

1. Preamble

By default, LitTeX typesets the file name on the title page. This is after the path has been removed. No other processing of the name is performed. So the name will be in the case the user typed on the command line. LitTeX.mac should be typeset slightly different. The filename is just littex.mac. So we override that definition here. Also this section is used to generate the version and date-time stamp. This is automatically done using the TLib version control system. This is done by using combination of %! and %# formatting lines. None of the commands actually appear in the formatted output.

LitTeX is a relatively simple macro that is based on the TeX macro.

```
\def\LitTeX{Lit\TeX}
```

Typesetting a backslash is a pain in text mode. \backslash is only defined to work in math mode. \BS simplifies the problem. It takes a parameter so it is easier to override the font that token names use. In many cases it is desirable to typeset them in monospaces. That can easily be done via a trivial modification to this macro. I prefer to let the token name typeset in roman.

```
\def\BS#1{{\tt\char'134}#1}
```

Just a little meaningless macro to test to see if active characters are properly indexed.

```
\def\IndexTestOfActive{Just~some~junk~text.}
```

The running head is currently a constant. It is just the \JobName. In a future version it may be desirable to print the current major or minor section in the running head.

```
\headline={\tenit%
  \ifodd\pageno%
    \hfil\JobName\rlap{\bf\quad\folio}%
  \else
    \llap{\bf\folio\quad}\JobName\hfil%
  \fi}
```

```
\footline={\hfil}
```

2 *LitTeX.mac*

2. **Fonts.**

Here is where any fonts that are needed are defined. Currently, LitTeX uses just the standard set of Computer Modern fonts loaded by plain. It is relatively easy to override this selection. If that is desired, here is the place to do it.

```
\font\titelfont=cmr7\magstep4
```

```
% title on the contents page
```

3. Character Definitions.

This section defines the replacement code sequences for the special characters in TeX. Some of the characters can be simple chardefs. Others require the character to be selected from a font that has them. In this case the monospaced font is used.

```

\def\A{{\tt\char"5C}}           % the backslash character "\
\def\B{{\tt\char"7B}}           % left curly brace "{"
\def\C{{\tt\char"7D}}           % right curly brace "}"

\chardef\D="24                   % dollar sign "\$
\chardef\E="26                   % ampersand "&"
\chardef\F="23                   % number sign "#"

\def\G{{\tt\char"0B}}           % up arrow "\uparrow"
\def\H{{\tt\char"5F}}           % underscore "\_"

\chardef\I="7E                   % tilde "\~"
\chardef\J="25                   % percent "\%"

% an escaped carriage return is relatively rare. To
% get one requires a line to end with an escape char.
\def\K{\hookleftarrow}         % the carriage return

```

4 *LitT_EX*.mac

4. Indentation.

Leading indentation requires a lot of special handling. The macros to actually output the result are quite simple.

```
\newdimen\IndentDimen
\IndentDimen=5pt
\def\1#1{\hskip#1\IndentDimen}
```

4.1. The Space Character. If a space or tab is embedded in a macro this needs to be shown. *cmtt10* has a character designed specifically for this purpose.

```
\def\S{\tt\char"20}
```

4.2. Blank Lines. Multiple blank lines are processed by the *LitT_EX* program. When it is appropriate, blank lines are added to the output via the `\BL` command. Currently this is let equal to a medium break.

```
\let\BL=\medbreak
```

5. Comments.

All percent characters, % are processed. Depending on the location in on the line comments are processed differently. If they occur in the left most position then the line is output using \ComL. Otherwise they are set left flush by \Com. Lines are prepared using the \L macro.

```
\def\ComL#1{\it\%#1}
```

```
\def\Com#1{\hfill\it\%#1}
```

```
\def\L#1{\hbox_{to}_{\hspace{#1\hfil}}
```

6. Rules.

Multiple percent characters, %, in the left hand margin are used to simulate rules. 64 or more will generate a 2 half point rules. 16 or more will generate one half point rule. 4 or more will generate a hairline rule. The macros for rules do play a few tricks. Each rule issues a negative penalty. This means that it is a good place to break the page. Because of the way the rules are generated, if the rule would fall at the top of the page it is suppressed. This is done via vertical leaders.

6.1. Hairline Rules. Hairline rules are only one tenth of a point thick. They have a penalty of -200.

```
\def\HairLine{%
  \par%
  \begingroup%
    \offinterlineskip%
    \setbox0=\vbox{\vskip2pt\line{\leaders\hrule_\height_\!1pt\hfil}\vskip2pt}%
    \penalty-200\bigskip%
    \cleaders\copy0\vskip\ht0%
    \bigskip%
  \endgroup}
```

6.2. SingleHeavyLine. The rules are half a point wide. The penalty is -300.

```
\def\SingleHeavyLine{%
  \par%
  \begingroup%
    \offinterlineskip%
    \setbox0=\vbox{\vskip2pt\line{\leaders\hrule_\height_\!5pt\hfil}\vskip2pt}%
    \penalty-300\bigskip%
    \cleaders\copy0\vskip\ht0%
    \bigskip%
  \endgroup}
```

6.3. DoubleHeavyLine. The rules are half a point wide. The penalty is -400.

```
\def\DoubleHeavyLine{%
  \par%
  \begingroup%
    \offinterlineskip%
    \setbox0=\vbox{\vskip2pt\line{\leaders\hrule_\height_\!1pt\hfil}\vskip2pt}%
    \penalty-400\bigskip%
    \cleaders\copy0\vskip2\ht0%
    \bigskip%
  \endgroup}
```

7. Table of Contents Setup.

To generate a table of contents an output file needs to be created.

```
\newwrite\TOCAux
\openout\TOCAux=\jobname.aux
\def\ReadTOC{\input\jobname.aux}
```

7.1. Table of Contents Tricks. Generating the table of contents requires a few minor tricks. First we need to globally fake some the macros that are used in the table of contents.

```
\let\TocMajor=\let
\let\TocMinor=\let
```

```
\def\TOCNumber#1#2{\hbox_{to_2em}{\hfil#1}.\hbox_{to_2.5em}{#2\hfil}}
\def\TOCLeaders{\leaders\hbox_{to_2.5em}{.\hfil}\hfil}
```

```
\def\RealTocMajor_#1[#2][#3]{%
  \line{\TOCNumber{#1}{#2}\TOCLeaders#3}}
```

```
\def\RealTocMinor_#1[#2][#3][#4]{%
  \line{\TOCNumber{#1}{#2}.\quad#3\TOCLeaders#4}}
```

8. Section Macros.

Major and Minor section macros are quite simple. Major sections force a page eject. Minor macros do a good break. Major sections force comments to start on the next line. Minor sections allow text to run-in.

The table of contents is generated using the Major and Minor section entries.

```
\def\MajorSection#1#2{%
  \vfil\eject%
                                     % little trick stolen from The TeXBook page 332, answer to 21.10
  {\let\the=0\edef\next{\write\TOCAux{\TocMajor[#1][#2][\the\count0]}\next}%
  \L{\bf_#1.\quad#2}\noindent\ignorespaces}
```

```
\def\MinorSection#1#2#3{%
  \bigbreak%
                                     % little trick stolen from The TeXBook page 332, answer to 21.10
  {\let\the=0\edef\next{\write\TOCAux{\TocMinor[#1][#2][#3][\the\count0]}\next}%
  \noindent{\bf_#1.#2.\quad#3}}
```

9. End of File.

`\EndOfFile` is the last line in the file. This macro closes the table of contents file, prepares the definitions to process the table of contents file, and reads in the table of contents file

```
\outer\def\EndOfFile{%
  \par\vfill\supereject%
  \let\TocMajor=\RealTocMajor%
  \let\TocMinor=\RealTocMinor%
  \headline={\hfil}
  \pageno=0
  \vbox\to\vsizel%
    \vfill%
    \centerline{\titlefont\JobName}%
    \centerline{\titlefont\Version}%
    \centerline{\titlefont\Dated}%
    \vfill%
    \ReadTOC\relax%
  \par\vfill}
\supereject%
\end}
```

% Now the contents appears on page 0.

% There needs to be a default value for Version and Dated.

```
\def\Version{}
\def\Dated{}
```

10. Index Macros.

The index processing quite simple. One possible enhancement would be to support printing the page number. This would require two passes over the data.

```

\def\Index{%
  \write\TOCAux{}                               % Ensure that the table of contents file isn't empty
  \closeout\TOCAux%
  \doublecolumns%
  \parindent=0pt}

\newdimen\ICalc

\def\IEntry#1#2{{\parfillskip=0pt\hangafter=1\hangindent=3em\relax_#1\TOCLeaders#2\par}}

\def\EndIndex{\singlecolumn}

```

11. Column Macros

Going from single to double column and back can be done many ways. This code is stolen from eplain.tex.

```
% make @ a character
\catcode'@=11

% define some variables and assign them default values

\newskip\abovedoublecolumnskip_\abovedoublecolumnskip_\bigskipamount
\newskip\belowdoublecolumnskip_\belowdoublecolumnskip_\bigskipamount
\newdimen\gutter_\gutter_\=2pc
\newdimen\doublecolumnhsize
\newbox\@partialpage
\newdimen\singlecolumnhsize
\newdimen\singlecolumnvsize
\newtoks\previousoutput
```

11.1. Double Columns Going from single columns to double columns is not difficult. First some simple calculations and assignments. The previous output routine is saved. Then the page is flushed. Then a new output routine is assigned as well as the the various sizes.

The eplain macros set the vertical size to twice the normal length and then split it at the end.

```
\def\doublecolumns{%
  \doublecolumnhsize_\= \hsize
  \par
  \previousoutput_\=\expandafter{\the\output}%
  \advance\doublecolumnhsize_\by_\- \gutter
  \divide\doublecolumnhsize_\by_2
  \output_\={%
    \global\setbox\@partialpage_\=
      \vbox{\unvbox255\vskip\abovedoublecolumnskip}%
  }%
  \pagegoal_\=\pagetotal
  \break
  \output_\={\doublecolumnoutput}%
  \singlecolumnhsize_\=\hsize
  \singlecolumnvsize_\=\vsize
  \hsize_\=\doublecolumnhsize
  \vsize_\=2\vsize
}%
```

*% If \hsize changed, get the new value.
% Shouldn't start in horizontal mode.*

% Now expand the \output just above.

11.2. Splitting the Column The long output column is split.

```
\def\@doublecolumnsplit{%
  \splittopskip_\=\topskip
  \splitmaxdepth_\=\maxdepth
  \dimen0_\=\singlecolumnvsize
  \advance\dimen0_\by_\- \ht\@partialpage
  \advance\dimen0_\by_\- \ht\footins
  \ifvoid\footins\else_\advance\dimen0_\by_\- \skip\footins_\fi
  \advance\dimen0_\by_\- \ht\topins
  \ifvoid\topins\else_\advance\dimen0_\by_\- \skip\topins_\fi
  \begingroup
  \vbadness_\=10000
  \global\setbox1_\=\vsplit255_\to_\dimen0
  \wd1_\=\hsize
```

```

    \global\setbox3=\vsplit255 to \dimen0
    \wd3=\hsize
  \endgroup
  \global\setbox4=\vbox{\unvbox255\penalty\outputpenalty}%
  \global\setbox255
    =\vbox{%
      \unvbox\@partialpage
      \hbox to \singlecolumnhsz{\box1\hfil\box3}%
    }%
}%

```

11.3. Double Column Output Routine. Splits the output and rebuilds the output box. Then calls the original output routine.

```

\def\doublecolumnoutput{%
  \@doublecolumnsplit
  \hsz=\singlecolumnhsz % Local to the \output group.
  \vsz=\singlecolumnvsz
  \the\previousoutput
  \unvbox4
}%

```

11.4. Single Column Restores from double column mode. First must flush out the partial double column page. Then restores it into the output stream. Finally restores the dimensions and original output routine.

```

\def\singlecolumn{%
  \par % Shouldn't start in horizontal mode.
  \output=\{\global\setbox1=\box255}%
  \pagegoal=\pagetotal
  \break % Exercise the page builder, i.e., \output.
  \setbox255=\box1 % Retrieve what the fake \output set.
  \begingroup
    \singlecolumnvsz=\ht\@partialpage
    \advance\singlecolumnvsz by \ht\footins
    \ifvoid\footins\else\advance\singlecolumnvsz by \skip\footins\fi
    \advance\singlecolumnvsz by \ht\topins
    \ifvoid\topins\else\advance\singlecolumnvsz by \skip\topins\fi
    \dimen0=\ht255\divide\dimen0 by 2
    \advance\singlecolumnvsz by \dimen0
    \advance\singlecolumnvsz by .5\baselineskip
    \@doublecolumnsplit
  \endgroup
  \hsz=\singlecolumnhsz
  \vsz=\singlecolumnvsz
  \output=\expandafter{\the\previousoutput}%
  \unvbox255
  \vskip\belowdoublecolumnskip
  \nointerlineskip
}%

```

% restore @ to being an other.

`\catcode'@=12`

% at signs are no longer letters

12. Index.

A complete cross reference of all the macros used.

<code>\@doublecolumnsplit</code>	11.2, 11.3, 11.4	<code>\font</code>	2
<code>\@partialpage</code>	11, 11.1, 11.2, 11.4	<code>\footins</code>	11.2, 11.4
<code>\A</code>	3	<code>\footline</code>	1
<code>\abovedoublecolumnskip</code>	11, 11.1	<code>\G</code>	3
<code>\advance</code>	11.1, 11.2, 11.4	<code>\global</code>	11.1, 11.2, 11.4
<code>\B</code>	3	<code>\gutter</code>	11, 11.1
<code>\baselineskip</code>	11.4	<code>\H</code>	3
<code>\begingroup</code>	6.1, 6.2, 6.3, 11.2, 11.4	<code>\HairLine</code>	6.1
<code>\belowdoublecolumnskip</code>	11, 11.4	<code>\hangafter</code>	10
<code>\bf</code>	1, 8	<code>\hangindent</code>	10
<code>\bigbreak</code>	8	<code>\hbox</code>	5, 7.1, 11.2
<code>\bigskip</code>	6.1, 6.2, 6.3	<code>\headline</code>	1, 9
<code>\bigskipamount</code>	11	<code>\hfil</code>	1, 5, 6.1, 6.2, 6.3, 7.1, 9, 11.2
<code>\BL</code>	4.2	<code>\hfill</code>	5
<code>\box</code>	11.2, 11.4	<code>\hookleftarrow</code>	3
<code>\break</code>	11.1, 11.4	<code>\hrule</code>	6.1, 6.2, 6.3
<code>\BS</code>	1	<code>\hspace</code>	5, 11.1, 11.2, 11.3, 11.4
<code>\C</code>	3	<code>\hskip</code>	4
<code>\catcode</code>	11, 11.4	<code>\ht</code>	6.1, 6.2, 6.3, 11.2, 11.4
<code>\centerline</code>	9	<code>\I</code>	3
<code>\char</code>	1, 3, 4.1	<code>\ICalc</code>	10
<code>\chardef</code>	3	<code>\IEntry</code>	10
<code>\cleaders</code>	6.1, 6.2, 6.3	<code>\ifodd</code>	1
<code>\closeout</code>	10	<code>\ifvoid</code>	11.2, 11.4
<code>\Com</code>	5	<code>\ignorespaces</code>	8
<code>\ComL</code>	5	<code>\IndentDimen</code>	4
<code>\copy</code>	6.1, 6.2, 6.3	<code>\Index</code>	10
<code>\count</code>	8	<code>\IndexTestOfActive</code>	1
<code>\D</code>	3	<code>\input</code>	7
<code>\Dated</code>	9	<code>\it</code>	5
<code>\def</code>	1, 3, 4, 4.1, 5, 6.1, 6.2, 6.3, 7, 7.1, 8, 9, 10, 11.1, 11.2, 11.3, 11.4	<code>\J</code>	3
<code>\dimen</code>	11.2, 11.4	<code>\JobName</code>	1, 9
<code>\divide</code>	11.1, 11.4	<code>\jobname</code>	7
<code>\doublecolumnhsize</code>	11, 11.1	<code>\K</code>	3
<code>\doublecolumnoutput</code>	11.1, 11.3	<code>\L</code>	5, 8
<code>\doublecolumns</code>	10, 11.1	<code>\leaders</code>	6.1, 6.2, 6.3, 7.1
<code>\DoubleHeavyLine</code>	6.3	<code>\let</code>	4.2, 7.1, 8, 9
<code>\E</code>	3	<code>\line</code>	6.1, 6.2, 6.3, 7.1
<code>\edef</code>	8	<code>\LitTeX</code>	1
<code>\eject</code>	8	<code>\llap</code>	1
<code>\else</code>	1, 11.2, 11.4	<code>\magstep</code>	2
<code>\end</code>	9	<code>\MajorSection</code>	8
<code>\endgroup</code>	6.1, 6.2, 6.3, 11.2, 11.4	<code>\maxdepth</code>	11.2
<code>\EndIndex</code>	10	<code>\medbreak</code>	4.2
<code>\EndOfFile</code>	9	<code>\MinorSection</code>	8
<code>\expandafter</code>	11.1, 11.4	<code>\newbox</code>	11
<code>\F</code>	3	<code>\newdimen</code>	4, 10, 11
<code>\fi</code>	1, 11.2, 11.4	<code>\newskip</code>	11
<code>\folio</code>	1	<code>\newtoks</code>	11
		<code>\newwrite</code>	7

<code>\next</code>	8	<code>\skip</code>	11.2, 11.4
<code>\noindent</code>	8	<code>\splitmaxdepth</code>	11.2
<code>\nointerlineskip</code>	11.4	<code>\splittopskip</code>	11.2
<code>\offinterlineskip</code>	6.1, 6.2, 6.3	<code>\supereject</code>	9
<code>\openout</code>	7	<code>\tenit</code>	1
<code>\outer</code>	9	<code>\TeX</code>	1
<code>\output</code>	11.1, 11.3, 11.4	<code>\the</code>	8, 11.1, 11.3, 11.4
<code>\outputpenalty</code>	11.2	<code>\titlefont</code>	2, 9
<code>\pagegoal</code>	11.1, 11.4	<code>\TOCAux</code>	7, 8, 10
<code>\pageno</code>	1, 9	<code>\TOCLeaders</code>	7.1, 10
<code>\pagetotal</code>	11.1, 11.4	<code>\TocMajor</code>	7.1, 8, 9
<code>\par</code>	6.1, 6.2, 6.3, 9, 10, 11.1, 11.4	<code>\TocMinor</code>	7.1, 8, 9
<code>\parfillskip</code>	10	<code>\TOCNumber</code>	7.1
<code>\parindent</code>	10	<code>\topins</code>	11.2, 11.4
<code>\penalty</code>	6.1, 6.2, 6.3, 11.2	<code>\topskip</code>	11.2
<code>\previousoutput</code>	11, 11.1, 11.3, 11.4	<code>\tt</code>	1, 3, 4.1
<code>\quad</code>	1, 7.1, 8	<code>\unvbox</code>	11.1, 11.2, 11.3, 11.4
<code>\ReadTOC</code>	7, 9	<code>\vbadness</code>	11.2
<code>\RealTocMajor</code>	7.1, 9	<code>\vbox</code>	6.1, 6.2, 6.3, 9, 11.1, 11.2
<code>\RealTocMinor</code>	7.1, 9	<code>\Version</code>	9
<code>\relax</code>	9, 10	<code>\vfil</code>	8
<code>\rlap</code>	1	<code>\vfill</code>	9
<code>\S</code>	4.1	<code>\vsize</code>	9, 11.1, 11.3, 11.4
<code>\setbox</code>	6.1, 6.2, 6.3, 11.1, 11.2, 11.4	<code>\vskip</code>	6.1, 6.2, 6.3, 11.1, 11.4
<code>\singlecolumn</code>	10, 11.4	<code>\vsplit</code>	11.2
<code>\singlecolumnhsize</code>	11, 11.1, 11.2, 11.3, 11.4	<code>\wd</code>	11.2
<code>\singlecolumnvsize</code>	11, 11.1, 11.2, 11.3, 11.4	<code>\write</code>	8, 10
<code>\SingleHeavyLine</code>	6.2	<code>~</code>	1, 3

Lit \TeX .mac
Version 0.1
(23-Feb-94 0:51:04)

1.	Preamble.....	1
2.	Fonts.....	2
3.	Character Definitions.....	3
4.	Indentation.	4
4.1.	The Space Character.....	4
4.2.	Blank Lines.....	4
5.	Comments.	5
6.	Rules.....	6
6.1.	Hairline Rules.....	6
6.2.	SingleHeavyLine.....	6
6.3.	DoubleHeavyLine.....	6
7.	Table of Contents Setup.	7
7.1.	Table of Contents Tricks.....	7
8.	Section Macros.....	8
9.	End of File.....	9
10.	Index Macros.....	10
11.	Column Macros.....	11
11.1.	Double Columns.....	11
11.2.	Splitting the Column.....	11
11.3.	Double Column Output Routine.....	12
11.4.	Single Column.....	12
12.	Index.....	13