Anatomy of a \TeX{} Macro Package

Arthur M. Keller
Stanford University

Abstract. A \TeX{} macro package is described that produces various formats of output. Parts of the macro package are based on the basic and book formats described in the \TeX{} manual as well as the format for the \TeX{} manual itself.

1 Introduction

This macro package was developed over approximately a year and a half. It started when I first began to use \TeX{}, and proceeded as I began to write macros that others found useful. Much of the development has been for a forthcoming book to be published by McGraw-Hill teaching PASCAL to students without a knowledge of computers.* Further, the work of others has been borrowed and adapted in preparing these macros.

It is assumed that the reader is familiar with \TeX{}. This documentation is intended to be read while perusing the code itself.

The macro package consists of a file called ARKTEX.\TeX{} which should reside in the \TeX{} system files area. This file refers to other files which are loaded if needed. This package was designed on the SU-AI system and uses the SAIL character set as described in the \TeX{} manual. Others not using the SAIL character set may find it useful to change the \texttt{chcode}'s at the start of the file as well as some of the one character macros.

2 ARKTEX.\TeX{}

The file ARKTEX.\TeX{} is divided up into about twenty sections separated by horizontal rules. In the file, these are usually on separate pages (\ie, separated by control-L's).

2.1 Standard Basic Stuff

This section consists primarily of text that appears in BASIC.\TeX{}. The \texttt{chcode}'s should be changed as necessary for your system.

2.2 Font Definitions and Related Macros

This section consists of macros for fonts of various fonts and sizes. The \texttt{chcode} on the first line is to all \texttt{@} to be parsed correctly on this page because elsewhere \texttt{@} has \texttt{chcode} of 13. The fonts on this page are primarily for eight, nine, and ten point typesetting. Other random fonts also exist. The \texttt{:} macro has been redefined to save the font letter in \texttt{fontcode}. This is used by the macros in “Definitions of Odd Characters” that produces output of different characters depending on which font is currently in use. Do not use the \texttt{usefont} macro, but the \texttt{curfont} macro may be used as an alternate to the \texttt{:} macro.

The \texttt{loadfont} macro is used to allow documents to use fonts that exist only on some systems to refer to these fonts symbolically. Ordinarily these fonts are not preloaded. However, a document attempting to use such a font for a particular output device for which the font exists may do so. For example, at Stanford, some of the fonts loaded by \texttt{loadfont} exist only on the XGP or on the Dover but not on the Alphatype.

---

*The book will probably be titled \textit{A First Course in Computer Programming Using PASCAL} and it will be available about January 1982.
The various sizes of type are referred to by the macros \tenpoint, \ninepoint, and \eightpoint to obtain ten, nine, and eight point type faces respectively. For each size there are the following fonts provided roman (\rm), slanted roman (\sl), boldface roman (\bf), italic (\it), math italic (\mi), teletype (\tt), and symbol (\sy). In addition, small caps (\sc) is provided for ten point. Each size of type also includes definitions for the width of a digit (\9) as well as complete math mode information.

To start text in one of these sizes specify \tenpoint, \ninepoint, or \eightpoint. Customised macros for various point sizes may be constructed by redefining the macros \usertenpoint, \userninepoint, and \usereightpoint. However, if the most of the document is to be in that type size, say \usetenpoint, etc., instead at the start of the document. If you use \startcode and \endcode, \fontsize will be used to determine what font size to return to. This macro may be defined at any time prior to such usage, but it can be done easily by saying \usetenpoint if desired.

2.3 Definitions of Odd Characters

This section includes definitions that will allow characters such as \ and \# to be parsed correctly in any mode. Characters that have no other usage, such as \ may appear without preceding a \ as they have a \chcode of 13. Characters that have other usages, such as \#, must be preceded by a \ if they are to appear as the character instead of being used for their standard purpose. However, because they are control sequences without an argument, you must put \ after following them to avoid the space after them getting ignored.

Characters in teletype mode are fixed width characters. Therefore, the \ttchar macro takes a specified character and puts it in the desired size box so that the remainder of the line will line up.

The \fontclassify macro determines which mode or typestyle is being used and chooses the correct argument to emit. This allows \# to produce a \# in the right font.

2.4 Redefinitions of One Character Macros

This section consists of redefinitions of one character macros so that they work in any mode. Users who redefine macros such as \leq to print \leq will also find the redefinitions useful. Macros starting with \m are defined to save the original definitions. Macros without an \m are defined to work in or out of math mode. These original macros are redefined to match the new set. The exception is \ as it means different things in math mode than in non-math mode.

2.5 Make Some Math Things Work Anywhere

This group of macros works just like those in the previous section.

2.6 Page Numbering

The section on page numbering is rather complicated by the existence of macros to defer text. The pertinent macro for users is \setpagecount which sets the page number on the following page to the specified number. The page number on the current page is set to one less than that number. This mess is done because the author likes to put out an extra page describing what is going on whenever changing page numbers. In particular, output devices that do not put out header pages permit several users output to appear without intervening pages. Putting out your own separator page reduces the chance that your output will be misfiled. Lastly, the \chapterbegin macro in book format ejects the page first, so the author usually puts fixed garbage on the previous page. It's also a good place to put copyright notice if the file is going to be copyrighted. Most importantly, using \settitle of book format on the ejected page guarantees that the next page will have the correct headings.

2.7 \output, Style, and Format Routines

These macros are the heart of the claim of providing many formats of output. They fall into four
categories: overhead, output, style, and format routines. Overhead routines are used in many places and are obvious. These are \normal, \resetsize, and \everyoutput.

The output routines set the \output macro. Also, they should also set the macros \normalhsize and \normalvsize. See OPLAIN.TEX for the minimum required in an output routine.

Style routines set such things are paragraph spacing. See OBLOCK.TEX for the minimum required in a style routine.

Format routines are simply style and output routines in the same file or macro. See the descriptions of the individual files for more information.

2.8 Footnotes

The footnote macro has gone through several generations. The latest one uses the \botsep to insert the horizontal bar. If your system does not yet support \botsep, you may have to be more clever about when to put in the bar and when to delete it. The author's previous method was to insert a bar if this was the first footnote on a page. The output routine would reset the first-footnote-on-the-page switch. However, this would occasionally fail in that the first footnote on the page would actually be generated before the output routine was called (e.g., if the paragraph is split on two pages). Then the \firstfootnote macro would be used which would hack the switches appropriately so that the next footnote would not get a bar. This involved setting a switch that the output routine cleared so that the output routine would not clear a second switch that indicated that a bar had already been output for that page. What a mess! Anyway, get a new version of \TeX.

The macro \footnote provides automatically numbered footnotes. The numbers are started at 1—the macro pre-increments it.

There are three macros of characters for using for footnotes. These are \upstar, \dagger, and \ddagger.

2.9 Paragraphs

This section consists of macros for various hanging paragraphs. The \hangbox macro creates a box of width based on argument 1 containing argument 2. The remaining lines of the paragraph will be indented the same width. For example, \hangbox to 30pt \{foo\}bar etc., will produce a paragraph containing bar etc., indented to 30 points with the first 30 points of the first line containing foo. The macros \levelone, \leveltwo, and \levelthree generate such hanging boxed paragraphs to 20 points, 40 points, and 60 points, respectively. However, the contents of the boxes are left justified in a twenty-point box that is right justified in the 40- or 60-point box. The following are uses of \levelone, \leveltwo, and \levelthree:

1. This is a short box followed by a long paragraph. Isn't it amazing to see what drivel can be published in the guise of an example. Put your ad here; to find out whether you can call 936-1212.
2. This is a medium box followed by a long paragraph. Isn't it amazing to see what drivel can be published in the guise of an example. For a good time call 767-8989.
3. This is a long box followed by a long paragraph. Isn't it amazing to see what drivel can be published in the guise of an example. For example, did you know that when the author finishes his Ph.D., he'll be looking for a teaching job? A reference to this article will pad his C.V.

The macros \number and \number create indented paragraphs with boxes of 20 and 50 points respectively. That's right, \number is just like \levelone.

The \indpar macro takes the argument and creates an paragraph indented on both sides to 40 points. Normal paragraph indentation or paragraph separation must be done by you. A \strut has been inserted to get the correct line spacing between the paragraph and preceding and following text to handle risers and descenders properly. However, no \parskip glue is inserted and 1 point is inserted for the assumed \lineskip.

The \hdr macro creates a centered boldface heading consisting of its argument that makes a good section heading if you are not using book format.
2.10 List Definitions

Now that you know all about the paragraph macros, you might expect macros for doing numbered lists automatically. There are three levels of numbering. The first level uses \list followed by an argument which is the initial number for counting. Then \item is used to precede each item. You may use \itemindent to indent the same amount as \item for continuing the following paragraph, for example. The macro \bitem gives a centered bullet in a 20 point box starting the hanging paragraph.

The second level of counting is in roman numerals. Put the number you want to start counting from after the \sublist. Note that this number should be positive, so \-3 gives \"iii.\" As you might expect, there are \subitem and \subitemindent.

The third level of counting is letters. Put the letter you want to start counting from after the \subsublist. And there are \subsubitem and \subsubitemindent.

2.11 Underlining and Boxes

This section consists of macros for doing various kinds of under- and overlining as well as lined boxes. The \undertext and \overtex macros underline and overline in horizontal mode. And \leaderline gives a leader of a rule.

The \boxit macro is from exercise 21.3 of the \TeX\ manual. However, \sizeboxit makes the box a specific size. Also, \boxitin\glue boxes the box without 3 points of space on all sides. If you want to put \boxit's in a \valign or put straight text inside, use \Boxit or \Boxitin\glue, which reverse horizontal and vertical mode. To put corner L's around a box, use \Lboxit.

The demonstrate interactive system output, it is useful to display the user entered data underlined. To underline the second half of a line, say \type prompt\>underlined text. Use \tttype the same way for indented dialogue.

2.12 Penalties

Aren't these obvious. They do save space in macros over their expansions.

2.13 \nofill \endnofill

This is the first of the verbatim mode set. To use it, say \nofill followed by the text, followed by \endnofill. Line breaks appear exactly where they do in the input text. Exactly as many spaces appear in the output as in the input. The code is listed verbatim without page breaks. To allow page breaks, say \allowbreak. A blank line is generated if there is no page break. To have no space generated if there is no page break, use \allowbreaknoglue.

Tabs are not allowed in verbatim mode. This is because it is not clear how many spaces to generate for a tab. If you think you know better, say \def\tab{definition}.

2.14 \startcode \endcode

The \startcode code \endcode sequence produces verbatim code in \displayfont. See the previous section and the code for more details.

In \startcode mode, \lesssim and \gtrsim produce \leq and \geq, respectively. These revert back to their former meanings at the end.

2.15 Verbatim Mode Using $$\halign$$

Verbatim mode is just like \startcode mode except that the calling conventions are different and it may appear in an \halign. To use it precede the code with \halign\ and follow the code with a right brace on its own line. To allow a page break, code \breakhere\%. Note that the % is required.
The \threecol macro generates \verbatim mode except with three columns instead of one.
To put a box around verbatim code, say \Boxit(verbatim\{code\}).

2.16 Notes

Notes are useful to provide descriptions of things that you want to fix. The description of the file MNOTES.TEX appears later.

2.17 Index Macros

An index package is described in Vol. 1, No. 1 of TUGboat.

2.18 Defer Mode

Defer mode is useful for specifying an entire page that is to appear as soon as possible. The description of the file DEFER.TEX appears later.

2.19 Table of Contents

This set of macros generates a table of contents compatible with book format. The description of the file MTOFC.TEX appears later.

2.20 Interesting Hacks

This section consists of interesting macro hacks that are useful for one and all.
The \ifnum macro determines if argument 1 is null. If so it expands argument 2; otherwise, argument 3 is expanded. To call say: \ifnum\text\then\{code\}\else\{code\}.
The \brace, \dnbrace, and \upbrace macros are from page 103 of the \TeX\ manual. On the other hand, \blackslug is from page 167.
The \boxtop macro sets the baseline at the top of the box. This is useful for lining up variable sized boxes at the top. For example, to line up \hbox par's in a \halign, use \boxtop around the \hbox par's.
The \topspace and \magnify macros is from the new version of BASIC.TEX.
There are two macros for playing with counters. Use \setq to set the control sequence which is the first argument to the counter number in the second argument. Use \advcountq to increment the counter in the control sequence. With these macros, you can save counters for what they are really needed for: setting up the correct numbers for output routines.
The \done macro goes at the end of the document.
The \capitalpar macro creates a paragraph like the "Gentle reader" at the start of the \TeX\ manual.

2.21 Default Options

See \startcode and \endcode for a description of \displayfont. See section 1.6 for a description of page numbering.

2.22 Documentation

Every macro package should have some.

3 MBOOK.TEX

The book format macro package is the most developed of the macro formats. The \bookoutput routine handles \titlepage (it sets \tpage to \text), as well as proper placement of the page numbers. The page
numbers will appear on the top of the page if \pagenumberarea is T, on the bottom if B; otherwise, no page numbers appear. It defaults to T.

The \pagenumberregion macro defines the format of the page heading (or footing). It defaults to the macro \boxpagenumberregion, which produces the format in the \TeX manual and this document. The right and left headings are specified by \titlemark{right}-\{left\}.

To get a page without a page heading, say \titlepage.

Chapters, sections, subsections, and diminished sections are all numbered automatically. To use, say \chapterbegin, \sectionbegin, \subsectionbegin, or \dimsectionbegin, all followed by the chapter or section name in braces. For an unnumbered chapter, such as an appendix or a table of contents, use \specialbegin followed by the name in braces. Use \settite to set the left and right headings if you aren't using the other macros in this paragraph.

To get data for a table of contents, use \inittoc{filename}.

4 MACACM.TEX

This produces 25% oversized output for the ACM camera-ready copy specifications. At the start of the paper, code \useacmformat. Then define \title and \authors. Then define the title portion of the paper. Next, say \endoftitle. When done with the paper, say \endofpaper\end.

Note that \defer does not work with this format.

5 OPLAIN.TEX

This produces output unadorned with page numbers or anything.

6 OBASIC.TEX

This produces output with page numbers as in BASIC.TEX.

7 OWOODS.TEX

This produces output with page numbers on the bottom of the page with hyphens around the numbers as popularized by Don Woods at Stanford.

8 SBLOCK.TEX

This produces block style paragraphs with about 6 points of space between paragraphs.

9 SBASIC.TEX

This produces indented paragraphs as in BASIC.TEX.

10 MNOTES.TEX

This file contains macros to generate notes to the writer. Say \initnotes to create the file. Say \endnotes\{text\} to output text. The \putnotes macro takes the notes and outputs them in the listing.

11 DEFER.TEX

Defer mode is used to produce a floating figure that takes one or more whole pages. Like \topinsert for floating figures, it is used in vertical mode. However, defer mode handles multiple page figures and will as keeping several figures in the order specified.

To use, say \defer followed by the figure followed by \enddefer.

Defer mode does not work with multicolumn formats.
12 \texttt{MTOFC.TEX}

The table-of-contents package produces a table of contents based on the data files produced if \texttt{inittotfc} is used in book format. To use, say \texttt{begintotc} followed by \texttt{chaptertotc}, \texttt{sectiontotc}, etc., macros each followed by a title in braces and a page number terminated by a period. Use \texttt{endtotc} at the end of the table of contents.

13 Acknowledgments

This paper could not have been written and published without \TeX\ designed by Donald E. Knuth. He also provided some advice and encouragement, as did Luis Trabb-Pardo. Many of the macros were written with the help of Jim Boyce. Brent Hailpern designed the original defer mode. Max Diaz provided additional suggestions and several macros as well. Denny Brown suggested some of the macros and the idea of supporting many formats. My advisor, Gio Wiederhold, gave encouragement and enlightened criticism through his attempts to use these macros. The staff of the American Mathematical Society were very helpful and interested in getting this manuscript written and published.

14 The Macros

The source for the macro package follows.

\begin{verbatim}
#

\chcode '45+5 % % **** N.B. this must be first
\chcode '173+1 % {
\chcode '176+2 % }
\chcode '44+3 % $
\chcode '26+4 % @
\chcode '43+6 % #
\chcode '136+7 % ↑
\chcode '1+8 % ▲

% Shorthands for certain definitions
\def \trace{\chparO*) \trace'1400345} %
\def \jpar{\chpar1*}
\def \hpen{\chpar2*}
\def \ragged{\chpar8*}

% centerings
\def \lfti{\hfill #1}
\def \ctri{\hfill #1\hfill}
\def \rti{\hfill #1}
\def \topi{\hfill #1}
\def \midii{\hfill #1\vfill}
\def \boti{\vfill #1}

\def \ljustline#1{\hbox to size(#1\hss)}
\def \crlnsl#1{\hbox to size(#1\hss\hss)}
\def \rjustline#1{\hbox to size(#1\hss\hss)}
\end{verbatim}
Font Definitions and Related Macros

\chcode'100+12 % allow 0 on this page to be parsed correctly

% font definitions for 8, 9, and 10 point fonts and friends
\font @+cmath
\font a+cmr10 \font b+cmr9 \font c+cmr8 \font d+cmr7 \font e+cmr6 \font f+cmr5
\font g+cmr10 \font h+cmr9 \font i+cmr8 \font j+cmr7 \font k+cmr6 \font l+cmr5
\font G+cmr10 \font H+cmr9 \font I+cmr8 \font J+cmr7 \font K+cmr6 \font L+cmr5
\font M+cmm10 \font N+cmm9 \font O+cmm8 \font P+cmm7 \font Q+cmm6 \font R+cmm5
\font T+cmm10 \font U+cmm9 \font V+cmm8 \font W+cmm7 \font X+cmm6 \font Y+cmm5
\font Z+cmm5

% font definitions for random desired fonts
\font ;+cmtitl
\font <+cmssb \font =+cmss12 \font >+cmss8 \font ?+cmss8

% font request macros
\let \usefont=:
\def \curfont #1{\usefont #1\def\fontcode(#1)}
\let \

% font name macros
\def \loadfont#2#3{\font #1+#2 \gdef #3{\#:1}\gdef #3{\#:2}\gdef #3{\#:3}
\def \big{\loadfont D{cmr12}{\big}(\big)}
\def \ms25{\loadfont A{ms25}{\ms25}(\ms25)}
\def \msn{\loadfont B{non}{\msn}(\msn)}
\def \peni{\loadfont P{peni}{\peni}(\peni)}
\def \stan{\loadfont S{stan70}{\stan}(\stan)}
\def \biggfnt{\loadfont C{cmmr10 at 20pt}{\biggfnt}(\biggfnt)}
\def \bigggfnt{\loadfont E{cmmr10 at 30pt}{\bigggfnt}(\bigggfnt)}
\def \cmrten{\a}
\def \cmrnine{\b}
\def \cmreight{\c}
\def \cmrseven{\d}
\def \cmsix{\e}
\def \cmrfive{\f}
\def \cmseven{\g}
\def \cmsix{\h}
\def \cmfive{\i}
\def \cmten{\j}
\def \cmss{\k}
\def \cmstencil{\l}
\def \cmstencil{\m}
\def \cmss{\n}
\def \cmss{\o}
\def \cmss{\p}
\def \cmseight{\q}
\def \cmseight{\r}
\def \cmseight{\s}
\def \cmseven{\t}
\def \cmtit{\u}
\def \cmseven{\v}
\def \cmtit{\w}
\def \cmseven{\x}
\def \cmtit{\y}
\def \cmseven{\z}
\def \cmseven{\a}
\def \cmseven{\b}
\def \cmseven{\c}
\def \cmseven{\d}
\def \cmseven{\e}
\def \cmseven{\f}
\def \cmseven{\g}
\def \cmseven{\h}
\def \cmseven{\i}
\def \cmseven{\j}
\def \cmseven{\k}
\def \cmseven{\l}
\def \cmseven{\m}
\def \cmseven{\n}
\def \cmseven{\o}
\def \cmseven{\p}
\def \cmseven{\q}
\def \cmseven{\r}
\def \cmseven{\s}
\def \cmseven{\t}
\def \cmseven{\u}
\def \cmseven{\v}
\def \cmseven{\w}
\def \cmseven{\x}
\def \cmseven{\y}
\def \cmseven{\z}
\def \vbox to 12pt()\i.e., \lower 1pt+.25\vnbox to 2pt+1em()
\def \rm\n\def \s\n\def \bf\n\def \it\n\def \mi\n\def \tt\n\def \sy\n\def \sc\n\def \mathrm\n\def \mathit\n\def \mathsf\n\def \mathbf\n\def \mathcal\n\def \vcenter\n\def \hbox\n\def \mathopen\n\def \mathclose\n\def \hskip 5pt\n\def \usertenpoint\n\def \usetenpoint\n\def \usetenpoint{\gdef\fontsize{\tenpoint}{\usertenpoint}}

% font family definitions
\def \tenpoint{\baselineskip 12pt
  \dispskip 12pt plus 9pt minus 9pt
  \dispaskip Opt plus 4pt
  \dispbskip 7pt plus 3pt minus 4pt
  \def \strut{\lower 3.5pt
  \hbox to 12pt()\i.e., \lower 1pt+.25\vnbox to 2pt+1em()}
  \def \rm\n  \def \s\n  \def \bf\n  \def \it\n  \def \mi\n  \def \tt\n  \def \sy\n  \def \sc\n  \def \mathrm\n  \def \mathit\n  \def \mathsf\n  \def \mathbf\n  \def \mathcal\n  \def \vcenter\n  \def \hbox\n  \def \mathopen\n  \def \mathclose\n  \def \hskip 5pt\n  \def \usertenpoint\n  \def \usetenpoint{\gdef\fontsize{\tenpoint}{\usertenpoint}}
  \def \usetenpoint{\gdef\fontsize{\tenpoint}{\usertenpoint}}
\def \ninepoint{\baselineskip 11pt
  \dispskip 11pt plus 3pt minus 8pt
  \dispaskip 0pt plus 3pt
  \dispbskip 8pt plus 3pt minus 8pt
  \def \strut{\lower 3.25pt\vbox to 11pt()}% see tempoint for explanation
  \def \rm{\:b}
  \def \sf{\:o}
  \def \bf{\:r}
  \def \it{\:i}
  \def \mi{\:h}
  \def \tt{\:t}
  \def \sy{\:v}
  \def \biglp{\mathopen{\hbox{a()}}}
  \def \bigrp{\mathclose{\hbox{a()}}}
  \def \g{\hskip 4.625pt}
  \mathrm bef
  \mathit kkl
  \mathsy wyy
  \rm \:userninepoint
  \def \userninepoint{}
  \def \userninepoint{\\gdef\\textsize{\\ninepoint}{\\ninepoint}}

\def \eigbtpoint{\baselineskip 9.5pt
  \dispskip 9.5pt plus 3pt minus 2pt
  \dispaskip 0pt plus 3pt
  \dispbskip 5pt plus 3pt minus 2pt
  \def \strut{\lower 2.75pt\vbox to 9.5pt()}% see tempoint for explanation
  \def \rm{\:c}
  \def \sl{\:p}
  \def \bf{\:s}
  \def \it{\:i}
  \def \mi{\:j}
  \def \tt{\:u}
  \def \sy{\:w}
  \def \biglp{\mathopen{\hbox{a()}}}
  \def \bigrp{\mathclose{\hbox{a()}}}
  \def \g{\hskip 4.25pt}
  \mathrm cef
  \mathit kkl
  \mathsy wyy
  \rm \:usereightpoint
  \def \usereightpoint{}
  \def \usereightpoint{\\gdef\\textsize{\\eightpoint}{\\eightpoint}}

\mathex 0

% definitions of large parentheses
\def \biglp{\mathopen{\vcenter{\hbox{\char*22}}}}
\def \bigrp{\mathclose{\vcenter{\hbox{\char*23}}}}
\def \bigglp{\mathopen{\vcenter{\hbox{\char*40}}}}
\def \bigggrp{\mathclose{\vcenter{\hbox{\char*41}}}}
% definitions of glue
\def \qquad{\quad}
\def \xskip{\hskip 7pt plus 3pt minus 4pt}
\def \yskip{\vskip 3pt plus 3pt minus 2pt}
\def \yyskip{\vskip 6pt plus 6pt minus 4pt}

Definitions of Odd Characters
\chcode'272'3072 \% this makes formulas like "$x:=x+1$" and "$f: X \to Y$" work

\% ttchar puts the char into a \tt fixed width box
\def \ttchar#1{\save \hbox{#1}\hbox to lwdl(\hskipOpt plusl000pt minusl000Opt #1\hskipOpt plusl000Opt minusl000Opt)}

\% fontclassify selects the right char based on what the current font is
\def \fontclassify#1#2#3{\ifmmode\else\if T\fontcode\else\if U\fontcode\else\if #3#3\else#3\fi\fi\fi\fi\fi\fi\fi
\%#1 is math, #2 is tt, #3 is others

% char macro definitions

\def \down{\fontclassify{\mathrel{\char'43}}{\ttchar{\sy\char'43}}{\sy\char'43}}
Anatomy of a \TeX\ Macro Package
\def \%{{\char'45}}
\def \@{\fontclassify{\char'574}{\char'100}{\sy\char'174}}
  \chcode'100+13
\def \{}{\fontclassify{\mathbin{\char'404}}{\char'134}{\sy\char'404}}
\def \up{\fontclassify{\mathrel{\char'442}}{\char'136}{\sy\char'42}}
\def \larrow{\fontclassify{\mathrel{\char'440}}{\char'137}{\sy\char'40}}
\def \lbrace{\fontclassify{\mathopen{\char'546610}}{\char'173}{\sy\char'146}}
\def \bar{\fontclassify{\char'552614}{\char'174}{\sy\char'152}}
\def \rbrace{\fontclassify{\mathclose{\char'547811}}{\char'176}{\sy\char'147}}
\def \uparrow{\$\up$
\def \sharp{\$
\def \seal{\{\st\}\
\let \space=\% for defining \ to be \hbox{space} in \tt
\def \sp{\tt\char'40}}

\begin{tabular}{|c|c|} 
\hline
Redefinitions of One Character Macros
\hline
\end{tabular}

\begin{tabular}{|c|c|}
\hline
\let \space=\% for defining \ to be \hbox{space} in \tt
\end{tabular}

\begin{tabular}{|c|c|}
\hline
\let \thinspace=\relax
\let \thick=\relax
\let \negthinspace=\relax
\let \negthick=\relax
\let \mathopen=\relax
\let \mathclose=\relax
\let \mathbin=\relax
\let \mathrel=\relax
\let \mathord=\relax
\let \mathpunct=\relax
\let \mathchoice=\relax
\let \mathaccent=\relax
\hline
\end{tabular}

\begin{tabular}{|c|c|}
\hline
% new long names work anywhere
\def \thinspace{\if\mode\@\else \relax\fi}
\def \thickspace{\if\mode\@\else \relax\fi}
\def \negthinspace{\if\mode\@\else \relax\fi}
\def \negthickspace{\if\mode\@\else \relax\fi}
\hline
\end{tabular}

\begin{tabular}{|c|c|}
\hline
\begin{tabular}{|c|c|}
\hline
\end{tabular}
\end{tabular}
\section*{Page Numbering}

% uses two flags:
% \indentmode is T when in \defernode
% \deferredpage is T when there is a piece of a page being deferred

\def\advpagecount{\if \indentmode \advpagecount \setcounto \highestpagenumber \else \if \deferredpage \setcounto \savedpagecount \gdef \deferredpage (F) \else \advpagecount (F) \fi \fi}
\def\deferredpage (F)
\def\indentmode (F)
\def\incpagecount{\gdef \advpagecount \setcounto \highestpagenumber \setq \highestpagenumber 9 \advcount by 1}
\def \decpagecount{\gdef \advpagecount \setcounto \highestpagenumber \setq \highestpagenumber \advcount 9 by 1}
"output, Style, Format Routines"

\def\normal{\resetsize \fontsize \parstyle}
\def\resetsize{\normalsize \normalsize}
\def\everyoutput{} % this is something that is in every output routine

% start of format descriptions
\def\usebookformat{\input mbook}
\def\usebasicformat{\usebasicstyle \usebasicoutput}

% ACM oversize format for Versatec (camera ready copy)
\def\useacmformat{\input macacm}
% To use, code \useacmformat at the start of the paper.
% Then define \title and \authors
% then define the title portion, followed by \endoftitle
% when you are all done \endofpaper\end

\def\useplainoutput{\input plain}
\def\usebasicoutput{\input obasic}
\def\usewoodsoutput{\input owoods}
\def\useblockstyle{\input sblock}
\def\usebasicstyle{\input sbasic}

% look at \useplainoutput and \useblockstyle for the minimum needed
% in output and style routines
% Format routines are simply output and style together. Note that
% other related macros and definitions may be included also.

Footnotes

% normal footnote
\def\footnote#1{\botinsert{\eightpoint\hbox par size(#1\#2)}}

% numbered footnote
\def\footnotenumber#1{\advcountq{\footnotenumber}!\$\footnotenumber\$!}
\def\footnotenumber#1{\botinsert{\eightpoint\hbox par size($#1$)}}
\def\footnotenumber{0}
Anatomy of a \TeX\ Macro Package

\begin{verbatim}
\botsep{\vskip15pt \hrule width5pc\vskip 8pt}

% footnote mark characters
\def\upstar{\lower3pt \hbox{$^{\text{\textasteriskcentered}}}$}}
\def\dagger{\lower2pt \hbox{$^{\text{\textdaggercentered}}}$}}
\def\ddagger{\lower2pt \hbox{$^{\text{\textdaggernamed}}}$}}

\def\hangbox to #1 #2{\par\hangindent #1\noindent
  \hbox to #1(#2)}
\def\levelone#1{\hangbox to 20pt {#1}\hangfill}
\def\leveltwo#1{\hangbox to 40pt {#1}\hangfill}
\def\levelthree#1{\hangbox to 60pt {#1}\hangfill}
\def\number#1{\levelone{#1}}
\def\number#1{\hangbox to 50pt {#1}\hangfill}
\def\indpar#1{\par
  \vspace9\hbox to \textwidth\hfill\vspace{-40pt}
  \vskip9\hbox{\boxQ\hfill\vskip-40pt
  \hsize lwd9
  \vskiplpt
  \leveltwo\par
  \normalhsize
  \vskiplpt}
\def \hdr#1{\par\goodbreak\yyskip\ctrlin{(bf \#1)}\posthdrskip}
\def \posthdrskip{\par\badbreak\vskip24pt plus 12pt minus 6pt}

\def \list#1{\vskip15pt % this centers the bullet. see \levelone
  \vskip9\hbox{\boxQ\hfill\bullet\hfill
  \hsize lwd9
  \vskiplpt
  \leveltwo\par
  \normalhsize
  \vskiplpt}
\def \bitem#1{\levelone{\hangfill\bullet{#1}}}
\def \bitem#1{\vskip9\hbox{\boxQ\hfill\bullet{#1}\hfill
  \hsize lwd9
  \vskiplpt
  \leveltwo\par
  \normalhsize
  \vskiplpt}
\def \subitem#1{\vskip15pt % leaves count in \count9
  \setcounter{\sublistcounter}{\value{\listcounter}+1}
  \leveltwo{\count9}}
\def \subitem#1{\vskip9\hbox{\boxQ\hfill\bullet{#1}\hfill
  \hsize lwd9
  \vskiplpt
  \leveltwo{\count9}}}
\def \subitemindent{\leveltwo}
\def \subitemindent{\leveltwo}
\def \subsubitem#1{\vskip15pt % should be a letter
  \setcounter{\subsublistcounter}{\value{\sublistcounter}+1}
  \leveltwo{\count9}}
\def \subsubitem#1{\vskip9\hbox{\boxQ\hfill\bullet{#1}\hfill
  \hsize lwd9
  \vskiplpt
  \leveltwo{\count9}}}
\def \subsubitemindent{\leveltwo}
\end{verbatim}

\end{verbatim}
\def \subsubitema\{advcountq{\subsublistcounter}\} \\
        \levelthree{\char\subsublistcounter.}\}

\def \subsubitemendindent{\levelthree{}}

* Underlining and Boxes

\def\undertext #1{$\underline{\hbox{#1}}}$ % underline in horizontal mode
\def\overtext #1{$\overline{\hbox{#1}}}$ % overline in horizontal mode
\def \leaderline{(leaders\ rule\ hfill)}

\def\boxit{$\vbox{\hrule\hbox{\vrule\hskip8pt \vbox{\vskip3pt\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule}}}$
\def\boxitnoglue{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

\def\boxitnoindent{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

\def\boxitnoindent{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

\def\Lboxit{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

\def\Lboxitnoglue{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

\def\Lboxitnoindent{\vbox{\hrule\hbox{\vrule\hbox{\hskip3pt\vrule}}\vskip3pt\hrule\vrule}}$

% Lboxit puts L's around box instead of rules
\def\Lboxit to #1 by #2 \#3{\def\hsplitrule{\vbox{\hrule width .25in\hfill \vbox{\hrule width .25in}}}}
\def\vsplitrule{\vbox to #2{\hbox{\vrule height .25in\vfill \hbox{\vrule height .25in}}}}
\vbox{\lineskip Opt \baselineskip Opt \hsplitrule \vbox to #2{\hbox to #1{\vsplitrule \hfill \vbox to #2{\vfill#3\vfill} \hfill \vsplitrule}}} $

\def\type #1\#2{\par\indpar{\displayfont #1{\under(#2}}} % type a line (as in dialogue)
% the second argument is underlined, good for prompts
\def\type #1\#2{\par\noindent{\displayfont #1{\under(#2)\par}} % type a line (as in dialogue)
Penalties

\def\badbreak{\penalty1000}
\def\goodbreak{\penalty-100}
\def\excellentbreak{\penalty-1000}

"nofill "endnofill

% To use, code:
% \nofill  % statements
% \endnofill  
%
% The code is listed verbatim without any page breaks.
% To allow a page break, put \allowbreak on a line. If there
% is no break, a blank line is generated.
%
% Note that \fontsize must be defined to be your normal size of type, such
% as \tenpoint
%
% Font is not changed

% Use of tabs in verbatim mode will give an error message.

% Define \cr to be \CR when enabled
\chcode'15+12\def\{
\chcode'15+5 %

\def\nofill{\par\skip Opt}
\chcode'11+13  % define tab to give an error
\chcode'15+13  % define <return> to generate <cr
\chcode'40+13  % define space to generate <space> (a real space)
\gdef\{\hbox{\space}}  % make space exactly one unshrinkable space
\gdef\CR{\par\goodbreak\noindent\hbox{!}}

\def\endnofill{\par\skip Opt}
\vskip-1pt
\chcode'11+10  % define tab to be a space
\chcode'15+5  % define <return> be a end of line
\chcode'40+10  % define space to be a space
\let\ =\space  % make " \ " as normal
\normal

\def\goodgels{\chcode'34+13
  % \le
\let \le=\le
\chcode'35+13
  % \ge
\let \ge=\ge
Verbatim Mode "startcode and "endcode

\def\normalgle{\chode'34+12} % ≤
\let \le=\negcondthinspace
\chode'35+12 % ≥
\let \ge=\condthinspace

\% To use, code:
\% \startcode
\% statements
\% \endcode
\%
\% The code is listed verbatim without any page breaks.
\% To allow a page break, put \allowbreak on a line. If there
\% is no break, a blank line is generated.
\%
\% \startcode supplies 4 pt of glue
\% \endcode supplies 5 pt of glue
\% The code is printed in \displayfont mode
\% To avoid glue, code \startcodenoglue or \endcodenoglue

\% Note that \fontsize must be defined to be your normal size of type, such
\% as \tenpoint

\% Use of tabs in verbatim mode will give an error message.

\% Define \<<tab>> to be \tab when enabled
\chode'11+12\def\{\tab\chode'11+10\% will cause an error message unless \tab is defined

\def\startcodenoglue{\par
\displayfont
\nofill
\goodgele
}
\def\endcodenoglue{\endnofill
\normalgle
\fontsize}
\def\startcode{\par\Excellentbreak\vsip 5pt plus 1pt minus 1pt\startcodenoglue}
\def\endcode{\endcodenoglue\Excellentbreak\vsip 5pt plus 1pt minus 1pt}
\def\startoutput{\par\Excellentbreak\vsip 5pt plus 1pt minus 1pt\\tenpoint
  \$\down\quad\down\quad\down\quad\down\quad\down\quad\down\quad\down\quad\down\quad\down\quad\par}

\def\allowbreaknoglue{\par\badbreak\vsip-1ipt\Excellentbreak}
\def\allowbreak{\allowbreaknoglue\vsip 1ipt plus 1pt}
Verbatim Mode Using ftft "halignftft"

% To use, code the following:
% \verbimat{
% follow with code
% } terminates verbatim mode.
% Note that \verbimat stuff will not be broken across page boundaries.

% To allow a break, use \noalign{\excellentbreak}%
% or \breakhere%
% Note the absence of spaces in the above.
% Note that the \% is necessary to avoid an extra line generated.

% Note that \fontsize must be defined to be your normal size of type, such
% as \tenpoint

% These macros rely upon the definitions of \cr and \tab on the previous page.
% Use of tabs in verbatim mode will give an error message.

\def\verbimat{\nofill
%\def\CR{\cr\noalign{\tabbreak}}
\goodgele
\verbatingnerate}

\def\verbatingenerating#1{%\displayfont$\halign to size(#1)$
\endnofill
\normalgele
}

\def\breakhere{%\noalign{\excellentbreak\vskip 11 pt}}

\def\threecol{\nofill
%\def\CR{\cr\noalign{\tabbreak}}
\goodgele
\threecolgenerate}

\def\threecolgenerating#1{%\displayfont\halign{#1}$\hfill\hfill\hfill\hfill\cr}$
\endnofill
\normalgele
}

Notes

% \sendnotes creates a list of entries which will be output when
% \putnotes is used. This should be at the end of the manuscript.
% use \initnotes to initialize notes

\def \initnotes{\input mnotes }
Index Macros
\def \initindex{\input index}
% see TUGboat (Vol. 1, No. 1) for an index package.

Defer Mode
\def defer{\input defer}

Table of Contents
\def \begintoc{\input stoc}

Interesting Hacks
\def \ifnull#1\then#2\else#3{\def \jnk{#1}\if ?#2\else\jnk\fi\else#3}\fi
% to use \ifnull {#1}{<true clause>} {<false clause>}
\def \leaders\hrule height 1.5pt \hfill
\def \dnbrace{$\char'772$\bracex$\char'775$
\char'774$\bracex\char'773$
\char'772$\bracex\char'775$}
\def \upbrace{$\char'734$\bracex$\char'773$
\char'774$\bracex\char'772$\bracex$\char'775$}
\def \hbox{\relax \magnify#1\chpar12=#1
% operand is magnification times 1000
\def \done{\par \vfill \end}
% To put a big capital letter beginning a paragraph; #1 = indent for (2 or 3) lines, #2 = letter, #3 = paragraph
\def\capitalpar\#1\#2\#3{\save9\hbox par size(\ragged 1000000
   \if2lllC€l \linebreak 233
   \else(Cl \linebreak 2
   \vskip -1ht9
   \saveQ\hbox<\bigggiat t2)
   \hbox par sizeC\hangindent 1.Wd9 for $1-OP3))
\}

\def\ie{\{sl i.e.\})
\def\eg{\{sl e.g.\})

\begin{itemize}
  \item Default Options
\end{itemize}
\begin{description}
  \item[\displayfont{ninepoint\tt}]
  \setpagecount{1}
\end{description}

\begin{itemize}
  \item Documentation of Use of Counters and Boxes
\end{itemize}
\begin{description}
  \item[Counters and use]
    \% 0 \quad the page number to appear on current page. Valid only in \output,\send,\mark
    \% 1 \quad unused
    \% 2 \quad unused
    \% 3 \quad unused
    \% 4 \quad unused
    \% 5 \quad unused
    \% 6 \quad unused
    \% 7 \quad unused
    \% 8 \quad unused
    \% 9 \quad work value, use this for temporary calculations in a macro
\begin{description}
  \item[Boxes]
    \% 0 \quad unused
    \% 1 \quad used by defer output and macacm
    \% 2 \quad used by defer output and macacm
    \% 3 \quad unused
    \% 4 \quad unused
    \% 5 \quad unused
    \% 6 \quad unused
    \% 7 \quad unused
    \% 8 \quad unused
    \% 9 \quad for temp macro use: \boxtop
The following section consists of external files that are only loaded when needed. As described in the text, this saves on the amount of space needed by these macros in "TEX" itself.

**MBOOK.TEX**

% Book Format
\def \bookoutput{\vbox to 9truein
  {\baselineskip Opt\lineskip Opt % beginning of output routine, resets skips
   \advpagecount % use the correct page number in \send
   \everyoutput
   \if T\tpage % the next is used when tpage is "T" (title pages)
     {\gdef\tpage{F} % reset tpage
     \vskip .7truein % blank space in place of headlines
     \page} % insert the page contents, no page #
   \else\if T\index{\indexoutput}
   \else{\if T\pagenumberarea{\pagenumberregion\vfill}{\else}
     \page % insert the page contents
     {\if B\pagenumberarea{\vfill\pagenumberregion}{\else{}}
   \else{\if F\vfill{}}\else{}}
   \else\if F\vfill{}}
   \if F\vfill{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\else{}}
   \else{}}\el

\def \bookstyle{\maxdepth 2pt
  \parindent 20pt
  \parskip Opt plus 1 pt
  \lineskip 1pt plus 0pt
  \topskip 24pt plus 6pt minus 10pt
  \ botskip 15pt plus 3pt minus 9pt
  \topbaseline Opt
}
% page number definitions
\def\boxpagenumberregion{\moveleft \httruein\vbox to \httruein\hrule\hbox to 6.75truein\trule
% horizontal rule at top of page
\hbox to \httruein\trule
% 20pt((1+sqrt(5))/2=32.50)\pt\fill\count0\cmrten\hfill\count0\cmsss12\topmark\hfill\trule
\ifeven\hbox to 32.361pt{\cmrten\hfill\count0\cmsss12\botmark\hfill\count0\cmrten\trule\fill\count0\cmsss12\topmark\hfill\count0\cmrten\trule}
\else\hbox to 32.361pt{\cmrten\hfill\count0\cmsss12\botmark\hfill\count0\cmrten\trule\fill\count0\cmsss12\topmark\hfill\count0\cmrten\trule}
\trule}
% horizontal rule under the headline
\def\trule{\vrule height 13.5pt depth 6.5pt} % used at top of page
\def\titlemark#12{\mark{\ifeven\titlemark{\else\titlemark{}}}}
\def\pagenumberregion{\boxpagenumberregion}

% "global variables"
\def\tpage{F}
\def\index{F}
\def\titlepage{\gdef\tpage{T}} % \titlepage sets tpage to T

% enable book format
\def\usebookformat{\gdef\standardoutput{\output{\bookoutput}}
\standardoutput
\gdef\parstyle{\bookstyle}
\gdef\normalhsize{\hsize 6.5truein}
\gdef\normalvsize{\vsize 8.3truein}
\normal}
\usebookformat

% chapter section
\def\chapternumber{0}
\def\chapterbegin#1{\par
\gdef\footnotenumber{0}
\gdef\countq{\chapternumber}
\gdef\sectionnumber{0}
\gdef\wholesectionnumber{Chapter \chapternumber}
\titlemark{wholesectionnumber}{\sectionname}
\vfill\eject
\gdef\sectionname{#1}
\titlemark{wholesectionnumber}{#1}
(\noindent \cmss12 \wholesectionnumber \#1}
{\noindent \cmss12 \wholesectionnumber \#1}
{\noindent \cmss12 \wholesectionnumber \#1}
\gdef\footnotenumber{0}
\gdef\countq{\chapternumber}
\gdef\sectionnumber{0}
\gdef\wholesectionnumber{Chapter \chapternumber}
\titlemark{wholesectionnumber}{\sectionname}
\vfill\eject
\gdef\sectionname{#1}
\titlemark{wholesectionnumber}{#1}
(\noindent \cmss12 \wholesectionnumber \#1}
\if T\writetoc{\send2{\chaptertoc{#1}\count0.}}\else{}
\posthdrskip}
% automatic table of contents generation
\def\inittocf#1{\open2 #1
\gdef\writetocf(T)\{ % write tocf info
\def\writetocf(F}
MACACM.TEX

% ACM two column format for Versatec
\def\acmoutput{\eve.youtput
\if T\tpage
 \{\if T\column
 {\gdef\normalhsize{\hsize 4.25truein}
 \gdef\normalvsize{\vsize 8.9truein}
 \normalhsize\normalvsize
 \save\page\gdef\column{L}
 }
 \else{\if L\column
 {\save\page\gdef\column{R})
 {\else{\vbox to 11.9truein{\box1\vskip -1000pt plus 1000000pt
 \hbox to 9 truein{\box2\hfill\page})
 \advcount 0
 \gdef\column{L}
 \gdef\tpage{F}
 \gdef\normalvsize{\vsize 11.5truein}
 \normalvsize
 }
 }
 \} \else{\if L\column
 {\save\page\gdef\column{R})
 \else{\vbox to 11.9 truein{\hbox to 9truein\n ninepoint{\ifeven0
 {\rm\lastnames\hfill\sl\title
 \else{\sl\title\hfill\rm\lastnames}
 \vfill
 \hbox to 9 truein{\box2\hfill\page})
 \advcount0
 \gdef\column{L}
 }
 \} \}
 {\def \acmstyle{\maxdepth 2pt
 \parindent 20pt
 \parskip 0pt plus 1 pt
 \lineskip 1pt plus 0pt
 \topskip 24pt plus 6pt minus 10pt
 \botskip 15pt plus 3pt minus 0pt
 \topbaseline 0pt
 }

 \def\endoftitle{\par\vfill\eject}
 \def\endofpaper{\par\vfill\if L\column{\eject\hbox{}\vfill}\else{}\}}

% To use, code \useacmformat at the start of the paper.
% Then define \title and \authors
% then define the title portion, followed by \endoftitle
% when you are all done \endofpaper\end

% enable acm format
\def\standardoutput{\output{\acmoutput}}
\def\useacmformat{\standardoutput
  \gdef\parstyle{\bookstyle}
  \gdef\normalhsize{\hsize 9truein}
  \gdef\smallhsize{\hsize 9truein} % too small to indent right
  \gdef\normalvsize{\vsize 9truein}
  % these sizes redefined in \endoftitle
  % and \acmoutput

\normal
\gdef\tpage{T}
\gdef\column{T}
}

\useacmformat

---

**OPLAIN.TEX**

% Plain Output routine

\def\plainoutput{\advpagecount % use the correct page number in \send
  \page
  \everyoutput}

\def\standardoutput{\output{\plainoutput}}

\def\useplainoutput{\standardoutput
  \gdef\normalhsize{\hsize 6.5truein}
  \gdef\normalvsize{\vsize 9truein}
  \normal
}

\useplainoutput

---

**OBASIC.TEX**

% Basic output routine

\def\basicoutput{\advpagecount % use the correct page number in \send
  \vbox to 9truein{\page
    \vfill
    \ctrline{\cmrten \countO}}
  \everyoutput}

\def\standardoutput{\output{\basicoutput}}

\def\usebasicoutput{\standardoutput
  \gdef\normalhsize{\hsize 6.5truein}
  \gdef\normalvsize{\vsize 8.75truein}
  \normal
}


\usebasicoutput

---

WOODS.TEX

% Woods output
% (To look like previous versions of the annual report.)
\def\Woodsoutput{\advpagecount
  \vbox to 9truein{\ctrline€\ninepoint\sl -\count0-}
  \vfill
  \page}
\everyoutput>

\def\standardoutput{\output{\Woodsoutput}}

\def\useWoodsoutput{\standardoutput}
  \gdef\normalhsizes{\hsize 8.5truein}
  \gdef\normalvsize{\vsize 8.75truein}
  \normal
}

\useWoodsoutput

---

SBLOCK.TEX

% Block Style
\def\blockstyle{\maxdepth 2pt
  \parindent 0pt
  \parskip 6 pt plus 6 pt minus 2 pt  % Skip a line between paragraphs.
  \lineskip 1pt plus 0pt
  \topskip 24pt plus 6pt minus 10pt
  \botskip 15pt plus 3pt minus 9pt
  \topbaseline 0pt
}

\def\useblockstyle{\gdef\parstyle{\blockstyle}
  \normal
}

\useblockstyle
SBASIC.TEX

% Basic Style
\def\basicstyle{\maxdepth 2pt
  \parindent 20pt
  \parskip Opt
  \lineskip ipt plus Opt
  \topskip 24pt plus 6pt minus 10pt
  \botskip 15pt plus 3pt minus 6pt
  \topbaseline Opt
}
\def\usebasicstyle{\gdef\parstyle{\basicstyle}
  \normal
}
\usebasicstyle

MNOTES.TEX

% notes
\openi=fixnot.tex
\def \sendnotes#1{\sendi{Page \count0. #1\par}}
\def \putnotes{\specialbegin{Fixup Notations}
  \openi=dumyi.tmp % Close the fixnot file
  \input fixnot.tex % Now put text here.
}

DEFER.TEX

% defermode based on that written by Brent Hailpern and Jim Boyce
% box 1 is slop on current page
% box 2 is extra slop on current page that will go on following page
\def\defer{\save2\vbox{)
  % no extra slop yet
  \output{\save1\page}\output{\save2\page}} % cause stuff to be saved
  \eject % flush out current page
  \standardoutput
  \ifdimen lht2>0pt{\unbox\save1\box2}\else{} % comment below
  % put out full page and copy partial page
  \if F\deferredpage{\gdef\deferredpage{T}
    \addvspace{\count0\unfootnotesskipamount}
    \savevspace{\count0\unfootnotesskipamount}
  \else{}
}}
\gdef\indefermode{T}
}
\def\enddefer{\eject
 \unbox
 \gdef\indefermode{F}
}
\def\savethepagecount{\setq{\savedpagecount}{9}}
\defer \% do it this time too!

MTOFC.TEX

\% table of contents
\def\begin{toc}{\gdef\chapternumber{0}}
 \setpagecount{-1} \% initial page number for cover page
 \special\begin{Table of Contents}

\def\chaptertoc{1}#1.\par
 \advcountq{\chapternumber}
 \gdef\sectionnumber{0}
 \hbox to size{\hbox to 30pt{\bf chapternumber\ hfill}\#1
 \leaders\hrule\hfill\hbox to 20pt{\hfill\#1}}

\def\sectiontoc{1}#1.\par
 \advcountq{\sectionnumber}
 \gdef\subsectionnumber{0}
 \hbox to size{\hbox to 45pt{\bf chapternumber.\ sectionnumber\ hfill}\#1
 \leaders\hrule\hfill\hbox to 20pt{\hfill\#1}}

\def\subsectiontoc{1}#1.\par
 \advcountq{\subsectionnumber}
 \gdef\dimsectionnumber{0}
 \hbox to size{\hbox to 60pt{\bf chapternumber.\ sectionnumber.\ subsectionnumber\ hfill}\#1
 \leaders\hrule\hfill\hbox to 20pt{\hfill\#1}}

\def\dimsectiontoc{1}#1.\par
 \advcountq{\dimsectionnumber}
 \hbox to size{\hbox to 75pt{\bf chapternumber.\ sectionnumber.\ subsectionnumber.\ dimsectionnumber
 \hfill}\#1}
 \leaders\hrule\hfill\hbox to 20pt{\hfill\#1}}

\def\end{toc}{\par\vfill\eject \% put out this page before screwing up page \#
 \gdef\chapternumber{0}
 \setpagecount{0}
}
\begin{toc} \% do it now too!