Progress of Theoretical Physics, Vol. 95, No. 4, April 1996

Instruction for Making LaTeX Compuscripts Using $\mathcal{PTP}$TeX

Shin-Ichiro TOMONAGA*) and Hideki YUKAWA*,**)

Physics Department, Tokyo Bunrika University, Tokyo 113
** Yukawa Institute for Theoretical Physics, Kyoto University, Kyoto 606-01*

(Received April 2, 1996)

This is a manual for making LaTeX compuscripts for *Progress of Theoretical Physics* using the style file $\mathcal{PTP}$TeX.sty. In particular we explain some useful macros and environments which are specially prepared in $\mathcal{PTP}$TeX.sty. The source file of this manual itself is designed to give a template which can be used for writing your own compuscripts.

§1. Introduction

We have in part started the printing of *Progress of Theoretical Physics* (PTP) directly using LaTeX [1)] manuscripts from April, 1996. When you prepare LaTeXmanuscript (compuscript) to submit to PTP, please use our style file $\mathcal{PTP}$TeX.sty. In $\mathcal{PTP}$TeX, you can use all the usual LaTeX commands of course, and also some useful macros and environments in addition which are specially prepared. This manual explains about those commands special to $\mathcal{PTP}$TeX from Sect.2 on.

1.1. *Submission*

Use mainly **E-mail** to submit the LaTeX compuscript written by using $\mathcal{PTP}$TeX style file. The address is the Editorial Office of PTP:

`ptp@yukawa.kyoto-u.ac.jp`

If the compuscript consists of two or more files, *e.g.*, by containing ps/eps figure files other than LaTeX file itself, please first combine and pack them into a single file in either way of the following two and send it to the above address:

1) Use (P. Ginsparg's) `uufiles` command on UNIX, or,
   do 'manually' `tar+gzip(compress)+uuencode`;
2) Use `lha+ish`.

Don't forget to write your name, institution and address in the mail. Generally when you send E-mail message to PTP Editorial Office, please indicate, in the Subject line of the mail header, Subject: <type of message> <manuscript Ref.Number> <first author's last name>; for instance,

`submit PTP Kobayashi` : Submission of a new manuscript
`final 5Z03L Sato` : Final manuscript after accepted for publication
`status 5X31 Maskawa` : Status query
`report 6425 Yamada` : Referee report

In case E-mail is not available, we also accept floppy disk. In this case, however,

*) A friend of Schwinger because of bearing 'swing' commonly in their names.
**) A common friend of Fermi and Bose. E-mail address: yukawa@yukawa.kyoto-u.ac.jp

do the initial submission by sending two hard copies of the manuscript and a letter by conventional mail as before. Indicate "LaTeX submission" clearly in your letter. At the stage when the paper is accepted for publication, we ask you to send your final LaTeX compuscript by floppy disk to the PTP office. Use then **3.5 inch** floppy disk with DOS/V or NEC formatted.

In either case, the communication between Editorial Office and the author such as the information of referee reports or the author's reply to it, is performed via *conventional mail* using the *hard copy* of the manuscript as before. If the author makes a revised manuscript in this process, send it by *conventional mail* together with the old manuscript.

When the paper is accepted for publication, the PTP Editorial Office makes corrections/additions to the **hard copy** of the manuscript and send it back to the author. The author is asked to examine the suggested corrections/additions there and to make necessary corrections to the LaTeX source file. Then send the completed LaTeX file as the final manuscript to the PTP office either by E-mail or by conventional mail (floppy disk).

1.2. *How to get $\mathcal{PTP}$TeX style file*

$\mathcal{PTP}$TeX style file can be obtained by accessing to WWW Home Page of Progress of Theoretical Physics:

`PTP Home Page: http://www2.yukawa.kyoto-u.ac.jp/~ptpwww/`

Please try to get the most recent version since it is revised from time to time.

The file `ptptex08.uu` placed there is a 'uuencoded-gzipped-tar' file, which is automatically unpacked and yield the following eight files if you simply execute `% csh ptptex.uu` on your UNIX machine. (In the above directories, you can also find the other forms of packed files, 'lha+ish'-ed file `ptptex08.ish` and 'gzipped-tar' file `ptptex08.tar.gz`, as well as the un-packed files themselves. Take suitable ones depending on your system environment.)

1. manptp.tex Source file of this manual.
2. template.tex Template for making your $\mathcal{PTP}$TeX compuscript.
3. ptptex.sty Main style file of $\mathcal{PTP}$TeX.
4. ptp-text.sty Style file for making PTP text style output.
5. ptp-prep.sty Style file for making preprint style output.
6. wrapft.sty Macro for wrapfigure and wraptable environments.
7. mbf.sty Macro for using math-boldface (`\mbf`).
8. epsf.sty Macro for putting ps/eps figure files in LaTeX manuscript.

The macros 6 – 8 are optional. The last `epsf.sty` is a style file attached to the standard LaTeX system. It is, however, put here for reference since there seems to exist another different style file bearing the same name.

§2. Style and preamble

When you write a compuscript of your paper using $\mathcal{PTP}$TeX.sty, make use of **template.tex** file which is prepared for the users' convenience as a separate file. By

filling in it, you can easily make your compuscript a la PTPTeX.sty format.

The default style of PTPTeX.sty is designed for Regular Articles in *Progress of Theoretical Physics*. You can simply start your LaTeX source file with the line

```
\documentstyle{ptptex}
```

or,

```
\documentstyle[seceq]{ptptex}
```

Generally the argument of LaTeX command put in a square bracket `[ ]` is called an optional argument and can be omitted. The optional argument `[seceq]` in the second is a declaration to put the equation numbers with section numbers like (2·15). If you omit the option, then the sequential numbers are put to the equations throughout the paper, like (1), (2), $\cdots$.

In the usual LaTeX's `article.sty`, this first line is like

```
\documentstyle[12pt]{article}
```

If you rewrite it as above, then the output automatically becomes of the Article form of *Progress of Theoretical Physics*. Since PTPTeX.sty is designed to be as compatible with the usual LaTeX `article.sty` as possible, the source file which is written with LaTeX `article.sty` can also be compiled with PTPTeX.sty as well.

If you want an output in a preprint form for your private distribution, you should start with the line

```
\documentstyle[seceq,preprint]{ptptex}
```

If you want to write Letter article, Supplement article, Addenda or Errata, instead of the regular PTP article, please start with the line, respectively,

```
\documentstyle[letter]{ptptex}
\documentstyle[seceq,supplement]{ptptex}
\documentstyle[seceq,addenda]{ptptex}
\documentstyle[seceq,errata]{ptptex}
```

The part of the source file from the first `\documentstyle` line to the declaration line of

```
\begin{document}
```

is called *preamble*. One usually put in the preamble the definitions of personal macros and some style specifications. But, since the style specifications are quite unnecessary (and even harmful) in PTPTeX, please do not write them. In PTPTeX, you should fill in the following in the preamble:

```
\markboth{ }{ }    : Running head
\title{ }          : Title of the paper
\author{ }         : Author's name
\inst{ }           : Institution (address)
\abst{ }           : Abstract
\recdate{ }        : Received date
```

You can see how to write these if you use `template.tex` where an example is written. (Beware of the fact that the commands from `\author` to `\recdate` are macros particular to PTPTeX.sty. So, if you compile the file with those commands by LaTeX `article.sty`, you will get an error message '`! Undefined control sequence`'.)

In the preamble of `template.tex`, the following commands other than these are written (with comment out symbol `%` in front):

`\notypesetlogo`	:	Not to print out 'typeset using $\mathcal{PTP}$TEX.sty'
`\subfontMBF`	:	To use subfont for `\mbf` (explained later)
`\publishedin{ }`	:	Vol/Year/Page of the paper for which Addenda or Errata is written.
`\preprintnumber`[*width*]`{ }`	:	To write preprint numbers when [preprint]. [*width*] stands for the width of the preprint number column.

Use these when necessary. The usage will also be clear in the `template.tex`.

§3. Equations

Here we explain a few macros and environments which can be used in $\mathcal{PTP}$TEX in addition to the usual ones in LATEX.

3.1. *Mathematical italic bold*

A macro `\mib` is available to obtain math-bold (mathematical italic bold) fonts which are used, for instance, to denote 3-vectors. This macro is a very simple one which is defined as

```
\def\mib#1{\mbox{\boldmath $#1$}}
```

If you type, for instances, `\mib{\alpha}` and `\mib{kx}` in math-mode, you can obtain math-bold outputs $\boldsymbol{\alpha}$, $\boldsymbol{kx}$.

This macro has, however, a trouble that it does not make the letter in an appropriate size when used in a suffix, and moreover you cannot make it small even writing `\small` or `\scriptsize`.

We prepared a new font command `\mbf` free from this trouble which can be used just like other font commands `\it`, `\bf` and so on. If you write `\mbf`, then you enter in the math-bold mode and all the letters following are printed in math-bold fonts. (Therefore you have to specify the region by sandwiching by `{ }` like `{\mbf \alpha kx}`.) The suffices in the math-bold mode automatically becomes small. (Like in the usual boldface mode by `\bf`, the symbols `+ - < / > ( ) [ ]` do not become math-bold face. If you want to make them also math-bold, add `\mbfplus` and write like `\mbf\mbfplus`.) This command `\mbf` is realized by the `mbf.sty` file. So you have to include it in the optional argument of the `\documentstyle` in the first line:

```
\documentstyle[seceq,mbf]{ptptex}
```

By this you can write equations like

$$\phi(\boldsymbol{x}) = \sum_{\boldsymbol{k}} \left[\boldsymbol{a_k} e^{i\boldsymbol{kx}} + \boldsymbol{a_k}^{\dagger} e^{-i\boldsymbol{kx}} \right] \tag{3·1}$$

This command also has a trouble in fact. If you are working on a machine which does not possess many fonts, you will sometimes be warned as `No Font: Any of the above.` and cannot do preview nor print. In such a case, write

```
\subfontMBF
```

in the preamble. (This command is already written with comment-out symbol `%` in the template.tex.) This makes the preview and print possible by substituting the fonts by available ones temporarily. When you submit your LaTeX file to PTP or do preview and print on machines possessing enough fonts, therefore, please comment-out that line with `%`.

3.2. *Fractions*

By the usual `\frac{numerator}{denominator}` command, the numerator and the denominator are printed in a smaller font than usual. If you want the output in the usual size, you can use `\dfrac` command. The usage is quite the same as `\frac`, and you can write complex fractions such as

$$a_0 + \cfrac{b_1}{a_1 + \cfrac{b_2}{a_2 + \cfrac{b_3}{a_3}}} \,.$$

3.3. *Subeqnarray and subequations*

When you want to number the equations in the form like (3·2a), (3·2b), ⋯ in a set of arrayed equations, you can use `subeqnarray` environment. The usage is quite the same as the usual `eqnarray` environment: you can simply write `\begin{subeqnarray}` ⋯ `\end{subeqnarray}` in place of `\begin{eqnarray}` ⋯ `\end{eqnarray}`.

If you want to do the same thing for some equations apart from one another, you can use the `subequations` environments. The usage will be clear from the following simple example:

Input

```
\begin{subequations}
  \label{eq:1}
  An example of subequations:
  \begin{equation}
    \alpha + 2\beta + \gamma = 2
    \label{eq:1a}
  \end{equation}
  Here come some sentences,
  which can be of any length.
  \begin{eqnarray}
    \gamma &=& \nu (2-\eta) \\
    \delta &=& \mu (1+\rho)
  \end{eqnarray}
\end{subequations}
```

$\Longrightarrow$ gives

Output

An example of subequations:

$$\alpha + 2\beta + \gamma = 2 \qquad (3\cdot 2a)$$

Here come some sentences, which can be of any length.

$$\gamma = \nu(2 - \eta) \qquad (3\cdot 2b)$$
$$\delta = \mu(1 + \rho) \qquad (3\cdot 2c)$$

Here note that the `\label{eq:1}` just after the `\begin{subequations}` defines the group equation name so that `(\ref{eq:1})` gives "(3·2)", while `\label{eq:1a}` refers to the first equation (3·2a). On the other hand, in the previous `subeqnarray`

environment, the `\label` command refers only to the group equation name wherever it is placed inside the `subeqnarray` environment. In order to refer to the individual equation there, a command `\slabel` is available and is used in quite the same way as the usual `\label` command.

§4. References

References are cited using `\cite` command. All references should be listed continually within curly brackets using commas such as `\cite{rf:1,rf:3,rf:4,rf:5}`. (Don't put any space after the commas here. Otherwise LaTeX will regard the reference label as containing the space also.) When three or more consecutive reference numbers appear as in this example, the numbers are automatically printed in a compressed way like [1),3)-5)]. For cases of very long consecutive references, it would be very tedious to type this way. In such cases you can type as `\cite{rf:3}\tocite{rf:5}`; namely, you write the first reference using `\cite` and succeedingly cite the last reference using `\tocite` command. Then you obtain an output like this. [3)-5)] `\citen` command can be used to obtain citation numbers: *e.g.,* `Ref.~\citen{rf:3}` gives an output "Ref. 3)".

When you use `\cite` command at the end of sentences, put always the `\cite` command *after period, comma, colon or semicolon*, like `something.\cite{rf:5}`. If you continue the next sentence, type ␣\␣ (␣ is [space] key) to give a suitable space between sentences. For instance, type

```
This is $\cdots$ something.\cite{rf:5} \ Therefore we can $\cdots$
```

then the output becomes

This is $\cdots$ something. [5)] Therefore we can $\cdots$

The references are listed at the end of the file by using the `\thebibliography` environment and `\bibitem` command as usual. For example the references at the end of this manual are typed as follows:

```
\begin{thebibliography}{99}
 \bibitem{rf:1}
    Leslie Lamport, {\it LaTeX{\rm :} A Document Preparation System}
    (Addison-Wesley, New York, 1986).
 \bibitem{rf:2}
    S.~Weinberg, Phys.\ Rev.\ Lett.\ {\bf 19} (1967), 1264.
 \bibitem{rf:3}
    M.~Kobayashi and T.~Maskawa, \JL{Prog.\ Theor.\ Phys.,49,1973,652}.
 \bibitem{rf:4}
    D.~Gross and F.~Wilczek, \PRL{30,1973,1343}.
   \\
    H.~D.~Politzer, \PRL{30,1973,1346}.
 \bibitem{rf:5}
    G.~'t~Hooft, \NP{B33,1971,173}; \andvol{B35,1971,167}.
\end{thebibliography}
```

Please write your bibitems properly according to these examples in the PTP format. In particular, the volume numbers should be in boldface, the (year) should be fol-

lowed by a comma '`,`' and the line should end with a period '`.`'. Even when you cite two or more references of different authors in an item as in `rf:4`, end with a *period* and continue with a line-feed by `\\`. When you cite two or more references with the same author as in `rf:5`, continue with a semicolon '`;`' without line-feed. To make it easy to write the bibitem following this PTP format, we prepared macros `\JL` , `\andvol` , `\PRL` , $\cdots$, as used in `rf:3`, `rf:4` and `rf:5`:

```
o for general use
   \JL : general journals          \andvol : Vol (Year), Page
o for individual journal
   \PR  : Phys. Rev.               \PRL : Phys. Rev. Lett.
   \NP  : Nucl. Phys.              \PL  : Phys. Lett.
   \JMP : J. Math. Phys.           \CMP : Commun. Math. Phys.
   \PTP : Prog. Theor. Phys.       \JPSJ: J. Phys. Soc. Jpn.
   \JP  : J. of Phys.              \NC  : Nouvo Cim.
   \IJMP: Int. J. Mod. Phys.       \ANN : Ann. of Phys.
```

The usage is as follows:

```
\PR{D45,1990,345}             ⇒ Phys.\ Rev.\ {\bf D45} (1990), 345
\JL{Phys.\ Lett.,A30,1981,56} ⇒ Phys.\ Lett.\ {\bf A30} (1981), 56
\andvol{B123,1995,1020}       ⇒ {\bf B123} (1995), 1020
```

Here also you should not put any space after commas separating the volume, year and page numbers in the argument.

§5. Figures

5.1. *Post-Script (ps/eps) figure files*

It is best fitted to compuscripts to prepare figures in the form of epsf (Encapsulated PostScript Files). To put the epsf figures into the text, you should add `epsf.sty` as an input file by writing in the first line like

```
\documentstyle[seceq,epsf]{ptptex}
```

then write for instance as follows at the place where you want to put the figure:

```
\begin{figure}
    \epsfxsize = WIDTH cm   %or \epsfysize = HEIGHT cm
    \centerline{\epsfbox{FILENAME.eps}}
\caption{Explanation of the figure.}
\label{fig:1}
\end{figure}
```

Here `\epsfxsize = WIDTH cm` specifies the horizontal size and the vertical size is automatically calculated by the proportion of height to width of the original epsf figure. (You can conversely specify the height by writing `\epsfysize = HEIGHT cm`, then the width is calculated.) `\epsfbox{FILENAME.eps}` in the next line is the command to place the figure with file name `FILENAME.eps`. There seems to exist a different style file bearing the same name `epsf.sty`. If you cannot get the desired

result with these commands, the `epsf.sty` file may be such a different one. Then try to input the attached (world-standard) `epsf.sty` file.

Instead, if you prepare conventional figures drawn by ink, then send the original ones to PTP editorial office by conventional mail separately. In that case, to allocate a space for the figure in the LaTeX source file, the width and height of the figure should be given by `\figurebox` command. Here is an example:

```
\begin{figure}
\figurebox{WIDTH}{HEIGHT}
\caption{Here is the caption}
\label{fig:1}
\end{figure}
```

If you want to delete the frame rule of this `figurebox`, use the `\nofigureboxrule` command in the preamble written in `template.tex` by eliminating `%` in front.

5.2. *Wrapfigure environment*

Narrow figures should be put in the half text-width (7cm). `wrapfigure` environment is available for this purpose. We give an example of the `wrapfigure` in Fig. 1. Beware that this `wrapfigure` environment is *not a floating* one but gives the figure exactly at the place specified (irrespectively of whether the space exists enough or not). For instance, the Fig. 1 on the right hand side is obtained by putting the following commands on the top of this paragraph as shown shortly below. Be careful, therefore, that this environment should not be used at the edge of section or subsection or page, since it does not work properly at such boundaries. Because of this "non-floating" nature, it is better to put those `wrapfigure`'s at the final stage where no more additions will be made to the text.

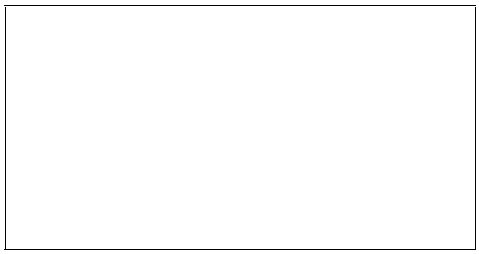

Fig. 1. A figure given by `wrapfigure` environment.

```
\begin{wrapfigure}{r}{6.6cm}   % r: RIGHT, 6.6cm: WIDTH
  \figurebox{60mm}{3cm}
  \caption{A figure given by {\tt wrapfigure} environment.}
  \label{fig:2}
\end{wrapfigure}
Narrow figures should be put in the half text-width (7cm)....
```

To use this environment, you have to add `wrapft.sty` as an input file by writing in the first line like

```
\documentstyle[seceq,epsf,wrapft]{ptptex}
```

The basic format of the `wrapfigure` environment is:

```
\begin{wrapfigure}[number]{position}{width}
   <figure> etc.
```

```
    \caption{ <caption> }
\end{wrapfigure}
```

Here the optional argument `[number]` is to specify the number of the narrow text lines wrapping around the figure. Since it is automatically calculate well, skip it usually. Write it explicitly only when you want to specify the number by force. (For instance, when the figure is placed at the bottom of a page, then the text lines at the beginning of the next page are sometimes made narrow. You can avoid such a thing by explicitly specifying the `[number]`.) The argument {`width`} specifies the width of the space for the figure (and of the figure caption). PTP admits only the size of half textwidth (7cm) from aesthetic viewpoint. So write **always {6.6cm}** or {`\halftext`} as this width by taking a space between the text and figure). `\halftext` is defined to be 0.471 times `\textwidth` and becomes 6.6cm in the case of PTP text style. The argument {`position`} is {`r`} or {`l`} when you put the figure on the right or left hand side. You can write {`c`} (center) there when you dislike the floating nature of the usual `figure` environment: if you write `\begin{wrapfigure}{c}{width}` with putting `{c}`, then it works just like the `figure` environment except that it gives the figure exactly at that place. The {`width`} specifies the caption width in this case. In the `<figure>` part can come epsf figures like in the above example, `picture` environment or anything else.

When you want to put a figure at the beginning of a paragraph like in the above example of Fig. 1, place the commands of `wrapfigure` environment before the first sentence of the paragraph. If you want to put it in the middle of a paragraph, it can be done rather manually; First comment out the `wrapfigure` environment part from the file and compile it. Then, in the preview screen, find and remember the word which comes at the end of the line below which you want to put the figure. Going back to the file, place the `wrapfigure` environment just after that word. A more detailed usage can be found in the beginning of the `wrapft.sty` file.

If you do not like any such complications, then put all the figures simply by using the usual `figure` environment alone. The editorial office will remake them in a most appropriate form.

5.3. *Putting figures side by side*

Either in `figure` or `wrapfigure` environment, it is easy to put two or more figures in a column (vertically). Repeat simply the `<figure>` part in the environment. Repeat also the `\caption` part if you want caption for each figure.

It is not so trivial to put two figures side by side. There would be many ways to do this. We here cite an example using `\parbox` command:

```
\begin{figure}[htb]
    \parbox{\halftext}{%   %\def\halftext{.471\textwidth}
        \figurebox{6cm}{2cm}
        \caption{The first figure on the left.}}
    \hspace{8mm}
    \parbox{\halftext}{
        \figurebox{6cm}{2cm}
```

```
        \caption{The second figure on the right.}}
    \end{figure}
```

This gives the following output:

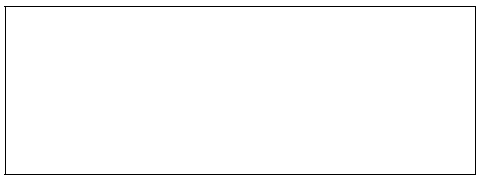

Fig. 2. The first figure on the left.

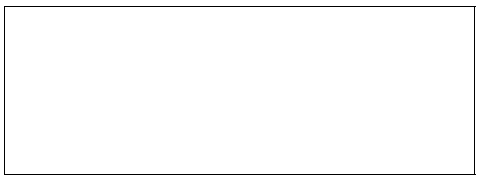

Fig. 3. The second figure on the right.

§6. Tables

Table I. An example of small table given by `\begin{wraptable}{l}{\halftext}`.

temperature	energy	specific heat
0.1	0.24	2.46
0.2	0.80	4.62
0.3	1.11	3.27

The standard LaTeX `table` environment can of course be used. We here give only an example of small table using `wraptable` environment in Table I. This environment is also supported by the `wrapft.sty` file explained above for figures and the usage is quite the same as `wrapfigure` environment. The format for Tables in PTP is, as in this example, to put **caption above the table** and to draw **double line only for the top line** of the table using `\hline \hline`. Please follow this way of writing tables. For reference we cite the source for the Table I:

```
\begin{wraptable}{l}{\halftext}
  \caption{An example of small table given by
           {\tt \BS begin\{wraptable\}\{l\}\{\BS halftext\}}.}
  \label{table:1}
  \begin{center}
    \begin{tabular}{ccc} \hline \hline
     temperature & energy  & specific heat \\ \hline
         0.1     &  0.24   &  2.46  \\
         0.2     &  0.80   &  4.62  \\
         0.3     &  1.11   &  3.27  \\ \hline
    \end{tabular}
  \end{center}
\end{wraptable}
The standard \LaTeX\ \verb+table+ environment can ......
```

The original `tabular` environment is rewritten in $\mathcal{PTP}$TEX.sty so as to print in the footnotesize fonts suitable for tables. If you want to use it in the usual text, therefore, please add a line of a sentence

```
\let\tabularsize\normalsize
```

to obtain normalsize output. But take care that the size change affects only *locally*.

§7. Comments

An error which sometimes occurs but is difficult to find the cause is the case that one cites equations by `\ref{..}` in the captions of Figure and Tables, for instance. This type of error can often be avoided if you put `\protect` in front of `\ref{..}`. The `\protect` command is also worth trying when you meet some unknown errors about the argument of `\section` and `\title`.

When you use PTPTeX, you may find possible bugs in `ptptex.sty`. If you find such bugs, please inform us of them to the address

`ptp@yukawa.kyoto-u.ac.jp` or `kugo@gauge.scphys.kyoto-u.ac.jp`

with writing `ptptex.sty bug` at the Subject column of your mail header.

A comment on LaTeX2ϵ

Our math-bold font command `\mbf` is now made to work also in LaTeX2ϵ. In LaTeX2ϵ case, however, the standard command `\boldsymbol` can also be used to obtain the math-bold fonts. The command `\boldsymbol` becomes available by adding `amsbsy.sty` as an input style file. Then, an input like `$\boldsymbol{a_k}$`, for instance, would give an output $\boldsymbol{a_k}$.

Acknowledgements

Acknowledgements can be written using the LaTeX standard command `\section*{Acknowledgements}`.

The style files and the guides of JJAP and JPSJ were of much help in making PTPTeX style file and writing this manual. We acknowledge the staffs of JJAP and JPSJ making those. At the final part of our style file `ptptex.sty`, we have included free-ware style files `subeqn.sty` (by Stephen Gildea), `subeqna.sty` (by Johannes Braams), `cite.sty` (by Donald Arseneau) as they stand. And `wrapft.sty` is merely a modified version of Tetsuo Iwakuma's extension `wrapfloat.sty` of the original `wrapfig.sty` by Donald Arseneau. We express sincere thanks to those authors.

Appendix A
—— *How to write Appendix* ——

The appendices can be written by the standard LaTeX commands as

```
\appendix
\section{How to write Appendix}
  ...
\section{Second Appendix}
  ...
```

If you write two or more appendices, the appendix numbers automatically become **A**, **B**, *etc.* Of course, the equations can be used and look like

$$S_q^z = \frac{1}{L}\sum_{j=1}^{L} S_j^z e^{iqj}. \qquad \text{(A·1)}$$

Appendix B
—— Converting PS file to EPS file ——

EPSF (Encapsulated PostScript Files) are best suited for the figures to be put in the TEX(LATEX) compuscripts. If you have only the usual PS (PostScript) files, you can easily convert them into EPSF as follows.

EPS file is merely a PS file which has an additional information on the region where the figure is drawn. This specification of the figure region is done by the command line "`%%BoundingBox: x1 y1 x2 y2`", which is usually written within ten lines from the top of the EPS file. The four numbers here represent two coordinates $(x1, y1)$ and $(x2, y2)$ standing respectively for the bottom left and top right corners of the figure region. (Even the ordinary PS file has usually this command line `%%BoundingBox:`. But, then, the coordinates $(x1, y1)$ and $(x2, y2)$ represent the region of the whole page. If such a line is missing, you have to add one to convert it into EPSF.)

To know the correct values for the coordinates $(x1, y1)$ and $(x2, y2)$, open the PS figure first using some viewer like ghostview (or GSview). If you move the cursor on the viewer screen, the coordinate of the cursor will be shown on the top (left) part of the viewer every moment. So you can read the desired coordinates $(x1, y1)$ and $(x2, y2)$ by putting your cursor on the bottom left and top right corners of the figure region.

Next, open the PS file by your editor software. You will see the top parts written like

```
%!PS-Adobe-2.0
%%Creator: gnuplot
%%DocumentFonts: Helvetica
%%BoundingBox: 50 50 410 302
%%EndComments
```

Now rewrite the four numbers (50 50 410 302 in this example) in the command `%%BoundingBox:` into the coordinates $(x1, y1)$ and $(x2, y2)$ which you have read above. If they are $(70, 80)$, $(100, 120)$, for instance, you will have to rewrite as

```
%!PS-Adobe-2.0 EPSF-2.0
%%Creator: gnuplot
%%DocumentFonts: Helvetica
%%BoundingBox: 70 80 100 120
%%EndComments
```

We have also changed the first line adding `EPSF-2.0`. Although this change seems irrelevant to the use in the TEX with the style file `epsf.sty`, it will cause a difference when viewing the file by ghostview in showing the whole page or the figure region only. It will also be better to change the file name from `xxx.ps` to `xxx.eps`. That

is all for the change of PS into EPS files.

Appendix

If you want to write an appendix without title like this appendix, simply write as `\section{}` leaving the argument empty or blank. Even then, the appendix section counter is working and the equation look like

$$A = B. \tag{C·1}$$

So, the use of appendix without title is not recommended exept for the case having a single appendix.

References

1) Leslie Lamport, *LaTeX: A Document Preparation System* (Addison-Wesley, New York, 1986).
2) S. Weinberg, Phys. Rev. Lett. **19** (1967), 1264.
3) M. Kobayashi and T. Maskawa, Prog. Theor. Phys. **49** (1973), 652.
4) D. Gross and F. Wilczek, Phys. Rev. Lett. **30** (1973), 1343.
H. D. Politzer, Phys. Rev. Lett. **30** (1973), 1346.
5) G. 't Hooft, Nucl. Phys. **B33** (1971), 173; **B35** (1971), 167.