
```

```
%           \@put{-\@ovro}{-\@ovro}{\@tempboxa}
%      else
%           \@circ{\@tempdimb}{96}
%    fi
%   \endgroup
%   END
%
% \circle*{DIAM}  ==  \@dot{DIAM} == \@circ{DIAM*\unitlength}{112}
%
% \@circ{DIAM}{CHAR} ==
%  BEGIN
%   \@tempcnta := integer coercion of (DIAM + .5pt)/1pt.
%   if \@tempcnta > 15 then \@tempcnta := 15 fi
%   if \@tempcnta > 1  then \@tempcnta := \@tempcnta - 1 fi
%   \@tempcnta := \@tempcnta + CHAR
%   \@circlefnt
%   \char \@tempcnta
%  END
%

\newif\if@ovt
\newif\if@ovb
\newif\if@ovl
\newif\if@ovr
\newdimen\@ovxx
\newdimen\@ovyy
\newdimen\@ovdx
\newdimen\@ovdy
\newdimen\@ovro
\newdimen\@ovri

%% \advance\@tempdima 2pt\relax added 1 Nov 88 to fix bug in which
%% size of drawn circle not monotonic function of argument of \circle,
%% caused by different rounding for dimensions of large and small circles.
%
\def\@getcirc#1{\@tempdima #1\relax \advance\@tempdima 2\p@
  \@tempcnta\@tempdima
  \@tempdima 4\p@ \divide\@tempcnta\@tempdima
  \ifnum \@tempcnta >10\relax \@tempcnta 10\relax\fi
  \ifnum \@tempcnta >\z@ \advance\@tempcnta\m@ne
    \else \@warning{Oval too small}\fi
  \multiply\@tempcnta 4\relax
  \setbox \@tempboxa \hbox{\@circlefnt
  \char \@tempcnta}\@tempdima \wd \@tempboxa}

\def\@put#1#2#3{\raise #2\hbox to\z@{\hskip #1#3\hss}}

\def\oval(#1,#2){\@ifnextchar[{\@oval(#1,#2)}{\@oval(#1,#2)[]}}

\def\@oval(#1,#2)[#3]{\begingroup\boxmaxdepth \maxdimen
  \@ovttrue \@ovbtrue \@ovltrue \@ovrtrue
  \@tfor\@tempa :=#3\do{\csname @ov\@tempa false\endcsname}\@ovxx
  #1\unitlength \@ovyy #2\unitlength
  \@tempdimb \ifdim \@ovyy >\@ovxx \@ovxx\else \@ovyy \fi
  \advance \@tempdimb -2\p@
  \@getcirc \@tempdimb
  \@ovro \ht\@tempboxa \@ovri \dp\@tempboxa
  \@ovdx\@ovxx \advance\@ovdx -\@tempdima \divide\@ovdx \tw@
  \@ovdy\@ovyy \advance\@ovdy -\@tempdima \divide\@ovdy \tw@
  \@circlefnt \setbox\@tempboxa
  \hbox{\if@ovr \@ovvert32\kern -\@tempdima \fi
```

```
  \if@ovl \kern \@ovxx \@ovvert01\kern -\@tempdima \kern -\@ovxx \fi
  \if@ovt \@ovhorz \kern -\@ovxx \fi
  \if@ovb \raise \@ovyy \@ovhorz \fi}\advance\@ovdx\@ovro
  \advance\@ovdy\@ovro \ht\@tempboxa\z@ \dp\@tempboxa\z@
  \@put{-\@ovdx}{-\@ovdy}{\box\@tempboxa}%
  \endgroup}

\def\@ovvert#1#2{\vbox to\@ovyy{%
    \if@ovb \@tempcntb \@tempcnta \advance \@tempcntb #1\relax
      \kern -\@ovro \hbox{\char \@tempcntb}\nointerlineskip
    \else \kern \@ovri \kern \@ovdy \fi
    \leaders\vrule \@width \@wholewidth\vfil \nointerlineskip
    \if@ovt \@tempcntb \@tempcnta \advance \@tempcntb #2\relax
      \hbox{\char \@tempcntb}%
    \else \kern \@ovdy \kern \@ovro \fi}}

\def\@ovhorz{\hbox to \@ovxx{\kern \@ovro
    \if@ovr \else \kern \@ovdx \fi
    \leaders \hrule \@height \@wholewidth \hfil
    \if@ovl \else \kern \@ovdx \fi
    \kern \@ovri}}

\def\circle{\@ifstar{\@dot}{\@circle}}
\def\@circle#1{\begingroup \boxmaxdepth \maxdimen \@tempdimb #1\unitlength
   \ifdim \@tempdimb >15.5\p@ \@getcirc\@tempdimb
      \@ovro\ht\@tempboxa
     \setbox\@tempboxa\hbox{\@circlefnt
      \advance\@tempcnta\tw@ \char \@tempcnta
      \advance\@tempcnta\m@ne \char \@tempcnta \kern -2\@tempdima
      \advance\@tempcnta\tw@
      \raise \@tempdima \hbox{\char\@tempcnta}\raise \@tempdima
        \box\@tempboxa}\ht\@tempboxa\z@ \dp\@tempboxa\z@
      \@put{-\@ovro}{-\@ovro}{\box\@tempboxa}%
   \else  \@circ\@tempdimb{96}\fi\endgroup}

\def\@dot#1{\@tempdimb #1\unitlength \@circ\@tempdimb{112}}

\def\@circ#1#2{\@tempdima #1\relax \advance\@tempdima .5\p@
   \@tempcnta\@tempdima \@tempdima \p@
   \divide\@tempcnta\@tempdima
   \ifnum\@tempcnta >15\relax \@tempcnta 15\relax \fi
   \ifnum \@tempcnta >\z@ \advance\@tempcnta\m@ne\fi
   \advance\@tempcnta #2\relax
   \@circlefnt \char\@tempcnta}


%INITIALIZATION
\thinlines

\newcount\@xarg
\newcount\@yarg
\newcount\@yyarg
\newcount\@multicnt
\newdimen\@xdim
\newdimen\@ydim
\newbox\@linechar
\newdimen\@linelen
\newdimen\@clnwd
\newdimen\@clnht
\newdimen\@dashdim
\newbox\@dashbox
```

```
\newcount\@dashcnt



 \message{theorem,}
%       ****************************************
%       *         THEOREM ENVIRONMENTS         *
%       ****************************************
%
%  The user creates his own theorem-like environments with the command
%      \newtheorem{NAME}{TEXT}[COUNTER]  or
%      \newtheorem{NAME}[OLDNAME]{TEXT}
%  This defines the environment NAME to be just as one would expect a
%  theorem environment to be, except that it prints ``TEXT'' instead of
%  ``Theorem''.
%
%  If OLDNAME is given, then environments NAME and OLDNAME use the same
%  counter, so using a NAME environment advances the number of the next
%  NAME environment, and vice-versa.
%
%  If COUNTER is given, then environment NAME is numbered within COUNTER.
%  E.g., if COUNTER = subsection, then the first NAME in subsection 7.2
%  is numbered TEXT 7.2.1.
%
%  The way NAME environments are numbered can be changed by redefining
%  \theNAME.
%
%  DOCUMENT STYLE PARAMETERS
%
%  \@thmcounter{COUNTER} : A command such that
%                \edef\theCOUNTER{\@thmcounter{COUNTER}}
%         defines \theCOUNTER to produce a number for a theorem environment.
%         The default is:
%            BEGIN \noexpand\arabic{COUNTER} END
%
%  \@thmcountersep : A separator placed between a theorem number and
%         the number of the counter within which it is numbered.
%         E.g., to make the third theorem of section 7.2 be numbered
%         7.2-3, \@thmcountersep should be \def'ed to '-'.  Its
%         default is '.'.
%
%  \@begintheorem{NAME}{NUMBER} : A command that begins a theorem
%         environment for a 'theorem' named 'NAME NUMBER' --
%         e.g., \@begintheorem{Lemma}{3.7} starts Lemma 3.7.
%
%  \@opargbegintheorem{NAME}{NUMBER}{OPARG} : A command that begins a theorem
%         environment for a 'theorem' named 'NAME NUMBER' with optional
%         argument OPARG -- e.g., \@begintheorem{Lemma}{3.7}{Jones}
%         starts `Lemma 3.7 (Jones):'.
%
%  \@endtheorem : A command that ends a theorem environment.
%
% \newtheorem{NAME}{TEXT}[COUNTER] ==
%   BEGIN
%     if \NAME is definable
%       then \@definecounter{NAME}
%            if COUNTER present
%              then \@addtoreset{NAME}{COUNTER} fi
%                   \theNAME ==  BEGIN \theCOUNTER \@thmcountersep
%                                      eval\@thmcounter{NAME}      END
%              else \theNAME ==  BEGIN eval\@thmcounter{NAME} END
```

```
% \NAME == \@thm{NAME}{TEXT}
% \endNAME == \@endtheorem
%       else  error
%     fi
%   END
%
% \newtheorem{NAME}[OLDNAME]{TEXT}==
%   BEGIN
%     if \NAME is definable
%       then \theNAME == \theOLDNAME
%            \NAME == \@thm{OLDNAME}{TEXT}
%            \endNAME == \@endtheorem
%       else  error
%     fi
%   END
%
% \@thm{NAME}{TEXT} ==
%   BEGIN
%    \refstepcounter{NAME}
%    if next char = [
%       then \@ythm{NAME}{TEXT}
%       else \@xthm{NAME}{TEXT}
%    fi
%   END
%
% \@xthm{NAME}{TEXT} ==
%   BEGIN
%    \@begintheorem{TEXT}{\theNAME}
%    \ignorespaces
%   END
%
% \@ythm{NAME}{TEXT}[OPARG] ==
%   BEGIN
%    \@opargbegintheorem{TEXT}{\theNAME}{OPARG}
%    \ignorespaces
%   END
%
\def\newtheorem#1{\@ifnextchar[{\@othm{#1}}{\@nthm{#1}}}

\def\@nthm#1#2{%
\@ifnextchar[{\@xnthm{#1}{#2}}{\@ynthm{#1}{#2}}}

\def\@xnthm#1#2[#3]{\expandafter\@ifdefinable\csname #1\endcsname
{\@definecounter{#1}\@addtoreset{#1}{#3}%
\expandafter\xdef\csname the#1\endcsname{\expandafter\noexpand
  \csname the#3\endcsname \@thmcountersep \@thmcounter{#1}}%
\global\@namedef{#1}{\@thm{#1}{#2}}\global\@namedef{end#1}{\@endtheorem}}}

\def\@ynthm#1#2{\expandafter\@ifdefinable\csname #1\endcsname
{\@definecounter{#1}%
\expandafter\xdef\csname the#1\endcsname{\@thmcounter{#1}}%
\global\@namedef{#1}{\@thm{#1}{#2}}\global\@namedef{end#1}{\@endtheorem}}}

%% RmS 92/01/10: check for existence of theorem environment #2
\def\@othm#1[#2]#3{%
  \@ifundefined{c@#2}{\@latexerr{No theorem environment `#2' defined}\@eha}%
  {\expandafter\@ifdefinable\csname #1\endcsname
  {\global\@namedef{the#1}{\@nameuse{the#2}}%
\global\@namedef{#1}{\@thm{#2}{#3}}%
\global\@namedef{end#1}{\@endtheorem}}}}

```

```
\def\@thm#1#2{\refstepcounter
    {#1}\@ifnextchar[{\@ythm{#1}{#2}}{\@xthm{#1}{#2}}}

\def\@xthm#1#2{\@begintheorem{#2}{\csname the#1\endcsname}\ignorespaces}
\def\@ythm#1#2[#3]{\@opargbegintheorem{#2}{\csname
       the#1\endcsname}{#3}\ignorespaces}

%DEFAULT VALUES
\def\@thmcounter#1{\noexpand\arabic{#1}}
\def\@thmcountersep{.}
%deleted September 2, 1986 MDK
%\def\@makethmnumber#1#2{\bf #1 #2:}

%% RmS 91/08/14 Moved \it after \item to make it work with NFSS
\def\@begintheorem#1#2{\trivlist \item[\hskip \labelsep{\bf #1\ #2}]\it}
\def\@opargbegintheorem#1#2#3{\trivlist
      \item[\hskip \labelsep{\bf #1\ #2\ (#3)}]\it}
\def\@endtheorem{\endtrivlist}


 \message{lengths,}
%     ****************************************
%     *              LENGTHS                 *
%     ****************************************
%
% USER COMMANDS:
%
%   \newlength{\NAME}          == \newskip\NAME
%   \setlength{\NAME}{VALUE}   == \NAME :=L VALUE
%   \addtolength{\NAME}{VALUE} == \NAME :=L \NAME + VALUE
%   \settowidth{\NAME}{TEXT}   == \NAME :=L width of \hbox{TEXT}
%
\def\newlength#1{\@ifdefinable#1{\newskip#1}}
\def\setlength#1#2{#1#2\relax}
\def\addtolength#1#2{\advance#1 #2\relax}
\def\settowidth#1#2{\setbox\@tempboxa\hbox{#2}#1\wd\@tempboxa\relax}
   %% \relax added 24 Mar 86

 \message{title,}
%     ****************************************
%     *              THE TITLE               *
%     ****************************************
%
% The user defines the title, author, date by the declarations \title{NAME},
% \author{NAME} and \date{DATE}.  Inside these, he can use the \thanks
% command to make footnoted acknowledgements, notice of address, etc.
% The \maketitle command produces the actual title.  Note: multiple authors
% are separated with the \and command.

\def\title#1{\gdef\@title{#1}}

\def\author#1{\gdef\@author{#1}}

\def\date#1{\gdef\@date{#1}}
\gdef\@date{\today}          %Default is today's date

\def\thanks#1{\footnotemark\begingroup
\def\protect{\noexpand\protect\noexpand}\xdef\@thanks{\@thanks
  \protect\footnotetext[\the\c@footnote]{#1}}\endgroup}

\def\@thanks{}
```

```
5918
5919 \def\and{%%                              % \begin{tabular}
5920 \end{tabular}\hskip 1em plus.17fil\begin{tabular}[t]{c}%% \end{tabular}
5921 }
5922
5923
5924
5925  \message{sectioning,}
5926 %     ****************************************
5927 %     *              SECTIONING              *
5928 %     ****************************************
5929 %
5930 %
5931 % \@startsection {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}
5932 %            optional * [ALTHEADING]{HEADING}
5933 %    Generic command to start a section.
5934 %    NAME       : e.g., 'subsection'
5935 %    LEVEL      : a number, denoting depth of section -- e.g., chapter=1,
5936 %                 section = 2, etc.
5937 %    INDENT     : Indentation of heading from left margin
5938 %    BEFORESKIP : Absolute value = skip to leave above the heading.
5939 %                 If negative, then paragraph indent of text following
5940 %                 heading is suppressed.
5941 %    AFTERSKIP  : if positive, then skip to leave below heading, else
5942 %                 negative of skip to leave to right of run-in heading.
5943 %    STYLE      : commands to set style
5944 %  If '*' missing, then increments the counter.  If it is present, then
5945 %  there should be no [ALTHEADING] argument.
5946 %  Uses the counter 'secnumdepth' whose value is the highest section
5947 %  level that is to be numbered.
5948 %
5949 %  WARNING: The \@startsection command should be at the same or higher
5950 %  grouping level as the text that follows it.  For example, you
5951 %  should NOT do something like
5952 %      \def\foo{ \begingroup ...
5953 %                  \paragraph{...}
5954 %                \endgroup}
5955 %
5956 % \@startsection {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE} ==
5957 %    BEGIN
5958 %     IF  @noskipsec = T  THEN  \leavevmode  FI  % true if previous section
5959 %                                                % had no body.
5960 %     \par
5961 %     \@tempskipa  := BEFORESKIP
5962 %     @afterindent := T
5963 %     IF \@tempskipa < 0  THEN  \@tempskipa  := -\@tempskipa
5964 %                               @afterindent := F
5965 %     FI
5966 %     IF @nobreak = true
5967 %       THEN \everypar == null
5968 %       ELSE \addpenalty{\@secpenalty}
5969 %            \addvspace{\@tempskipa}
5970 %     FI
5971 %     IF * next
5972 %       THEN \@ssect{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}
5973 %       ELSE \@dblarg{\@sect
5974 %                        {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}}
5975 %     FI
5976 % END
5977 %
5978 % \@sect{NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}[ARG1]{ARG2} ==
```

```
%   BEGIN
%    IF LEVEL > \c@secnumdepth
%      THEN \@svsec :=L null
%      ELSE \refstepcounter{NAME}
%           \@svsec :=L BEGIN \theNAME END
%    FI
%    IF AFTERSKIP > 0
%      THEN \begingroup
%             STYLE
%             \@hangfrom{\hskip INDENT\@svsec}
%             {\interlinepenalty 10000 ARG2\par}
%           \endgroup
%           \NAMEmark{ARG1}
%           \addcontentsline{toc}{NAME}
%              { IF  LEVEL > \c@secnumdepth
%                  ELSE \protect\numberline{\theNAME}  FI
%                ARG1 }
%      ELSE \@svsechd == BEGIN  STYLE
%                                \hskip INDENT\@svsec
%                                ARG2
%                                \NAMEmark{ARG1}
%                                \addcontentsline{toc}{NAME}
%                                   { IF  LEVEL > \c@secnumdepth
%                                       ELSE \protect\numberline{\theNAME}  FI
%                                     ARG1 }
%                        END
%    FI
%    \@xsect{AFTERSKIP}
% END
%
% \@xsect{AFTERSKIP} ==
%  BEGIN
%    IF AFTERSKIP > 0
%      THEN \par \nobreak
%           \vskip AFTERSKIP
%           \@afterheading
%      ELSE @nobreak :=G F
%           @noskipsec :=G T
%           \everypar{ IF @noskipsec = T
%                        THEN @noskipsec :=G F
%                             \clubpenalty :=G 10000
%                             \hskip -\parindent
%                             \begingroup
%                               \@svsechd
%                             \endgroup
%                             \unskip
%                             \hskip -AFTERSKIP \relax %% relax added 14 Jan 91
%                        ELSE \clubpenalty :=G \@clubpenalty
%                             \everypar := NULL
%                      FI
%                    }
%    FI
%
%   END
%
% \@ssect{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}{ARG} ==
%    BEGIN
%     IF AFTERSKIP > 0
%       THEN \begingroup
%              STYLE
%              \@hangfrom{\hskip INDENT}{\interlinepenalty 10000 ARG\par}
```

```
%           \endgroup
%      ELSE \@svsechd == BEGIN STYLE
%                              \hskip INDENT
%                              ARG
%                         END
%    FI
%    \@xsect{AFTERSKIP}
%   END
%
% \@afterheading ==
%  BEGIN
%    @nobreak :=G true
%    \everypar := BEGIN  IF @nobreak = T
%                          THEN @nobreak  :=G false
%                               \clubpenalty :=G 10000
%                               IF @afterindent = F
%                                 THEN remove \lastbox
%                               FI
%                          ELSE \clubpenalty :=G \@clubpenalty
%                               \everypar := NULL
%                       FI
%                 END
%  END
%
% \@secpenalty : The penalty (usually negative) put before a section
%                heading unless it immediately follows another one.
%
\newcount\@secpenalty
\@secpenalty = -300


\newif\if@noskipsec \@noskipsectrue


\def\@startsection#1#2#3#4#5#6{\if@noskipsec \leavevmode \fi
   \par \@tempskipa #4\relax
   \@afterindenttrue
   \ifdim \@tempskipa <\z@ \@tempskipa -\@tempskipa \@afterindentfalse\fi
   \if@nobreak \everypar{}\else
     \addpenalty{\@secpenalty}\addvspace{\@tempskipa}\fi \@ifstar
     {\@ssect{#3}{#4}{#5}{#6}}{\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}}

\def\@sect#1#2#3#4#5#6[#7]#8{\ifnum #2>\c@secnumdepth
     \let\@svsec\@empty\else
     \refstepcounter{#1}\edef\@svsec{\csname the#1\endcsname\hskip 1em}\fi
     \@tempskipa #5\relax
      \ifdim \@tempskipa>\z@
        \begingroup #6\relax
          \@hangfrom{\hskip #3\relax\@svsec}{\interlinepenalty \@M #8\par}%
        \endgroup
       \csname #1mark\endcsname{#7}\addcontentsline
         {toc}{#1}{\ifnum #2>\c@secnumdepth \else
                      \protect\numberline{\csname the#1\endcsname}\fi
                    #7}\else
        \def\@svsechd{#6\hskip #3\relax  %% \relax added 2 May 90
                   \@svsec #8\csname #1mark\endcsname
                      {#7}\addcontentsline
                           {toc}{#1}{\ifnum #2>\c@secnumdepth \else
                             \protect\numberline{\csname the#1\endcsname}\fi
                       #7}}\fi
     \@xsect{#5}}
```

```
6101
6102 \def\@xsect#1{\@tempskipa #1\relax
6103       \ifdim \@tempskipa>\z@
6104        \par \nobreak
6105        \vskip \@tempskipa
6106        \@afterheading
6107     \else \global\@nobreakfalse \global\@noskipsectrue
6108        \everypar{\if@noskipsec \global\@noskipsecfalse
6109                     \clubpenalty\@M \hskip -\parindent
6110                     \begingroup \@svsechd \endgroup \unskip
6111                     \hskip -#1\relax  % relax added 14 Jan 91
6112                    \else \clubpenalty \@clubpenalty
6113                      \everypar{}\fi}\fi\ignorespaces}
6114
6115 \def\@ssect#1#2#3#4#5{\@tempskipa #3\relax
6116    \ifdim \@tempskipa>\z@
6117      \begingroup #4\@hangfrom{\hskip #1}{\interlinepenalty \@M #5\par}\endgroup
6118    \else \def\@svsechd{#4\hskip #1\relax #5}\fi
6119     \@xsect{#3}}
6120
6121 \newif\if@afterindent \@afterindenttrue
6122
6123 \def\@afterheading{\global\@nobreaktrue
6124       \everypar{\if@nobreak
6125                    \global\@nobreakfalse
6126                    \clubpenalty \@M
6127                    \if@afterindent \else {\setbox\z@\lastbox}\fi
6128                  \else \clubpenalty \@clubpenalty
6129                     \everypar{}\fi}}
6130
6131
6132 % \@hangfrom{TEXT} : Puts TEXT in a box, and makes a hanging indentation
6133 %    of the following material up to the first \par.  Should be used
6134 %    in vertical mode.
6135 %
6136 \def\@hangfrom#1{\setbox\@tempboxa\hbox{#1}%
6137       \hangindent \wd\@tempboxa\noindent\box\@tempboxa}
6138
6139 \newcount\c@secnumdepth
6140 \newcount\c@tocdepth
6141
6142 % \secdef{UNSTARCMDS}{STARCMDS} :
6143 %    When defining a \chapter or \section command without using
6144 %    \@startsection, you can use \secdef as follows:
6145 %       \def\chapter { ... \secdef \CMDA \CMDB }
6146 %       \def\CMDA    [#1]#2{ ... }  % Command to define \chapter[...]{...}
6147 %       \def\CMDB    #1{ ... }      % Command to define \chapter*{...}
6148
6149 \def\secdef#1#2{\@ifstar{#2}{\@dblarg{#1}}}
6150
6151 % Initializations
6152 %
6153 \def\sectionmark#1{}
6154 \def\subsectionmark#1{}
6155 \def\subsubsectionmark#1{}
6156 \def\paragraphmark#1{}
6157 \def\subparagraphmark#1{}
6158
6159  \message{contents,}
6160 %     ***************************************
6161 %     *        TABLE OF CONTENTS, ETC.        *
```

```
%     ****************************************
%
% CONVENTIONS:
%   \tf@foo = file number for output for table foo.  The file is
%             opened only if @filesw = true.
%
%   \contentsline{TYPE}{ENTRY}{PAGE}
%       Macro to produce a TYPE entry in a table of contents, etc.
%       It will appear in the .TOC or other file.  For example,
%       The entry for subsection 1.4.3 in the table of contents might
%       be produced by:
%       \contentsline{subsection}{\makebox{30pt}[r]{1.4.3} Gnats and Gnus}{22}
%       The \protect command causes command sequences to be written
%       without expanding them.
%
%   \l@TYPE{ENTRY}{PAGE}
%       Macro defined by document style for making an entry of
%       type TYPE in a table of contents, etc.  E.g., the document
%       style should define \l@chapter, \l@section, etc.
%
%   \addcontentsline{TABLE}{TYPE}{ENTRY}
%      User command for adding his own entry to a table of contents, etc.
%      It adds the entry
%         \contentsline{TYPE}{ENTRY}{page}
%      to the .TABLE file.
%
%   \addtocontents{TABLE}{TEXT} : Adds TEXT to the .TABLE file, with no
%      page number.
%
%  Note: When used in the ENTRY or TEXT of one of the above commands,
%  \protect causes the following control sequence to be written
%  on the file without being expanded.  The sequence will be expanded
%  when the table of contents entry is processed.
%
%  SURPRISE: \index, \glossary,  and \label are no-ops inside an
%  \addcontentsline or \addtocontents command argument.  This could cause a
%  problem if the user puts an \index or \label into one of the commands he
%  writes, or into the optional 'short version' argument of a \section or
%  \caption command.
%
% \addcontentsline{TABLE}{TYPE}{ENTRY}  ==
%   BEGIN
%     if @filesw = true
%       then  \begingroup
%               \index == \label == \glossary == \@gobble
%               \protect{ARG} == \string\string\string ARG \string\space\space
%               \@temptokena := \thepage
%               \@tempa  == write \string\contentsline
%                                 {TYPE}{ENTRY}{\the\@temptokena}
%               \@tempa
%               IF vmode and @nobreak = true  THEN  \nobreak FI
%             \endgroup
%     fi
%   END
%
% \@starttoc{EXT} : Used to define \tableofcontents, \listoffigures, etc.--
%      e.g., \@starttoc{lof} is used in \listoffigures.  This command reads
%      the .EXT file and sets up to write the new .EXT file.
%
% \@starttoc{EXT} ==
%   BEGIN
```

```
%     \begingroup
%        \makeatletter
%        read file \jobname.EXT
%        IF @filesw = true
%          THEN  open \jobname.EXT as file \tf@EXT
%        FI
%        @nobreak :=G FALSE  %% added 24 May 89
%     \endgroup
%   END

%% RmS 92/01/14: added \immediate to \openout as all \write commands
%%               are also executed \immediate
\def\@starttoc#1{\begingroup
  \makeatletter
  \@input{\jobname.#1}\if@filesw \expandafter\newwrite\csname tf@#1\endcsname
             \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
  \fi \global\@nobreakfalse \endgroup}


\let\protect=\relax

\def\addcontentsline#1#2#3{\if@filesw \begingroup
\let\label\@gobble \let\index\@gobble \let\glossary\@gobble
\def\protect##1{\string\string\string##1\string\space
   \space}\@temptokena{\thepage}%
\edef\@tempa{\write \@auxout{\string\@writefile{#1}{\protect
     \contentsline{#2}{#3}{\the\@temptokena}}}}\@tempa
   \if@nobreak \ifvmode\nobreak\fi\fi\endgroup\fi}

\long\def\addtocontents#1#2{\if@filesw \begingroup
\let\label\@gobble \let\index\@gobble \let\glossary\@gobble
\def\protect##1{\string\string\string##1\string\space\space}%
\edef\@tempa{\write \@auxout {\string\@writefile{#1}{#2}}}\@tempa
  \if@nobreak \ifvmode\nobreak\fi\fi\endgroup\fi}

\def\contentsline#1{\csname l@#1\endcsname}

% \@dottedtocline{LEVEL}{INDENT}{NUMWIDTH}{TITLE}{PAGE} :
%   Macro to produce a table of contents line with the following
%   parameters:
%     LEVEL    : If LEVEL > \c@tocdepth, then no line produced.
%     INDENT   : Total indentation from the left margin.
%     NUMWIDTH : Width of box for number if the TITLE has a
%                \numberline command.
%                As of 25 Jan 88, this is also the amount of extra indentation
%                added to second and later lines of a multiple line entry.
%     TITLE    : Contents of entry.
%     PAGE     : Page number.
%
%  Uses the following parameters, which must be set by the document style.
%  They should be defined with \def's.
%    \@pnumwidth : Width of box in which page number is set.
%    \@tocrmarg  : Right margin indentation for all but last line of
%                  multiple-line entries.
%    \@dotsep    : Separation between dots, in mu units.  Should be \def'd to
%                  a number like 2 or 1.7
%

%% RmS 91/09/29: added \reset@font for page number
\def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
  \vskip \z@ plus.2\p@
```

```
  {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
    \parindent #2\relax\@afterindenttrue
   \interlinepenalty\@M
   \leavevmode
   \@tempdima #3\relax \advance\leftskip \@tempdima \hbox{}\hskip -\leftskip
    #4\nobreak\leaders\hbox{$\m@th \mkern \@dotsep mu.\mkern \@dotsep
       mu$}\hfill \nobreak
           \hbox to\@pnumwidth{\hfil\reset@font\rm #5}\par}\fi}


%%% Note: \nobreak's added 7 Jan 86 to prevent bad line break that
%%% left the page number dangling by itself at left edge of a new line.
%%%
%%% Changed 25 Jan 88 to use \leftskip instead of \hangindent so
%%% leaders of multiple-line contents entries would line up properly.

% \numberline{NUMBER} : For use in a \contentsline command.
%   It puts NUMBER flushleft in a box of width \@tempdima
%   (Before 25 Jan 88 change, it also added \@tempdima to the hanging
%   indentation.)

\def\numberline#1{\hbox to\@tempdima{#1\hfil}}


 \message{index,}
%       ***************************************************
%       *            INDEX COMMANDS AND GLOSSARY          *
%       ***************************************************
%
% \makeindex ==
%   BEGIN
%    if \@filesw = T
%      then  open file \jobname.IDX as \@indexfile
%             \index ==  BEGIN \@bsphack
%                              \begingroup
%                                 \protect{X} == \string X\space
%                                       %% added 3 Feb 87 for \index commands
%                                       %% in \footnotes
%                                  re-\catcode special characters to 'other'
%                                  \@wrindex
%    fi
%   END
%
%  \@wrindex{ITEM} ==
%    BEGIN
%        write of {\indexentry{ITEM}{page number}}
%      \endgroup
%      \@esphack
%    END

%  INITIALIZATION:
%
%  \index == BEGIN \@bsphack
%                  \begingroup
%                     re-\catcode special characters (in case '%' there)
%                     \@index
%            END
%
%  \@index{ITEM} == BEGIN \endgroup \@esphack END
%
% Changes made 14 Apr 89 to write \glossaryentry's instead of
```

```
% \indexentry's on the .glo file.

\def\makeindex{\if@filesw \newwrite\@indexfile
  \immediate\openout\@indexfile=\jobname.idx
  \def\index{\@bsphack\begingroup
             \def\protect####1{\string####1\space}\@sanitize
             \@wrindex}\typeout
  {Writing index file \jobname.idx }\fi}

\def\@wrindex#1{\let\thepage\relax
   \edef\@tempa{\write\@indexfile{\string
      \indexentry{#1}{\thepage}}}\expandafter\endgroup\@tempa
   \if@nobreak \ifvmode\nobreak\fi\fi\@esphack}

\def\index{\@bsphack\begingroup \@sanitize\@index}

\def\@index#1{\endgroup\@esphack}

\def\makeglossary{\if@filesw \newwrite\@glossaryfile
  \immediate\openout\@glossaryfile=\jobname.glo
  \def\glossary{\@bsphack\begingroup\@sanitize\@wrglossary}\typeout
  {Writing glossary file \jobname.glo }\fi}

\def\@wrglossary#1{\let\thepage\relax
   \edef\@tempa{\write\@glossaryfile{\string
      \glossaryentry{#1}{\thepage}}}\expandafter\endgroup\@tempa
   \if@nobreak \ifvmode\nobreak\fi\fi\@esphack}

\def\glossary{\@bsphack\begingroup\@sanitize\@index}

 \message{bibliography,}
%      ****************************************
%      *            BIBLIOGRAPHY              *
%      ****************************************
%
%  A bibliography is created by the bibliography environment, which
%  generates a title such as ``References'', and a list of entries.
%  The BIBTeX program will create a file containing such an environment,
%  which will be read in by the \bibliography command.  With
%  BIBTeX, the following commands will be used.
%
%  \bibliography{FILE1,FILE2, ... ,FILEn} : specifies
%     the bibdata files.  Writes a \bibdata entry on the .aux file
%     and tries to read in mainfile.BBL.
%
%  \bibliographystyle{STYLE} : Writes a \bibstyle entry on the .aux file.
%
%  The thebibliography environment is a list environment.  To save the
%  use of an extra counter, it should use  enumiv  as the item counter.
%  Instead of using \item, items in the bibliography are produced by the
%  following commands:
%    \bibitem{NAME}             : Produces a numbered entry cited as NAME.
%    \bibitem[LABEL]{NAME}      : Produces an entry labeled by LABEL and
%                                   cited by NAME.
%  The former is used for bibliographies with citations like [1], [2], etc.;
%  the latter is used for citations like [Knuth82].
%
%  The document style must define the thebibliography environment.  This
%  environment has a single argument, which is the widest bibliography
%  label-- e.g., if the [Knuth67] is the widest entry, then thist argument
%  will be Knuth67.  The \thebibliography command must begin a list
```

```
%  environment, which the \endthebibliography command ends.
%
%  Entries are cited by the command \cite{NAME}.
%
%  PARAMETERS
%
%   \@cite     : A macro such that \@cite{LABEL1,LABEL2}{NOTE}
%                produces the output for a \cite[NOTE]{FOO1,FOO2} command,
%                where entry FOOi is defined by \bibitem[LABELi]{FOOi}.
%                The switch @tempswa is true if the optional NOTE argument
%                is present.
%                The default definition is :
%                  \@cite{LABELS}{NOTE} ==
%                     BEGIN [LABELS
%                           IF @tempswa = T THEN , NOTE FI
%                           ]
%                     END
%
%   \@biblabel : A macro to produce the label in the bibliography
%                entry.  For \bibitem[LABEL]{NAME}, the label is
%                generated by \@biblabel{LABEL}.  It has the default
%                definition \@biblabel{LABEL} -> [LABEL].
%  CONVENTION
%
%  \b@FOO : The name or number of the reference created by \cite{FOO}
%           E.g., if \cite{FOO} -> [17] , then \b@FOO -> 17.
%
%

\def\bibitem{\@ifnextchar[{\@lbibitem}{\@bibitem}}

%% RmS 92/02/26: Added \hfill to restore left-alignment of
%%               bibliography labels in alpha style
\def\@lbibitem[#1]#2{\item[\@biblabel{#1}\hfill]\if@filesw
      {\def\protect##1{\string ##1\space}\immediate
       \write\@auxout{\string\bibcite{#2}{#1}}}\fi\ignorespaces}
%% Placement of `}' in def of \@lbibitem corrected 29 Apr 87
%% (Error found by Arthur Ogawa.)

%% RmS 91/11/13: Changed counter enumi to enumiv,
%%               as it says in the comment above
%% RmS 92/01/10: Changed \c@enumiv to \value{\@listctr}.
\def\@bibitem#1{\item\if@filesw \immediate\write\@auxout
       {\string\bibcite{#1}{\the\value{\@listctr}}}\fi\ignorespaces}

\def\bibcite#1#2{\global\@namedef{b@#1}{#2}}

\let\citation\@gobble

\def\cite{\@ifnextchar [{\@tempswatrue\@citex}{\@tempswafalse\@citex[]}}

% \penalty\@m added to definition of \@citex to allow a line
% break after the `,' in citations like [Jones80,Smith77]
% (Added 23 Oct 86)
%
% space added after the `,' (21 Nov 87)
%
%% RmS 91/10/25: added \reset@font, suggested by Bernd Raichle.
%% RmS 91/11/06: added code to remove a leading blank
\def\@citex[#1]#2{\if@filesw\immediate\write\@auxout{\string\citation{#2}}\fi
  \let\@citea\@empty
```

```
  \@cite{\@for\@citeb:=#2\do
    {\@citea\def\@citea{,\penalty\@m\ }%
     \def\@tempa##1##2\@nil{\edef\@citeb{\if##1\space##2\else##1##2\fi}}%
     \expandafter\@tempa\@citeb\@nil
     \@ifundefined{b@\@citeb}{{\reset@font\bf ?}\@warning
       {Citation `\@citeb' on page \thepage \space undefined}}%
     \hbox{\csname b@\@citeb\endcsname}}}{#1}}

\let\bibdata=\@gobble
\let\bibstyle=\@gobble

\def\bibliography#1{\if@filesw\immediate\write\@auxout{\string\bibdata{#1}}\fi
  \@input{\jobname.bbl}}

\def\bibliographystyle#1{\if@filesw\immediate\write\@auxout
    {\string\bibstyle{#1}}\fi}

% \nocite{CITATIONS} : puts information on .AUX file to cause
%   BibTeX to include the CITATIONS list in the bibliography,
%   but puts nothing in the text.  (Added 14 Jun 85)

\def\nocite#1{\@bsphack
  \if@filesw\immediate\write\@auxout{\string\citation{#1}}\fi
  \@esphack}



%DEFAULT DEFINITIONS

\def\@cite#1#2{[{#1\if@tempswa , #2\fi}]}
%% RmS 92/01/14: removed \hfill in definition of \@biblabel
\def\@biblabel#1{[#1]}

 \message{floats,}
%     ****************************************
%     *               FLOATS                 *
%     ****************************************
%
%  The different types of floats are identified by a TYPE name, which is
%  the name of the counter for that kind of float.  For example, figures
%  are of type 'figure' and tables are of type 'table'.  Each TYPE has
%  associated a positive TYPE NUMBER, which is a power of two.  E.g.,
%  figures might be have type number 1, tables type number 2, programs
%  type number 4, etc.
%
%  The locations where a float can go are specified by a PLACEMENT
%  SPECIFIER, which is a list of the possible locations, each denoted
%  by a letter as follows:
%     h : here   - at the current location in the text.
%     t : top    - at the top of a text page.
%     b : bottom - at the bottom of a text page.
%     p : page   - on a separate float page.
%  For example, 'pht' specifies that the float can appear in any of three
%  locations: page, here or top.
%
% Where floats may appear on a page, and how many may appear there
% are specified by the following float placement parameters.  The
% numbers are named like counters so the user can set them with
% the ordinary counter-setting commands.
%
%  \c@topnumber            : Number of floats allowed at the top of a column.
```

```
%  \topfraction            : Fraction of column that can be devoted to floats.
%  \c@dbltopnumber, \dbltopfraction : Same as above, but for double-column
%                           floats.
%  \c@bottomnumber, \bottomfraction : Same as above for bottom of page.
%  \c@totalnumber          : Number of floats allowed in a single column,
%                           including in-text floats.
%  \textfraction         : Minimum fraction of column that must contain text.
%  \floatpagefraction    : Minimum fraction of page that must be taken
%                           up by float page.
%  \dblfloatpagefraction : Same as above, for double-column floats.
%
% The document style must define the following.
%
%    \fps@TYPE   : The default placement specifier for floats of type TYPE.
%
%    \ftype@TYPE : The type number for floats of type TYPE.
%
%    \ext@TYPE   : The file extension indicating the file on which the
%                  contents list for float type TYPE is stored.  For example,
%                  \ext@figure = 'lof'.
%
%    \fnum@TYPE  : A macro to generate the figure number for a caption.
%                  For example, \fnum@TYPE == Figure \thefigure.
%
%    \@makecaption{NUM}{TEXT} : A macro to make a caption, with NUM the value
%                  produced by \fnum@... and TEXT the text of the caption.
%                  It can assume it's in a \parbox of the appropriate width.
%
% \@float{TYPE}[PLACEMENT] : This macro begins a float environment for a
%     single-column float of type TYPE with PLACEMENT as the placement
%     specifier.  The default value of PLACEMENT is defined by \fps@TYPE.
%     The environment is ended by \end@float.
%     E.g., \figure == \@float{figure}, \endfigure == \end@float.
%
%  \caption ==
%    BEGIN
%     \refstepcounter{\@captype}
%     \@dblarg{\@caption{\@captype}}
%    END
%
%% In following definition, \par moved from after \addcontentsline to
%% before \addcontentsline because the \write could cause
%% an extra blank line to be added to the paragraph above the
%% caption.  (Change made 12 Jun 87)
%
%  \@caption{TYPE}[STEXT]{TEXT} ==
%   BEGIN
%     \par
%     \addcontentsline{\ext@TYPE}{TYPE}{\numberline{\theTYPE}{STEXT}}
%     \begingroup
%       \@parboxrestore
%       \normalsize
%       \@makecaption{\fnum@TYPE}{TEXT}
%       \par
%     \endgroup
%   END
%
%  \@float{TYPE}[PLACEMENT] ==
%   BEGIN
%     if hmode then \@bsphack
%                   \@floatpenalty := -10002
```

```
%              else \@floatpenalty := -10003
%     fi
%     \@captype ==L TYPE
%     if inner
%       then LaTeX Error: 'Not in outer paragraph mode.'
%            \@floatpenalty := 0
%       else if \@freelist nonempty
%              then \@currbox  :=L head of \@freelist
%                   \@freelist :=G tail of \@freelist
%                   \count\@currbox :=G 32*\ftype@TYPE + 16 +
%                                       bits determined by PLACEMENT
%              else \@floatpenalty := 0
%                   LaTeX Error: 'Too many unprocessed floats'
%            fi
%     fi
%     \@currbox :=G \vbox{  %% 15 Dec 87 -- removed \boxmaxdepth :=L 0pt
%                           %% that made box zero depth because it screwed
%                           %% things up.  Instead, added \vskip 0pt at end
%                     \hsize = \columnwidth
%                     \@parboxrestore
%   END
%
%  \end@float ==
%    BEGIN
%       \vskip 0pt %% makes 0 depth box -- added 15 Dec 87
%       }
%      if \@floatpenalty < 0
%        then add \@currbox to \@currlist
%             if \ht\@currbox > \textheight
%               then \ht\@currbox :=G \textheight fi
%             if \@floatpenalty < -10002
%               then \penalty -10004
%                    \vbox{}
%                    \penalty \@floatpenalty
%               else \vadjust{\penalty -10004
%                             \vbox{}
%                             \penalty \@floatpenalty}
%                    \@Esphack
%      fi    fi
%    END
%
%  \@dblfloat{TYPE}[PLACEMENT] : Macro to begin a float environment for a
%     double-column float of type TYPE with PLACEMENT as the placement
%     specifier.  The default value of PLACEMENT is 'tp'
%     The environment is ended by \end@dblfloat.
%     E.g., \figure* == \@dblfloat{figure}, \endfigure* == \end@dblfloat.
%
%  \@dblfloat{TYPE}[PLACEMENT] ==
%     Identical to \@float{TYPE}[PLACEMENT] except \hsize and \linewidth
%     are set to \textwidth.
%
%  \end@dblfloat ==
%     BEGIN    %%% { BRACE MATCHING
%       \vskip 0pt %% makes 0 depth box -- added 15 Dec 87
%       }
%       if \@floatpenalty < 0
%         then \@dbldeferlist :=G \@dbldeferlist * \@currbox
%       fi
%       if \@floatpenalty = -10002 then \@Esphack  fi
%     END
%
```

```
\newcount\@floatpenalty

\def\caption{\refstepcounter\@captype \@dblarg{\@caption\@captype}}


\long\def\@caption#1[#2]#3{\par\addcontentsline{\csname
  ext@#1\endcsname}{#1}{\protect\numberline{\csname
  the#1\endcsname}{\ignorespaces #2}}\begingroup
    \@parboxrestore
    \normalsize
    \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
  \endgroup}

\def\@float#1{\@ifnextchar[{\@xfloat{#1}}{\edef\@tempa{\noexpand\@xfloat
    {#1}[\csname fps@#1\endcsname]}\@tempa}}

\def\@xfloat#1[#2]{\ifhmode \@bsphack\@floatpenalty -\@Mii\else
   \@floatpenalty-\@Miii\fi\def\@captype{#1}\ifinner
      \@parmoderr\@floatpenalty\z@
    \else\@next\@currbox\@freelist{\@tempcnta\csname ftype@#1\endcsname
       \multiply\@tempcnta\@xxxii\advance\@tempcnta\sixt@@n
       \@tfor \@tempa :=#2\do
                        {\if\@tempa h\advance\@tempcnta \@ne\fi
                         \if\@tempa t\advance\@tempcnta \tw@\fi
                         \if\@tempa b\advance\@tempcnta 4\relax\fi
                         \if\@tempa p\advance\@tempcnta 8\relax\fi
         }\global\count\@currbox\@tempcnta}\@fltovf\fi
    \global\setbox\@currbox\vbox\bgroup
%    \boxmaxdepth\z@     % commented out 15 Dec 87
    \hsize\columnwidth \@parboxrestore}

\def\end@float{\par\vskip\z@\egroup %% \par\vskip\z@ added 15 Dec 87
   \ifnum\@floatpenalty <\z@
     \@cons\@currlist\@currbox
     \ifdim \ht\@currbox >\textheight
%% RmS 91/11/06 added warning message
% perhaps we should use an error message
        \@warning{Float larger than \string\textheight}%
        \ht\@currbox\textheight \fi
     \ifnum\@floatpenalty <-\@Mii
        \penalty -\@Miv
        \@tempdima\prevdepth    %% saving and restoring \prevdepth added
        \vbox{}%                %% 26 May 87 to prevent extra vertical
        \prevdepth \@tempdima   %% space when used in vertical mode
        \penalty\@floatpenalty
%% RmS 92/03/18 changed \@esphack to \@Esphack
      \else \vadjust{\penalty -\@Miv \vbox{}\penalty\@floatpenalty}\@Esphack
     \fi\fi}


\def\@dblfloat{\if@twocolumn\let\@tempa\@dbflt\else\let\@tempa\@float\fi
  \@tempa}

\def\@dbflt#1{\@ifnextchar[{\@xdblfloat{#1}}{\@xdblfloat{#1}[tp]}}

\def\@xdblfloat#1[#2]{\@xfloat{#1}[#2]\hsize\textwidth\linewidth\textwidth}

\def\end@dblfloat{\if@twocolumn
     \par\vskip\z@\egroup %% \par\vskip\z@ added 15 Dec 87\egroup
     \ifnum\@floatpenalty <\z@
% make sure that we never exceed \textheight, otherwise float
```

```
% will never get typeset =91/03/15 FMi=
       \ifdim\ht\@currbox >\textheight
% perhaps we should use an error message
         \@warning{Float larger than \string\textheight}%
         \ht\@currbox\textheight \fi
        \@cons\@dbldeferlist\@currbox\fi
%% RmS 92/03/18 changed \@esphack to \@Esphack
      \ifnum \@floatpenalty =-\@Mii \@Esphack\fi\else\end@float\fi}

\newcount\c@topnumber
\newcount\c@dbltopnumber
\newcount\c@bottomnumber
\newcount\c@totalnumber

\def\@floatplacement{\global\@topnum\c@topnumber
   \global\@toproom \topfraction\@colht
   \global\@botnum  \c@bottomnumber
   \global\@botroom \bottomfraction\@colht
   \global\@colnum  \c@totalnumber
   \@fpmin   \floatpagefraction\@colht}

\def\@dblfloatplacement{\global\@dbltopnum\c@dbltopnumber
   \global\@dbltoproom \dbltopfraction\@colht
   \@fpmin \dblfloatpagefraction\textheight
   \@fptop \@dblfptop
   \@fpsep \@dblfpsep
   \@fpbot \@dblfpbot}

%   MARGINAL NOTES:
%
%   Marginal notes use the same mechanism as floats to communicate
%   with the \output routine.  Marginal notes are distinguished from
%   floats by having a negative placement specification.  The command
%   \marginpar [LTEXT]{RTEXT} generates a marginal note in a parbox,
%   using LTEXT if it's on the left and RTEXT if it's on the right.
%   (Default is RTEXT = LTEXT.)  It uses the following parameters.
%
%   \marginparwidth : Width of marginal notes.
%   \marginparsep   : Distance between marginal note and text.
%        the page layout to determine how to move the marginal
%        note into the margin.   E.g., \@leftmarginskip ==
%        \hskip -\marginparwidth \hskip -\marginparsep .
%   \marginparpush  :  Minimum vertical separation between \marginpar's
%
%  Marginal notes are normally put on the outside of the page
%  if @mparswitch = true, and on the right if @mparswitch = false.
%  The command \reversemarginpar reverses the side where they
%  are put.  \normalmarginpar undoes \reversemarginpar.
%  These commands have no effect for two-column output.
%
%  SURPRISE: if two marginal notes appear on the same line of
%  text, then the second one could appear on the next page, in
%  a funny position.
%
%
%  \marginpar [LTEXT]{RTEXT} ==
%   BEGIN
%     if hmode then \@bsphack
%                   \@floatpenalty := -10002
%              else \@floatpenalty := -10003
%     fi
```

```
%     if inner
%       then LaTeX Error: 'Not in outer paragraph mode.'
%            \@floatpenalty := 0
%       else if \@freelist has two elements:
%              then get \@marbox, \@currbox  from \@freelist
%                   \count\@marbox :=G -1
%              else \@floatpenalty := 0
%                   LaTeX Error: 'Too many unprocessed floats'
%                   \@currbox, \@marbox := \@tempboxa    %%use \def
%            fi
%     fi
%     if optional argument
%       then %% \@xmpar ==
%            \@savemarbox\@marbox{LTEXT}
%            \@savemarbox\@currbox{RTEXT}
%       else %% \@ympar ==
%            \@savemarbox\@marbox{RTEXT}
%            \box\@currbox :=G \box\@marbox
%    fi
%    %% \@xympar ==
%    if \@floatpenalty < 0 then add \@marbox to \@currlist fi
%    \setbox\@tempboxa =L     %% added 3 Jan 88 to correct bug introduced
%       { \end@float %%%% BRACE MATCHING}        %% by 15 Dec 87 change
%   END
%
% \@savemarbox\BOX{TEXT} ==
%   BEGIN
%     \BOX  :=G \vtop{ \hsize = \marginparwidth
%                      \@parboxrestore
%                      TEXT
%                    }
%   END
%
% \reversemarginpar == BEGIN \@mparbottom   :=G 0
%                            @reversemargin :=G true
%                      END
%
% \normalmarginpar  == BEGIN \@mparbottom   :=G 0
%                            @reversemargin :=G false
%                      END
%

\def\marginpar{\ifhmode \@bsphack\@floatpenalty -\@Mii\else
   \@floatpenalty-\@Miii\fi\ifinner
      \@parmoderr\@floatpenalty\z@
    \else\@next\@currbox\@freelist{}{}\@next\@marbox\@freelist{\global
      \count\@marbox\m@ne}{\@floatpenalty\z@ \@fltovf\def\@currbox{\@tempboxa
           }\def\@marbox{\@tempboxa}}\fi
     \@ifnextchar [{\@xmpar}{\@ympar}}

\long\def\@xmpar[#1]#2{\@savemarbox\@marbox{#1}\@savemarbox\@currbox
   {#2}\@xympar}

\long\def\@ympar#1{\@savemarbox\@marbox{#1}\global\setbox\@currbox
     \copy\@marbox\@xympar}

\long\def\@savemarbox#1#2{\global\setbox#1\vtop{\hsize\marginparwidth
   \@parboxrestore #2}}

\def\@xympar{\ifnum\@floatpenalty <\z@\@cons\@currlist\@marbox\fi
     \setbox\@tempboxa\vbox   %% added 3 Jan 88
```

```
%% RmS 92/03/18 added \global\@ignorefalse
     \bgroup\end@float\global\@ignorefalse\@esphack}

\def\reversemarginpar{\global\@mparbottom\z@ \@reversemargintrue}
\def\normalmarginpar{\global\@mparbottom\z@ \@reversemarginfalse}


 \message{footnotes,}
%       ***************************************
%       *             FOOTNOTES               *
%       ***************************************
%
%   \footnote{NOTE}       : User command to insert a footnote.
%
%   \footnote[NUM]{NOTE}  : User command to insert a footnote numbered
%                           NUM, where NUM is a number -- 1, 2,
%                           etc.  For example, if footnotes are numbered
%                           *, **, etc. within pages, then \footnote[2]{...}
%                           produces footnote '**'.  This command does not
%                           step the footnote counter.
%
%   \footnotemark[NUM]    : Command to produce just the footnote mark in
%                           the text, but no footnote.  With no argument,
%                           it steps the footnote counter before generating
%                           the mark.
%
%   \footnotetext[NUM]{TEXT} : Command to produce the footnote but no
%                              mark.  \footnote is equivalent to
%                              \footnotemark \footnotetext .
%
%   As in PLAIN, footnotes use \insert\footins, and the following parameters:
%
%   \footnotesize   : Size-changing command for footnotes.
%
%   \footnotesep    : The height of a strut placed at the beginning of
%                     every footnote.
%   \skip\footins   : Space between main text and footnotes.  The rule
%                     separating footnotes from text occurs in this space.
%                     This space lies above the strut of height \footnotesep
%                     which is at the beginning of the first footnote.
%   \footnoterule   : Macro to draw the rule separating footnotes from text.
%                     It is executed right after a \vspace of \skip\footins.
%                     It should take zero vertical space--i.e., it should to
%                     a negative skip to compensate for any positive space
%                     it occupies.  (See PLAIN.TEX.)
%
%   \interfootnotelinepenalty : Interline penalty for footnotes.
%
%   \thefootnote : In usual LaTeX style, produces the footnote number.
%                  If footnotes are to be numbered within pages, then the
%                  document style file must include an \@addtoreset command
%                  to cause the footnote counter to be reset when the page
%                  counter is stepped.  This is not a good idea, though,
%                  because the counter will not always be reset in time
%                  to ensure that the first footnote on a page is footnote
%                  number one.
%
%   \@thefnmark : Holds the current footnote's mark--e.g., \dag or '1' or 'a'.
%
%   \@mpfnnumber     : A macro that generates the numbers for \footnote
%                      and \footnotemark commands. It == \thefootnote
```

```
% outside a minipage environment, but can be changed
% inside to generate numbers for \footnote's.
%
%   \@makefnmark : A macro to generate the footnote marker from \@thefnmark
%                  The default definition is \hbox{$^\@thefnmark$}.
%
%   \@makefntext{NOTE} :
%        Must produce the actual footnote, using \@thefnmark as the mark
%        of the footnote and NOTE as the text.  It is called when effectively
%        inside a \parbox, with \hsize = \columnwidth.  For example, it might
%        be as simple as
%               $^{\@thefnmark}$ NOTE
%
% In a minipage environment, \footnote and \footnotetext are redefined
% so that
%    (a) they use the counter mpfootnote
%    (b) the footnotes they produce go at the bottom of the minipage.
% The switch is accomplished by letting \@mpfn == footnote or mpfootnote
% and \thempfn == \thefootnote or \thempfootnote, and by redefining
% \@footnotetext to be \@mpfootnotetext in the minipage.
%
% \footnote{NOTE}  ==
%  BEGIN
%    \stepcounter{\@mpfn}
%    \@thefnmark :=G eval (\thempfn)
%    \@footnotemark
%    \@footnotetext{NOTE}
%  END
%
% \footnote[NUM]{NOTE} ==
%  BEGIN
%    begingroup
%       counter \@mpfn :=L NUM
%       \@thefnmark :=G eval (\thempfn)
%    endgroup
%    \@footnotemark
%    \@footnotetext{NOTE}
%  END
%
% \@footnotetext{NOTE} ==
%  BEGIN
%    \insert into \footins
%       {\footnotesize
%        \interlinepenalty :=L \interfootnotelinepenalty
%        \splittopskip     :=L \footnotesep
%        \splitmaxdepth    :=L \dp\strutbox
%        \floatingpenalty  :=L 20000
%        \hsize :=L \columnwidth
%        \@parboxrestore
%        set \@currentlabel to make \label command work right
%        \@makefntext{\rule{0pt}{\footnotesep} NOTE}
%       }
%  END
%
% \footnotemark      ==
%  BEGIN \stepcounter{footnote}
%        \@thefnmark :=G eval(\thefootnote)
%        \@footnotemark
%  END
%
% \footnotemark[NUM] ==
```

```
%   BEGIN
%       begingroup
%         footnote counter :=L NUM
%        \@thefnmark :=G eval(\thefootnote)
%       endgroup
%       \@footnotemark
%   END
%
% \@footnotemark ==
%   BEGIN
%    \leavevmode
%    IF hmode THEN \@x@sf := \the\spacefactor FI
%    \@makefnmark          % put number in main text
%    IF hmode THEN \spacefactor := \@x@sf FI
%   END
%
% \footnotetext      ==
%    BEGIN \@thefnmark :=G eval (\thempfn)
%          \@footnotetext
%    END
%
% \footnotetext[NUM] ==
%    BEGIN begingroup  counter \@mpfn :=L NUM
%                      \@thefnmark :=G eval (\thempfn)
%          endgroup
%          \@footnotetext
%    END
%

\@definecounter{footnote}
\def\thefootnote{\arabic{footnote}}

\@definecounter{mpfootnote}
\def\thempfootnote{\alph{mpfootnote}}

% Default definition
\def\@makefnmark{\hbox{$^{\@thefnmark}\m@th$}}

\newdimen\footnotesep

%% RmS 91/11/01: Added \let\protect\noexpand in \footnote, \footnotemark,
%%               and \footnotetext, since \xdef is used.
%% RmS 91/11/22: Added \let\protect\noexpand in \@xfootnote, \@xfootnotemark,
%%               and \@xfootnotetext.

\def\footnote{\@ifnextchar[{\@xfootnote}{\stepcounter{\@mpfn}%
     \begingroup\let\protect\noexpand
       \xdef\@thefnmark{\thempfn}\endgroup
     \@footnotemark\@footnotetext}}

\def\@xfootnote[#1]{\begingroup \csname c@\@mpfn\endcsname #1\relax
   \let\protect\noexpand
   \xdef\@thefnmark{\thempfn}\endgroup
   \@footnotemark\@footnotetext}

%% RmS 91/09/29: added \reset@font
\long\def\@footnotetext#1{\insert\footins{\reset@font\footnotesize
    \interlinepenalty\interfootnotelinepenalty
    \splittopskip\footnotesep
    \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
    \hsize\columnwidth \@parboxrestore
```

```
    \edef\@currentlabel{\csname p@footnote\endcsname\@thefnmark}\@makefntext
     {\rule{\z@}{\footnotesep}\ignorespaces
       #1\strut}}}

\def\footnotemark{\@ifnextchar[{\@xfootnotemark}{\stepcounter{footnote}%
      \begingroup\let\protect\noexpand
        \xdef\@thefnmark{\thefootnote}\endgroup
      \@footnotemark}}

\def\@xfootnotemark[#1]{\begingroup \c@footnote #1\relax
    \let\protect\noexpand
    \xdef\@thefnmark{\thefootnote}\endgroup \@footnotemark}

\def\@footnotemark{\leavevmode\ifhmode
   \edef\@x@sf{\the\spacefactor}\fi \@makefnmark
    \ifhmode\spacefactor\@x@sf\fi\relax}

\def\footnotetext{\@ifnextchar [{\@xfootnotenext}%
    {\begingroup\let\protect\noexpand
       \xdef\@thefnmark{\thempfn}\endgroup
     \@footnotetext}}

\def\@xfootnotenext[#1]{\begingroup \csname c@\@mpfn\endcsname #1\relax
    \let\protect\noexpand
    \xdef\@thefnmark{\thempfn}\endgroup \@footnotetext}

\def\@mpfn{footnote}
\def\thempfn{\thefootnote}

 \message{initial,}
%          ****************************************
%          *    INITIAL DECLARATION COMMANDS      *
%          ****************************************
%
%                    DOCUMENT STYLE
%                    --------------
%
% The user starts his file with the command
%       \documentstyle [OPTION1, ... ,OPTIONk]{STYLE}
% which saves the OPTION's and \input's the file STYLE.STY.  When the
% STYLE.STY file issues the command \@options, the following happens
% for each i :
%     IF \ds@OPTIONi is defined
%       THEN execute \ds@OPTIONi
%       ELSE save OPTIONi on a list of unprocessed options.
%     FI
% After STYLE.STY has been executed, the file OPTIONi.STY is read for
% each OPTIONi on the list of unprocessed options.
%
% \documentstyle ==
%   BEGIN
%      IF next char = [
%        THEN  \@documentstyle
%        ELSE  \@documentstyle[]
%      FI
%   END
%
% \@documentstyle[OPTIONS]{STYLE} ==
%   BEGIN
%     \makeatletter
%     \@optionlist  :=  OPTIONS
```

```
%     \@optionfiles :=G null
%     \input STYLE.STY
%     \@elt == \input
%     \@optionfiles
%     \@elt == \relax
%     \makeatother
%   END
%
%  \@options ==
%    BEGIN
%      \@elt := \relax
%      FOR \@tempa := \@optionlist
%         DO  IF \ds@[eval(\@tempa)] defined
%               THEN  \ds@[eval(\@tempa)]
%               ELSE  \@optionfiles :=G \@optionfiles *
%                                          \@elt eval(\@tempa) \relax
%         OD  FI
%    END
%
%              PAGE STYLE COMMANDS
%              -------------------
%  \pagestyle{STYLE}     : sets the page style of the current and succeeding
%                          pages to STYLE
%
%  \thispagestyle{STYLE} : sets the page style of the current page only
%                          to STYLE
%
%  To define a page style STYLE, you must define \ps@STYLE to set the page
%  style parameters.
%
%  HOW A PAGE STYLE MAKES RUNNING HEADS AND FEET:
%
% The \ps@... command defines the macros \@oddhead, \@oddfoot,
% \@evenhead, and \@evenfoot to define the running heads and feet.
% (See output routine.)  To make headings determined by the sectioning
% commands, the page style defines the commands \chaptermark,
% \sectionmark, etc., where \chaptermark{TEXT} is called by \chapter to
% set a mark.  The \...mark commands and the \...head macros are defined
% with the help of the following macros.  (All the \...mark commands
% should be initialized to no-ops.)
%
% MARKING CONVENTIONS:
% LaTeX extends TeX's \mark facility by producing two kinds of marks
% a 'left' and a 'right' mark, using the following commands:
%     \markboth{LEFT}{RIGHT} : Adds both marks.
%     \markright{RIGHT}      : Adds a 'right' mark.
%     \leftmark  : Used in the output routine, gets the current 'left'  mark.
%                  Works like TeX's \botmark.
%     \rightmark : Used in the output routine, gets the current 'right' mark.
%                  Works like TeX's \firstmark.
% The marking commands work reasonably well for right marks 'numbered
% within' left marks--e.g., the left mark is changed by a \chapter command and
% the right mark is changed by a \section command.  However, it does
% produce somewhat anomalous results if 2 \markboth's occur on the same page.
%
% Commands like \tableofcontents that should set the marks in some page styles
% use a \@mkboth command, which is \let by the pagestyle command (\ps@...)
% to \markboth for setting the heading or to \@gobbletwo to do nothing.

\def\documentstyle{\@ifnextchar[{\@documentstyle}{\@documentstyle[]}}

```

```
\def\@documentstyle[#1]#2{\makeatletter
   \def\@optionlist{#1}\gdef\@optionfiles{}\input #2.sty\relax
   \let\@elt\input \@optionfiles \let\@elt\relax \makeatother}

\def\@options{\let\@elt\relax
    \@for\@tempa:=\@optionlist\do
        {\@ifundefined{ds@\@tempa}{\xdef\@optionfiles{\@optionfiles
             \@elt \@tempa.sty\relax}}{\csname ds@\@tempa\endcsname}}}

\def\pagestyle#1{\@nameuse{ps@#1}}
\def\thispagestyle#1{\global\@specialpagetrue\gdef\@specialstyle{#1}}

% \head : An obsolete command that was used in the 'myheadings'
%         page style.   (Removed 14 Jun 85)
% \def\head{\@ifnextchar[{\@xhead}{\@yhead}}
% \def\@xhead[#1]#2{\if #1l \def\@lhead{#2}\else \def\@rhead{#2}\fi}
% \def\@yhead#1{\def\@lhead{#1}\def\@rhead{#1}}

% Initialization
%
%\def\@lhead{} %% RmS 91/09/29: removed since no longer used
%\def\@rhead{} %%  ditto


% Default Initializations
%
\def\ps@empty{\let\@mkboth\@gobbletwo\let\@oddhead\@empty\let\@oddfoot\@empty
\let\@evenhead\@empty\let\@evenfoot\@empty}

\def\ps@plain{\let\@mkboth\@gobbletwo
     \let\@oddhead\@empty\def\@oddfoot{\reset@font\rm\hfil\thepage
     \hfil}\let\@evenhead\@empty\let\@evenfoot\@oddfoot}

\def\@leftmark#1#2{#1}
\def\@rightmark#1#2{#2}

%% test for @nobreak added 15 Apr 86 in \markboth and \markright
%% letting \label and \index to \relax added 22 Feb 86 so these
%%   commands can appear in sectioning command arguments
%% RmS 91/06/21 Same for \glossary
%%

\def\markboth#1#2{\gdef\@themark{{#1}{#2}}{\let\protect\noexpand
     \let\label\relax \let\index\relax \let\glossary\relax
     \mark{\@themark}}\if@nobreak\ifvmode\nobreak\fi\fi}
\def\markright#1{{\let\protect\noexpand
     \let\label\relax \let\index\relax \let\glossary\relax
     \expandafter\@markright\@themark
     {#1}\mark{\@themark}}\if@nobreak\ifvmode\nobreak\fi\fi}

\def\@markright#1#2#3{\gdef\@themark{{#1}{#3}}}
\def\leftmark{\expandafter\@leftmark\botmark}
\def\rightmark{\expandafter\@rightmark\firstmark}

% Initialization
%
\def\@themark{{}{}}


%  OTHER
%  -----
```

```
%
%   \raggedbottom : Typesets pages with no vertical stretch, so they have
%                   their natural height instead of all being exactly the
%                   same height.  (Uses a space of .0001fil to avoid
%                   interfering with the 1fil space of \newpage.)
%
%   \flushbottom  : Inverse of \raggedbottom - makes all pages the same
%                   height.
%
%   \sloppy : Resets TeX's parameters so it accepts worse line and page
%             breaks, and slightly more overfull boxes.
%
%   \fussy  : Resets TeX's parameters to their normal finnicky values.
%

\def\raggedbottom{\def\@textbottom{\vskip \z@ plus.0001fil}\let\@texttop\relax}
\def\flushbottom{\let\@textbottom\relax \let\@texttop\relax}

% Default definitions
%  \sloppy will never (well, hardly ever) produce overfull boxes, but may
%  produce underfull ones.  (14 June 85)
%  A sloppypar environment is equivalent to {\par \sloppy ... \par}.
\def\sloppy{\tolerance \@M \hfuzz .5\p@ \vfuzz .5\p@}
\def\sloppypar{\par\sloppy}
\def\endsloppypar{\par}
\def\fussy{\tolerance 200 \hfuzz .1\p@ \vfuzz .1\p@}



% LaTeX default is no overfull box rule.  Changed by document
% style option

\overfullrule 0pt

 \message{output,}
%     ****************************************
%     *               OUTPUT                 *
%     ****************************************
%
%
%  PAGE LAYOUT PARAMETERS
%
%   \topmargin      : Extra space added to top of page.
%   @twoside        : boolean.  T if two-sided printing
%   \oddsidemargin  : IF @twoside = T
%                         THEN extra space added to left of odd-numbered
%                              pages.
%                         ELSE extra space added to left of all pages.
%   \evensidemargin : IF @twoside = T
%                         THEN extra space added to left of even-numbered
%                              pages.
%   \headheight     : height of head
%   \headsep        : separation between head and text
%   \footskip       : distance separation between baseline of last
%                     line of text and baseline of foot.
%                     Note difference between \footSKIP and \headSEP.
%   \textheight     : height of text on page, excluding head and foot
%   \textwidth      : width of printing on page
%   \columnsep      : IF @twocolumn = T
%                        THEN width of space between columns
%   \columnseprule  : IF @twocolumn = T
```

```
%                       THEN width of rule between columns (0 if none).
%   \columnwidth    : IF @twocolumn = T
%                       THEN (\textwidth - \columnsep)/2
%                       ELSE \textwidth
%                     It is set by the \@maketwocolumn and \@makeonecolumn
%                     commands.
%   \@textbottom    : Command executed at bottom of vbox holding text of page
%                     (including figures).  The \raggedbottom command
%                     almost \let's this to \vfil (actually sets it to
%                     \vskip \z@ plus.0001fil). %expanded 18 Jun 86
%
%   \@texttop       : Command executed at top of vbox holding text of page
%                     (including figures).  Used by letter style; can also
%                     be used to produce centered pages.  Is \let to \relax
%                     by \raggedbottom and \flushbottom.
%
%   Page layout must also initialize \@colht and \@colroom to \textheight.
%
%  PAGE STYLE PARAMETERS:
%
%   \floatsep       : Space left between floats.
%   \textfloatsep   : Space between last top float or first bottom float
%                     and the text.
%   \topfigrule     : Command to place rule (or whatever) between floats
%                     at top of page and text.  Executed in inner vertical
%                     mode right before the \textfloatsep skip separating
%                     the floats from the text.  Must occupy zero vertical
%                     space.  (See \footnoterule.)
%   \botfigrule     : Same as \topfigrule, but put after the \textfloatsep
%                     skip separating text from the floats at bottom of page.
%   \intextsep      : Space left on top and bottom of an in-text float.
%   \@maxsep        : The maximum of \floatsep, \textfloatsep and \intextsep
%   \dblfloatsep    : Space between double-column floats.
%   \dbltextfloatsep : Space between top or bottom double-column floats
%                      and text.
%   \dblfigrule     : Similar to \topfigrule, but for double-column floats.
%   \@dblmaxsep     : The maximum of \dblfloatsep and \dbltexfloatsep
%   \@fptop         : Glue to go at top of float column -- must be 0pt +
%                     stretch
%   \@fpsep         : Glue to go between floats in a float column.
%   \@fpbot         : Glue to go at bottom of float column -- must be 0pt +
%                     stretch
%   \@dblfptop, \@dblfpsep, \@dblfpbot
%                   : Analogous for double-column float page in two-column
%                     format.
%
%  FOOTNOTES: As in PLAIN, footnotes use \insert\footins.
%
%  PAGE LAYOUT SWITCHES AND MACROS
%
%   @twocolumn      : Boolean.  T if two columns per page.
%
%  PAGE STYLE MACROS AND SWITCHES
%
%   \@oddhead        : IF @twoside = T
%                            THEN macro to generate head of odd-numbered pages.
%                            ELSE macro to generate head of all pages.
%   \@evenhead       : IF @twoside = T
%                            THEN macro to generate head of even-numbered pages.
%   \@oddfoot        : IF @twoside = T
%                            THEN macro to generate foot of odd-numbered pages.
```

```
%                           ELSE macro to generate foot of all pages.
%   \@evenfoot       : IF @twoside = T
%                           THEN macro to generate foot of even-numbered pages.
%   @specialpage    : boolean.  T if current page is to have a special format.
%   \@specialstyle  : If its value is  foo then
%                       IF @specialpage = T
%                         THEN the command \ps@foo is executed to temporarily
%                              reset the page style parameters before composing
%                              the current page.  This command should execute
%                              only \def's and \edef's, making only local
%                              definitions.
%
%  FLOAT PLACEMENT PARAMETERS
%
% The following parameters are set by the macro \@floatplacement.
% When \@floatplacement is called,
% \@colht is the height of the page or column being built.  I.e.:
%        * For single-column page it equals \textheight.
%        * For double-column page it equals \textheight - height
%          of double-column floats on page.
% Note that some are set globally and some locally:
%    \@topnum  :=G Maximum number of floats allowed on the top of a column.
%    \@toproom :=G Maximum amount of top of column devoted to floats--
%                  excluding \textfloatsep separation below the floats and
%                  \floatsep separation between them.  For two-column
%                  output, should be computed as a function of \@colht.
%    \@botnum, \@botroom
%                : Analogous to above.
%    \@colnum  :=G Maximum number of floats allowed in a column, including
%                  in-text floats.
%    \@textmin :=L Minimum amount of text (excluding footnotes) that must
%                  appear on a text page.  %% 27 Sep 85 : made local to
%                                          %% \@addtocurcol and \@addtonextcol
%    \@fpmin   :=L Minimum height of floats in a float column.
%
% The macro \@dblfloatplacement sets the following parameters.
%    \@dbltopnum  :=G Maximum number of double-column floats allowed at the
%                     top of a two-column page.
%    \@dbltoproom :=G Maximum height of double-column floats allowed at
%                     top of two-column page.
%    \@fpmin      :=L Minimum height of floats in a float column.
% It should also perform the following local assignments where necessary
% -- i.e., where the new value differs from the old one:
%      \@fptop       :=L \@dblfptop
%      \@fpsep       :=L \@dblfpsep
%      \@fpbot       :=L \@dblfpbot
%
%  OUTPUT ROUTINE VARIABLES
%
%  \@colht : The total height of the current column.  In single column
%            style, it equals \textheight.  In two-column style, it is
%            \textheight minus the height of the double-column floats
%            on the current page.  MUST BE INITIALIZED TO \textheight.
%
%  \@colroom : The height available in the current column for text and
%              footnotes.  It equals \@colht minus the height of all
%              floats committed to the top and bottom of the current
%              column.
%
%  \footins : Footnote insertion number.
```

```
%
%  \@maxdepth : Saved value of TeX's \maxdepth.  Must be set
%               when any routine sets \maxdepth.
%
%            CALLING THE OUTPUT ROUTINE
%            --------------------------
%
% The output routine is called either by TeX's normal page-breaking
% mechanism, or by a macro putting a penalty < or = -10000 in the output
% list.  In the latter case, the penalty indicates why the output
% routine was called, using the following code.
%
%   penalty   reason
%   -------   ------
%   -10000    \pagebreak
%             \newpage
%   -10001    \clearpage (called with \penalty -10000 \vbox{} \penalty -10001
%   -10002    float insertion, called from horizontal mode
%   -10003    float insertion, called from vertical mode.
%   -10004    float insertion.
%
% Note: A float or marginpar puts the following sequence in the output
%       list:  (i) a penalty of -10004,
%              (ii) a null \vbox
%             (iii) a penalty of -10002 or -10003.
%       This solves two special problems:
%         1. If the float comes right after a \newpage or \clearpage,
%            then the first penalty is ignored, but the second one
%            invokes the output routine.
%         2. If there is a split footnote on the page, the second 'page'
%            puts out the rest of the footnote.
%
%             THE OUTPUT ROUTINE
%             ------------------
%
% FUNCTIONS USED IN THE OUTPUT ROUTINE:
%
% \@outputpage : Produces an output page with the contents of box
%              \@outputbox as the text part.  Also sets
%              \@colht :=G \textheight.   The page style is determined
%              as follows.
%                IF  @thispagestyle = true
%                  THEN  use \thispagestyle style
%                  ELSE  use ordinary page style.
%
% \@tryfcolumn\FLIST : Tries to form a float column composed of floats from
%              \FLIST with with the following parameters:
%                \@colht : height of box
%                \@fpmin : minimum height of floats in the box
%                \@fpsep : interfloat space
%                \@fptop : glue at top of box
%                \@fpbot : glue at bottom of box.
%              If it succeeds, then it does the following:
%                * \@outputbox :=L the composed float box.
%                * @fcolmade   :=L true
%                * \FLIST      :=G \FLIST - floats put in box
%                * \@freelist  :=G \@freelist + floats put in box
%              If it fails, then:
%                * @fcolmade :=L false
%           NOTE: BIT MUST BE A SINGLE TOKEN!
%
```

```
% \@makefcolumn \FLIST : Same as \@tryfcolumn except that it
%             fails to make a float column only if \FLIST is empty.
%             Otherwise, it makes a float column containing at least
%             the first box in \FLIST, disregarding \@fpmin.
%
% \@startcolumn :
%       Calls \@tryfcolumn\@deferlist{8}.  If \@tryfcolumn returns with
%       @fcolmade = false, then:
%                * Globally sets \@toplist and \@botlist to floats
%                  from \@deferlist to go at top and bottom of column,
%                  deleting them from \@deferlist.  It does
%                  this using \@colht as the total height, the page
%                  style parameters \@floatsep and \@textfloatsep, and
%                  the float placement parameters \@topnum, \@toproom,
%                  \@botnum, \@botroom, \@colnum and \textfraction.
%                * Globally sets \@colroom to \@colht minus the height
%                  of the added floats.
%
% \@startdblcolumn :
%      Calls \@tryfcolumn\@dbldeferlist{8}.  If \@tryfcolumn returns
%      with @fcolmade = false, then:
%                * Globally sets \@dbltoplist to floats from \@dbldeferlist
%                  to go at top and bottom of column, deleting them from
%                  \@dbldeferlist.  It does this using \textheight as the
%                  total height, and the parameters \@dblfloatsep, etc.
%                * Globally sets \@colht to \textheight minus the height
%                  of the added floats.
%
% \@combinefloats : Combines the text from box
%             \@outputbox with the floats from \@toplist and \@botlist,
%             putting the new box in \@outputbox.  It uses \floatsep and
%             \textfloatsep for the appropriate separations.  It puts the
%             elements of \TOPLIST and \BOTLIST onto \@freelist, and makes
%             those lists null.
%
% \@makecol : Makes the contents of \box255 plus the accumulated
%             footnotes, plus the floats in \@toplist and \@botlist,
%             into a single column of height \@colht, which it puts
%             into box \@outputbox.  It puts boxes in \@midlist back
%             onto \@freelist and restores \maxdepth.
%
% \@opcol : Outputs a column whose text is in box \@outputbox
%             If @twocolumn = false, then it calls \@outputpage,
%             sets \@colht :=G \textheight, and calls \@floatplacement.
%
%             If @twocolumn = true, then:
%                 If @firstcolumn = true, then it puts box \@outputbox
%                 into \@leftcolumn and sets @firstcolumn :=G false.
%
%                 If @firstcolumn = false, then it puts out the current
%                 two-column page, any possible two-column float pages,
%                 and determines \@dbltoplist for the next page.
%
% \@opcol ==
%  BEGIN
%   \@mparbottom :=G 0pt
%   if @twocolumn = true
%     then %% \@outputdblcol ==
%          if @firstcolumn = true
%            then @firstcolumn :=G false
%                 \@leftcolumn :=G \@outputbox
```

```
%            else @firstcolumn :=G true
%                 \@outputbox  := \vbox{
%                                   \hbox to \textwidth{
%                                     \hbox to\columnwidth{\box\@leftcolumn
%                                                          \hss}
%                                     \hfil \vrule width \columnseprule \hfil
%                                     \hbox to\columnwidth{\box\@outputbox}
%                                                          \hss}            }
%                 \@combinedblfloats
%                 \@outputpage
%                 \begingroup
%                    \@dblfloatplacement
%                    \@startdblcolumn
%                    while @fcolmade = true
%                      do  \@outputpage
%                          \@startdblcolumn  od
%                 \endgroup
%          fi
%     else
%       \@outputpage
%       \@colht :=G \textheight
%   fi
%  END
%
%  \@makecol ==
%    BEGIN
%     ifvoid \insert\footins
%        then  \@outputbox := \box255
%        else  \@outputbox := \vbox {\boxmaxdepth :=L \maxdepth
%                                                  %added 21 Jan 87
%                                    \unvbox255
%                                    \vskip \skip\footins
%                                    \footnoterule
%                                    \unvbox\footins
%                                   }
%    fi
%    \@freelist :=G \@freelist * \@midlist
%    \@midlist  :=G empty
%    \@combinefloats
%    \@outputbox := \vbox to \@colht{\boxmaxdepth := \maxdepth
%                                    \@texttop
%                                    temp :=L \dp\@outputbox
%                                    \unvbox\@outputbox
%                                    \vskip -temp
%                                    \@textbottom}
%    \maxdepth :=G \@maxdepth
%    END
%
% \@outputpage ==
%   BEGIN
%     \begingroup          %%% added 11 Jun 85 to keep special page
%                          %%%  declarations local to this output page
%     \catcode`\  := 10       %%make sure space is really a space
%     \- := \@dischyph     %%% Added 4 Aug 88 in event output routine
%     \' := \@acci         %%% called inside a tabbing environment.
%     \` := \@accii
%     \= := \@acciii
%     if @specialpage = T
%       then @specialpage :=G F
%            execute \ps@[eval(\@specialstyle)]  fi
%     if \@twoside = T
```

```
%       then if \count0 odd
%                       \@thehead       ==L \@oddhead
%                       \@thefoot       ==L \@oddfoot
%                       \@themargin     ==L \oddsidemargin
%                 else \@thehead        ==L \@evenhead
%                       \@thefoot       ==L \@evenfoot
%                       \@themargin     ==L \evensidemargin  fi fi
%     \shipout\vbox
%       {\normalsize          % set fonts size for head and foot
%        \baselineskip :=L \lineskip :=L 0pt
%        \par :=L \@@par       %% added 15 Sep 87 for robustness
%        \vskip \topmargin
%        \moveright\@themargin\vbox
%              { \box\@tempboxa := \vbox to \headheight{\vfil
%                                    \hbox to \textwidth
%                                       {\index == \label ==
%                                         \glossary == \@gobble
%                                            %% Added 22 Feb 87 as bug fix
%                                            %% RmS 91/06/21 \glossary added
%                                          \@thehead}}
%                \dp\@tempboxa := 0pt   % Don't skip space for descenders in
%                \box\@tempboxa         % running head.
%                \vskip \headsep
%                \box\@outputbox
%                \baselineskip\footskip
%                \hbox to \textwidth{\index == \label == \glossary == \@gobble
%                                      %%% added 22 Feb 87 as bug fix
%                                      %%% RmS 91/06/21 \glossary added
%                                   \@thefoot}
%               }
%       }
%     \@colht :=G \textheight
%     \endgroup                 %% added 11 Jun 85
%     \stepcounter{page}
%     \firstmark ==L \botmark   %% So marks work properly on float
%                               %% pages. (14 Jun 85)
%   END
%
% \@startcolumn ==
%  BEGIN
%    \@colroom :=G \@colht
%    if \@deferlist is empty
%      then @fcolmade := false
%      else \@tryfcolumn\@deferlist       %% else clause == \@xstartcol
%           if @fcolmade = false
%             then \begingroup
%                    \@tempb     :=L \@deferlist
%                    \@deferlist :=G empty
%                    \@elt \BOX  ==  BEGIN \@currbox == \BOX      % use \gdef
%                                          \@addtonextcol
%                                    END  == \@scolelt
%                    \@tempb
%                  \endgroup
%    fi    fi
%  END
%
% \@startdblcolumn ==
%  BEGIN
%    \@colht :=G \textheight
%      \@tryfcolumn\@dbldeferlist      %% else clause == \@xstartcol
%      if @fcolmade = false
```

```
%        then \begingroup
%               \@tempb        :=L \@dbldeferlist
%               \@dbldeferlist :=G empty
%               \@elt \BOX     ==  BEGIN \@currbox == \BOX      % use \gdef
%                                        \@addtodblcol
%                                    END  == \@sdblcolelt
%               \@tempb
%             \endgroup
%    fi    fi
%  END
%
% \output ==
%  BEGIN
%   case of \outputpenalty
%      > -10001  -> \@makecol
%                   \@opcol
%                   \@floatplacement
%                   \@startcolumn
%                   while  @fcolmade = true
%                      do \@opcol
%                         \@startcolumn
%                      od
%
%  %%% \@specialoutput ==
%
%      -10001   ->  %% \@doclearpage ==
%                      if there are no footnote insertions
%                        then unbox the \writes at the head of \box255
%                               and throw away the rest
%                             \@deferlist :=G \@toplist * \@botlist
%                                             * \@deferlist
%                             \@toplist :=G \@botlist :=G empty
%                             \@colroom :=G \@colht
%                             if \@currlist not empty
%                               then  LaTeX error: float(s) lost
%                                     \@currlist :=G empty
%                             fi
%                             \@makefcolumn\@deferlist
%                             while  @fcolmade = true
%                                  do \@opcol
%                                     \@makefcolumn\@deferlist
%                                  od
%                             if @twocolumn
%                               then
%                                  if @firstcolumn = true
%                                    then \@dbldeferlist :=G \@dbltoplist *
%                                                           \@dbldeferlist
%                                         \@dbltoplist   :=G empty
%                                         \@colht :=G \textheight
%                                         \begingroup
%                                            \@dblfloatplacement
%                                            \@makefcolumn\@dbldeferlist
%                                            while  @fcolmade = true
%                                              do \@outputpage
%                                                 \@makefcolumn\@dbldeferlist
%                                              od
%                                         \endgroup
%                                    else  \vbox{} \clearpage
%                             fi fi
%                        else \box255 := \vbox{\box255\vfil}
%                             \@makecol
```

```
%                              \@opcol
%                              \clearpage
%                         fi
%    < -10001     ->
%               if \outputpenalty < -10003
%                 then if \outputpenalty <-20000  %% true only at end
%                         then \deadcycles := 0
%                      fi
%                      box \@holdpg :=G box255
%                 else throw away box 255
%                      \@pagedp :=L natural depth of box \@holdpg
%                      \@pageht :=L natural ht of box \@holdpg
%                      \unvbox box \@holdpg  %% put text back
%                      if \@currlist nonempty
%                         then \@currbox  :=L head of \@currlist
%                              \@currlist :=G tail of \@currlist
%                              if \count\@currbox > 0
%                            %% Changed 28 Feb 88 so \@pageht and \@pagedp
%                            %% aren't changed for a marginal note
%                                then  %% this is a float
%                                  if there are footnote insertions
%                                    then advance \@pageht and \@pagedp and
%                                    reinsert footnotes
%                                  fi
%                                  \@addtocurcol
%                                else  %% this is a marginal note
%                                  if there are footnote insertions
%                                    reinsert footnotes
%                                  fi
%                                  \@addmarginpar
%                              fi
%                         else  THIS SHOULDN'T HAPPEN
%                      fi
%                      if \outputpenalty < 0           %% TO PERMIT PAGE BREAK
%                        then \penalty\interlinepenalty fi %% IF \@addtocurcol
%                                                  %%  DIDN'T INSERT A PENALTY
%               fi
%   end case
%  \vsize :=G if \outputpenalty > -10004 then \@colroom  %%normal case
%                                        else \maxdimen  %%processing float
%              fi
%  END
%
% \@combinefloats ==
%  BEGIN
%    if \@toplist nonempty
%      then %%\@cfla ==
%           \@elt\BOX == BEGIN  \@tempbox := \vbox{\unvbox\@tempbox
%                                                 \box\BOX
%                                                 \vskip \floatsep}
%                        END  ==  \@comflelt
%            \@tempbox := null
%            \@toplist
%            \@outputbox := \vbox{\boxmaxdepth :=L \maxdepth
%                                              %added 21 Jan 87
%                                 \unvbox\@tempbox
%                                 \vskip - \floatsep
%                                 \topfigrule
%                                 \vskip \textfloatsep
%                                 \unvbox\@outputbox               }
%            \@elt == \relax
```

```
%            \@freelist :=G \@freelist * \@toplist
%            \@toplist  :=G null
%    fi
%    if \@botlist nonempty
%      then  %%\@cflb ==
%            \@elt == \@comflelt
%            \@tempbox   := null
%            \@botlist
%            \@outputbox := \vbox{ \unvbox\@outputbox
%                                  \vskip \textfloatsep
%                                  \botfigrule
%                                  \unvbox\@tempbox
%                                  \vskip  - \floatsep   }
%            \@elt == \relax
%            \@freelist :=G \@freelist * \@botlist
%            \@botlist  :=G null
%    fi
%  END
%
% \@combinedblfloats ==
%  BEGIN
%    if \@dbltoplist nonempty
%      then \@elt == \@comdblflelt
%            \@tempbox := null
%            \@dbltoplist
%            \@outputbox := \vbox to \textheight
%                               {\boxmaxdepth :=L \maxdepth
%                                \unvbox\@tempbox
%                                \vskip - \dblfloatsep
%                                \dblfigrule
%                                \vskip \dbltextfloatsep
%                                \box\@outputbox                          }
%            \@elt == \relax
%            \@freelist :=G \@freelist * \@dbltoplist
%            \@dbltoplist  :=G null
%    fi
%  END
%
%
%            USER COMMANDS THAT CALL OR AFFECT THE OUTPUT ROUTINE
%            ---------------------------------------------------
%
% \newpage == BEGIN \par\vfil\penalty -10000 END
%
% \clearpage == BEGIN \newpage
%                     \write -1{}    % Part of hack to make sure no
%                     \vbox{}        % \write's get lost.
%                     \penalty -10001
%               END
%
% \cleardoublepage == BEGIN \clearpage
%                           if @twoside = true and c@page is even
%                             then \hbox{} \newpage fi
%                     END
%
% \twocolumn ==
%  BEGIN
%    \clearpage
%    \columnwidth :=G .5(\textwidth - \columnsep)
%    \hsize       :=G \columnwidth
%    @twocolumn   :=G true
```

```
%    @firstcolumn :=G true
%    \@dblfloatplacement
%  END
%
% \onecolumn ==
%  BEGIN
%    \clearpage
%    \columnwidth :=G \textwidth
%    \hsize       :=G \columnwidth
%    @twocolumn   :=G false
%    \@floatplacement
%  END
%
%
% \topnewpage{BOX} : starts a new page and puts BOX in a parbox of width
%     \textwidth across the top.  Useful for full-width titles for
%     double-column pages.
%     SURPRISE: The stretch from \@dbltextfloatsep will be inserted
%     between the BOX and the top of the two columns.
%
% \topnewpage{BOX} ==
%  BEGIN
%   \clearpage
%   Take \@currbox from \@freelist
%   \box\@currbox :=G \parbox{BOX \par
%                             \vskip - \@dbltextfloatsep}
%   \count\@currbox :=G 2
%   \@dbltopnum     :=G 1
%   \@dbltoproom    :=G maxdimension
%   \@addtodblcol
%   \vsize          :=G \@colht
%   \@colroom       :=G \@colht
%  END


%            FLOAT-HANDLING MECHANISMS
%            -------------------------
%
% The float environment obtains an insertion number B from the
% \@freelist (see below for a description of list manipulation), puts
% the float into box B and sets \count B to a FLOAT SPECIFIER.  For
% a normal (not double-column) float, it then causes a page break
% in one of the following two ways:
%   - In outer hmode: \vadjust{\penalty -10002}
%   - In vmode :      \penalty -10003.
% For a double-column float, it puts B onto the \@dbldeferlist.
% The float specifier has two components:
%    * A PLACEMENT SPECIFICATION, describing where the float may
%      be placed.
%    * A TYPE, which is a power of two--e.g., figures might be
%      type 1 floats, tables type 2 floats, programs type 4 floats, etc.
% The float specifier is encoded as follows, where bit 0 is the least
% significant bit.
%
%  Bit    Meaning
%  ---    -------
%   0     1 iff the float may go where it appears in the text.
%   1     1 iff the float may go on the top of a page.
%   2     1 iff the float may go on the bottom of a page.
%   3     1 iff the float may go on a float page.
%   4     always 1
```

```
%   5     1 iff a type 1 float
%   6     1 iff a type 2 float
%   etc.
%
%  A negative float specifier is used to indicate a marginal note.
%
%     MACROS AND DATA STRUCTURES FOR PROCESSING FLOATS
%     -----------------------------------------------
%
%  A FLOAT LIST consisting of the floats in boxes \boxa ... \boxN has the form:
%         \@elt \boxa ... \@elt \boxN
%  where  \boxI is defined by
%         \newinsert\boxI
%  Normally, \@elt is \let to \relax.  A test can be performed on the entire
%  float list by locally \def'ing \@elt appropriately and executing
%  the list.  This is a lot more efficient than looping through the list.
%
%  The following macros are used for manipulating float lists.
%
%  \@next \CS \LIST {NONEMPTY}{EMPTY} ==  %% NOTE: ASSUME \@elt = \relax
%    BEGIN  assume that \LIST == \@elt \B1 ... \@elt \Bn
%           if n = 0
%             then  EMPTY
%             else  \CS    :=L \B1
%                   \LIST  :=G \@elt \B2 ... \@elt \Bn
%                   NONEMPTY
%           fi
%    END
%
%
%  \@bitor\NUM\LIST : Globally sets switch @test to the disjunction for all I
%        of bit  log2 \NUM of the float specifiers of all the floats in
%        \LIST.  I.e., @test is set to true iff there is at least one
%        float in \LIST having bit  log2 \NUM  of its float specifier
%        equal to 1.
%
%  Note: log2 [(\count I)/32] is the bit number corresponding to the
%  type of float I.  To see if there is any float in \LIST having
%  the same type as float I, you run \@bitor with \NUM = [(\count I)/32] * 32.
%
% \@bitor\NUM\LIST ==
%   BEGIN
%      @test :=G false
%      { \@elt \CTR ==  if \count\CTR / \NUM is odd
%                        then  @test := true       fi
%        \LIST
%      }
%   END
%
%
% \@cons\LIST\NUM : Globally sets \LIST := \LIST * \@elt \NUM
%
% \@cons\LIST\NUM ==
%   BEGIN {  \@elt == \relax
%            \LIST :=G \LIST \@elt \NUM
%         }
%
%  BOX LISTS FOR FLOAT-PLACEMENT ALGORITHMS
%
%    \@freelist     : List of empty boxes for placing new floats.
%    \@toplist      : List of floats to go at top of current column.
```

```
%    \@midlist      : List of floats in middle of current column.
%    \@botlist      : List of floats to go at bottom of current column.
%    \@deferlist    : List of floats to go after current column.
%    \@dbltoplist   : List of double-col. floats to go at top of current page.
%    \@dbldeferlist : List of double-column floats to go on subsequent pages.
%
%  FLOAT-PLACEMENT ALGORITHMS
%
% \@tryfcolumn \FLIST ==
%  BEGIN
%   @fcolmade    :=G false
%   \@trylist    :=G \FLIST
%   \@failedlist :=G empty
%   \begingroup
%    \@elt == \@xtryfc
%    \@trylist
%   \endgroup
%   if @fcolmade = true
%     then  \@vtryfc \FLIST
%   fi
%  END
%
%  \@vtryfc ==
%    BEGIN
%      \@outputbox :=G \vbox{}
%      \@elt\BOX == BEGIN
%                     \@outputbox :=L \vbox{ \unvbox \@outputbox
%                                            \vskip \@fpsep
%                                            \box\BOX            }
%                   END  == \@wtryfc
%      \@flsucceed
%      \@outputbox :=G \vbox to \@colht{ \vskip \@fptop
%                                        \vskip -\@fpsep
%                                        \unvbox \@outputbox
%                                        \vskip \@fpbot      }
%      \@elt      ==  \relax
%      \@freelist :=G \@freelist * \@flsucceed
%      \FLIST     :=G \@failedlist * \@flfail
%    END
%
% \@xtryfc \BOX ==
%  BEGIN
%    remove first element from \@trylist
%    \@currtype := (\count\BOX / 32) * 32
%    \@bitor \@currtype \@failedlist    % @test := true if type on list
%    \@testfp \BOX                      % @test := true if no p-option
%    if ht of \BOX > \@colht
%      then @test :=G true
%    fi
%    if @test = true
%      then add \BOX to \@failedlist
%      else \@ytryfc \BOX
%    fi
%  END
%
% \@ytryfc ==
%  BEGIN
%   \begingroup
%     \@flsucceed :=G \@elt\BOX
%     \@flfail    :=G empty
%     \@tempdima  := \ht\BOX
```

```
%     \@elt == \@ztryfc
%     \@trylist
%     if \@tempdima > \@fpmin
%       then @fcolmade :=G true
%       else add \BOX to \@failedlist
%     fi
%   \endgroup
%   if @fcolmade = true  then  \@elt == \@gobble fi
%  END
%
% \@ztryfc \BOX ==
%  BEGIN
%   \@tempcnta := (\count\BOX / 32) * 32
%   \@bitor \@tempcnta {\@failedlist \@flfail}  % @test := true if on a list
%   \@testfp \BOX                               % @test := true if not p-option
%   \@tempdimb := \@tempdima + ht of \BOX + \@fpsep
%   if \@tempdimb > \@colht
%     then @test :=G true
%   fi
%   if @test = true
%     then add \BOX to \@flfail
%     else add \BOX to \@flsucceed
%          \@tempdima := \@tempdimb
%   fi
%  END
%
% \@testfp \BOX == BEGIN if bit 3 of \count\BOX = 0
%                          then @test :=G true      fi
%                 END
%
% \@makefcolumn \FLIST ==
%  BEGIN
%   \begingroup
%     \@fpmin =:L 0
%     \@testfp == \@gobble
%     \@tryfcolumn \FLIST
%   \endgroup
%  END
%
%  \@addtobot : Tries to put insert \@currbox on \@botlist.  Called only when:
%                  * \ht BOX + \@maxsep < \@colroom
%                  * type of \@currbox not on \@deferlist
%                  * \@colnum > 0
%                  * @insert = false
%               If it succeeds, then:
%                  * sets @insert true
%                  * decrements \@botroom by \ht BOX
%                  * decrements \@botnum and \@colnum by 1
%                  * decrements \@colroom by \ht BOX + either \floatsep
%                    or \textfloatsep, as appropriate.
%                  * sets \maxdepth to 0pt
%
%  \@addtotoporbot : Tries to put insert \@currbox on \@toplist or \@botlist.
%                    Called only under same conditions as \@addtobot.
%                    If it succeeds, then:
%                       * sets @insert true
%                       * decrements either \@toproom or \@botroom by \ht BOX
%                       * decrements \@colnum and either \@topnum or
%                         \@botnum by 1
%                       * decrements \@colroom by \ht BOX + either \floatsep
%                         or \textfloatsep, as appropriate.
```

```
%
% \@addtocurcol : Tries to add \@currbox to current column, setting @insert
%                 true if it succeeds, false otherwise.  It will add
%                 \@currbox to top only if bit 0 of \count \@currbox is 0, and
%                 to the bottom only if bit 0 = 0 or an earlier float of
%                 the same type is put on the bottom.
%                 If the float is put in the text, then
%                 \penalty\interlinepenalty is put
%                 right after the float, before the following \vskip, and
%                 \outputpenalty :=L 0.
%
% \@addtonextcol : Tries to add \@currbox to the next column, setting @insert
%                 true if it succeeds, false otherwise.
%
% \@addtodblcol : Tries to add \@currbox to the next double-column page,
%                 adding it to \@dbltoplist if it succeeds and \@dbldeferlist
%                 if it fails.
%
%  \@addtobot ==
%    BEGIN
%      if bit 2 of \count \@currbox = 1
%        then  if \@botnum > 0
%                 then if \@botroom > \ht \@currbox
%                        then \@botnum   :=G \botnum - 1
%                             \@colnum   :=G \@colnum - 1
%                             \@tempdima  :=L - \ht\@currbox -
%                                             if \@botlist empty
%                                                 then \textfloatsep
%                                                 else \floatsep
%                                             fi
%                             \@botroom  :=G \@botroom + \@tempdima
%                             \@colroom  :=G \@colroom + \@tempdima
%                              add \@currbox to \@botlist
%                              \maxdepth :=G 0pt
%                              @insert :=L true
%      fi      fi       fi
%    END
%
%  \@addtotoporbot ==
%    BEGIN
%      if bit 1 of \count \@currbox = 1
%        then if \@topnum > 0
%               then if \@toproom > \ht \@currbox
%                       then if \@currtype not on \@midlist or \@botlist
%                               then \@topnum    :=G \topnum - 1
%                                    \@colnum    :=G \@colnum - 1
%                                    \@tempdima  :=L - \ht\@currbox -
%                                                   if \@toplist empty
%                                                       then \textfloatsep
%                                                       else \floatsep
%                                                   fi
%                                    \@toproom   :=G \@toproom + \@tempdima
%                                    \@colroom   :=G \@colroom + \@tempdima
%                                    add \@currbox to \@toplist
%                                    @insert :=L true
%      fi     fi      fi      fi
%      if @insert = false then \@addtobot fi
%    END
%
% \@addtocurcol ==
%  BEGIN
```

```
%    @insert :=L false
%    \@textmin := \textfraction\@colht         %% added 27 Sep 85
%    if \@colroom > \ht \@currbox + max(\@pageht+\@pagedp, \@textmin)
%                            + \@maxsep
%       then if \@colnum > 0
%               then \@currtype := type of \@currbox
%                    if \@currtype not on \@deferlist
%                      then if \@currtype on \@botlist
%                             then \@addtobot
%                             else if bit0 of \count \@currbox = 1
%                                    then  decrement \@colnum
%                                          put \@currbox on \@midlist
%                                          add \@currbox + space +
%                                            \penalty \interlinepenalty to text
%                                          \outputpenalty :=L 0
%                                          @insert := true
%                                    else  \@addtotoporbot
%    fi      fi      fi     fi     fi
%    if @insert = false
%      then add \@currbox to \@deferlist
%    fi
%  END
%
% \@addtonextcol ==
%  BEGIN
%    @insert :=L false
%    \@textmin := \textfraction\@colht         %% added 27 Sep 85
%    if \@colroom > \ht \@currbox + \@textmin + \@maxsep
%       then if \@colnum > 0
%               \@currtype := type of \@currbox
%               then if \@currtype not on \@deferlist
%                      then \@addtotoporbot
%    fi      fi      fi
%    if @insert = false
%      then add \@currbox to \@deferlist
%    fi
%  END
%
%  \@addtodblcol ==
%    BEGIN
%      @insert :=L false
%      if bit 1 of \count \@currbox = 1
%        then  if \@dbltopnum > 0
%                then if \@dbltoproom > \ht \@currbox
%                       then if type of \@currbox not on \@dbldeferlist
%                              then \@dbltopnum   :=G \@dbltopnum - 1
%                                   \@tempdima := -\ht\@currbox -
%                                                 if \@dbltoplist empty
%                                                    then \dbltextfloatsep
%                                                    else \dblfloatsep
%                                                 fi
%                                   \@dbltoproom  :=G \@dbltoproom+\@tempdima
%                                   \@colht       :=G \@colht+\@tempdima
%                                   add \@currbox to \@dbltoplist
%                                    @insert :=L true
%      fi      fi       fi    fi
%      if @insert = false then add \@currbox to \@dbldeferlist
%    END
%
%  \@addmarginpar ==
%   BEGIN
```

```
%     if \@currlist nonempty
%       then remove \@marbox from \@currlist  %% NOTE: \@currbox = left box
%            add \@marbox and \@currbox to \@freelist
%       else LaTeX error: ?  %% shouldn't happen
%     fi
%     \@tempcnta := 1     %% 1 = right, -1 = left
%     if @twocolumn = true
%       then if @firstcolumn = true
%              then \@tempcnta := -1
%            fi
%       else if @mparswitch = true
%              then if count0 odd
%                     else \@tempcnta := -1
%                   fi
%            fi
%            if @reversemargin = true
%               then \@tempcnta := -\@tempcnta
%            fi
%     fi
%     if \@tempcnta < 0 then \box\@marbox :=G \box\@currbox fi
%     \@tempdima   :=L maximum(\@mparbottom - \@pageht + ht of \@marbox, 0)
%     if \@tempdima > 0 then LaTeX warning: 'marginpar moved' fi
%     \@mparbottom :=G \@pageht + \@tempdima + depth of \@marbox
%                           + \marginparpush
%     \@tempdima   :=L \@tempdima - ht of \@marbox
%     height of \@marbox :=G depth of \@marbox :=G 0
%     \vskip -\@pagedp
%     \vskip \@tempdima
%     \nointerlineskip
%     \hbox{ if @tempcnta > 0 then \hskip \columnwidth
%                                  \hskip \marginparsep
%                             else \hskip -\marginparsep
%                                  \hskip -\marginparwidth
%            fi
%            \box\@marbox
%            \hss
%          }
%     \vskip -\@tempdima
%     \nointerlineskip
%     \hbox{\vrule height 0 width 0 depth \@pagedp}
%   END


\maxdeadcycles = 100 % floats and \marginpar's add a lot of dead cycles

\let\@elt\relax

\def\@next#1#2#3#4{\ifx#2\@empty #4\else
   \expandafter\@xnext #2\@@#1#2#3\fi}

\def\@xnext \@elt #1#2\@@#3#4{\def#3{#1}\gdef#4{#2}}

\newif\if@test

\def\@bitor#1#2{\global\@testfalse {\let\@elt\@xbitor
   \@tempcnta #1\relax #2}}

%% RmS 91/11/22: Added test for \count#1 being 0.
%%               Suggested by Chris Rowley.
\def\@xbitor #1{\@tempcntb \count#1
   \ifnum \@tempcnta =\z@
```

```
    \else
      \divide\@tempcntb\@tempcnta
      \ifodd\@tempcntb \global\@testtrue\fi
    \fi}

% DEFINITION OF FLOAT BOXES:
\newinsert\bx@A
\newinsert\bx@B
\newinsert\bx@C
\newinsert\bx@D
\newinsert\bx@E
\newinsert\bx@F
\newinsert\bx@G
\newinsert\bx@H
\newinsert\bx@I
\newinsert\bx@J
\newinsert\bx@K
\newinsert\bx@L
\newinsert\bx@M
\newinsert\bx@N
\newinsert\bx@O
\newinsert\bx@P
\newinsert\bx@Q
\newinsert\bx@R



\gdef\@freelist{\@elt\bx@A\@elt\bx@B\@elt\bx@C\@elt\bx@D\@elt\bx@E
                \@elt\bx@F\@elt\bx@G\@elt\bx@H\@elt\bx@I\@elt\bx@J
                 \@elt\bx@K\@elt\bx@L\@elt\bx@M\@elt\bx@N
                 \@elt\bx@O\@elt\bx@P\@elt\bx@Q\@elt\bx@R}

\gdef\@toplist{}
\gdef\@botlist{}
\gdef\@midlist{}
\gdef\@currlist{}
\gdef\@deferlist{}
\gdef\@dbltoplist{}
\gdef\@dbldeferlist{}

% PAGE LAYOUT PARAMETERS
\newdimen\topmargin
\newdimen\oddsidemargin
\newdimen\evensidemargin
\let\@themargin=\oddsidemargin
\newdimen\headheight
\newdimen\headsep
\newdimen\footskip
\newdimen\footheight % even though it never gets used.
\newdimen\textheight
\newdimen\textwidth
\newdimen\columnwidth
\newdimen\columnsep
\newdimen\columnseprule
\newdimen\@maxdepth    \@maxdepth = \maxdepth
\newdimen\marginparwidth
\newdimen\marginparsep
\newdimen\marginparpush

% PAGE STYLE PARAMETERS
\newskip\floatsep
```

```
\newskip\textfloatsep
\newskip\intextsep
\newdimen\@maxsep
\newskip\dblfloatsep
\newskip\dbltextfloatsep
\newdimen\@dblmaxsep
\newskip\@fptop
\newskip\@fpsep
\newskip\@fpbot
\newskip\@dblfptop
\newskip\@dblfpsep
\newskip\@dblfpbot
\let\topfigrule=\relax
\let\botfigrule=\relax
\let\dblfigrule=\relax

% INTERNAL REGISTERS

\newcount\@topnum
\newdimen\@toproom
\newcount\@dbltopnum
\newdimen\@dbltoproom
\newcount\@botnum
\newdimen\@botroom
\newcount\@colnum
\newdimen\@textmin
\newdimen\@fpmin
\newdimen\@colht
\newdimen\@colroom
\newdimen\@pageht
\newdimen\@pagedp
\newdimen\@mparbottom \@mparbottom\z@
\newcount\@currtype
\newbox\@outputbox
\newbox\@leftcolumn
\newbox\@holdpg

\newif\if@insert
\newif\if@fcolmade
\newif\if@specialpage \@specialpagefalse
\newif\if@twoside     \@twosidefalse
\newif\if@firstcolumn \@firstcolumntrue
\newif\if@twocolumn   \@twocolumnfalse
\newif\if@reversemargin \@reversemarginfalse
\newif\if@mparswitch  \@mparswitchfalse

\def\@thehead{\@oddhead} % initialization
\def\@thefoot{\@oddfoot}

\def\newpage{\par\vfil\penalty -\@M}

\def\clearpage{\newpage \write\m@ne{}\vbox{}\penalty -\@Mi}

\def\cleardoublepage{\clearpage\if@twoside \ifodd\c@page\else
    \hbox{}\newpage\if@twocolumn\hbox{}\newpage\fi\fi\fi}

\def\twocolumn{\clearpage \global\columnwidth\textwidth
   \global\advance\columnwidth -\columnsep \global\divide\columnwidth\tw@
   \global\hsize\columnwidth \global\linewidth\columnwidth
   \global\@twocolumntrue \global\@firstcolumntrue
   \@dblfloatplacement\@ifnextchar[{\@topnewpage}{}}
```

```
8358
8359 \def\onecolumn{\clearpage\global\columnwidth\textwidth
8360      \global\hsize\columnwidth \global\linewidth\columnwidth
8361      \global\@twocolumnfalse \@floatplacement}
8362
8363 \long\def\@topnewpage[#1]{\@next\@currbox\@freelist{}{}%
8364     \global\setbox\@currbox\vbox{\hsize\textwidth \@parboxrestore
8365      #1\par\vskip -\dbltextfloatsep}\global\count\@currbox\tw@
8366     \global\@dbltopnum\@ne \global\@dbltoproom\maxdimen\@addtodblcol
8367     \global\vsize\@colht \global\@colroom\@colht}
8368
8369 %% RmS 91/09/29: added reset of \par to the output routine.
8370 %%               This avoids problems when the output routine is
8371 %%               called within a list where \par may be a no-op.
8372
8373 \output{\let\par\@@par
8374   \ifnum\outputpenalty <-\@M\@specialoutput\else
8375   \@makecol\@opcol\@floatplacement\@startcolumn
8376   \@whilesw\if@fcolmade \fi{\@opcol\@startcolumn}\fi
8377   \global\vsize\ifnum\outputpenalty >-\@Miv \@colroom
8378                  \else \maxdimen\fi}
8379
8380 % CHANGES TO \@specialoutput:
8381 % * \penalty\z@ changed to \penalty\interlinepenalty so \samepage
8382 %   works properly with figure and table environments.
8383 %   (Changed 23 Oct 86)
8384 %
8385 % * Definition of \@specialoutput changed 26 Feb 88 so \@pageht and \@pagedp
8386 %   aren't changed for a marginal note.  (Change suggested by
8387 %   Chris Rowley.)
8388 %
8389 \def\@specialoutput{\ifnum\outputpenalty >-\@Mii
8390     \@doclearpage
8391   \else
8392     \ifnum \outputpenalty <-\@Miii
8393        \ifnum\outputpenalty<-\@MM \deadcycles\z@\fi
8394        \global\setbox\@holdpg\vbox{\unvbox\@cclv}%
8395     \else \setbox\@tempboxa\box\@cclv
8396         \@pagedp\dp\@holdpg \@pageht\ht\@holdpg
8397         \unvbox\@holdpg
8398         \@next\@currbox\@currlist{\ifnum\count\@currbox >\z@
8399            \ifvoid\footins\else\advance\@pageht\ht\footins
8400              \advance\@pageht\skip\footins \advance\@pagedp\dp\footins
8401              \insert\footins{\unvbox\footins}\fi
8402             \@addtocurcol\else
8403            \ifvoid\footins\else\insert\footins{\unvbox\footins}\fi
8404             \@addmarginpar\fi}\@latexbug
8405     \ifnum \outputpenalty <\z@ \penalty\interlinepenalty\fi
8406   \fi\fi}
8407
8408
8409 \def\@doclearpage{\ifvoid\footins
8410         \setbox\@tempboxa\vsplit\@cclv to\z@ \unvbox\@tempboxa
8411         \setbox\@tempboxa\box\@cclv
8412         \xdef\@deferlist{\@toplist\@botlist
8413             \@deferlist}\gdef\@toplist{}\gdef\@botlist{}\global\@colroom\@colht
8414              \ifx\@currlist
8415               \@empty\else\@latexerr{Float(s)
8416                  lost}\@ehb\gdef\@currlist{}\fi
8417        \@makefcolumn\@deferlist
8418         \@whilesw\if@fcolmade \fi{\@opcol
```

```
                                  \@makefcolumn\@deferlist}\if@twocolumn
           \if@firstcolumn
             \xdef\@dbldeferlist{\@dbltoplist
                 \@dbldeferlist}\gdef\@dbltoplist{}\global\@colht\textheight
              \begingroup \@dblfloatplacement \@makefcolumn\@dbldeferlist
               \@whilesw\if@fcolmade \fi{\@outputpage
                                         \@makefcolumn\@dbldeferlist}\endgroup
           \else \vbox{}\clearpage
        \fi\fi
     \else\setbox\@cclv\vbox{\box\@cclv\vfil}\@makecol\@opcol
      \clearpage
     \fi}

\def\@opcol{\global\@mparbottom\z@\if@twocolumn\@outputdblcol\else
    \@outputpage \global\@colht\textheight \fi}

\def\@outputdblcol{\if@firstcolumn \global\@firstcolumnfalse
    \global\setbox\@leftcolumn\box\@outputbox
  \else \global\@firstcolumntrue
    \setbox\@outputbox\vbox{\hbox to\textwidth{\hbox to\columnwidth
      {\box\@leftcolumn \hss}\hfil \vrule width\columnseprule\hfil
       \hbox to\columnwidth{\box\@outputbox \hss}}}\@combinedblfloats
       \@outputpage \begingroup \@dblfloatplacement \@startdblcolumn
       \@whilesw\if@fcolmade \fi{\@outputpage\@startdblcolumn}\endgroup
    \fi}

% Extra \@texttop somehow found its way into \@makecol.  Deleted
% 1 Dec 86.  (Found by Mike Harrison)
%% RmS 91/10/22: Replaced \dimen128 by \dimen@.
\def\@makecol{\ifvoid\footins \setbox\@outputbox\box\@cclv
   \else\setbox\@outputbox
     \vbox{\boxmaxdepth \maxdepth
     \unvbox\@cclv\vskip\skip\footins\footnoterule\unvbox\footins}\fi
     \xdef\@freelist{\@freelist\@midlist}\gdef\@midlist{}\@combinefloats
     \setbox\@outputbox\vbox to\@colht{\boxmaxdepth\maxdepth
        \@texttop\dimen@\dp\@outputbox\unvbox\@outputbox
        \vskip-\dimen@\@textbottom}%
     \global\maxdepth\@maxdepth}

\let\@texttop=\relax
\let\@textbottom=\relax

\def\@outputpage{\begingroup\catcode`\ =10
     \let\-\@dischyph \let\'\@acci \let\`\@accii \let\=\@acciii
    \if@specialpage
     \global\@specialpagefalse\@nameuse{ps@\@specialstyle}\fi
     \if@twoside
       \ifodd\count\z@ \let\@thehead\@oddhead \let\@thefoot\@oddfoot
            \let\@themargin\oddsidemargin
          \else \let\@thehead\@evenhead
          \let\@thefoot\@evenfoot \let\@themargin\evensidemargin
     \fi\fi
     \shipout
     \vbox{\reset@font %% RmS 91/08/15
           \normalsize \baselineskip\z@ \lineskip\z@
           \let\par\@@par %% 15 Sep 87
           \vskip \topmargin \moveright\@themargin
           \vbox{\setbox\@tempboxa
                   \vbox to\headheight{\vfil \hbox to\textwidth
                                       {\let\label\@gobble \let\index\@gobble
                                        \let\glossary\@gobble %% 21 Jun 91
```

```
                                        \@thehead}}% %% 22 Feb 87
                  \dp\@tempboxa\z@
                  \box\@tempboxa
                  \vskip \headsep
                  \box\@outputbox
                  \baselineskip\footskip
                  \hbox to\textwidth{\let\label\@gobble
                           \let\index\@gobble  %% 22 Feb 87
                           \let\glossary\@gobble %% 21 Jun 91
                           \@thefoot}}\global\@colht\textheight
            \endgroup\stepcounter{page}\let\firstmark\botmark}


\def\@combinefloats{\boxmaxdepth\maxdepth \ifx\@toplist\@empty\else\@cfla\fi
    \ifx\@botlist\@empty\else\@cflb\fi}

\def\@cfla{\let\@elt\@comflelt \setbox\@tempboxa\vbox{}\@toplist
    \setbox\@outputbox\vbox{\unvbox\@tempboxa\vskip-\floatsep
    \topfigrule\vskip\textfloatsep \unvbox\@outputbox}\let\@elt\relax
    \xdef\@freelist{\@freelist\@toplist}\gdef\@toplist{}}

\def\@cflb{\let\@elt\@comflelt \setbox\@tempboxa\vbox{}\@botlist
    \setbox\@outputbox\vbox{\unvbox\@outputbox \vskip\textfloatsep
    \botfigrule\unvbox\@tempboxa \vskip-\floatsep}\let\@elt\relax
    \xdef\@freelist{\@freelist\@botlist}\gdef\@botlist{}}

\def\@comflelt#1{\setbox\@tempboxa
      \vbox{\unvbox\@tempboxa\box #1\vskip\floatsep}}

\def\@combinedblfloats{\ifx\@dbltoplist\@empty\else
    \let\@elt\@comdblflelt \setbox\@tempboxa\vbox{}\@dbltoplist
    \setbox\@outputbox\vbox to\textheight
      {\boxmaxdepth\maxdepth
       \unvbox\@tempboxa\vskip-\dblfloatsep
       \dblfigrule\vskip\dbltextfloatsep \box\@outputbox}\let\@elt\relax
    \xdef\@freelist{\@freelist\@dbltoplist}\gdef\@dbltoplist{}\fi}


\def\@comdblflelt#1{\setbox\@tempboxa
      \vbox{\unvbox\@tempboxa\box #1\vskip\dblfloatsep}}


\def\@startcolumn{\global\@colroom\@colht
    \ifx\@deferlist\@empty\global\@fcolmadefalse\else\@xstartcol\fi}

\def\@xstartcol{\@tryfcolumn\@deferlist \if@fcolmade\else
   \begingroup\edef\@tempb{\@deferlist}\gdef\@deferlist{}\let\@elt\@scolelt
   \@tempb\endgroup\fi}

\def\@scolelt#1{\def\@currbox{#1}\@addtonextcol}

\def\@startdblcolumn{\global\@colht\textheight
   \@tryfcolumn\@dbldeferlist \if@fcolmade\else
     \begingroup
       \edef\@tempb{\@dbldeferlist}\gdef\@dbldeferlist{}\let\@elt\@sdblcolelt
       \@tempb\endgroup\fi}

\def\@sdblcolelt#1{\def\@currbox{#1}\@addtodblcol}

\def\@tryfcolumn #1{\global\@fcolmadefalse \xdef\@trylist{#1}\xdef\@failedlist
   {}\begingroup \let\@elt\@xtryfc \@trylist \endgroup
```

```
    \if@fcolmade \@vtryfc #1\fi}

\def\@vtryfc #1{\global\setbox\@outputbox\vbox{}\let\@elt\@wtryfc
       \@flsucceed \global\setbox\@outputbox\vbox to\@colht{\vskip \@fptop
       \vskip -\@fpsep \unvbox \@outputbox \vskip \@fpbot}\let\@elt\relax
       \xdef #1{\@failedlist\@flfail}\xdef\@freelist{\@freelist\@flsucceed}}

\def\@wtryfc #1{\global\setbox\@outputbox\vbox{\unvbox\@outputbox
    \vskip\@fpsep\box #1}}


\def\@xtryfc #1{\@next\@tempa\@trylist{}{}\@currtype
  \count #1\divide\@currtype\@xxxii \multiply\@currtype\@xxxii
  \@bitor \@currtype \@failedlist \@testfp #1\ifdim
    \ht #1>\@colht \global\@testtrue\fi
    \if@test \@cons\@failedlist #1\else \@ytryfc #1\fi}

\def\@ytryfc #1{\begingroup \gdef\@flsucceed{\@elt #1}\gdef\@flfail
  {}\@tempdima\ht #1\let\@elt\@ztryfc \@trylist \ifdim \@tempdima >\@fpmin
     \global\@fcolmadetrue \else \@cons\@failedlist #1\fi
  \endgroup \if@fcolmade \let\@elt\@gobble \fi}

\def\@ztryfc #1{\@tempcnta\count #1\divide\@tempcnta\@xxxii
    \multiply\@tempcnta\@xxxii \@bitor \@tempcnta {\@failedlist
    \@flfail}\@testfp #1\@tempdimb\@tempdima \advance\@tempdimb\ht #1\advance
    \@tempdimb\@fpsep \ifdim \@tempdimb >\@colht \global\@testtrue\fi
    \if@test \@cons\@flfail #1\else \@cons\@flsucceed #1\@tempdima\@tempdimb
    \fi}

\def\@testfp #1{\@tempcnta\count #1\divide\@tempcnta 8\relax
   \ifodd\@tempcnta \else \global\@testtrue\fi}

\def\@makefcolumn #1{\begingroup  \@fpmin\z@ \let\@testfp\@gobble
   \@tryfcolumn #1\endgroup}

\def\@addtobot{\@tempcnta\count\@currbox\divide\@tempcnta4 \ifodd\@tempcnta
   \ifnum \@botnum >\z@ \ifdim \@botroom >\ht\@currbox
    \global\advance\@botnum\m@ne
    \global\advance\@colnum\m@ne
    \@tempdima -\ht\@currbox
    \advance\@tempdima -\ifx\@botlist\@empty \textfloatsep
       \else\floatsep\fi
    \global\advance\@botroom \@tempdima
    \global\advance\@colroom \@tempdima
    \@cons\@botlist\@currbox \global\maxdepth\z@
    \@inserttrue\fi\fi\fi}

\def\@addtotoporbot{\@tempcnta\count\@currbox \divide\@tempcnta\tw@
   \ifodd\@tempcnta \ifnum \@topnum >\z@ \ifdim\@toproom >\ht\@currbox
     \@bitor\@currtype{\@midlist\@botlist}\if@test\else
        \global\advance\@topnum\m@ne
        \global\advance\@colnum\m@ne
        \@tempdima-\ht\@currbox
        \advance\@tempdima
           -\ifx\@toplist\@empty \textfloatsep \else\floatsep\fi
        \global\advance\@toproom \@tempdima
        \global\advance\@colroom \@tempdima
        \@cons\@toplist\@currbox
        \@inserttrue
\fi\fi\fi\fi
\if@insert\else\@addtobot \fi}
```

```
8602
8603 \def\@addtonextcol{\@insertfalse \@textmin \textfraction\@colht
8604    \@tempdima\ht\@currbox
8605    \advance\@tempdima\@textmin\advance\@tempdima\@maxsep
8606    \ifdim\@colroom >\@tempdima
8607      \ifnum\@colnum >\z@
8608         \@currtype\count\@currbox \divide\@currtype\@xxxii
8609         \multiply\@currtype\@xxxii
8610         \@bitor\@currtype\@deferlist
8611         \if@test\else
8612           \@addtotoporbot
8613     \fi\fi\fi
8614     \if@insert\else \@cons\@deferlist\@currbox\fi}
8615
8616 \def\@addtodblcol{\@insertfalse
8617    \@tempcnta\count\@currbox \divide\@tempcnta\tw@
8618    \ifodd\@tempcnta
8619       \ifnum\@dbltopnum >\z@
8620          \ifdim\@dbltoproom >\ht\@currbox
8621            \@currtype\count\@currbox \divide\@currtype\@xxxii
8622                \multiply\@currtype\@xxxii
8623            \@bitor\@currtype\@dbldeferlist
8624            \if@test\else
8625               \global\advance\@dbltopnum\m@ne
8626               \@tempdima -\ht\@currbox
8627               \advance\@tempdima -\ifx\@dbltoplist\@empty
8628                  \dbltextfloatsep\else\dblfloatsep\fi
8629               \global\advance\@dbltoproom \@tempdima
8630               \global\advance\@colht \@tempdima
8631               \@cons\@dbltoplist\@currbox
8632               \@inserttrue
8633    \fi\fi\fi\fi
8634    \if@insert\else \@cons\@dbldeferlist\@currbox \fi}
8635
8636 % CHANGE TO \@addtocurcol:
8637 % \penalty\z@ changed to \penalty\interlinepenalty so \samepage
8638 % works properly with figure and table environments.
8639 % (Changed 23 Oct 86)
8640 %
8641 \def\@addtocurcol{\@insertfalse \@textmin \textfraction\@colht
8642    \@tempdima\@pageht \advance\@tempdima\@pagedp
8643    \ifdim \@textmin >\@tempdima \@tempdima\@textmin \fi
8644        \advance\@tempdima\ht\@currbox \advance\@tempdima\@maxsep
8645    \ifdim\@colroom >\@tempdima
8646       \ifnum\@colnum >\z@
8647          \@currtype\count\@currbox \divide\@currtype\@xxxii
8648                \multiply\@currtype\@xxxii
8649          \@bitor\@currtype\@deferlist
8650          \if@test\else
8651             \@bitor\@currtype\@botlist
8652             \if@test \@addtobot \else
8653                \ifodd\count\@currbox
8654                  \global\advance\@colnum\m@ne
8655                  \@cons\@midlist\@currbox
8656                  \vskip\intextsep \box\@currbox
8657                  \penalty\interlinepenalty \vskip\intextsep
8658                  \ifnum\outputpenalty <-\@Mii \vskip -\parskip\fi
8659                  \outputpenalty\z@
8660                  \@inserttrue
8661                \else \@addtotoporbot
8662    \fi\fi\fi\fi\fi
```

```
   \if@insert\else\@cons\@deferlist\@currbox\fi}

\def\@addmarginpar{\@next\@marbox\@currlist{\@cons\@freelist\@marbox
    \@cons\@freelist\@currbox}\@latexbug\@tempcnta\@ne
    \if@twocolumn
        \if@firstcolumn \@tempcnta\m@ne \fi
    \else
      \if@mparswitch
         \ifodd\c@page \else\@tempcnta\m@ne \fi
      \fi
      \if@reversemargin \@tempcnta -\@tempcnta \fi
    \fi
    \ifnum\@tempcnta <\z@  \global\setbox\@marbox\box\@currbox \fi
    \@tempdima\@mparbottom \advance\@tempdima -\@pageht
       \advance\@tempdima\ht\@marbox \ifdim\@tempdima >\z@
       \@@warning{Marginpar on page \thepage\space moved}\else\@tempdima\z@ \fi
    \global\@mparbottom\@pageht \global\advance\@mparbottom\@tempdima
       \global\advance\@mparbottom\dp\@marbox
       \global\advance\@mparbottom\marginparpush
    \advance\@tempdima -\ht\@marbox
    \global\ht\@marbox\z@ \global\dp\@marbox\z@
    \vskip -\@pagedp \vskip\@tempdima\nointerlineskip
    \hbox to\columnwidth
      {\ifnum \@tempcnta >\z@
          \hskip\columnwidth \hskip\marginparsep
        \else \hskip -\marginparsep \hskip -\marginparwidth \fi
       \box\@marbox \hss}\nobreak   %% RmS 91/06/21 \nobreak added
    \vskip -\@tempdima
    \nointerlineskip
    \hbox{\vrule \@height\z@ \@width\z@ \@depth\@pagedp}}

 \message{debugging}
%     ****************************************
%     * DEBUGGING AND TEST INITIALIZATIONS  *
%     ****************************************
%
% DEBUGGING
\def\showoverfull{\tracingonline=1}
\tracingstats1   % SHOWS HOW MUCH STUFF TeX HAS USED
\def\showoutput{\tracingonline1\tracingoutput1
    \showboxbreadth99999\showboxdepth99999\errorstopmode}
\def\makeatletter{\catcode`\@=11\relax}
\def\makeatother{\catcode`\@=12\relax}

\newcount\@lowpenalty
\newcount\@medpenalty
\newcount\@highpenalty

% LIST

% ENUMERATION

% ITEMIZE

% ARRAY AND TABULAR

% THE PICTURE ENVIRONMENT

\unitlength = 1pt
\fboxsep = 3pt
\fboxrule = .4pt
```

```
8724
8725 %% FOOTNOTES
8726
8727 %\def\footnoterule{}  % INITIALIZED BY PLAIN
8728 %\skip\footins{}      % INITIALIZED BY PLAIN
8729 %\interfootnotelinepenalty % INITIALIZED BY PLAIN
8730
8731 \@maxdepth       = \maxdepth
8732
8733 % \vsize initialized because a \clearpage with \vsize < \topskip
8734 %  causes trouble.
8735 % \@colroom and \@colht also initialized because \vsize may be
8736 %  set to them if a \clearpage is done before the \begin{document}
8737
8738 \vsize = 1000pt
8739 \@colroom = \vsize
8740 \@colht = \vsize
8741
8742 \endinput
```

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols

A

B

C

D

E

F

G

H

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z