

Brief description of the package: This package is suitable to add chords to lyrics. The main advantages of using this package is that chords are written before the text they refer to and L^AT_EX takes care of typesetting them above the lyrics in the right place: thanks to this we avoid the task of correctly aligning chords above the lyrics, which is tedious in the case of a monospaced font and absolutely non-trivial in the case of variable-width fonts.

1 Usage

1.1 Chords

The following commands typeset a chord above a given text:

| | | | | | | |
|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| A | B | C | D | E | F | G |
| <code>\A</code> | <code>\B</code> | <code>\C</code> | <code>\D</code> | <code>\E</code> | <code>\F</code> | <code>\G</code> |
| A ^b | B ^b | C ^b | D ^b | E ^b | F ^b | G ^b |
| <code>\Ab</code> | <code>\Bb</code> | <code>\Cb</code> | <code>\Db</code> | <code>\Eb</code> | <code>\Fb</code> | <code>\Gb</code> |
| A [#] | B [#] | C [#] | D [#] | E [#] | F [#] | G [#] |
| <code>\As</code> | <code>\Bs</code> | <code>\Cs</code> | <code>\Ds</code> | <code>\Es</code> | <code>\Fs</code> | <code>\Gs</code> |
| A ^m | B ^m | C ^m | D ^m | E ^m | F ^m | G ^m |
| <code>\Am</code> | <code>\Bm</code> | <code>\Cm</code> | <code>\Dm</code> | <code>\Em</code> | <code>\Fm</code> | <code>\Gm</code> |
| A ^b m | B ^b m | C ^b m | D ^b m | E ^b m | F ^b m | G ^b m |
| <code>\Abm</code> | <code>\Bbm</code> | <code>\Cbm</code> | <code>\Dbm</code> | <code>\Ebm</code> | <code>\Fbm</code> | <code>\Gbm</code> |
| A [#] m | B [#] m | C [#] m | D [#] m | E [#] m | F [#] m | G [#] m |
| <code>\Asm</code> | <code>\Bsm</code> | <code>\Csm</code> | <code>\Dsm</code> | <code>\Esm</code> | <code>\Fsm</code> | <code>\Gsm</code> |

For example `\Bb{text}` produces the following: ^{B^b}text.

Each command takes an optional argument, in order to allow users to typeset variations of the above chords.

For example `\G[7sus4]{text}` produces the following: ^{G7sus4}text.

An Example: London Bridge is Falling Down

| | |
|--|--|
| ^G London bridge is falling down | <code>\G</code> London bridge is falling down |
| ^{D7} Falling down, | <code>\D[7]</code> Falling down, |
| ^G falling down, | <code>\G</code> falling down. |
| ^G London bridge is falling down, | <code>\G</code> London bridge is falling down, |
| ^D my fair lady. | <code>\D</code> my fair <code>\G</code> lady. |

1.2 Operations on chords

It is sometimes useful to transpose a song by a certain number of half-tones. This package provides the following commands to do this:

| | | |
|---------------------------------|---------------------------------|----------------------------------|
| <code>\transposeOneUp</code> | <code>\transposeTwoUp</code> | <code>\transposeThreeUp</code> |
| <code>\transposeFourUp</code> | <code>\transposeFiveUp</code> | <code>\transposeSixUp</code> |
| <code>\transposeOneDown</code> | <code>\transposeTwoDown</code> | <code>\transposeThreeDown</code> |
| <code>\transposeFourDown</code> | <code>\transposeFiveDown</code> | <code>\transposeSixDown</code> |

Things can be brought back to normal via the command `\notranspose`.

Another possibility is to convert all chords with a \flat to the corresponding chords with a \sharp or vice-versa: this is achieved with the commands `\sharponly` and `\flatonly`.

Finally it is possible to avoid all occurrences of $E\sharp$, $F\flat$, $B\sharp$ and $C\flat$ and replace them with F, E, C and B respectively by using the command `\normalize`.

1.3 Package Options

All commands from §1.2 can be run at the beginning of the document by using the desired ones as package options — the initial backslash has to be omitted of course!

1.4 Changing the Way Things Look

The line spacing in presence of a chord is controlled by the length `\crdheight` — the default setting is `3ex`.

The font of the chord is controlled by the command `\crdfont` — the default setting is `\footnotesize` `\sffamily`.

All chords are specific instances of the command `\CHORD`, which takes two mandatory arguments and an optional one: the first argument is concatenated to the optional argument (respectively the chord and its variation) and is typeset above the text given as second argument.

It is possible to change the way a chord is typeset by redefining this command — and this does not compromise the functionality of the commands in §1.2 as long as the meaning of the arguments of `\CHORD` is not changed.

2 Contacts

If you have comments or want to report any bug, please send a mail to gtrcrd@tex.nopkoguonet.net.