HOW TO LaTeX A THESIS

by

Eric R. L. Benedict

A preliminary report submitted in partial fulfillment of
the requirements for the degree of

Master TeXnician

(Electrical Engineering)

at the

UNIVERSITY OF WISCONSIN–MADISON

2000

To my pet rock, Skippy.

ACKNOWLEDGMENTS

I thank the many people who have done lots of nice things for me.

DISCARD THIS PAGE

TABLE OF CONTENTS

Page

DISCARD THIS PAGE

LIST OF TABLES

Table Page

DISCARD THIS PAGE

LIST OF FIGURES

DISCARD THIS PAGE

NOMENCLATURE

TEX	a typesetting system by Donald Knuth [2]. It also refers to the "plain" format. The proper pronounciation rhymes with "heck" and "peck" and does not sound like "hex" or "Rex."
LATEX	a set of TEX macros originally written by Leslie Lamport [3]. The proper pronunciation is `lā·tek'` and not `lā'·teks` (see above).
BIBTEX	a bibliography generation program by Oren Patashnik [3] that can be used with either plain TEX or LATEX.
C_1	Constant 1
V	Voltage
$	US Dollars

HOW TO LaTeX A THESIS

Eric R. L. Benedict

Under the supervision of Assistant Professor Bucky J. Badger

At the University of Wisconsin-Madison

This is not a thesis or dissertation and Master TeXnician is not a degree granted at the University of Wisconsin-Madison.

This explains the basics for using LaTeX to typeset a dissertation, thesis or masters project or preliminary report for the University of Wisconsin-Madison. Chapter 1 talks briefly about the thesis formatting at UW-Madison. Chapter 2 gives an overview of the "essentials" of LaTeX and was written by Jon Warbrick. Chapter 3 talks about figures and tables and what a *float* is. Chapter 4 briefly introduces the BibTeX program. And finally, Chapter 5 discusses some of the details for using the `withesis` style file. The material in Chapters 2-4 basically are a review of fundamental LaTeX usage and form a reasonable basic tutorial.

The style discussed in this manual was originally written by Dave Kraynie and edited by James Darrell McCauley as the `puthesis` style for Purdue University's theses. This style was modified to form the `withesis` style. This manual is largely based on a similar manual by James Darrell McCauley and Scott Hucker. Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is here by granted. This software and its documentation is provided "as is" without any express or implied warranty.

Bucky J. Badger

ABSTRACT

This is not a thesis or dissertation and Master TEXnician is not a degree granted at the University of Wisconsin-Madison.

This explains the basics for using LATEX to typeset a dissertation, thesis or masters project or preliminary report for the University of Wisconsin-Madison. Chapter 1 talks briefly about the thesis formatting at UW-Madison. Chapter 2 gives an overview of the "essentials" of LATEX and was written by Jon Warbrick. Chapter 3 talks about figures and tables and what a *float* is. Chapter 4 briefly introduces the BIBTEX program. And finally, Chapter 5 discusses some of the details for using the `withesis` style file. The material in Chapters 2-4 basically are a review of fundamental LATEX usage and form a reasonable basic tutorial.

The style discussed in this manual was originally written by Dave Kraynie and edited by James Darrell McCauley as the `puthesis` style for Purdue University's theses. This style was modified to form the `withesis` style. This manual is largely based on a similar manual by James Darrell McCauley and Scott Hucker. Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is here by granted. This software and its documentation is provided "as is" without any express or implied warranty.

Chapter 1

Introducing the `withesis` LaTeX Style Guide

This manual is was written to test the `withesis` style file and to provide documentation for this style file.

1.1 History

The idea for this came from a similar manual written by James Darrell McCauley and Scott Hucker in 1993 for the Purdue University thesis style file. Content ideas were liberally borrowed from this document. The `withesis` style file is based on the Purdue thesis file written by Dave Kraynie and edited by Darrell McCauley. This base was edited to meet the format requirements of the University of Wisconsin–Madison and several additional new commands were created. In addition, environments from the UW Mathematics Department were also incorporated.

1.2 Producing Your Thesis or Dissertation

The `withesis` style file will take care of most of the formatting requirements for submitting your thesis or dissertation at the University of Wisconsin-Madison. There are some requirements on the printing of your document. From the Graduate School's *UW-Madison Guide To Preparing Your Doctoral Dissertation*,

> Print your dissertation on a laser printer. (Some high quality dot-matrix printers may be acceptable.) The printer must produce output that meets all format and legibility requirements. A professional copy shop can produce an acceptable copy to be submitted to the Graduate School. Some copiers enlarge the original between one and two percent. To avoid problems with margins, produce the original copy with margins larger than the required minimum. Look carefully at the copy before paying for the

services and ask for pages to be recopied if necessary. Common flaws are: smudges, copy lines, specks, missing pages, margin shifts, slanting of the printed image on the page, and poor paper quality.

1.2.1 Required Paper

The paper which is used for PhD Dissertations should be:

- 8-1/2 x 11 inches
- High-quality, white
- 20 pound weight, bond

While for Masters Theses, the paper should be:

- 8-1/2 x 11 inches
- White
- Acid-free or pH neutral
- 20 pound weight
- 25% cotton bond minimum

Paper that meets these requirements can be purchased at book and stationery stores.

1.2.2 Copyright Page

If you choose to retain and register copyright of the dissertation, prepare a copyright page using the `withesis \copyrightpage` command. Center the text in the bottom third of the page within the dissertation margins. This page is not numbered. There is an additional fee for copyrighting your dissertation which is payable at the bursars office along with the microfilming and binding fee.

1.2.3 Prechecks

The Graduate School has reserved 9:00-9:30 each morning to answer specific formatting questions (for example: use of tables, graphs and charts). You may bring in 8-10 pages to be reviewed. No appointment is necessary.

1.2.4 Final Checks

For information about the final Graduate School review and about depositing your dissertation in the library, see *The Three D's: Deadlines, Defending, Depositing Your Doctoral Dissertation* or look at the web site

`http://www.wisc.edu/grad/gs/degrees/ddd.html`

1.3 Disclaimer

This software and documentation is provided "as is" without any express or implied warranty. While care has been taken by the authors of this style file such that the final product will probably meet the University of Wisconsin's formatting requirements this is not guaranteed.

Chapter 2

Essential LaTeX

This chapter introduces some key ideas behind LaTeX and give you the "essential" items of information. This chapter is an edited form of the paper "Essential LaTeX" by Jon Warbrick, Plymouth Polytechnic.

2.1 Introduction

This document is an attempt to give you all the essential information that you will need in order to use the LaTeX Document Preparation System. Only very basic features are covered, and a vast amount of detail has been omitted. In a document of this size it is not possible to include everything that you might need to know, and if you intend to make extensive use of the program you should refer to a more complete reference. Attempting to produce complex documents using only the information found below will require much more work than it should, and will probably produce a less than satisfactory result.

The main reference for LaTeX is *The LaTeX User's guide and Reference Manual* by Leslie Lamport. This contains most of the information that you will ever need to know about the program, and you will need access to a copy if you are to use LaTeX seriously. You should also consider getting a copy of *The LaTeX Companion*

2.2 How does LaTeX work?

In order to use LaTeX you generate a file containing both the text that you wish to print and instructions to tell LaTeX how you want it to appear. You will normally create this file using your

system's text editor. You can give the file any name you like, but it should end "`.TEX`" to identify the file's contents. You then get LaTeX to process the file, and it creates a new file of typesetting commands; this has the same name as your file but the "`.TEX`" ending is replaced by "`.DVI`". This stands for '*D*evice *I*ndependent' and, as the name implies, this file can be used to create output on a range of printing devices. Your *local guide* will go into more detail.

Rather than encourage you to dictate exactly how your document should be laid out, LaTeX instructions allow you describe its *logical structure*. For example, you can think of a quotation embedded within your text as an element of this logical structure: you would normally expect a quotation to be displayed in a recognisable style to set it off from the rest of the text. A human typesetter would recognise the quotation and handle it accordingly, but since LaTeX is only a computer program it requires your help. There are therefore LaTeX commands that allow you to identify quotations and as a result allow LaTeX to typeset them correctly.

Fundamental to LaTeX is the idea of a *document style* that determines exactly how a document will be formatted. LaTeX provides standard document styles that describe how standard logical structures (such as quotations) should be formatted. You may have to supplement these styles by specifying the formatting of logical structures peculiar to your document, such as mathematical formulae. You can also modify the standard document styles or even create an entirely new one, though you should know the basic principles of typographical design before creating a radically new style.

There are a number of good reasons for concentrating on the logical structure rather than on the appearance of a document. It prevents you from making elementary typographical errors in the mistaken idea that they improve the aesthetics of a document—you should remember that the primary function of document design is to make documents easier to read, not prettier. It is more flexible, since you only need to alter the definition of the quotation style to change the appearance of all the quotations in a document. Most important of all, logical design encourages better writing. A visual system makes it easier to create visual effects rather than a coherent structure; logical design encourages you to concentrate on your writing and makes it harder to use formatting as a substitute for good writing.

2.3 A Sample LaTeX file

Have a look at the example LaTeX file in Figure 2.1. It is a slightly modified copy of the standard LaTeX example file `SMALL.TEX`. The line numbers down the left-hand side are not part of the file, but have been added to make it easier to identify various portions.

Try entering this file (without the line numbers), save the text as `small.tex`, next run LaTeX on it, and then view the output:

```
% latex small
% xdvi small                # displays the output on the screen
% dvips -o small.ps small   # to create a PostScript file, small.ps
% lp -d<printer> small.ps   # to print
```

2.3.1 Running Text

Most documents consist almost entirely of running text—words formed into sentences, which are in turn formed into paragraphs—and the example file is no exception. Describing running text poses no problems, you just type it in naturally. In the output that it produces, LaTeX will fill lines and adjust the spacing between words to give tidy left and right margins. The spacing and distribution of the words in your input file will have no effect at all on the eventual output. Any number of spaces in your input file are treated as a single space by LaTeX, it also regards the end of each line as a space between words (see lines 15–17). A new paragraph is indicated by a blank line in your input file, so don't leave any blank lines unless you really wish to start a paragraph.

LaTeX reserves a number of the less common keyboard characters for its own use. The ten characters

```
# $ % & ~ _ ^ \ { }
```

should not appear as part of your text, because if they do LaTeX will get confused.

```
 1: % SMALL.TEX -- Released 5 July 1985
 2: % USE THIS FILE AS A MODEL FOR MAKING YOUR OWN LaTeX INPUT FILE.
 3: % EVERYTHING TO THE RIGHT OF A  %  IS A REMARK TO YOU AND IS IGNORED
 4: % BY LaTeX.
 5: %
 6: % WARNING!  DO NOT TYPE ANY OF THE FOLLOWING 10 CHARACTERS EXCEPT AS
 7: % DIRECTED:        &   $   #   %   _   {   }   ^   ~   \
 8:
 9: \documentclass[11pt,a4]{article}  % YOUR INPUT FILE MUST CONTAIN THESE
10: \begin{document}                  % TWO LINES PLUS THE \end COMMAND AT
11:                                   % THE END
12:
13: \section{Simple Text}          % THIS COMMAND MAKES A SECTION TITLE.
14:
15: Words are separated by one or    more      spaces.  Paragraphs are
16:     separated by one or more blank lines.  The output is not affected
17: by adding extra spaces or extra blank lines to the input file.
18:
19:
20: Double quotes are typed like this: ``quoted text''.
21: Single quotes are typed like this: `single-quoted text'.
22:
23: Long dashes are typed as three dash characters---like this.
24:
25: Italic text is typed like this: {\em this is italic text}.
26: Bold   text is typed like this: {\bf this is  bold  text}.
27:
28: \subsection{A Warning or Two}        % THIS MAKES A SUBSECTION TITLE.
29:
30: If you get too much space after a mid-sentence period---abbreviations
31: like etc.\ are the common culprits)---then type a backslash followed by
32: a space after the period, as in this sentence.
33:
34: Remember, don't type the 10 special characters (such as dollar sign and
35: backslash) except as directed!  The following seven are printed by
36: typing a backslash in front of them:  \$  \&  \#  \%  \_  \{  and  \}.
37: The manual tells how to make other symbols.
38:
39: \end{document}                   % THE INPUT FILE ENDS LIKE THIS
```

Figure 2.1 A Sample LaTeX File

2.3.2 LaTeX Commands

There are a number of words in the file that start '\' (see lines 9, 10 and 13). These are LaTeX *commands* and they describe the structure of your document. There are a number of things that you should realize about these commands:

- All LaTeX commands consist of a '\' followed by one or more characters.
- LaTeX commands should be typed using the correct mixture of upper- and lower-case letters. `\BEGIN` is *not* the same as `\begin`.
- Some commands are placed within your text. These are used to switch things, like different typestyles, on and off. The `\em` command is used like this to emphasize text, normally by changing to an *italic* typestyle (see line 25). The command and the text are always enclosed between '`{`' and '`}`'—the '`{\em`' turns the effect on and and the '`}`' turns it off.
- There are other commands that look like

  ```
  \command{text}
  ```

  In this case the text is called the "argument" of the command. The `\section` command is like this (see line 13). Sometimes you have to use curly brackets '`{}`' to enclose the argument, sometimes square brackets '`[]`', and sometimes both at once. There is method behind this apparent madness, but for the time being you should be sure to copy the commands exactly as given.
- When a command's name is made up entirely of letters, you must make sure that the end of the command is marked by something that isn't a letter. This is usually either the opening bracket around the command's argument, or it's a space. When it's a space, that space is always ignored by LaTeX. We will see later that this can sometimes be a problem.

2.3.3 Overall structure

There are some LaTeX commands that must appear in every document. The actual text of the document always starts with a `\begin{document}` command and ends with an `\end{document}` command (see lines 10 and 39). Anything that comes after the `\end{document}` command is ignored. Everything that comes before the `\begin{document}` command is called the *preamble*. The preamble can only contain LaTeX commands to describe the document's style.

One command that must appear in the preamble is the `\documentclass` command (see line 9). This command specifies the overall style for the document. Our example file is a simple technical document, and uses the `article` class. The document you are reading was produced with the `withesis` class. There are other classes that you can use, as you will find out later on in this document.

2.3.4 Other Things to Look At

LaTeX can print both opening and closing quote characters, and can manage either of these either single or double. To do this it uses the two quote characters from your keyboard: ` and '. You will probably think of ' as the ordinary single quote character which probably looks like ´ or ' on your keyboard,

and ` as a "funny" character that probably appears as `. You type these characters once for single quote (see line 21), and twice for double quotes (see line 20). The double quote character " itself is almost never used and should not be used unless you want your text to look "funny" (compare the quote in the previous sentence).

LaTeX can produce three different kinds of dashes. A long dash, for use as a punctuation symbol, as is typed as three dash characters in a row, like this '---' (see line 23). A shorter dash, used between numbers as in '10–20', is typed as two dash characters in a row, while a single dash character is used as a hyphen.

From time to time you will need to include one or more of the LaTeX special symbols in your text. Seven of them can be printed by making them into commands by proceeding them by backslash (see line 36). The remaining three symbols can be produced by more advanced commands, as can symbols that do not appear on your keyboard such as †, ‡, §, £, ©, ♯ and ♣.

It is sometimes useful to include comments in a LaTeX file, to remind you of what you have done or why you did it. Everything to the right of a % sign is ignored by LaTeX, and so it can be used to introduce a comment.

2.4 Document Classes and Class Options

There are four standard document classes available in LaTeX:

`article` intended for short documents and articles for publication. Articles do not have chapters, and when `\maketitle` is used to generate

a title (see Section 2.9) it appears at the top of the first page

rather than on a page of its own.

`report` intended for longer technical documents. It is similar to `article`, except that it contains chapters and the title appears on a page of its own.

`book` intended as a basis for book publication. Page layout is adjusted assuming that the output will eventually be used to print on both sides of the paper.

`letter` intended for producing personal letters. This style will allow you to produce all the elements of a well laid out letter: addresses, date, signature, etc.

An additional document class, the one used for this document and for University of Wisconsin–Madison theses, is `withesis`.

These standard classes can be modified by a number of *class options*. They appear in square brackets after the `\documentclass` command. Only one class can ever be used but you can have more than one class option, in which case their names should be separated by commas. The standard style options are:

`11pt` prints the document using eleven-point type for the running text rather that the ten-point type normally used. Eleven-point type is about ten percent larger than ten-point.

`12pt` prints the document using twelve-point type for the running text rather than the ten-point type normally used. Twelve-point type is about twenty percent larger than ten-point.

`twoside` causes documents in the article or report styles to be formatted for printing on both sides of the paper. This is the default for the book style.

`twocolumn` produces two column on each page.

`titlepage` causes the `\maketitle` command to generate a title on a separate page for documents in the `article` style. A separate page is always used in both the `report` and `book` styles.

The University of Wisconsin–Madison thesis style, `withesis` also has some class options defined. These class options are for ten-point type (`10pt`), tweleve-point type (`12pt`), two-sided printing (`twoside`), Master Thesis margins (`msthesis`) and an option to print a small black box on lines which exceed the margins (`margincheck`).

2.5 Environments

We mentioned earlier the idea of identifying a quotation to LaTeX so that it could arrange to typeset it correctly. To do this you enclose the quotation between the commands `\begin{quotation}` and `\end{quotation}`. This is an example of a LaTeX construction called an *environment*. A number of special effects are obtained by putting text into particular environments.

2.5.1 Quotations

There are two environments for quotations: `quote` and `quotation`. `quote` is used either for a short quotation or for a sequence of short quotations separated by blank lines:

```
US presidents ... remarks:
\begin{quote}
The buck stops here.

I am not a crook.
\end{quote}
```

US presidents have been known for their pithy remarks:

> The buck stops here.
>
> I am not a crook.

Use the `quotation` environment for quotations that consist of more than one paragraph. Paragraphs in the input are separated by blank lines as usual:

```
Here is some advice to remember:
\begin{quotation}
Environments for making
...other things as well.

Many problems
...environments.
\end{quotation}
```

Here is some advice to remember:

> Environments for making quotations can be used for other things as well.
>
> Many problems can be solved by novel applications of existing environments.

2.5.2 Centering and Flushing

Text can be centered on the page by putting it within the `center` environment, and it will appear flush against the left or right margins if it is placed within the `flushleft` or `flushright` environments.

Text within these environments will be formatted in the normal way, in particular the ends of the lines that you type are just regarded as spaces. To indicate a "newline" you need to type the `\\` command. For example:

```
\begin{center}
one
two
three \\
four \\
five
\end{center}
```

one two three

four

five

2.5.3 Lists

There are three environments for constructing lists. In each one each new item is begun with an `\item` command. In the `itemize` environment the start of each item is given a marker, in the `enumerate` environment each item is marked by a number. These environments can be nested within each other in which case the amount of indentation used is adjusted accordingly:

```
\begin{itemize}
\item Itemized lists are handy.
\item However, don't forget
  \begin{enumerate}
  \item The 'item' command.
  \item The 'end' command.
  \end{enumerate}
\end{itemize}
```

- Itemized lists are handy.
- However, don't forget
  1. The 'item' command.
  2. The 'end' command.

The third list making environment is `description`. In a description you specify the item labels inside square brackets after the `\item` command. For example:

```
Three animals that you should
know about are:
\begin{description}
  \item[gnat] A small
            animal...
  \item[gnu] A large
           animal...
  \item[armadillo] A ...
\end{description}
```

Three animals that you should know about are:

gnat A small animal that causes no end of trouble.

gnu A large animal that causes no end of trouble.

armadillo A medium-sized animal.

2.5.4 Tables

Because LaTeX will almost always convert a sequence of spaces into a single space, it can be rather difficult to lay out tables. See what happens in this example

```
\begin{flushleft}
Income  Expenditure Result   \\
20s 0d  19s 11d     happiness \\
20s 0d  20s 1d      misery  \\
\end{flushleft}
```

Income Expenditure Result
20s 0d 19s 11d happiness
20s 0d 20s 1d misery

The `tabbing` environment overcomes this problem. Within it you set tabstops and tab to them much like you do on a typewriter. Tabstops are set with the `\=` command, and the `\>` command

moves to the next stop. The \\ command is used to separate each line. A line that ends \kill produces no output, and can be used to set tabstops:

```
\begin{tabbing}
Income \=Expenditure \=    \kill
Income \>Expenditure \>Result \\
20s 0d \>19s 11d \>Happiness \\
20s 0d \>20s 1d  \>Misery    \\
\end{tabbing}
```

Income	Expenditure	Result
20s 0d	19s 11d	Happiness
20s 0d	20s 1d	Misery

Unlike a typewriter's tab key, the \> command always moves to the next tabstop in sequence, even if this means moving to the left. This can cause text to be overwritten if the gap between two tabstops is too small.

2.5.5 Verbatim Output

Sometimes you will want to include text exactly as it appears on a terminal screen. For example, you might want to include part of a computer program. Not only do you want LaTeX to stop playing around with the layout of your text, you also want to be able to type all the characters on your keyboard without confusing LaTeX. The verbatim environment has this effect:

```
The section of program in
question is :
\begin{verbatim}
{ this finds %a & %b }

for i := 1 to 27 do
  begin
  table[i] := fn(i);
  process(i)
  end;
\end{verbatim}
```

The section of program in question is:

```
{ this finds %a & %b }

for i := 1 to 27 do
   begin
   table[i] := fn(i);
   process(i)
   end;
```

The withesis document style also provides the command \verbatimfile{foo.fe} which will read in the file foo.fe into the document in verbatim format with the font \tt. See Appendix A for an example.

2.6 Type Styles

We have already come across the `\em` command for changing typeface. Here is a full list of the available typefaces:

`\rm`	Roman	`\it`	*Italic*	`\sc`	SMALL CAPS
`\em`	*Emphatic*	`\sl`	*Slanted*	`\tt`	`Typewriter`
`\bf`	**Boldface**	`\sf`	Sans Serif		

Remember that these commands are used *inside* a pair of braces to limit the amount of text that they effect. In addition to the eight typeface commands, there are a set of commands that alter the size of the type. These commands are:

`\tiny`	`\small`	`\large`	`\huge`
`\scriptsize`	`\normalsize`	`\Large`	`\Huge`
`\footnotesize`		`\LARGE`	

2.7 Sectioning Commands and Tables of Contents

Technical documents, like this one, are often divided into sections. Each section has a heading containing a title and a number for easy reference. LaTeX has a series of commands that will allow you to identify different sorts of sections. Once you have done this LaTeX takes on the responsibility of laying out the title and of providing the numbers.

The commands that you can use are:

`\chapter`	`\subsection`	`\paragraph`
`\section`	`\subsubsection`	`\subparagraph`

The naming of these last two is unfortunate, since they do not really have anything to do with ‘paragraphs’ in the normal sense of the word; they are just lower levels of section. In most document styles, headings made with `\paragraph` and `\subparagraph` are not numbered. `\chapter` is not available in document style `article`. The commands should be used in the order given, since sections are numbered within chapters, subsections within sections, etc.

A seventh sectioning command, `\part`, is also available. Its use is always optional, and it is used to divide a large document into series of parts. It does not alter the numbering used for any of the other commands.

Including the command `\tableofcontents` in you document will cause a contents list to be included, containing information collected from the various sectioning commands. You will notice that each time your document is run through LaTeX the table of contents is always made up of the headings from the previous version of the document. This is because LaTeX collects information for the table as it processes the document, and then includes it the next time it is run. This can sometimes mean that the document has to be processed through LaTeX twice to get a correct table of contents.

2.8 Producing Special Symbols

You can include in you LaTeX document a wide range of symbols that do not appear on you your keyboard. For a start, you can add an accent to any letter:

ò	`\‘{o}`	õ	`\~{o}`	ǒ	`\v{o}`	o̧	`\c{o}`	ó	`\’{o}`
ō	`\={o}`	ő	`\H{o}`	ọ	`\d{o}`	ô	`\^{o}`	ȯ	`\.{o}`
o͡o	`\t{oo}`	o̲	`\b{o}`						
ö	`\"{o}`	ŏ	`\u{o}`						

A number of other symbols are available, and can be used by including the following commands:

†	`\dag`	§	`\S`	©	`\copyright`
‡	`\ddag`	¶	`\P`	£	`\pounds`
œ	`\oe`	Œ	`\OE`	æ	`\AE`
Æ	`\AE`	å	`\aa`	Å	`\AA`
ø	`\o`	Ø	`\O`	ł	`\l`
Ł	`\E`	ß	`\ss`	¿	`?‘`
¡	`!‘`	...	`\ldots`	LaTeX	`\LaTeX`

There is also a `\today` command that prints the current date. When you use these commands remember that LaTeX will ignore any spaces that follow them, so that you can type ‘`\pounds 20`’ to

get '£20'. However, if you type '`LaTeX is wonderful`' you will get 'LaTeXis wonderful'—notice the lack of space after LaTeX. To overcome this problem you can follow any of these commands by a pair of empty brackets and then any spaces that you wish to include, and you will see that `\LaTeX{} really is wonderful!` (LaTeX really is wonderful!).

2.9 Titles

Most documents have a title. To title a LaTeX document, you include the following commands in your document, usually just after `begin{document}`.

```
\title{required title}
\author{required author}
\date{required date}
\maketitle
```

If there are several authors, then their names should be separated by `\and`; they can also be separated by `\\` if you want them to be centred on different lines. If the `\date` command is left out, then the current date will be printed.

```
\title{Essential \LaTeX}
\author{J Warbrick \and An Other}
\date{14th February 1988}
\maketitle
```

Essential LaTeX

J Warbrick A N Other

14th February 1988

The exact appearance of the title varies depending on the document style. In styles `report` and `book` the title appears on a page of its own. In the `article` style it normally appears at the top of the first page, the style option `titlepage` will alter this (see Section 2.4). In the `withesis` style, the title is created on a seperate page in the format appropriate to a UW-Madison thesis or dissertation.

2.10 Errors

When you create a new input file for LaTeX you will probably make mistakes. Everybody does, and it's nothing to be worried about. As with most computer programs, there are two sorts of

mistake that you can make: those that LaTeX notices and those that it doesn't. To take a rather silly example, since LaTeX doesn't understand what you are saying it isn't going to be worried if you mis-spell some of the words in your text. You will just have to accurately proof-read your printed output. On the other hand, if you mis-spell one of the environment names in your file then LaTeXwon't know what you want it to do.

When this sort of thing happens, LaTeX prints an error message on your terminal screen and then stops and waits for you to take some action. Unfortunately, the error messages that it produces are rather user-unfriendly and a little frightening. Nevertheless, if you know where to look they will probably tell you where the error is and went wrong.

Consider what would happen if you mistyped `\begin{itemize}` so that it became `\begin{itemie}`. When LaTeX processes this instruction, it displays the following on your terminal:

```
LaTeX error.  See LaTeX manual for explanation.
              Type  H <return>  for immediate help.
! Environment itemie undefined.
\@latexerr ...for immediate help.}\errmessage {#1}
                                                  \endgroup
l.140 \begin{itemie}

?
```

After typing the '?' LaTeX stops and waits for you to tell it what to do.

The first two lines of the message just tell you that the error was detected by LaTeX. The third line, the one that starts '!' is the *error indicator*. It tells you what the problem is, though until you have had some experience of LaTeX this may not mean a lot to you. In this case it is just telling you that it doesn't recognise an environment called `itemie`. The next two lines tell you what LaTeX was doing when it found the error, they are irrelevant at the moment and can be ignored. The final line is called the *error locator*, and is a copy of the line from your file that caused the problem. It start with a line number to help you to find it in your file, and if the error was in the middle of a line it will be shown broken at the point where LaTeX realised that there was an error. LaTeX can

sometimes pass the point where the real error is before discovering that something is wrong, but it doesn't usually get very far.

At this point you could do several things. If you knew enough about LaTeX you might be able to fix the problem, or you could type 'X' and press the return key to stop LaTeX running while you go and correct the error. The best thing to do, however, is just to press the return key. This will allow LaTeX to go on running as if nothing had happened. If you have made one mistake, then you have probably made several and you may as well try to find them all in one go. It's much more efficient to do it this way than to run LaTeX over and over again fixing one error at a time. Don't worry about remembering what the errors were—a copy of all the error messages is being saved in a *log* file so that you can look at them afterwards.

If you look at the line that caused the error it's normally obvious what the problem was. If you can't work out what you problem is look at the hints below, and if they don't help consult Chapter 6 of the manual [3]. It contains a list of all of the error messages that you are likely to encounter together with some hints as to what may have caused them.

Some of the most common mistakes that cause errors are

- A mispelled command or environment name.
- Improperly matched '`{`' and '`}`'—remember that they should always come in pairs.
- Trying to use one of the ten special characters `# $ % & _ { } ~ ^` and `\` as an ordinary printing symbol.
- A missing `\end` command.
- A missing command argument (that's the bit enclosed in '`{`' and '`}`').

One error can get LaTeX so confused that it reports a series of spurious errors as a result. If you have an error that you understand, followed by a series that you don't, then try correcting the first error—the rest may vanish as if by magic. Sometimes LaTeX may write a `*` and stop without an error message. This is normally caused by a missing `\end{document}` command, but other errors can cause it. If this happens type `\stop` and press the return key.

Finally, LaTeX will sometimes print *warning* messages. They report problems that were not bad enough to cause LaTeX to stop processing, but nevertheless may require investigation. The most common problems are ‘overfull’ and ‘underfull’ lines of text. A message like:

```
Overfull \hbox (10.58649pt too wide) in paragraph at lines 172--175
[]\tenrm Mathematical for-mu-las may be dis-played. A dis-played
```

indicates that LaTeX could not find a good place to break a line when laying out a paragraph. As a result, it was forced to let the line stick out into the right-hand margin, in this case by 10.6 points. Since a point is about 1/72nd of an inch this may be rather hard to see, but it will be there none the less.

This particular problem happens because LaTeX is rather fussy about line breaking, and it would rather generate a line that is too long than generate a paragraph that doesn’t meet its high standards. The simplest way around the problem is to enclose the entire offending paragraph between `\begin{sloppypar}` and `\end{sloppypar}` commands. This tells LaTeX that you are happy for it to break its own rules while it is working on that particular bit of text.

Alternatively, messages about “Underfull `\hbox’es’’` may appear. These are lines that had to have more space inserted between words than LaTeX would have liked. In general there is not much that you can do about these. Your output will look fine, even if the line looks a bit stretched. About the only thing you could do is to re-write the offending paragraph!

2.11 A Final Reminder

You now know enough LaTeX to produce a wide range of documents. But this document has only scratched the surface of the things that LaTeX can do. This entire document was itself produced with LaTeX (with no sticking things in or clever use of a photocopier) and even it hasn’t used all the features that it could. From this you may get some feeling for the power that LaTeX puts at your disposal.

Please remember what was said in the introduction: if you **do** have a complex document to produce then **go and read the manual**. You will be wasting your time if you rely only on what you have read here.

Chapter 3

Figures and Tables

This chapter[1] shows some example ways of incorporating tables and figures into LaTeX. Special environments exist for tables and figures and are special because they are allowed to *float*—that is, LaTeX doesn't always put them in the exact place that they occur in your input file. An algorithm is used to place the floating environments, or floats, at locations which are typographically correct. This may cause endless frustration if you want to have a figure or table occur at a specific location. There are a few methods for solving this.

You can exert some influence on LaTeX's float placement algorithm by using *float position specifiers*. These specifiers, listed below, tell LaTeX what you prefer.

`h`	"here"	do not move this object
`p`	"page"	put this object on a page of floats
`b`	"bottom"	put this object at the bottom of a page
`t`	"top"	put this object at the top of a page

Any combination of these can be used:

```
\begin{figure}[htbp]
 ...
\caption{A Figure!}
\end{figure}
```

In this example, we asked LaTeX to "put the figure 'here' if possible. If it is not possible (according to the rule encoded in the float algorithm), put it on the next float page. A float page is

[1] Most of the text in this chapter's introduction is from *How to TeX a Thesis: The Purdue Thesis Styles*

a page which contains nothing but floating objects, *e.g.* a page of nothing but figures or tables. If this isn't possible, try to put it at the 'top' of a page. The last thing to try is to put the figure at the 'bottom' of a page."

The remainder of this chapter deals with some examples of what to put into the figure, the ellipsis (. . .) in the example above.

3.1 Tables

Table 3.1 is an example table from the UW Math Department. The code to generate it is as

Table 3.1 PDE solve times, $15^3 + 1$ equations.

Precond.	Time	Nonlinear Iterations	Krylov Iterations	Function calls	Precond. solves
None	1260.9u (21:09)	3	26	30	0
FFT	983.4u (16:31)	2	5	8	7

follows:

```
\begin{table}[htbp]
\centering
\caption{PDE solve times, $15^3+1$
equations.\label{pde.tab1}}
\begin{tabular}{||l|l|l|l|l|l||}\hline
Precond. & Time & Nonlinear & Krylov
& Function & Precond. \\
 & & Iterations & Iterations & calls & solves \\ \hline
None & 1260.9u & 3 & 26 & 30 & 0  \\
 &(21:09) & & & &  \\ \hline
FFT  & 983.4u & 2  & 5  & 8  & 7 \\
&(16:31) & & & & \\ \hline
\end{tabular}
\end{table}
```

3.2 Figures

There are many different ways to incorporate figures into a LaTeX document. LaTeX has an internal `picture` environment and some programs will generate files which are in this format and can be simply `included`. In addition to LaTeX native `picture` format, additional packages can be loaded in the `\documentstyle` command (or using the `input` command) to allow LaTeX to process non-native formats such as PostScript.

3.2.1 `gnuplot`

The graph of Figure 3.1 was created by gnuplot. For simple graphs this is a great utility. For example, if you want a sin curve in your thesis try the following:

```
(terminal window): gnuplot
(in gnuplot):
                set terminal latex
                set output "foo.tex"
                plot sin(x)
                quit
```

This will generate a file called `foo.tex` which can be read in with the following statements.

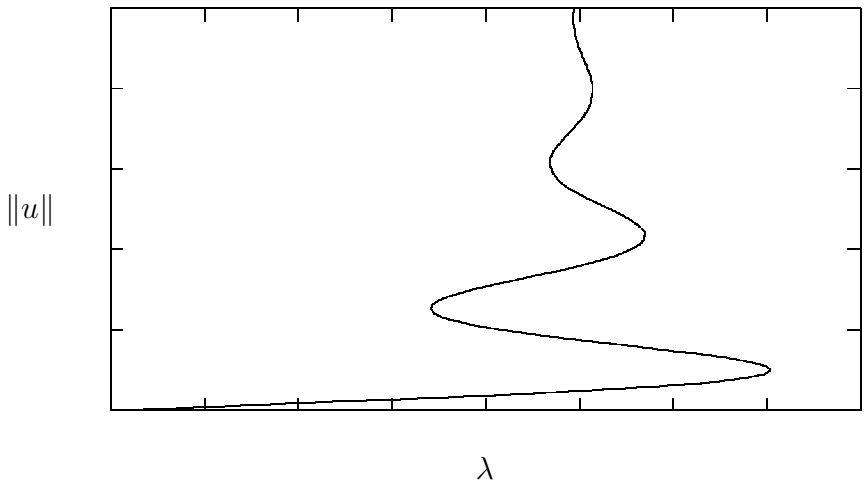

Figure 3.1 Gelfand equation on the ball, $3 \leq n \leq 9$.

```
\begin{figure}[htbp]
\centering
\input{fig2.tex}
\caption{Gelfand equation on the ball, $3\leq n \leq 9$.
\label{gelfand.fig2}}
\end{figure}
```

One advantage to using the native LaTeX `picture` environment is that the fonts will be assured to agree and the pictures can be viewed in the `.dvi` viewer.

3.2.2 PostScript

Many drawing applications now allow the export of a graphic to the *Encapsulated PostScript* format. These files have a suffix of `.EPS` or `.EPSF` and are similar to a regular PostScript file except that they contain a *bounding box* which describes the dimensions of the figure.

In order to include PostScript figures, the `epsfig` (or `psfig` depending on the system you are using) style file must be included in either the `\documentstyle` command or the preamble using the `input` command.

Figure 3.2 is a plot from Matlab. The commands to include this figure are

```
\begin{figure}[htbp]
\centerline{
\psfig{figure=vwcontr.ps,width=5in,angle=0}
          }
\caption{$\sigma$ as a Function of Voltage and Speed, $\alpha = 20$}
\label{vwcontr}
\end{figure}
```

Observe that the `\psfig` command allows the scaling of the figure by setting either the `width` or `height` of the figure. If only one dimension is specified, the other is computed to keep the same aspect ratio. The figure can also be rotated by setting `angle` to the desired value in degrees.

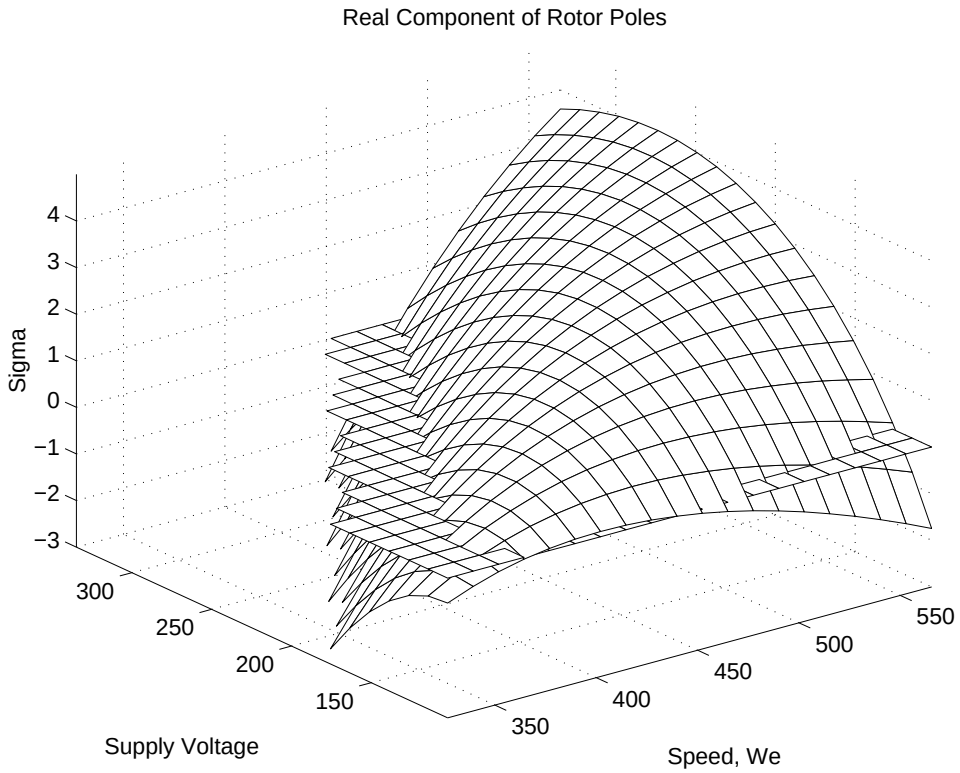

Figure 3.2 σ as a Function of Voltage and Speed, $\alpha = 20$

Chapter 4

Citations and Bibliographies

This chapter is an edited form of the same chapter from *How to TEX a Thesis: The Purdue Thesis Styles* by James Darrell McCauley and Scott Hucker.

The task of compiling and formatting the sources cited in papers can be quite tedious, especially for large documents like theses. A program separate from LaTeX, called "BibTeX,"can be used to automate this task [3].

4.1 The Citation Command

When referring to the work of someone else, the `\cite` command is used. This generates the citation in the text for you. In the above paragraph, the command `\cite{lamport}` was used after the word "task." The formatting of your citation is handled by either the document style or a style option. The default citation style uses the number system (a number in square brackets). Other citation styles may use the author-date system, (Lamport, 1986) or the superscript[3] system.

4.2 Bibliography Styles

The way that a reference is formatted in your bibliography depends on the bibliography style, which is specified near the beginning of your document with the `\bibliographstyle{file}` command. The file `file.bst` is the name of the bibliography style file. Standard BibTeX bibliography style files include `plain`, `unsrt`, `alpha`, and `abbrev`. The bibliography style governs whether or not references are sorted, whether first names or initials are used for authors, whether or not last names are listed first, the location of the year in the references

(after the author or at the end of the reference), *etc.*. You may be required by your department or major professor to follow as style for a particular journal. If so, then you will need to find a BIBTEX style file to suit your needs. Most major journals have style files. If you cannot locate an appropriate BIBTEX style file, then choose the one which is closest and then edit the `.bbl` file by hand. See Section 4.4 for a brief discussion on the `.bbl` file. Some common, but non-standard BIBTEX styles include

`acm.bst`	The Association for Computing Machinery
`ieeetr.bst`	The *IEEE Transactions* style
`jacs-new.bst`	*Journal of the American Chemical Society*

4.3 The Database

The `\bibliography{file}` command is placed in your input file at the location where the "List of References" section[1] would be. It specifies the name (or names) of your bibliographic data base, `file.bib`. An example entry in a BIBTEX database is:

```
@book{ lamport86 ,
     author =    "Leslie Lamport" ,
     title =     "\LaTeX: A Document Preparation System" ,
     publisher = "Addison--Wesley Pub.\ Co." ,
     year =      "1986" ,
     address =   "Reading, MA"
}
```

The citation key is the first field in this entry— citing this book in a LATEX file would look like

```
According to Lamport~\cite{lamport86} ...
```

The tilde (`~`) is used to tie the word "Lamport" to the citation generated. The space between these words is then unbreakable—the word "Lamport" and the citation [3] will not be split across two lines if they happen to occur near the end of a line.

A listing of all entry types with their required and optional fields is given in Appendix B. There are several tools which exist to help in editing a BIBTEX file, however, their use is beyond the scope

[1] or "Bibliography" if `\altbibtitle` has been specified in the preamble.

of this manual and can be found by searching the net. You can simply use a plain text editor like `vi` or `WordPad` to edit and create the database files.

There are several rules which you must follow when creating your database. Authors are always listed by their full names, first name first, and multiple authors are separated by `and`. For example

```
author = "John Jay Park and Frederick Gene Watson and
          Michelle Catherine Smith",
```

If you were using `abbrv` as your `bibliographystyle`, a reference for these authors may look like:

> J.J. Park, F.G. Watson, and M.C. Smith ...

Some styles only capitalize the first word of the title. If you use any acronyms or other words that should always be capitalized in titles, then they should be enclosed in {}'s (*e.g.*, {`Fortran`}, {`N`}`ewton`). This protects the case of these characters.

There are several other rules for BIBTEX listed in [3] which should be referred to because they are not discussed here.

4.4 Putting It All Together

To aid the reader in understanding how all of this works together, the following excerpt was taken from Lamport [3]:

> When you ran LATEX with the input file `sample.tex`, you may have noticed that LATEX created a file named `sample.aux`. This file, called an *auxiliary* file, contains cross-referencing information. Since `sample.tex` contains no cross-referencing commands, the auxiliary file it produces has no information. However, suppose that LATEX is run with an input file named `myfile.tex` that has citations and bibliography-making [or referencing] commands. The auxiliary file `myfile.aux` that it produces will contain all of the citation keys and the arguments of the `\bibliography` and `\bibliographystyle` commands. When BIBTEX is run, it reads this information from the auxiliary file and produces a file named `myfile.bbl` containing LATEX commands to produce the source list ... The next time LATEX is run on `myfile.tex`, the `\bibliography` command reads the `bbl` file (`myfile.bbl`), which generates the source list.

Thus, the command sequence for a source file called `main.tex` which is going to use BibTeX would be:

```
latex main.tex
bibtex main
latex main
latex main
```

The first LaTeX is to collect all of the citations for BibTeX. Then BibTeX is run to generate the bibliography. LaTeX is run again to incorporate the bibliography into the document and the run the last time to update any references (like pages in the Table of Contents) which changed when the bibliography was included.

Chapter 5

Using the `withesis` Style

You can get a copy of the LaTeX style for creating a University of Wisconsin–Madison thesis or dissertation from:

`http://www.cae.wisc.edu/~benedict/LaTeX.html`

After somehow unpacking it, you will have the style files (`withesis.sty` `withe10.sty`, and `withe12.sty`) as well the files used to create this document. The files used for this document can be copied and used as a template for your own thesis or dissertation.

The final printed form of this document is useful, but the combination of the source code and final copy form a much more valuable reference. Keeping a working copy of the this document can be helpful when you are later working on your thesis or disseration and want to know how to do something. If you find a similar example in this document, then you can simply look at the corresponding source code and add it to your document. Because many parts of this document were written by different people, the styles and techniques are also different and provide different ways of achieving the same or similar results.

Because of the typical size of theses, it makes sense to break the document up into several smaller files. Usually this is done at the chapter level. These files can then be `\included` in a *root* file. It is the *root* file that you will run LaTeX on. For this manual, the root file is called `main.tex`.

5.1 The Root File and the Preamble

The `\documentclass` command is used to tell LaTeX that you will be using the `withesis` document class and it is the first command in your root file. Class options such as `10pt`, `12pt`, `msthesis` or `margincheck` are specified here:

```
\documentclass[12pt,msthesis]{withesis}
```

The class option `msthesis` sets the margins to be appropriate for depositing with the UW library, namely a 1.25 inch left margin with the remaining margins 1 inch. The defaults for the title page are also defined for a thesis and for a Master of Science degree.

The class option `margincheck` will place a small black square at the end of each line which exceeds the margins.[1] This is visible both in the `.dvi` file as well as in the `.ps` file.

The area immediately following this command is called the *preamble* and is used for things like including different style packages, defining new macros and declaring the page style.

The style packages can be used to easily change the thesis font. For example, this document is set in Times Roman instead of the LaTeXdefault of Computer Modern. This change was performed by including the `times` package:

```
\usepackage{times}
```[2]

Remember that if you change the fonts from the default Computer Modern to PostScript (*e.g.* Times Roman) then in order to correctly see the document, you will need to convert the `*.dvi` output into a `*.ps` file and view the document with a PostScript viewer. This is required since most `*.dvi` previewer programs cannot display PostScript fonts. Usually, the previewer will substitute default fonts so the document may be viewed; however, since the alternate fonts may not be the same size, the formatting of the document may appear to be incorrect.

The style package for including Postscript figures, `epsfig`, is included with

```
\usepackage{epsfig}
```

[1] In reality, the square is placed at the end of lines which exceed their `\hbox`. This usually (but not always) indicates a margin violation on the right margin. Left margin violations aren't indicated and if the margin violation is large enough, there isn't room for the black box to be visiable.

[2] In this document, the typewriter font `\tt` was redefined to use the Computer Modern font with the command `\renewcommand{\ttdefault}{cmtt}`. For more information, see [1].

If multiple style packages are required, then they can be combined into one statement as follows:

```
\usepackage{epsfig,times}
```

Many different style packages are available. For more information, see [1].

The page styles are defined using a similar method. A special style is defined for the `withesis` style:

```
\pagestyle{thesisdraft}
```

This style causes the footer text to become:

```
 DRAFT: Do Not Distribute        <time><Date>        <input file name>
```

This appears at the bottom of every page.

In addition to the page style command, the `withesis` has defined several useful commands which are specified in the preamble. They include `\draftmargin`, `\draftscreen`, `\noappendixtables`, and `\noappendixfigures`.

The command `\draftmargin` draws a PostScript box with the dimensions of the margins. This makes it easy to check that the margins are correct and to see if any of the text or figures are outside of the required margins. This box is only visible in the `.ps` file since it is a PostScript special.

The command `\draftscreen` draws a PostScript screen with the word *DRAFT* in light grey and diagonally across the page. This screen is only visible in the `.ps` file since it is a PostScript special.

The commands `\noappendixtables` and/or `\noappendixfigures` should be used if the appendix does not have either tables or figures respectively. These commands inhibit the Appendix Table or Appendix Figure titles in the List of Tables or List of Figures.

If you have specified the `psfig` or `epsfig` document style package, then a useful command is `\psdraft`. This command will show the bounding box that the figure would occupy (instead of actually including the figure). This speeds up the draft copy printing, reduces toner usage and the drawn box is visible in the `.dvi` file.

The next usual command is `\begin{document}`. The following example is part of the root file used for this manual.

```
\bibliographystyle{plain}
\include{prelude}        % Title page, abstract, table of contents, etc
\include{intro}          % Chapter 1
\include{essentials}     % Edited ‘‘Essential LaTeX’’ by Jon Warbrick
\include{figs}           % Chapter 3 Edited from UW Math Dept’s Sample Thesis
\include{bibs}           % Chapter 4 From PU Thesis styles, by J.D. McCauley
\include{usage}          % Chapter 5 Strongly based on similar by J.D. McCauley
\bibliography{refs}      % Make the bibliography
\begin{appendices}       % Start of the Appendix Chapters.  If there is only
                         % one Appendix Chapter, then use \begin{appendix}
\include{code}         % Including computer code listings
\include{bibref}         % a BibTeX reference
\include{math}           % Complex Equations from the UW Math Department
\include{acro}           % A discussion on generating PDF files.
\end{appendices}         % End of the Appendix Chapters.  ibid on \end{appendix}
%\include{vita}          % Optional Vita, use \begin{vita} vita text \end{vita}
\end{document}
```

5.2 Prelude

After the `\begin{document}` comes the preliminary information found in theses. In this manual, the information is kept in the file `prelude.tex` (see above). These pages will need to be numbered with roman numerals, so use

```
\clearpage\pagenumbering{roman}
```

Next, comes your thesis or dissertation title, your name, date of graduation, department and degree.

```
\title{How to \LaTeX\ a Thesis}
\author{Eric R. Benedict}
\date{2000}
%   - The default degree is ‘‘Doctor of Philosophy’’
%     Degree can be changed using the command \degree{}
%\degree{New Degree}
%   - for a PhD dissertation (default), specify \dissertation
%\dissertation
%   - for a masters project report, specify \project
%\project
%   - for a preliminary report, specify \prelim
%\prelim
%   - for a masters thesis, specify \thesis
%\thesis
```

```
%   - The default department is ``Electrical Engineering''
%     The department can be changed using the command \department{}
%\department{New Department}
```

If you specified the class option `msthesis`, then the degree is changed to *Master of Science* and the `\thesis` option is specified. If you want to have the masters margins with another document, then the `\degree` and `\dissertation`, `\project`, *etc.* can be specified as needed.

Once the above are all defined, use `\maketitle` to generate the title page.

```
\maketitle
```

If you wish to include a copyright page (see Section 1.2.2 for information on registering the copyright.), then add the command

```
\copyrightpage
```

This will generate the proper copyright page and will use the name and date specified in `\author{}` and `\date{}`.

Next are the dedications and acknowledgements:

```
\begin{dedication}
To my pet rock, Skippy.
\end{dedication}

\begin{acknowledgments}
I thank the many people who have done lots of nice things for me.
\end{acknowledgments}
```

You must tell LaTeX to generate a table of contents, a list of tables and a list of figures:

```
\tableofcontents
\listoftables
\listoffigures
```

If you wish to have a nomenclature, list of symbols or glossary it can go here.

```
\begin{nomenclature}
%\begin{listofsymbols}
%\begin{glossary}
```

```
\begin{tabular}{ll}
$C_1$ & Constant 1\\
\ldots
\end{tabular}
%\end{glossary}
%\end{listofsymbols}
\end{nomenclature}
```

If your abstract will be microfilmed by Bell and Howell (formerly UMI), then you will need to generate an abstract of less than 350 words. This abstract can be created using the `umiabstract` environment. This environment requires that you define your advisor and your advisor's title using `\advisorname{}` and `\advisortitle{}`.

```
\advisorname{Bucky J. Badger}
\advisortitle{Assistant Professor}
% ABSTRACT
\begin{umiabstract}
\noindent       % Don't indent first paragraph.
This explains the basics for using \LaTeX\ to typeset a
dissertation, thesis or project report for the University of
Wisconsin-Madison.

...

\end{umiabstract}
```

This will place your name, title and required text at the top of the page and follow the abstract text with your advisor's name at the bottom for your advisor's signature. This page is not numbered and would be submitted separately.

If you will have an abstract as part of your document, then the `abstract` environment should be used.

```
\begin{abstract}
\noindent       % Don't indent first paragraph.
This explains the basics for using \LaTeX\ to typeset a
dissertation, thesis or project report for the University of
Wisconsin-Madison.

...
```

```
\end{abstract}
```

This will generate a page number and it will be included in the Table of Contents.

If you will have both the UMI and regular abstracts like this document, then you will probably want to write the abstract once and save it in a seperate file such as `abstract.tex`. Then, you can use the same abstract for both purposes.

```
\begin{umiabstract}
  \input{abstract}
\end{umiabstract}


\begin{abstract}
  \input{abstract}
\end{abstract}
```

Finally, the page numbers must be changed to arabic numbers to conclude the preliminary portion of the document.

```
\clearpage\pagenumbering{arabic}
```

5.3 The Body

At the beginning of `intro.tex` there is the following command:

```
\chapter{Introducing the {\tt withesis} \LaTeX{} Style Guide}
```

Following that is the text of the chapter. The body of your thesis is separated by sectioning commands like `\chapter{}`. For more information on the sectioning commands, see Section 2.7.

Remember the basic rule of outlining you learned in grammar school:

> You cannot have an ‘A’ if you do not have a ‘B’

Take care to have at least two `\sections` if you use the command; have two `\subsections`, *etc*.

5.4 Additional Theorem Like Environments

The `withesis` style adds numerous additional theorem like environments. These environments were included to allow compatibility with the University of Wisconsin's Math Department's style file. These environments are `theorem`, `assertion`, `claim`, `conjecture`, `corollary`, `definition`, `example`, `figger`, `lemma`, `prop` and `remark`.

As an example, consider the following.

Lemma 5.1 Assuming that $\partial\Omega_2 = \emptyset$ and that $h(t) = 1$, we have

$$\Delta u = f, \quad x \in \Omega,$$
$$u = g_1, \quad x \in \partial\Omega.$$

which was produced with the following:

```
\begin{lemma}
Assuming that $\partial\Omega_2 = \emptyset$ and that $h(t) = 1$, we
have $$
\begin{array}{lr}
\Delta u = f, &  x\in\Omega ,\\[2pt]
 u =  g_1, &  x\in\partial\Omega .
\end{array}
$$
\end{lemma}
```

5.5 Bibliography or References

As a final note, the default title for the references chapter is "LIST OF REFERENCES." Since some people may prefer "BIBLIOGRAPHY", the command `\altbibtitle` has been added to change the chapter title.

5.6 Appendices

There are two commands which are available to suppress the writing of the auxiliary information (to the `.lot` and `.lof` files). They are:

```
\noappendixtables               % Don't have appendix tables
\noappendixfigures              % Don't have appendix figures
```

These commands should be in the preamble. See Section 5.1.

There are two environments for doing the appendix chapter: `appendix` and `appendices`. If you have only one chapter in the appendix, use the `appendix` environment. If you have more than one chapter, like this manual, use the `appendices` environment.

```
\begin{appendices}  % Start of the Appendix Chapters.  If there is only
                    % one Appendix Chapter, then use \begin{appendix}
\include{code}      % Including computer code listings
\include{bibref}    % a BibTeX reference
\include{math}      % Complex Equations from the UW Math Department

\end{appendices}    % End of the Appendix Chapters. ibid on \end{appendix}
```

The difference between these two environments is the way that the chapter header is created and how this is listed in the table of contents.

LIST OF REFERENCES

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeXCompanion*. Addison–Wesley Pub. Co., Reading, MA, 1994.

[2] Donald E. Knuth. *The TeXbook*. Addison–Wesley Pub. Co., Reading, MA, 1984.

[3] Leslie Lamport. *LaTeX: A Document Preparation System*. Addison–Wesley Pub. Co., Reading, MA, 1986.

[4] F. Rellich. Darstellung der Eigenwerte von $\Delta u + \lambda u = 0$ durch ein Randintegral. *Math Z.*, 46:635–636, 1940.

[5] E. Zeidler. *Nonlinear Functional Analysis*, volume IIa. Springer Verlag, New York, 1988.

Appendix A: Matlab Code

This is an example of a Matlab m-file.

```
function dPsi = derivs(t,Psi);
% derivs computes the state derivatives for an induction machine
% The machine is in the synchronous reference frame.
% Eric Benedict, Spring 1996

% Machine parameters...
global Rs Rqr Rdr Xls Xlr Xm Xms J P We Wb Vs

% rename state vector...
Phqs=Psi(1);
Phds=Psi(2);
Phqr=Psi(3);
Phdr=Psi(4);
Wr=Psi(5);
Theta=Psi(6);


% Select Reference Frame...
% Assume no angle differnce between a-axis and q-axis at t=0

% Synchronous
W=We;
ThetaR=W*t;

% Compute Machine Currents...
Phmq=Xms*Phqs/Xls + Xms*Phqr/Xlr;
Phmd=Xms*Phds/Xls + Xms*Phdr/Xlr;

Iqs=(Phqs - Phmq)/Xls;
Ids=(Phds - Phmd)/Xls;
Iqr=(Phqr - Phmq)/Xlr;
                                    Idr=(Phdr - Phmd)/Xlr;


% Compute Torques...
Te=3*P*(Phds*Iqs-Phqs*Ids)/(4*Wb);    %  Electrical Torque
Tl=0;                                 %  Load Torque
```

```
% Compute Voltages...
Vqs=Vs*cos(We*t - ThetaR);
Vds=-1*Vs*sin(We*t - ThetaR);
Vqr=0;
Vdr=0;

% Compute new derivative of state vector...
dPsi(1) = Vqs - Rs*(Phqs - Phmq)/Xls - W*Phds/Wb;          % PHqs/Wb
dPsi(2) = Vds - Rs*(Phds - Phmd)/Xls + W*Phqs/Wb;          % PHds/Wb
dPsi(3) = Vqr - Rqr*(Phqr - Phmq)/Xlr - (W-Wr)*Phdr/Wb;    % PHqr/Wb
dPsi(4) = Vdr - Rdr*(Phdr - Phmd)/Xlr + (W-Wr)*Phqr/Wb;    % PHdr/Wb

dPsi(5) = (Te - Tl)*P/(2*J*Wb);           % Wr/Wb

dPsi(6) = Wr/Wb;                          % Theta/Wb

dPsi=Wb*dPsi;
```

Appendix B: BibTEX Entries

The following shows the fields required in all types of BibTEX entries. Fields with OPT prefixed are optional (the three letters OPT should not be used). If an optional field is not used, then the entire field can be deleted.

```
@Unpublished{,
  author =       "",
  title =        "",
  note =         "",
  OPTyear =      "",
  OPTmonth =     ""
}

@TechReport{,
  author =       "",
  title =        "",
  institution =  "",
  year =         "",
  OPTtype =      "",
  OPTnumber =    "",
  OPTaddress =   "",
  OPTmonth =     "",
  OPTnote =      ""
}

@Proceedings{,
  title =            "",
  year =             "",
  OPTeditor =        "",
  OPTpublisher =     "",
  OPTorganization =  "",
  OPTaddress =       "",
  OPTmonth =         "",
  OPTnote =          ""
}
```

```
@Manual{,
  title =            "",
  OPTauthor =        "",
  OPTorganization =  "",
  OPTaddress =       "",
  OPTedition =       "",
  OPTyear =          "",
  OPTmonth =         "",
  OPTnote =          ""
}

@InProceedings{,
  author =           "",
  title =            "",
  booktitle =        "",
  year =             "",
  OPTeditor =        "",
  OPTpages =         "",
  OPTorganization =  "",
  OPTpublisher =     "",
  OPTaddress =       "",
  OPTmonth =         "",
  OPTnote =          ""
}
```

```
@PhDThesis{,
  author =      "",
  title =       "",
  school =      "",
  year =        "",
  OPTaddress =  "",
  OPTmonth =    "",
  OPTnote =     ""
}
```

```
@Misc{,
  OPTauthor =       "",
  OPTtitle =        "",
  OPThowpublished = "",
  OPTyear =         "",
  OPTmonth =        "",
  OPTnote =         ""
}
}
```

```
@MastersThesis{,
  author =      "",
  title =       "",
  school =      "",
  year =        "",
  OPTaddress =  "",
  OPTmonth =    "",
  OPTnote =     ""
}
```

```
@InCollection{,
  author =          "",
  title =           "",
  booktitle =       "",
  publisher =       "",
  year =            "",
  OPTeditor =       "",
  OPTchapter =      "",
  OPTpages =        "",
  OPTaddress =      "",
  OPTmonth =        "",
  OPTnote =         ""
}
```

```
@InCollection{,
  author =          "",
  title =           "",
  chapter =         "",
  publisher =       "",
  year =            "",
  OPTeditor =       "",
  OPTpages =        "",
  OPTvolume =       "",
  OPTseries =       "",
  OPTaddress =      "",
  OPTedition =      "",
  OPTmonth =        "",
  OPTnote =         ""
}
```

```
@Article{,
  author =          "",
  title =           "",
  journal =         "",
  year =            "",
  OPTvolume =       "",
  OPTnumber =       "",
  OPTpages =        "",
  OPTmonth =        "",
  OPTnote =         ""
 }
```

Appendix C: Mathematics Examples

This appendix provides an example of LaTeX's typesetting capabilities. Most of text was obtained from the University of Wisconsin-Madison Math Department's example thesis file.

C.1 Matrices

The equations for the *dq*-model of an induction machine in the synchronous reference frame are

$$\begin{bmatrix} v_{qs}^e \\ v_{ds}^e \\ v_{qr}^e \\ v_{dr}^e \end{bmatrix} = \begin{bmatrix} r_s + x_s\frac{\rho}{\omega_b} & \frac{\omega_e}{\omega_b}x_s & x_m\frac{\rho}{\omega_b} & \frac{\omega_e}{\omega_b}x_m \\ -\frac{\omega_e}{\omega_b}x_s & r_s + x_s\frac{\rho}{\omega_b} & -\frac{\omega_e}{\omega_b}x_m & x_m\frac{\rho}{\omega_b} \\ x_m\frac{\rho}{\omega_b} & \frac{\omega_e-\omega_r}{\omega_b}x_m & r_r' + x_r'\frac{\rho}{\omega_b} & \frac{\omega_e-\omega_r}{\omega_b}x_r' \\ -\frac{\omega_e-\omega_r}{\omega_b}x_m & x_m\frac{\rho}{\omega_b} & -\frac{\omega_e-\omega_r}{\omega_b}x_r' & r_r' + x_r'\frac{\rho}{\omega_b} \end{bmatrix} \begin{bmatrix} i_{qs}^e \\ i_{ds}^e \\ i_{qr}^e \\ i_{dr}^e \end{bmatrix} \quad [C.1]$$

$$T_e = \frac{3}{2}\frac{P}{2}\frac{x_m}{\omega_b}\left(i_{qs}^e i_{dr}^e - i_{ds}^e i_{qr}^e\right) \quad [C.2]$$

$$T_e - T_l = \frac{2J\omega_b}{P}\frac{d}{dt}\left(\frac{\omega_r}{\omega_b}\right). \quad [C.3]$$

C.2 Multi-line Equations

LaTeX has a built-in equation array feature, however the equation numbers must be on the same line as an equation. For example:

$$\begin{aligned} \Delta u + \lambda e^u &= 0 \quad u \in \Omega, \\ u &= 0 \quad u \in \partial\Omega. \end{aligned} \quad [C.4]$$

Alternatively, the number can be centered in the equation using the following method.

$$\begin{aligned} \Delta u + \lambda e^u = 0, \quad u \in \Omega, \\ u = 0, \quad u \in \partial\Omega. \end{aligned} \quad [C.5]$$

The previous equation had a label. It may be referenced as equation (C.5).

C.3 More Complicated Equations

Rellich's identity

Standard developments of Pohozaev's identity used an identity by Rellich [4], reproduced here.

Lemma C.1 (Rellich) Given L in divergence form and a, d defined above, $u \in C^2(\Omega)$, we have

$$\int_\Omega (-Lu)\nabla u \cdot (x-\overline{x})\,dx = (1-\frac{n}{2})\int_\Omega a(\nabla u,\nabla u)\,dx - \frac{1}{2}\int_\Omega d(\nabla u,\nabla u)\,dx \qquad \text{[C.6]}$$

$$+\frac{1}{2}\int_{\partial\Omega} a(\nabla u,\nabla u)(x-\overline{x})\cdot\nu\,dS - \int_{\partial\Omega} a(\nabla u,\nu)\nabla u\cdot(x-\overline{x})\,dS.$$

Proof:

There is no loss in generality to take $\overline{x}=0$. First rewrite L:

$$Lu = \frac{1}{2}\left[\sum_i\sum_j \frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial u}{\partial x_j}\right) + \sum_i\sum_j \frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial u}{\partial x_j}\right)\right]$$

Switching the order of summation on the second term and relabeling subscripts, $j\to i$ and $i\to j$, then using the fact that $a_{ij}(x)$ is a symmetric matrix, gives the symmetric form needed to derive Rellich's identity.

$$Lu = \frac{1}{2}\sum_{i,j}\left[\frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial u}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left(a_{ij}\frac{\partial u}{\partial x_i}\right)\right]. \qquad \text{[C.7]}$$

Multiplying $-Lu$ by $\frac{\partial u}{\partial x_k}x_k$ and integrating over Ω, yields

$$\int_\Omega (-Lu)\frac{\partial u}{\partial x_k}x_k\,dx = -\frac{1}{2}\int_\Omega \sum_{i,j}\left[\frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial u}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left(a_{ij}\frac{\partial u}{\partial x_i}\right)\right]\frac{\partial u}{\partial x_k}x_k\,dx$$

Integrating by parts (for integral theorems see [5, p. 20]) gives

$$= \frac{1}{2}\int_\Omega \sum_{i,j} a_{ij}\left[\frac{\partial u}{\partial x_j}\frac{\partial^2 u}{\partial x_k\partial x_i} + \frac{\partial u}{\partial x_i}\frac{\partial^2 u}{\partial x_k\partial x_j}\right]x_k\,dx$$

$$+\frac{1}{2}\int_\Omega \sum_{i,j} a_{ij}\left[\frac{\partial u}{\partial x_j}\delta_{ik} + \frac{\partial u}{\partial x_i}\delta_{jk}\right]\frac{\partial u}{\partial x_k}\,dx$$

$$-\frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left[\frac{\partial u}{\partial x_j}\nu_i + \frac{\partial u}{\partial x_i}\nu_j\right]\frac{\partial u}{\partial x_k}x_k\,dx$$

$= I_1 + I_2 + I_3$, where the unit normal vector is ν. One may rewrite I_1 as

$$I_1 = \frac{1}{2}\int_\Omega \sum_{i,j} a_{ij} \frac{\partial}{\partial x_k}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) x_k \, dx$$

Integrating the first term by parts again yields

$$I_1 = -\frac{1}{2}\int_\Omega \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) dx + \frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) x_k \nu_k \, dS$$

$$-\frac{1}{2}\int_\Omega \sum_{i,j}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) x_k \frac{\partial a_{ij}}{\partial x_k}\, dx.$$

Summing over k gives

$$\int_\Omega (-Lu)(\nabla u \cdot x)\, dx = -\frac{n}{2}\int_\Omega \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) dx$$

$$+\frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right)(x\cdot\nu)\, dS - \frac{1}{2}\int_\Omega \sum_{i,j}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right)(x\cdot\nabla a_{ij})\, dx$$

$$+\frac{1}{2}\int_\Omega \sum_{i,j,k} a_{ij}\left[\frac{\partial u}{\partial x_j}\frac{\partial u}{\partial x_k}\delta_{ik} + \frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_k}\delta_{jk}\right] dx$$

$$-\frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left[\frac{\partial u}{\partial x_j}\nu_i + \frac{\partial u}{\partial x_i}\nu_j\right](\nabla u \cdot x)\, dS.$$

Combining the first and fourth term on the right-hand side simplifies the expression

$$\int_\Omega (-Lu)(\nabla u \cdot x)\, dx = (1-\frac{n}{2})\int_\Omega \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right) dx$$

$$+\frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right)(x\cdot\nu)\, dS - \frac{1}{2}\int_\Omega \sum_{i,j}\left(\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\right)(x\cdot\nabla a_{ij})\, dx$$

$$-\frac{1}{2}\int_{\partial\Omega} \sum_{i,j} a_{ij}\left[\frac{\partial u}{\partial x_j}\nu_i + \frac{\partial u}{\partial x_i}\nu_j\right](\nabla u \cdot x)\, dS.$$

Using the notation defined above, the result follows.

Appendix D: Adobe Acrobat (*.pdf) Files

The Adobe Acrobat file format has pretty much become the *de facto* standard for document sharing. As such, some faculty members and/or departments may be requiring a final copy of the thesis in Acrobat format (*.pdf).

There are several different methods of obtaining a *.pdf file from a LaTeX thesis; however, they are all very site specific. A couple of different methods which have been found to work are mentioned as suggested ideas to try as a starting point. Depending on what is installed at your site/location some of these may be applicable.

D.1 Converting from *.ps to *.pdf

One option to obtain the *.pdf file would be to generate the thesis in a normal manner and then use the Acrobat Distiller to convert the postscript file into a *.pdf file.

If the Distiller program is available and convenient to use, then this is quite easy to do.

Depending on the choice of document fonts, the results may not be satisfactory since some of the fonts may end up as bit-mapped fonts and will display poorly at any resolution other than what they were sampled on. Also, since the Distiller program is an expensive program to obtain, it is not always available.

An alternative to the Adobe Distiller program is the Alladin Ghostscript program. This is available for free from

http://www.cs.wisc.edu/~ghost/index.html

This program is available for most common operating systems as a compiled binary, but the source code is available for other systems. One drawback is that this conversion must be performed as a command line invocation and isn't very user friendly. This may be addressed in a future version of Ghostview, the program which provides a nice user interface to Ghostscript.

D.2 Converting from *.dvi to *.pdf

There are two programs available which will convert from *.dvi to *.pdf, dvipdf and dvipdfm. The dvipdfm program will be discussed here. In version 0.12, it can generate bookmarks, thumbnails (with assistance from Ghostscript), scaling and rotation, JPEG and PNG bitmaps and font encoding and re-encoding (to support fonts which aren't fully supported by the Acrobat suite). When Ghostscript is properly installed, dvipdfm will automatically convert any encapsulated PostScript figures into the required *.pdf format. This program behaved in a similar manner to the dvips program and was used to produce the *.pdf format of this document.

D.3 Generating *.pdf Initially

There are now some programs which are similar to TEX but instead of producing a .dvi output, they produce *.pdfas a native output. One such program, PDFTEX / PDFLATEX, is available from

http://www.tug.org/applications/pdftex

Note that as of this date, July 2000, PDFTEX / PDFLATEX while currently quite usable, it is still in a beta version. Look at the web site for more current information.

The present version was able to produce a *.pdf file of this document without any required changes, except for the Postscript figure inclusion (Figure 3.2). To properly include this figure, requires the conversion of the postscript figure into a *.pdf figure. The procedure is described in the manual for PDFTEX / PDFLATEX. Note that the figure conversion will require either Distiller or Ghostscript.