THE SPBMARK PACKAGE

v1.46f 2024/04/24

Customize superscript and subscript

Qu Yi

https://github.com/texno3/spbmark

toquyi@163.com

spbmark provides three commands `\super`, `\sub` and `\supersub` to improve the layout of superscript and subscript 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 `\sbpset`.

`text` = `true`|`false`

☆ Changed `\textsuperscript` and `\textsubscript` are equivalent to the `\super` and `\sub` commands. At this point, you can use the `\defspbstyle` command to define the style `textsp` and `textsb` to change the format of the text superscript and subscript.

`math` = `true`|`false`

☆ Changed `\sp` and `\sb` are equivalent to the `\super` and `\sub` commands. At this point, you can use the `\defspbstyle` command to define the style `mathsp` and `mathsb` to change the format of the math superscript and subscript.

`foot` = `true`|`false`

☆ Changed

The format of the footnote mark match the superscript global move and format settings. At this point, you can use the `\defspbstyle` command to define the superscript 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}
```

2. User commands

There are currently three commands to set superscript and subscript. Their format can be set temporarily using the optional parameters 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$^{\mathrm{bcd}}{}_{\mathrm{e}}$B

2A$^{\mathrm{bcd}}_{\mathrm{e}}$B

3A$_{\mathrm{e}}$B$^{\mathrm{bcd}}$

4A$^{\mathrm{bcd}}_{\;\mathrm{e}}$B

5A$^{\mathrm{bcd}}_{\;\;\mathrm{e}}$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 previous super-subscript command.

`\subwd`

Save the width of the subscript in the previous super-subscript command.

`\maxwd`

Save the maximum width of superscript and subscript in the previous super-subscript 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
```

```
 super
1AB
 examsuper
2AB
 sub
 super
3A   B
 examsuper
4A         B
 sub
 super
5A   B
 examsuper
6A         B
 sub
```

`\defspbstyle{⟨style name⟩}{⟨kv list⟩}`
Defines the style of the superscript or subscript 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$^{\text{b}}$

2A$_{b}$

The options common to ⟨*kv list*⟩ of the three commands are as follows. They can also be used in ⟨*key-value list*⟩ of the `\spbset` command. In this situation, in addition to `style` and `mode`, they will be set according to the type of the previous most recent command.

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

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

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

Horizontal move of superscript or subscript. Represents the common move of superscript and subscript in super-subscript 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}
```

1$\overset{\text{b}}{\text{A}}$

2A$^{\text{examsuper}}_{\text{sub}}$

`cmd` = {⟨*format cmds*⟩}

The format commands of superscript or subscript. The last command can take a parameter, which accepts superscript or subscript. Represents the format of superscript and subscript in the superscript and subscript commands.

`cmd+` = {⟨*format cmds*⟩}

Add code to the previous global superscript or subscript format commands.

`height` = {⟨*fixed length*⟩}

The distance between the superscript or subscript baseline and the previous line. It is recommended that the superscript adjusts the height and the subscript adjusts the depth.

`depth` = {⟨*fixed length*⟩}

The distance between the superscript or subscript baseline and the next line.

`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 superscript or subscript output can be `text` or `math` mode. The `match` option automatically matches output modes according to the current mode.

`thiswd` = `auto` | `keep`

☆ New

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} \\
3A\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$^{bcd}_{e}$Some texts$_{e}$

3A$^{bcd}_{e}$Some texts$_{e}$

3A$_{bcd}$Some texts$_{e}$

4A$_{bcde}$Some texts$_{e}$

`regex` = {⟨*regular expression*⟩}

☆ New

After using the 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}}
```

3. Global control interface

`\spbset`{⟨*key-value list*⟩}

spbmark uses the `\spbset` command to control the global default format of superscript and subscript. These options also apply to ⟨*kv list*⟩ of the above commands. The values set by it will be overwritten by the optional parameters of the superscript and other commands.

The following list of keys control the format both of superscript or subscript.

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

Extra vertical move of the superscript.

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

Extra horizontal move of the superscript.

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

Extra vertical move of the subscript.

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

Extra horizontal move of the subscript.

nohmove

Cancel the horizontal move of superscript and subscript at the same time.

novmove

Cancel the vertical move of superscript and subscript at the same time.

spcmd = {⟨*format cmds*⟩}

The format commands of superscript. The last command in the code can take an argument, which is a superscript.

spcmd+ = {⟨*format cmds*⟩}

Add code to the previous global superscript format commands.

sbcmd = {⟨*format cmds*⟩}

The format commands of subscript. The last command in the code can take an argument, which is a subscript.

sbcmd+ = {⟨*format cmds*⟩}

Add code to the previous global subcript format commands.

spheight = {⟨*fixed length*⟩}

The distance between the superscript baseline and the previous line.

spdepth = {⟨*fixed length*⟩}

The distance between the superscript baseline and the next line.

sbheight = {⟨*fixed length*⟩}

The distance between the subscript baseline and the previous line.

sbdepth = {⟨*fixed length*⟩}

The distance between the subscript baseline and the next line.

spthiswd = auto|keep

☆ New Width processing mode after using this superscript. See Section 2 for the meaning of this option.

`sbthiswd = auto|keep`

☆ New Width processing mode after using this subscript. See Section 2 for the meaning of this option.

`spregex = {`⟨*regular expression*⟩`}`

☆ New After the superscript 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*⟩`}`

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

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

The following list of keys control the format of superscript and subscript.

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

Extra vertical move of the superscript and the subscript.

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

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

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

Add code to the previous global superscript and subscript format commands.

`spbheight = {`⟨*fixed length*⟩`}`

The distance between the superscript baseline and the previous line.

`spbdepth = {`⟨*fixed length*⟩`}`

The distance between the subscript baseline and the next line.

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

The extra vertical distance between superscript and subscript. The front part is the movement amount of the superscript, and the latter part is the movement amount of the subscript. 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 superscript and subscript, which contains `l`, `c`, and `r` parameters respectively for left, center, and right alignment.

4. Examples of use

Here is a list of the three commands, please pay attention to the usage of optional parameters. 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 \\
5A\super{c\super{d}}B
```

1A $^{\text{exam}}$B

2$A_{\text{exam}}B$

3A$^{\texttt{examsuper}}_{\text{sub}}$B

4A^{c}B$_{d}$AB

5A$^{c^{d}}$B

4.1. siunitx

It can also be used with the siunitx package to output superscript and subscript in the unit:

```
\unit[mode=math]{kg.m/s\super[vmove=-1pt]{2}} \\
\qty[mode=text]{30}{A\supersub[hmove=1pt,cmd=\color{red}]{b}{c}} \\
\spbset{sbhmove=2pt}\unit[mode=text]{A\sub{b}}
```

---

kg m/s^{2}

30 A$^{b}_{c}$

A $_{b}$

4.2. realscripts

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

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

4.3. altsubsup

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

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

$A_{bcd}^{e}B$

4.4. 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.

5. 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 superscript in the text mode.

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

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

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

6. 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. 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.

[Tea22] The LaTeX Project Team. The LaTeX3 Interfaces. Feb. 24, 2022 (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.

B. Index