spbmark package

v1.46x   2025/02/08

Customize superscripts and subscripts

Qu Yi

https://github.com/texno3/spbmark

toquyi@163.com

spbmark provides three commands `\super`, `\sub` and `\supersub` to improve the layout of superscripts and subscripts which can be adjusted the relative position and format, and can be used in text and math mode.

Table of Contents

1. Macro package options

The following macro package options will redefine the script commands of the LaTeX kernel. If you do not specify the values of boolean options, they default to `true`. These options can also be changed in the preamble area and main body using the global control command `\spbset`.

`text` = `true`|`false`
   `\textsuperscript` and `\textsubscript` are equivalent to the `\super` and `\sub` commands. The command `\defspbstyle` can be used to define the style `textsp` and `textsb` to change the format of the text superscripts and subscripts.

`math` = `true`|`false`
   `\sp` and `\sb` are equivalent to the `\super` and `\sub` commands. The command `\defspbstyle` can be used to define the style `mathsp` and `mathsb` to change the format of the math superscripts and subscripts.

`math*` = `true`|`false`

Exp v1.46j

   The `^` and `_` tokens in the math mode are equivalent to the `\super` and `\sub` commands. The command `\defspbstyle` can be used to define the style `mathsp*` and `mathsb*` to change the format of the math superscripts and subscripts.

`foot` = `true` | `false`

The format of the footnote mark match the superscripts global move and format settings. `\defspbstyle` can be used to define the superscripts style `fnmark` to change the format of the footnote mark.

```
\defspbstyle{textsp}{vmove=-1pt,cmd=\bfseries}
\defspbstyle{mathsb}{hmove=0.5em}
\defspbstyle{fnmark}{cmd=\small\color{red}}
```

`both` = `true` | `false`

The values of `text` and `math` two options are `true` or `false` at the same time.

`all` = `true` | `false`

The values of `text`, `math` and `foot` three options are `true` or `false` at the same time.

```
\usepackage[both]{spbmark}
\usepackage[text,foot=true]{spbmark}
\spbset{all=false,math}
```

`spcore` = `trad` | `none`

New
v1.46k

Kernel command case for text and math modes superscripts. If `none` is selected, support for the superscripts kernel command is removed, and the `cmd` and `vmove` options need to be set to control the size and offset of the superscripts.

```
\usepackage[spcore=none]{spbmark}
\usepackage{graphicx}
\spbset{spcmd=\scalebox{0.6},spvmove=5pt}
```

`sbcore` = `trad` | `none`

New
v1.46k

Option for subscripts kernel command case, similar to the `spcore` option.

2. User commands

There are currently three commands to print superscripts and subscripts. Their format can be set locally using the optional parameter of the command, or set globally using a ⟨*key-value list*⟩, see Section 3.

`\super`*[⟨*kv list*⟩]{⟨*content*⟩}[⟨*kv list*⟩]

This is a superscript output command. The two ⟨*kv list*⟩ are equivalent.

`\sub`*[⟨*kv list*⟩]{⟨*content*⟩}[⟨*kv list*⟩]

This is a subscript output command. The two ⟨*kv list*⟩ are equivalent.

When the horizontal movement distance of the marker is non-negative, since the `\super` and `\sub` commands put the marker in a horizontal box, the horizontal position is staggered when using superscript and subscript consecutively, which is a different mechanism than the sequential use of `^` and `_` symbols in math mode. The following three commands save the width of the previous marker, use it for negative move of the `hmove` option of the next marker, and provide the corresponding alignment.

It should be noted that the following commands should save the longer width of the marker, that is, the longer width of the marker in the front, the shorter width of the marker in the back. This is due to the fact that horizontal movement uses a negative distance to leave the marker in a zero-width box state.

`\llastwd`

Save the width of the previous marker and provide left alignment. This is actually the width of the previous marker, and is the same as the value of `\rlastwd` after the next command is used.

`\clastwd`

Save the width of the previous marker and provide center alignment.

`\rlastwd`

Save the width of the previous marker and provide right alignment. Actually this is the width of the next marker.

```
1A\super{bcd}\sub{e}B \\
2A\super{bcd}\sub[hmove=-\llastwd]{e}B \\
3A\sub{e}\super{bcd}[hmove=-\llastwd]B \\
4A\super{bcd}\sub[hmove=-\clastwd]{e}B \\
5A\super{bcd}\sub[hmove=-\rlastwd]{e}B
```

$1A^{bcd}{}_{e}B$
$2A^{bcd}_{e}B$
$3A^{bcd}_{e}B$
$4A^{bcd}_{e}B$
$5A^{bcd}_{e}B$

`\supersub*[⟨kv list⟩]{⟨super cont⟩}{⟨sub cont⟩}[⟨kv list⟩]`

This is a command that outputs both superscript and subscript at the same time. You can also use the shorter command `\spb` instead of it. The two ⟨kv list⟩ are equivalent.

`\superwd`

Save the width of the superscript in the `\supersub` command.

`\subwd`

Save the width of the subscript in the `\supersub` command.

`\maxwd`

Save the maximum width of superscript and subscript in the `\supersub` command. That is, it is the larger of the `\superwd` and `\subwd` commands.

If horizontally move is negative and its absolute value is less than the maximum width of the marker, then the marker overlaps the subsequent text. To avoid this use the command with the asterisk parameter or adjust the horizontal distance with the marker length commands.

```
\spbset{spvmove=5pt,vsep=1.8ex,spcmd=\color{blue}}
1A\super[hmove=-8pt]{super}B \\
2A\supersub[hmove=-8pt]{examsuper}{sub}B \\
3A\super*[hmove=-8pt]{super}B \\
4A\supersub*[hmove=-8pt]{examsuper}{sub}B \\
5A\super[hmove=-8pt]{super}\hspace{\dimexpr(\llastwd-8pt)}B \\
6A\supersub[hmove=-8pt]{examsuper}{sub}\hspace{\dimexpr(\maxwd-8pt)}B
```

$1A\,^{super}B$
$2A\,^{examsuper}_{sub}B$
$3A\quad^{super}\ B$
$4A\quad^{examsuper}_{sub}\ B$
$5A\quad^{super}\ B$
$6A\quad^{examsuper}_{sub}\ B$

`\defspbstyle{⟨style name⟩}{⟨kv list⟩}`

Defines the style of the superscripts or subscripts used for the `style` option.

`\spbifmath{⟨math code⟩}{⟨text code⟩}`

In some cases, `math` or `text` output modes require different code for format or move. This command can be used when using the `match` option or changing the output mode locally, whitch should be used in the move or format options. It can switch the corresponding code according to different output modes.

`\spbshortkv{⟨short opt⟩}{⟨key value⟩}`

Converts any existing key-value pair to a shorthand option. The value of key #1 indicates that the shorthand option needs to be assigned a value.

```
\spbshortkv{x}{sphmove=#1}
\defspbstyle{fancy}{sbcmd=\color{blue},mode=math}
\spbshortkv{mysb}{style=fancy}
1A\super[x=2pt]{b} \\
2A\sub{b}[mysb]
```

$1A\,^{b}$
$2A_{b}$

The options common to ⟨kv list⟩ of the three commands are as follows. These options can also be used in ⟨key-value list⟩ of the `\spbset` command to represent the simultaneous setting of three commands. The symbol ☺ before options indicates that they are invalid for the `\supersub` command.

`style` = {⟨*style name*⟩}

Use the ⟨*style name*⟩ defined by the `\defspbstyle` command to make it work global or local.

`mode` = `text|math|` `match`

The mode of superscripts or subscripts output can be `text` or `math` mode. The `match` option automatically matches output modes according to the current mode.

`vmove` = {⟨*fixed length*⟩}   Default: `0pt`

Vertical move of superscripts or subscripts. Represents the extra vertical distance `vsep` between superscript and subscript in the `\supersub` command. The vertical movement starts at the marker horizontal baseline position.

`hmove` = {⟨*fixed length*⟩}   Default: `0pt`

Horizontal move of superscripts or subscripts. Represents the common move of superscript and subscript in the `\supersub` command. The starting point for horizontal movement is to the left of the marker. Moved values can be expressed mathematically:

```
1A\super{b}[vmove=0.4ex+4pt/2,hmove=2pt*3-11.5pt] \\
2A\supersub[vsep={3pt,6pt-2bp}]{examsuper}{sub}
```

$$1A^{b} \\ 2A^{examsuper}_{sub}$$

`cmd` = {⟨*format cmds*⟩}

The format commands of superscripts or subscripts. The last command can take a parameter, which accepts superscript or subscript. Represents the format option `spbcmd` of superscript and subscript in the `\supersub` command.

`cmd+` = {⟨*format cmds*⟩}

Add code to the previous option `cmd`.

`height` = {⟨*fixed length*⟩}

The height of the marker box above the baseline. It is recommended that the superscript adjusts the height and the subscript adjusts the depth.

`depth` = {⟨*fixed length*⟩}

The depth of the marker box below the baseline.

☺ `thiswd` = `auto|` `keep`

Sets how the length of the this marker is handled after this command is used. If it is `keep`, the width of the previous marker remains the same. If it is `auto`, if the next token is a marker command and the upper or lower position is the opposite of the this command, the width of the this marker is maintained, otherwise the width automatically returns to zero.

```
\spbset{sbhmove=-\llastwd,sbthiswd=auto,sbcmd=\color{red}}
1A\super{bcd}\sub[thiswd=keep]{e}Some texts\sub{e} \\
2A\super{bcd}\sub{e}Some texts\sub{e} \\
3A\sub{bcd}[thiswd=keep]\sub{e}Some texts\sub{e} \\
4A\sub{bcd}\sub{e}Some texts\sub{e}
```

$$1A^{bcd}_{e}\text{Some texts}_{e} \\ 2A^{bcd}_{e}\text{Some texts}_{e} \\ 3A_{bcd}\text{Some texts}_{e} \\ 4A_{bcde}\text{Some texts}_{e}$$

☺ `regex` = {⟨*regular expression*⟩}

After using a superscript or subscript command, this option is used to determine whether the following tokens match the ⟨*regular expression*⟩. If it does not, the width of the this marker returns to zero.

```
\newcommand{\mysp}[2][]{\super[regex=\c{mysb},#1]{#2}}
\newcommand{\mysb}[2][]{\sub[regex=\c{mysp},#1]{#2}}
```

☺ `nobox` = `true|` `false`

New
v1.46v

Causes the superscripts or subscripts not to be saved in the box, in which case the offset-related options are invalid. For example, there is a difference in nesting superscripts in math mode.

```
1A$\super{c\super{d}}$B \\
2A$\super{c\super[nobox]{d}}$B
```

$$1\mathrm{A}^{c^d}\mathrm{B}$$
$$2\mathrm{A}^{c^d}\mathrm{B}$$

3. Global control interface

\spbset{⟨*key-value list*⟩}

    spbmark uses the \spbset command to control the global default format of superscripts and subscripts. These options also apply to ⟨*kv list*⟩ of the above print commands. The values set by it will be overwritten by the optional parameter of superscripts and subscripts commands.

    Note that the relative length units in the options such as ex and em are for the body font size, not the marker itself.

    The following list of keys control the format both of superscripts or subscripts. They can also be used in the optional parameter of the \super or \sub command.

spvmove = {⟨*fixed length*⟩}   Default: 0pt
    Extra vertical move of the superscripts.

sphmove = {⟨*fixed length*⟩}   Default: 0pt
    Extra horizontal move of the superscripts.

sbvmove = {⟨*fixed length*⟩}   Default: 0pt
    Extra vertical move of the subscripts.

sbhmove = {⟨*fixed length*⟩}   Default: 0pt
    Extra horizontal move of the subscripts.

nohmove
    Cancel the horizontal move of superscripts and subscripts at the same time.

novmove
    Cancel the vertical move of superscripts and subscripts at the same time.

spcmd = {⟨*format cmds*⟩}
    The format commands of superscripts. The last command in the code can take an argument, which is a superscripts.

spcmd+ = {⟨*format cmds*⟩}
    Add code to the previous option spcmd.

sbcmd = {⟨*format cmds*⟩}
    The format commands of subscripts. Similar to the spcmd option, but for subscripts.

sbcmd+ = {⟨*format cmds*⟩}
    Add code to the previous option sbcmd.

spheight = {⟨*fixed length*⟩}
    The height of the superscript box above the baseline.

spdepth = {⟨*fixed length*⟩}
    The depth of the superscript box below the baseline.

sbheight = {⟨*fixed length*⟩}
    The height of the subscript box above the baseline.

sbdepth = {⟨*fixed length*⟩}
    The depth of the subscript box below the baseline.

spthiswd = auto | keep
    Width processing mode after using this superscripts. See Section 2 for the meaning of this option.

sbthiswd = auto | keep

Width processing mode after using this subscripts. See Section 2 for the meaning of this option.

spregex = {⟨*regular expression*⟩}

After the superscripts command is used, the regular expression rules of the following tokens are matched. See Section 2 for the meaning of this option.

sbregex = {⟨*regular expression*⟩}

Similar to the spregex option, but for subscripts.

If sphmove is positive, the superscripts or subscripts moves to the right, conversely it moves to the left. The positive direction of vertical offset of subscripts and superscripts is opposite. For superscripts, if spvmove is positive, the superscripts moves up, conversely it moves down. For subscripts, if spvmove is positive, the subscripts moves down, conversely it moves up.

The following list of keys control the format of superscripts and subscripts. They can also be used in the optional parameter of the \supersub command.

spbhmove = {⟨*fixed length*⟩}   Default: 0pt

Extra vertical move of the superscripts and the subscripts.

spbcmd = {⟨*super cmds*⟩,⟨*sub cmds*⟩}

The format commands of superscripts and subscripts. The first part is in superscripts format, and the latter part is in subscripts format. They are separated by commas, or only the first part exists.

spbcmd+ = {⟨*super cmds*⟩,⟨*sub cmds*⟩}

Add code to the previous option spbcmd.

spbheight = {⟨*fixed length*⟩}

The height of the superscript box above the baseline.

spbdepth = {⟨*fixed length*⟩}

The depth of the subscript box below the baseline.

vsep = {⟨*super move*⟩,⟨*sub move*⟩}   Default: 0.6ex

The extra vertical distance between superscripts and subscripts. The front part is the movement amount of the superscripts, and the latter part is the movement amount of the subscripts. If only the front part means that both are vertically shifted by one-half of the current value.

halign = l | c | r

The alignment of superscripts and subscripts, which contains l, c, and r parameter respectively for left, center, and right alignment.

4. Marker hooks

\AddToHookWithArguments{spb/super/before}[⟨*label*⟩]{⟨*code*⟩}
\AddToHookWithArguments{spb/super/after}[⟨*label*⟩]{⟨*code*⟩}
\AddToHookWithArguments{spb/sub/before}[⟨*label*⟩]{⟨*code*⟩}
\AddToHookWithArguments{spb/sub/after}[⟨*label*⟩]{⟨*code*⟩}

New
v1.46r

spbmark provides hooks to markers of \super and \sub with arguments in front and back positions, this feature requires support in the LaTeX kernel after **2023/06/01**. Argument #1 in the ⟨*code*⟩ represents super or sub, and argument #2 represents the marker content itself.

```
\AddToHookWithArguments{spb/super/before}{\spbifmath{(#1-#2)}{[#1-#2]}}
$\super{a}$\super{b}
\RemoveFromHook{spb/super/before}
```

$(super - a)^a$[super-b]^b

```
\AddToHookWithArguments{spb/super*/before}[⟨label⟩]{⟨code⟩}
\AddToHookWithArguments{spb/super*/after}[⟨label⟩]{⟨code⟩}
\AddToHookWithArguments{spb/sub*/before}[⟨label⟩]{⟨code⟩}
\AddToHookWithArguments{spb/sub*/after}[⟨label⟩]{⟨code⟩}
```
Hook name with an asterisk indicates that this is the hook for the marker in the \supersub command.

```
\AddToHookWithArguments{cmd+/super/after}[⟨label⟩]{⟨code⟩}
\AddToHookWithArguments{cmd+/sub/after}[⟨label⟩]{⟨code⟩}
```
The above hooks are located in the marker horizontal boxes, and these two hooks are located at the end of the group outside of the \super and \sub commands. Due to the addition of the peek command at the end of marker commands, using such as cmd/super/after hook directly would result in a peek error.

5. Examples of use

Here is a list of the three commands, please pay attention to the usage of optional parameter. Note when the horizontal move is negative, the starting point is at the right end of the mark.

```
\defspbstyle{fancy}{cmd=\color{purple}}
\spbset{spbcmd={\spbifmath{\mathtt}{\ttfamily},\color{blue}}}
1A\super[vmove=0.2ex,hmove=0.2em,cmd=\textcolor{red}]{exam}B \\
2$A\sub[style=fancy,cmd+=\mathsf,mode=math]{exam}B$ \\
3A\supersub[vsep=0.6ex,halign=c]{examsuper}{sub}B \\
4A\super{c}[vmove=5pt,hmove=-5.5pt]B\sub[vmove=5pt,hmove=-5pt]{d}AB
```

$1A^{\text{exam}}B$
$2A_{\text{exam}}B$
$3A^{\text{examsuper}}_{\text{c}}{}_{\text{sub}}B$
$4A\overset{}{\underset{\text{d}}{}}BAB$

5.1. siunitx

It can also be used with the siunitx package to output superscripts and subscripts in the unit:

```
\spbset{spcmd=\spbifmath{}{\color{purple}}}
\sisetup{text-superscript-command=\super}
1-\qty[mode=text]{10}{A^2} \\
2-\unit[mode=math]{kg.m/s\super[vmove=-1pt]{2}} \\
3-\qty[mode=text]{30}{A\supersub[hmove=1pt,cmd=\color{blue}]{b}{c}} \\
4-\spbset{sbhmove=2pt}\unit[mode=text]{A\sub{b}}
```

$1\text{-}10\,A^{2}$
$2\text{-kg}\,m/s^{2}$
$3\text{-}30\,A^{b}_{c}$
$4\text{-}A_{b}$

5.2. realscripts

If the OpenType text font you are using does not have optical sizes, the superscripts and subscripts may not appear correctly. Loading the realscripts package before the spbmark package fixes this behavior.

```
\usepackage{realscripts,spbmark}
```

5.3. altsubsup

The altsubsup package allows to write alternate superscripts and subscripts in math mode with ^[⟨mark⟩] and _[⟨mark⟩] dimensions. When the package is loaded using the spbmark option, the superscripts and subscripts mechanism of spbmark is used, you can use \defspbstyle to define its superscripts style altsup and subscripts style altsub.

Note that the mechanism of altsubsup package adds \text command to markers in math mode, which invalidates the match option. You can restore it by setting the format command to \relax.

```
\SetAltSubSupCommands{\relax}
\defspbstyle{altsup}{sphmove=-\llastwd,spthiswd=auto,spvmove=1pt}
$A_[bcd]^[e]B$
```
$A_{bcd}^{e}B$

5.4. footnote

spbmark also patches the footer markers for standard document class and KOMA-Script. You can format the footer markers by redefining the \fnmarkfont command. Note that extra horizontal move does not work with footnote markers.

6. Developer commands

If you need to use the original definitions of \textsuperscript, \textsubscript, \sp and \sb after using the text or math option, then you can use the following commands:

\spb@textsuperscript@save{⟨*content*⟩}
  Save the original definition of the \textsuperscript command, output superscripts in the text mode.

\spb@textsubscript@save{⟨*content*⟩}
  Save the original definition of the \textsubscript command, output subscripts in the text mode.

\spb@math@super@save{⟨*content*⟩}
  Save the original definition of the \sp command, output superscripts in the math mode.

\spb@math@sub@save{⟨*content*⟩}
  Save the original definition of the \sb command, output subscripts in the math mode.

7. Known issues

At present, the vertical and horizontal move are effective for the unit commands in the siunitx macro package. However, due to the special mechanism that the decimal point is not recognized correctly because it's converted to a space in the \unit command, it's recommended to use pt as the unit of move.

A. Changelog

1.46j (2024/05/19)

- Add option math*: The ^ and _ tokens in the math mode are equivalent to the \super and \sub commands.

1.46k (2024/06/21)

- Add options spcore and sbcore: Set kernel commands for superscripts and subscripts of text and math modes.

1.46r (2024/10/29) — Give hooks to markers with parameters at the front and back positions.

B. References

[Lab22] Julien Labbé. altsubsup. version 1.1, Mar. 15, 2022 (or newer).
         URL: https://ctan.org/pkg/altsubsup.

[Rob16] Will Robertson. realscripts. version 0.3d, Feb. 13, 2016 (or newer).
         URL: https://www.ctan.org/pkg/realscripts.

[Tea23] The LATEX Project Team.
The LATEX3 Interfaces. version 2023-12-11, Dec. 11, 2023 (or newer).
URL: https://ctan.org/pkg/l3kernel.

[Wri22] Joseph Wright. siunitx. version 3.0.48, Feb. 7, 2022 (or newer).
URL: https://www.ctan.org/pkg/siunitx.

C. Index