Abstract

This short document describes physics2 package for those who are used to the physics package. This document is only a simple reference manual for:

- Frequent users of the legacy physics package;
- Those who have to maintain a document written with physics;
- Users who failed to use unicode-math with physics.

It seems no reason for any other user to read this document instead of the package documentation of physics2, because this document cannot describe the package in detail.

In this document, the modules of physics2 will be introduced in the same order as the physics documentation.

Contents

1 Before you start 2
    1.1 Legacy problems with physics package . . . . . . . . . . . . . . . . 2
    1.2 Loading physics2 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

2 List of commands 3
    2.1 Automatic bracing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

2.2 Vector notation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
2.3 Operators . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
2.4 Quick quad text . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
2.5 Derivatives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
2.6 Dirac bra-ket notation . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
2.7 Matrix macros . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

*https://www.github.com/AlphaZTX/physics2
1 Before you start

1.1 Legacy problems with physics package

The physics package provides \qty command for automatic-sizing braces. The \qty command would cause conflict with the siunitx package, which provides a unified method to typeset numbers and units correctly.

Besides, after you loaded physics, when you type \homework you will get Maxwell equations and Schrödinger equation. The \homework command is “declared” in physics.sty but it was not described in the documentation. That is, if you have defined \homework before loading physics package, physics would overwrite the definition “silently”.

The vector-notation part of physics uses amsmath’s (more exactly, amsbsy.sty’s) \boldsymbol command to generate bold vectors. Commands for cross/dot product are defined with \boldsymbol. \boldsymbol uses \mathversion, a \LaTeX kernel command that works well with traditional TFM-based fonts but fails when using unicode-math.

In the definition of \imat, \xmat, \dmat and \admat commands from physics, there is a \newtoks command which allocates a token list register and two \newcount commands allocating two count registers. Every time you write a command like \imat in your document, then one token list register and two count registers will be wasted. What’s even worse is that, if you wrote really too many matrix commands from physics (for example, 32767 \imats in Lua\LaTeX), there’d be no room for a new \count.

physics integrated all the functions in one file (physics.sty), that is, you cannot load one of the total seven parts of functions; you have to load the seven parts altogether, even included the extra \homework command we mentioned in the first paragraph.

Moreover, the code of physics.sty “abuses” the g-type arguments of xparse package. Therefore the syntax of physics package looks kind of weird. See here for more.

1.2 Loading physics2

The physics2 package includes different modules, among which every module focuses on one single function.

Write the following line in the preamble to load physics2:

\usepackage{physics2}
But this is not enough. \texttt{physics2} contains different modules. If you want to load any module of \texttt{physics2}, write this line after loading \texttt{physics2} package:

\begin{verbatim}
\usephysicsmodule{(module list)}
\end{verbatim}

For example, "\texttt{\usephysicsmodule{ab,doubleprod}}" loads the \texttt{ab} module and the \texttt{doubleprod} module.

You can also load a module with options:

\begin{verbatim}
\usephysicsmodule[(option list)]{(module)}
\end{verbatim}

For example, "\texttt{\usephysicsmodule[legacy]{ab}}" loads \texttt{ab} with the option “\texttt{legacy}”.

\textbf{Attention}, if you used any font package in your document, remember that \texttt{physics2} requires to be loaded \textit{after} font packages.

\section{List of commands}

\subsection{Automatic bracing}

As mentioned in \S1.1, the \texttt{\qty} command from \texttt{physics} would cause conflicts with \texttt{siunitx}. The command for automatic braces in \texttt{physics2} is \texttt{\ab}, a shorthand for \texttt{a\utomatic\braces}.

The \texttt{\ab} command requires the \texttt{ab} module, so don’t forget to write \texttt{\usephysicsmodule{ab}} in the preamble after you loaded \texttt{physics2}. Always remember, \textit{do not put an \ab separately in the end of a math formula}. Take some examples:

\begin{verbatim}
\[
\begin{array}{c}
\text{(}\quad \text{\ab}\text{(}\frac{1}{2}\text{)}\quad \text{\ab}\text{[}\frac{1}{2}\text{]}\quad \text{\ab}\{\frac{1}{2}\} \\
\text{\ab}\text{[}\text{\frac{1}{2}}\text{]}\quad \text{\ab}\{\text{\frac{1}{2}}\} \\
{1\choose 2}\quad {1\choose 2}\quad {1\choose 2}
\end{array}
\end{verbatim}

\texttt{\ab} can modify a delimiter-braced subformula. But the delimiters should not be out of the range described by the following chart:

\begin{verbatim}
\begin{array}{c|c}
\text{(}, \text{)} & \text{\lbracket}, \text{\rbracket} \\
\text{[, ]} & \text{\llangle}, \text{\rrangle} \\
\text{\{}{}, \text{\}} & \text{\lvert}, \text{\rvert} \\
\text{\{}{}, \text{\}} & \text{\lVert}, \text{\rVert}
\end{array}
\end{verbatim}

For example, \texttt{$\ab\{\text{foo}$} and \texttt{$\ab\{\text{foo}$} are \texttt{illegal}, but \texttt{$\ab\{\text{foo$} and \texttt{$\ab\{\text{foo$} are \texttt{okay}; \texttt{$\ab\{\}$} is \texttt{okay} but \texttt{$\ab\{\}$} is \texttt{illegal}.
Attention, if you want to delimit a subformula with "{" and "}", you can only write \{, \} or \lbrack, \rbrack around it. \{ and \} are not supported in ab module.

Between \ab and the first delimiter can be a “biggg” command, that is, from \big to \Bigg. Actually, you can also write \biggg and \Biggg because physics2 defines these after you load it. For example,

\[
\begin{array}{c}
\ab \Big \quad \frac{1}{2} \quad \ab \Bigg \quad \frac{1}{2} \\
\end{array}
\]

Between \ab and the first delimiter can also be a star (*), which means “use the default size of delimiters”. But in this situation, you needn’t use the \ab command at all.

The physics package provides commands like \pqty, \bqty. In the ab module of physics2, these commands have changed to \pab, \bab, etc. The following example shows all the \Xab commands in ab module:

\[
\begin{array}{c}
\pab{\frac{1}{2}} \quad \bab{\frac{1}{2}} \quad \Bab{\frac{1}{2}} \\
\aab{\frac{1}{2}} \quad \vab{\frac{1}{2}} \quad \Vab{\frac{1}{2}} \\
\end{array}
\]

\Xab can take an optional star and an optional [\(\biggg\)] argument. For example,

\[
\begin{array}{c}
\pab[\Big]{\frac{1}{2}} \quad \bab[\big]{\frac{1}{2}} \\
\end{array}
\]

physics also provides the following commands:

\abs \norm \eval \order \comm \acomm \pb

These commands are not originally supported by physics2, but the first four commands can be used through the ab.legacy module of physics2:

\usephysicsmodule{ab.legacy}

For example,

\[
\begin{array}{c}
\abs{\frac{1}{2}} \quad \abs[\big]{\frac{1}{2}} \quad \abs[\biggg]{\frac{1}{2}} \\
\end{array}
\]
Users of the legacy \texttt{physics} package should notice that the syntax of \texttt{\eval} has been changed. The \texttt{ab.legacy} module abandoned the \texttt{\eval\{foo-}\texttt{like syntax. The new} \texttt{\eval}'s syntax is just like other commands in this module. There are also two variants of \texttt{\eval} — \texttt{\peval} and \texttt{\beval}. For example,

\begin{verbatim}
[2.1.6] \def\0{1+\frac12x}
\[ \eval\{\0\}_a^b \quad \peval*{\0}_a^b \quad \beval[big]{\0}_a^b \]
\end{verbatim}

\begin{align*}
1 + \frac{1}{2} x |^b_a (1 + \frac{1}{2} x |^b_a)
\end{align*}

The \texttt{\comm}, \texttt{\acomm} and \texttt{\pb} (Poisson bracket) are not supported. But you can write like \texttt{\ab[foo,baz]} or \texttt{\bab\{foo,baz\}} instead.

By the way, you can set the “order” symbol in \texttt{ab.legacy} through the \texttt{order} option like this:

\begin{verbatim}
\usephysicsmodule[order=O]{ab.legacy}
\end{verbatim}

Then $\\order(N)$ yields $O(N)$.

\section{2.2 Vector notation}

Unfortunately, there is not a plan for \texttt{physics2} to support this part of \texttt{physics} completely, but the rest of this section will show some methods to maintain the document written with \texttt{physics}.

The \texttt{\vb(*)}, \texttt{\va(*)} and \texttt{\vu(*)} are not supported in any module of \texttt{physics2}. But these commands can be defined by copying the following lines below and pasting them in the preamble:

\begin{verbatim}
\makeatletter
\newcommand\vb{\@ifstar\boldsymbol\mathbf}
\newcommand\va[1]{\@ifstar{\vec{#1}}{\vec{\mathrm{#1}}}}
\newcommand\vu[1]{\@ifstar{\hat{\boldsymbol{#1}}}{\hat{\mathbf{#1}}}}
\makeatother
\end{verbatim}

The \texttt{\boldsymbol} command requires the \texttt{amsmath} or \texttt{bm} package. If you prefer to use \texttt{bm}, you can also use the \texttt{\bm} command. What’s more, if you tried the commands above, you might find that, the result of \texttt{\va} above is different from that of \texttt{physics}. This is because, if you choose to present a vector in bold, there’s almost no need to put a \texttt{\vec} (\texttt{\vec{v}}) sign above it.

However, the method above may not work well with \texttt{unicode-math} because there are so many OpenType math fonts without a bold version. When using
\textbf{unicode-math}, it’s recommended to use `\symbf` and `\symbfit` for a separate vector. For example, $\symbf{0}$ yields $\mathbf{0}$, and $\symbfit{A}$ yields $\mathbf{A}$.

The `\vdot` and `\cross` commands are not supported in any module of \texttt{physics2}. Actually, there is no need to use a bold “\cdot” or “\times” for the products of two vectors. Using `\cdot` and `\times` is enough.

The commands related to “\nabla” are supported through \texttt{nabla.legacy} module. These commands are `\grad`, `\div` and `\curl`. These commands should not be put in the end of a math formula either (just like `\ab`). Notice that the former `\div` command for a “÷” (if there exists one) is redefined as `\divsymbol`. For example,

\begin{verbatim}
% \usephysicsmodule{nabla.legacy}
\[\grad F \quad \div\left[\frac G2\right] \quad \curl\left\{\frac Y2\right\}\]
\[2 \divsymbol 1\]
\end{verbatim}

Actually, the nabla-related commands end with `\ab`. Thus, the subformula after these commands can be delimited with `( )`, `[ ]` and `{ }`.

The \texttt{nabla.legacy} requires the \texttt{fixdif} package at least version 2.0 (file date: 2023/01/31 or after 2023/01/31).

By the way, if you are used to writing `\bm` for a vector but interested in \texttt{unicode-math}, the \texttt{bm-um.legacy} module would be a passable alternative to \texttt{bm} package. Notice that the `\bm` command from the \texttt{bm-um.legacy} module can only take one letter (or one digit) as its argument.

\section*{2.3 Operators}

There’s no plan for \texttt{physics2} to support this part of \texttt{physics} completely. The syntax in this part of \texttt{physics} (such as `\tan[2](x)`) abuses \texttt{xparse}.

It’s suggested to write like this if you used the \texttt{ab} module:

\begin{verbatim}
$ \sin^2 \ab\left(\frac\alpha2\right) $
\end{verbatim}

Rather than take the superscript as an optional argument of the command of log-like functions.

The \texttt{physics} package provides a bundle of commands for log-like functions that have not been defined in the \texttt{BibTeX2e} kernel. Those log-like functions can be used with the \texttt{op.legacy} module; this module does not support the syntax of \texttt{physics} either. For example:
The \texttt{\Re} and \texttt{\Im} commands are redefined as operators “\texttt{Re}” and “\texttt{Im}”, while \texttt{\Re} and \texttt{\Im} are reserved as \texttt{\Resymbol} and \texttt{\Imsymbol}. \Re and \Im are ordinary symbols but \texttt{Re} and \texttt{Im} are operators.

### 2.4 Quick quad text

The \texttt{qtext.legacy} module provides the \texttt{\q{foo}} commands for \texttt{\quad}-wrapped texts. These commands have the same syntax as \texttt{physics}. For example,

\begin{verbatim}
% \usephysicsmodule{qtext.legacy}
[\[ A \qq{foo bar} B \]\[ A \qq*(foo bar) B \]
[\[ C \qcc D \qcc* E \]\[ F \qif G \qthen H \]
\end{verbatim}

All the commands described in §2.4 of \texttt{physics documentation} are supported when using \texttt{qtext.legacy} module, but I don’t recommend using this module unless you are maintaining a document written with \texttt{physics’s} \texttt{\q{foo}} commands.

### 2.5 Derivatives

There is no plan for \texttt{physics2} to support this part of \texttt{physics}. If you want to typeset the differential operators on a better sense, you can try the \texttt{fixdif} package; if you want an easy way to type derivatives, you can try the \texttt{derivative} package. These two packages can be used together. For example,

\begin{verbatim}
% \usepackage{fixdif,derivative}
[\[ \pdv{f}{x,y,z} \d x \]
Math ($\d x$) v.s. Text ($\d x$)
\end{verbatim}

Here are the documentations of \texttt{fixdif} and \texttt{derivative}. \texttt{fixdif’s} commands behave better in superscripts and subscripts.

### 2.6 Dirac bra-ket notation

There are two solutions to Dirac bra-ket in \texttt{physics2} — \texttt{ab.braket} and \texttt{braket}. These two modules are not compatible and neither of them supports \texttt{physics’s} syntax completely. Click \texttt{here} to see the \texttt{ab.braket} module and \texttt{here} to see the \texttt{braket} module.
The **ab.braket** module  This module provides four commands — \(\bra\), \(\ket\), \(\braket\) and \(\ketbra\). After these commands can be a star (\(\ast\)) or a "biggg" command. These commands share similar syntaxes like ab’s syntax. But, **the bra-ket commands from ab.braket module are completely different from ab**. Their internal structures are different.

The argument of \(\bra\) should be delimited with < and |, that is,

\[ \bra < (\text{subformula}) | \]

For example,

\[
\begin{align*}
\text{(2.6.1)} & \quad \left\{ \frac{\phi}{2} \right\} \\
& \quad \left\{ \frac{\phi}{2} \right\} \\
& \quad \left\{ \frac{\phi}{2} \right\}
\end{align*}
\]

The argument of \(\ket\) should be delimited with | and >, that is,

\[ \ket | (\text{subformula}) > \]

For example,

\[
\begin{align*}
\text{(2.6.2)} & \quad \left\{ \frac{\psi}{2} \right\} \\
& \quad \left\{ \frac{\psi}{2} \right\} \\
& \quad \left\{ \frac{\psi}{2} \right\}
\end{align*}
\]

If you want to write “>” and “<” for relations in the argument of \(\bra\) and \(\ket\), you can write \(\mathrel{>}\) and \(\mathrel{<}\) (although there is almost no such need).

The argument of \(\braket\) should be delimited with < and >, that is,

\[ \braket < (\text{subformula}) > \]

In the \(\langle \text{subformula} \rangle\) argument, every “|” will be regarded as an extensible vertical bar. For example,
The argument of $\ketbra$ should be delimited with $|$ and $|$. In the argument, $>$ and $<$ will be regarded as extensible $)$ and $(. That is,

$$\ketbra | (\text{subformula}_1) > (\text{optional}) < (\text{subformula}_2) |$$

For example,

$$\def\0{\frac\phi2}
\begin{align*}
[ \ketbra & | \0 > \psi | ] \\
[ \ketbra* & | \0 > \psi | ] \\
[ \ketbra\text{Bigg} & | \0 > \psi | ]
\end{align*}$$

If you want to write "\textgreater" and "\textless" for relations in the argument of $\braket$ and $\ketbra$, you can write $\backslash >$ and $\backslash <$ (although there is almost no such need). It is quite different from $\mathrel{>}$ or $\mathrel{<}$ because in these commands' argument, $>$ and $<$ will be redefined.

The **braket** module This module contains four commands — $\bra$, $\ket$, $\braket$ and $\ketbra$. After these commands can be a star (*) or a square-bracket-delimited size option, the size option can take the following values:
big, Big, bigg, Bigg, biggg or Biggg.

Star stands for “do not size the bra-ket automatically”.

The argument(s) of these four commands are braced with \{ and \}. \bra and \ket take one mandatory argument. For example,

\def\0{\frac{\phi}{2}}
\begin{align*}
\bra \{ \emptyset \} & \quad \bra * \{ \emptyset \} \\
\ket \{ \emptyset \} & \quad \ket * \{ \emptyset \} \\
\end{align*}

The \braket command, in default, can take two arguments.

\def\0{\frac{\phi}{2}}
\begin{align*}
\braket \{ \emptyset \} \psi & \quad \braket * \{ \emptyset \} \psi \\
\end{align*}

If you want \braket to take one or three arguments, you can write the number of arguments in the square bracket. If you need to specify the size of bra-ket simultaneously, you need to separate the number and the size with a comma.

\def\0{\frac{\phi}{2}}
\begin{align*}
\braket \{ 1 \} \emptyset \psi & \quad \braket * \{ 1 \} \emptyset \psi \\
\braket \{ 3 \} \emptyset A \psi & \quad \braket \{ 3, \text{Big} \} \emptyset A \psi \\
\end{align*}

The \ketbra command takes two mandatory arguments. It can also take an optional argument between the two mandatory arguments. The optional argument will be placed between \rangle and \langle.

\def\0{\frac{\phi}{2}}
\begin{align*}
\ketbra \{ \emptyset \} \psi & \quad \ketbra * \{ \emptyset \} \psi \\
\ketbra \{ \text{Big} \} \psi & \quad \ketbra \{ \emptyset \} _x^y \psi \\
\end{align*}
2.7 Matrix macros

Unfortunately, physics2 do not support the \mqty command from physics. If you are used to this command, you can write like this:

\newcommand\mqty[1]{\begin{matrix}#1\end{matrix}}
\newcommand\pmqty[1]{\begin{pmatrix}#1\end{pmatrix}}
$\ab(\mqty{foo})$ or $\pmqty{foo}$

These are equal to physics’s \mqty(foo) (require amsmath).

physics2’s diagmat module provides \diagmat command for diagonal matrices. For example,

\[
\begin{array}{c}
\text{\diagmat \{ 1, 2, 3 \}} \\
\end{array}
\]

\[
\begin{array}{c}
\text{\pdiagmat [ empty = {} ] \{ a, b, c, d \}} \\
\end{array}
\]

\pdiagmat, \bdigmat, \bdigmat, \vdiagmat and \Vdiagmat are also available.

physics2’s xmat module provides \xmat command for matrices with formatted entries. For example,

\[
\begin{array}{c}
\text{\xmat(a)(2)(3)} \\
\end{array}
\]

\[
\begin{array}{c}
\text{% usephysicsmodule}
% \ [showleft=3,showtop=3]{xmat}
\text{\pxmat(X)(m)(n)} \\
\end{array}
\]

\[
\begin{array}{c}
\text{\xmat [showleft=2,showtop=2,format=textrtt[#1[#2][#3]]] \{x\}(m)(n)} \\
\end{array}
\]

\pxmat, \bxmat, \Bxmat, \vxmat and \Vxmat are also available.