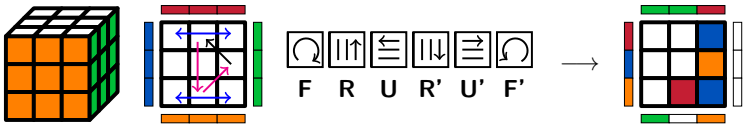
# The RUBIKCUBE package

RWD Nickalls (dick@nickalls.org)
A Syropoulos (asyropoulos@yahoo.com)

This file describes version 5.0 (2018/02/25)
www.ctan.org/pkg/rubik

**Abstract**

The RUBIKCUBE package provides LaTeX commands and macros for typesetting Rubik cube (3x3x3) notation, configurations, and rotation sequences using the TikZ graphic language. It is part of the Rubik 'bundle'.




## Contents

**1 Introduction** **7**
1.1 Requirements . . . 7
1.2 rubikrotation package . . . 8
1.3 rubikpatterns . . . 8
1.4 rubiktwocube . . . 8
1.5 Copyright . . . 9

**2 Installation** **9**
2.1 Generating the RUBIKCUBE files . . . 9
2.2 RubikExamples file . . . 10
2.3 Placing the files . . . 10
2.4 Usage . . . 10

**3 Command conventions** **11**
3.1 Keywords Rubik and Two . . . 11
3.2 Environments . . . 12
3.3 Capital letters . . . 12
3.4 XYZ argument ordering . . . 12
3.5 Trailing % on the end of commands . . . 13
3.6 Cubies, cubicles, faces and facelets . . . 13

**4 Rubik cube coordinates** **14**
4.1 Size of cube minipage . . . . . 14

**5 TikZ picture environment** **15**

**6 ShowCube command** **16**

**7 Optimum strategy** **17**

**8 Colour commands** **18**
8.1 Colour state of the cube . . . . . 19
8.2 RubikFace commands . . . . . 20
8.3 RubikSolvedConfig command . . . . . 21
8.4 RubikCubeSolved commands . . . . . 21
8.5 RubikCubeGrey.. commands . . . . . 23
8.6 RubikSlice commands . . . . . 23

**9 Rotation commands** **24**
9.1 Typesetting . . . . . 25
9.2 Face rotations . . . . . 25
9.3 Inner-slice rotations . . . . . 26
9.4 Outer-slice rotations . . . . . 27
9.5 Wide rotations . . . . . 27
9.6 Axis rotations . . . . . 28
9.7 Examples . . . . . 29
9.8 Backwards compatibility . . . . . 30
9.9 Listing of all rotation commands . . . . . 30
9.9.1 Randelshofer notation . . . . . 34
9.10 The rubikfont . . . . . 36

**10 Draw commands** **37**
10.1 Error message . . . . . 38
10.2 DrawCubie commands . . . . . 38
10.3 textCubie commands . . . . . 39
10.4 DrawRubikCube commands . . . . . 39
10.5 DrawRubikFace.. commands . . . . . 41
10.6 Drawing Sidebars (Face) . . . . . 41
10.7 Sidebar parameters . . . . . 42
10.8 NoSidebar command . . . . . 42
10.9 Drawing Sidebars (Cube) . . . . . 43
10.10 DrawRubikFlat commands . . . . . 44
10.11 DrawNCube . . . . . 44

**11 ShowSequence command** **45**
11.1 Trailing digits . . . . . 46

**12 SequenceBrace commands** **47**

**13 Arrows** **48**

**14 Final example** **50**
14.1 Without using \RubikRotation . . . . . 51

**15 Deprecated commands** **51**

**16 Known limitations** **51**

**17 Change history** **52**

**18 Acknowledgements** **54**

**19 References** **54**

**20 The code (rubikcube.sty)** **57**
20.1 Package heading . . . . . 57
20.2 Colours . . . . . 58
20.3 The rubikfont . . . . . 58
20.4 ShowCube command . . . . . 58
20.5 ShowSequence command . . . . . 59
20.6 SequenceBrace commands . . . . . 61
20.7 RubikFace commands . . . . . 61
20.8 RubikCubeGrey command . . . . . 64
20.9 SolvedConfig command . . . . . 64
20.10 RubikCubeGreyAll command . . . . . 64
20.11 RubikCubeSolved command . . . . . 65
20.12 Slice commands . . . . . 65
20.13 Cube drawing macros . . . . . 66
20.13.1 Viewing direction . . . . . 67
20.14 DrawRubikFlatX commands . . . . . 71
20.14.1 DrawRubikFaceXSide commands . . . . . 76
20.15 DrawRubikFace commands . . . . . 79
20.16 Sidebars (Face) . . . . . 80
20.16.1 Drawing and allocating a colour to a single facelet sidebar 81
20.17 Sidebars (Cube) . . . . . 83
20.17.1 Sidebars: RU view . . . . . 83
20.17.2 Sidebars: LU view . . . . . 85
20.17.3 Sidebars: RD view . . . . . 87
20.17.4 Sidebars: LD view . . . . . 89
20.18 DrawNCube command . . . . . 90
20.19 Drawing single cubies . . . . . 92
20.20 Text cubies . . . . . 93
20.21 Rotation commands . . . . . 94
20.21.1 Introduction . . . . . 94
20.21.2 DrawNotationBox . . . . . 95
20.21.3 Some useful internal commands . . . . . 95

20.21.4 Using \@join . . . 97
20.21.5 Rotation B . . . 98
20.21.6 Rotation Bp . . . 98
20.21.7 Rotation Bw . . . 98
20.21.8 Rotation Bwp . . . 98
20.21.9 Rotation Bs . . . 99
20.21.10 Rotation Bsp . . . 99
20.21.11 Rotation Ba . . . 99
20.21.12 Rotation Bap . . . 99
20.21.13 Rotation D . . . 100
20.21.14 Rotation Dp . . . 100
20.21.15 Rotation Dw . . . 101
20.21.16 Rotation Dwp . . . 101
20.21.17 Rotation Ds . . . 102
20.21.18 Rotation Dsp . . . 102
20.21.19 Rotation Da . . . 103
20.21.20 Rotation Dap . . . 103
20.21.21 Rotation E . . . 104
20.21.22 Rotation Ep . . . 104
20.21.23 Rotation F . . . 105
20.21.24 Rotation Fp . . . 105
20.21.25 Rotation Fw . . . 106
20.21.26 Rotation Fwp . . . 106
20.21.27 Rotation Fs . . . 107
20.21.28 Rotation Fsp . . . 107
20.21.29 Rotation Fa . . . 107
20.21.30 Rotation Fap . . . 107
20.21.31 Rotation L . . . 108
20.21.32 Rotation Lp . . . 108
20.21.33 Rotation Lw . . . 109
20.21.34 Rotation Lwp . . . 109
20.21.35 Rotation Ls . . . 110
20.21.36 Rotation Lsp . . . 110
20.21.37 Rotation La . . . 111
20.21.38 Rotation Lap . . . 111
20.21.39 Rotation M . . . 112
20.21.40 Rotation Mp . . . 112
20.21.41 Rotation R . . . 113
20.21.42 Rotation Rp . . . 113
20.21.43 Rotation Rw . . . 114
20.21.44 Rotation Rwp . . . 114
20.21.45 Rotation Rs . . . 115
20.21.46 Rotation Rsp . . . 115
20.21.47 Rotation Ra . . . 116
20.21.48 Rotation Rap . . . 116
20.21.49 Rotation S . . . 117

20.21.50 Rotation Sp . . . 117
20.21.51 Rotation Su . . . 117
20.21.52 Rotation Sup . . . 118
20.21.53 Rotation Sd . . . 118
20.21.54 Rotation Sdp . . . 118
20.21.55 Rotation Sl . . . 119
20.21.56 Rotation Slp . . . 119
20.21.57 Rotation Sr . . . 119
20.21.58 Rotation Srp . . . 119
20.21.59 Rotation Sf . . . 120
20.21.60 Rotation Sfp . . . 120
20.21.61 Rotation Sb . . . 120
20.21.62 Rotation Sbp . . . 121
20.21.63 Rotation U . . . 121
20.21.64 Rotation Uw . . . 121
20.21.65 Rotation Up . . . 122
20.21.66 Rotation Uwp . . . 122
20.21.67 Rotation Us . . . 123
20.21.68 Rotation Usp . . . 123
20.21.69 Rotation Ua . . . 124
20.21.70 Rotation Uap . . . 124
20.21.71 Rotations x and xp . . . 125
20.21.72 Rotations y and yp . . . 125
20.21.73 Rotations z and zp . . . 125
20.21.74 Rotations u and d . . . 126
20.21.75 Rotations l and r . . . 126
20.21.76 Rotations f and b . . . 126
20.22 Face(c) and Face(m) rotations . . . 127
20.22.1 Rotations Rc and Rcp . . . 127
20.22.2 Rotations Lc and Lcp . . . 128
20.22.3 Rotations Uc and Ucp . . . 128
20.22.4 Rotations Dc and Dcp . . . 128
20.22.5 Rotations Fc and Fcp . . . 128
20.22.6 Rotations Bc and Bcp . . . 129
20.22.7 Rotations Rm and Rmp . . . 129
20.22.8 Rotations Lm and Lmp . . . 130
20.22.9 Rotations Um and Ump . . . 130
20.22.10 Rotations Dm and Dmp . . . 131
20.22.11 Rotations Fm and Fmp . . . 131
20.22.12 Rotations Bm and Bmp . . . 132
20.23 Randelshofer superset ENG rotations (CMST) . . . 132
20.23.1 Rotations CR and CRp . . . 133
20.23.2 Rotations CL and CLp . . . 133
20.23.3 Rotations CU and CUp . . . 134
20.23.4 Rotations CD and CDp . . . 134
20.23.5 Rotations CF and CFp . . . 134

20.23.6 Rotations CB and CBp . . . . . 135
20.23.7 Rotations MR and MRp . . . . . 135
20.23.8 Rotations ML and MLp . . . . . 135
20.23.9 Rotations MU and MUp . . . . . 136
20.23.10 Rotations MD and MDp . . . . . 137
20.23.11 Rotations MF and MFp . . . . . 137
20.23.12 Rotations MB and MBp . . . . . 138
20.23.13 Rotations T (wide) . . . . . 138
20.23.14 Rotations SR and SRp (opposite slices) . . . . . 141
20.23.15 Rotations SL and SLp . . . . . 141
20.23.16 Rotations SU and SUp . . . . . 142
20.23.17 Rotations SD and SDp . . . . . 142
20.23.18 Rotations SF and SFp . . . . . 143
20.23.19 Rotations SB and SBp . . . . . 144
20.24 Axis rotations (textRubik versions) . . . . . 144

# 1 Introduction

The RUBIKCUBE package (part of the RUBIK 'bundle') provides a collection of LaTeX commands and macros for typesetting Rubik cube (3x3x3) configurations using the PGF/TikZ graphic languages. We have extended the rotation hieroglyph notation, originally developed by Garfath-Cox (1981), and improved by Duvoid (2010, 2011).

The RUBIKCUBE package is the 'base' package of the bundle, and is required by all of the Rubik packages; it deals primarily with typesetting 3x3x3 cube (Rubik cube) configurations. The RUBIKROTATION package processes rotation sequences and keeps track of the cube's configuration during rotations. The RUBIKPATTERNS package is a small database of 3x3x3 (Rubik) cube rotation sequences which generate well-known named cube configurations (patterns). The RUBIKTWOCUBE package allows the typesetting of 2x2x2 cube (Two cube) configurations.

Full 3x3x3 functionality requires the following packages to be loaded (TikZ first; RUBIKCUBE second), as follows:

```
\usepackage{tikz}
\usepackage{rubikcube,rubikrotation,rubikpatterns}
```

Full 2x2x2 functionality requires the RUBIKTWOCUBE package *in addition* to the packages listed above. Note that the `TikZ` package must be loaded *before* the RUBIKCUBE package.

The RUBIKCUBE package has been road-tested on a Microsoft platform (with MiKTeX), a GNU-Linux platform (Debian 8.2.0 and TeXLive 2017), and on a Solaris platform (OpenIndiana).

For the mathematics and group theory associated with the Rubik cube see Chen (2004), Davis (2006), Fung website, Golomb (1981, 1982), Hofstadter (1981), Hutchings (2011), Heise website, Joyner (2008), Kociemba website, Rokicki *et al.* (2013), Scherphius website, Tran (2005). Other useful websites are the Speedsolving website, and those maintained by Duvoid, by Fridrich, by Jelinek, by Reid, and by Vandenburgh. A useful online solver utility (based on an algorithm by Kociemba) is available at the RuWix website. Websites with good pages on patterns and symmetries are those by Fridrich, Kociemba, Longridge, Reid, Randelshofer, Scherphius (see References for details).

For historical and technical details regarding Rubik's cube see Sher (2014), and also the Wikipedia article *Rubik's Cube*.

## 1.1 Requirements

The RUBIKCUBE package requires the TikZ package, since it makes use of the TikZ picture environment and the `\pgfmathsetmacro` command. Consequently, the `TikZ` package must be loaded *before* the RUBIKCUBE package. The RUBIKROTATION package (see below) requires Perl to be installed.

## 1.2 Supporting tool—the rubikrotation package

The RUBIKROTATION package (also part of the RUBIK 'bundle'), is a dynamic extension to the RUBIKCUBE package. It consists of the Perl program `rubikrotation.pl` and the associated style option `rubikrotation.sty`. The RUBIKROTATION package implements rotation sequences on-the-fly using a `\RubikRotation{`⟨*rotation-sequence*⟩`}` command. It returns the new state in a form which can then be used by the RUBIKCUBE package. It also returns some useful strings associated with the rotation sequence which can be used by the RUBIKCUBE package—see also Section 11.

Since the `\RubikRotation` command works by CALLing the `rubikrotation.pl` program, it follows that the RUBIKROTATION package requires (a) Perl to be installed, and (b) the LaTeX engine needs to be run using the `--shell-escape` command-line option. Those wishing to use LuaLaTeX will also need to have access to the `shellesc` package (this can always be downloaded from CTAN directly). See the RUBIKROTATION documentation for further details. See also the examples in the file `RubikExamples.pdf`.

## 1.3 Supporting database—rubikpatterns package

The RUBIKPATTERNS.STY file (also part of the RUBIK 'bundle') is a small database of some well-known 3x3x3 cube (Rubik cube) rotation sequences, stored as named macros. For example, the 'fourspot' and 'sixspot' sequences are encoded in this package as follows:

```
\newcommand{\fourspot}{[fourspot],F2,B2,U,Dp,R2,L2,U,Dp,<(12q*, 8f*)>}
\newcommand{\sixspot}{[sixspot],U,Dp,R,Lp,F,Bp,U,Dp,<(8q*, 8f*)>}
```

These sequences can be processed by name (using the `\RubikRotation` command which also requires Perl to be installed—see Section 1.2), and then displayed (using the `\ShowCube` command in conjunction with various `\DrawRubikCube...` commands). So, for example, one could typeset the so-called 'fourspot' configuration using the following code:

```
\usepackage{tikz,rubikcube,rubikrotation,rubikpatterns}
...
\RubikCubeSolved
\RubikRotation{\fourspot} % this runs the Perl program \texttt{rubikrotation.pl}
\ShowCube{2.4cm}{0.6}{\DrawRubikCubeRU}
```

The sequence itself can be readily typeset using the `\ShowSequence` command (see Section 11). See also the RUBIKROTATION documentation—especially Section 5.1.1 *Sequences as macros*. See also the examples in the file `RubikExamples.pdf`.

## 1.4 Supporting 2x2x2 package—rubiktwocube package

The RUBIKTWOCUBE package carries the macros and commands necessary for processing and displaying 2x2x2 cubes (TwoCubes). The 2x2x2 commands are isomorphic with the 3x3x3 commands—i.e., the word '`Two`' has replaced the word

'Rubik' in commands. Consequently, users of this package will need to be familiar with the RUBIKCUBE package. There are lots of 2x2x2 examples in the file RubikExamples.pdf.

### 1.5 Copyright

Copyright 2014–2018 RWD Nickalls and A Syropoulos.

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3c of this license or any later version. The latest version of this licence is in www.latex-project.org/lppl.txt

## 2 Installation

The Rubik bundle consists of the four packages RUBIKCUBE, RUBIKROTATION, RUBIKPATTERNS and RUBIKTWOCUBE. Although installing the Rubik bundle will typically install everything automatically (eg., from the TEXLive DVD), each package can be installed separately if necessary. Here we detail only the RUBIKCUBE package.

### 2.1 Generating the rubikcube files

Place the file rubikcube.zip into a temporary directory, and unzip it. This will generate the following files:

```
rubikcube.ins
rubikcube.dtx
rubikcube.pdf          --documentation of the rubikcube package
rubik-doc-figA.pdf
rubik-doc-figB.pdf
rubik-doc-figC.pdf
rubik-doc-figD.pdf
rubik-doc-figE.pdf
rubik-doc-figF.pdf
rubikexamples.tex
rubikexamples.pdf
rubikexamples.sh
rubikexamples.bat
```

The style option rubikcube.sty is generated by running (pdf)LATEX on the file rubikcube.ins as follows:

```
pdflatex  rubikcube.ins
```

This documentation file (rubikcube.pdf) can then be generated using the following steps [1]:

[1] Several pdflatex runs are required, since the documentation includes an index as well as hyperef links (the package hypdoc is used). Prior to the first run it is a good idea to delete any relevant .toc, .aux, .out files.

```
pdflatex   rubikcube.dtx
pdflatex   rubikcube.dtx
makeindex -s gind.ist  rubikcube
makeindex -s gglo.ist -o rubikcube.gls  rubikcube.glo
pdflatex   rubikcube.dtx
pdflatex   rubikcube.dtx
```

## 2.2 RubikExamples file

Note that the package includes a 'rubikexamples' file (`rubikexamples.pdf`), as well as the source file (`rubikexamples.tex`), and associated `.sh` (Linux) and `.bat` (Microsoft) batch files, which can be used to facilitate processing the source `.tex` file. The file `rubikexamples.pdf` showcases both 3x3x3 (Rubik cube) and 2x2x2 (Two cube) examples.

Note that should you need to generate the file `rubikexamples.pdf` from the source file (`rubikexamples.tex`) you will require the RUBIKROTATION, RUBIKPATTERNS and RUBIKTWOCUBE packages to be installed, and will also need to use the `--shell-escape` command-line option (see Section 1.2 for details).

## 2.3 Placing the files

Place the files either in the local working directory, or where your system will find them. For a Linux system with a standard TEX Directory Structure (TDS), then:

*.sty → `/usr/local/texlive/texmf-local/tex/latex/rubik/`
*.pdf → `/usr/local/texlive/texmf-local/doc/rubik/`

Finally, (depending on your system) update the TEX file database. For example, on a Linux system one uses the `texhash` command.

## 2.4 Usage

Load the package by using the command `\usepackage{rubikcube}`. Note that the RUBIKCUBE package requires the TikZ package, and so always load TikZ before RUBIKCUBE as follows:

```
\usepackage{tikz}
\usepackage{rubikcube,rubikrotation,rubikpatterns,rubiktwocube}
```

However, the RUBIKCUBE package does check for the presence of TikZ, and will load it if TikZ is not already loaded.

While RUBIKCUBE is a stand-alone package, for full 3x3x3 functionality it is necessary to load the complementary packages RUBIKROTATION, RUBIKPATTERNS. For full 2x2x2 functionality you need to load all four packages.

# 3 Command conventions

All RUBIKCUBE package commands assume a 3x3x3 cube by default. Since all cubes are displayed or 'drawn' using the TikZ picture environment, it is useful (initially at least) to categorise commands with regard to this environment (and also with regard to the `\ShowCube..` command since this is simply a convenient wrapper for the TikZ picture environment). On this basis, we can distingush three conceptually useful categories, as follows:

1. `\Draw..` commands (which must *always* be used *inside* a TikZ picture environment),
2. 'parameter-allocation' commands (which can be used either inside or outside a TikZ environment); for example, `\RubikFace..` (for allocating facelet colours), and
3. commands which can be used in ordinary text; for example, `\rr{}` (for typesetting certain rotation codes).

From a functional point of view, however, we can view the Rubik bundle commands as splitting into the following groups:

1. those that allocate colour to faces, facelets etc., —these commands all start with `\Rubik` (for 3x3x3 cubes) or `\Two` (for 2x2x2 cubes [2]),
2. those that draw —these commands all start with `\Draw`,
3. those that typeset rotation codes or hieroglyphs; —there are just four of these for 3x3x3 cubes (these commands start with `\rr`, `\rrh`, `\Rubik`, and `\textRubik`), and an equivalent four commands for 2x2x2 cubes (these start with `\tr`, `\trh`, `\Two`, and `\textTwo`).

`\rubikcube` This command generates the logo RUBIKCUBE.

## 3.1 The keywords Rubik and Two in commands

In order to try and keep commands intuitive [3] we adopt the convention that the word 'Rubik' in a command reflects the fact that the command relates to a 3x3x3 cube (i.e., a 'Rubik' cube). Similarly, commands which relate to a 2x2x2 cube (a 'Two' cube) —see the RUBIKTWOCUBE package— use instead the word 'Two'. For example, the commands for drawing a 3x3x3 cube and a 2x2x2 cube from a RU viewpoint are respectively `\DrawRubikCubeRU` and `\DrawTwoCubeRU`.

Having packages now for both 3x3x3 and 2x2x2 cubes (v5) means we need to be more careful regarding command names, and try to make commands (a) as

[2] Requires the RUBIKTWOCUBE package

[3] This is a tricky problem given the large number of commands, so any feedback or ideas on how to avoid ambiguity, including pruning or revising 'bad' commands, is always welcome.

intuitive as possible, and (b) use the same command name format for equivalent 3x3x3 and 2x2x2 commands (as shown in the example above).

In keeping with this approach, some commands have had to be renamed. For example, in this new version we have therefore renamed the earlier `\DrawFace..` commands → `\DrawRubikFace..` (see Section 15).

## 3.2 Environments

Although the RUBIKCUBE package has been designed with TikZ in mind, it is important to appreciate that of all the various RUBIKCUBE package commands only the Rubik `\Draw...` commands and TikZ commands actually have to be used inside a TikZ picture environment.

Indeed, using RUBIKCUBE package commands which influence the Rubik colour state (configuration) outside the `tikzpicture`, `minipage` or `figure` environments can make for useful flexibility when a document is generating more than one figure or image. This is because the scope of any colours specified by commands inside these environments is constrained to be 'local' to that particular environment, and hence any change in the Rubik colour state brought about by such commands is not accessible globally (i.e., outside the environment) —see also Section 5 in the documentation of the RUBIKROTATION package.

Consequently users need to be mindful of the environments when drawing sequences of rotations across several figures; for example, keeping commands like `\RubikRotation`, `\RubikFace..`, `\RubikCubeSolved`, outside the environments keeps their effects global (an example of this problem is presented in the file `RubikExamples.pdf`).

## 3.3 Capital letters

Virtually all Rubik bundle commands start with a capital letter, primarily to avoid any confusion with TikZ commands (these generally start with lower-case letters). However, each 'word' in a Rubik bundle command (except the word 'text') also starts with a capital letter, primarily to facilitate readability. For example, `\DrawRubikCubeRU`, `\DrawCubieRU`. However, as with LaTeX, 'text..' commands start with a lower-case 't'; for example `\textCubieRU`. Letter arguments for colours (R, O, Y, G, B, W, X) are always written in upper-case letters.

## 3.4 XYZ argument ordering

Many commands have an appended two, three, or even six ordered arguments or letters which form some feature of the structure of a command; perhaps either face or colour code or a viewpoint direction.

We adopt the convention that where ordering of arguments is critical, then the arguments are ordered in the XYZ, +, − order. An XYZ code implies that the first letter in the code relates to an X-related parameter, for example, L (Left) or R (Right); the second letter relates to a Y-related parameter, for example, U (Up)

or D (Down); the third (if required) relates to a Z-related parameter, for example, F (Front) or B (Back) —see Figure 2.

Some commands have six arguments which adopt an (XYZ;+−) format. In this case, for example, the `\RubikSolvedConfig` command, for which the six colour arguments are ordered as X+, X−, Y+, Y−, Z+, Z−. Here the colour argument associated with a face positioned on the +ve axis is ordered before its −ve complement on the same axis.

Another example is the `\DrawCubieRU{G}{Y}{O}` command, which draws a cubie. Here the RU letters are XY ordered; i.e., RightUp viewpoint. The sequence of colour codes for the three visible faces are XYZ ordered, and hence result in the cube having a Green Right face, Yellow Up face and Orange Front face.

## 3.5 Trailing % on the end of commands

Since the all the output of this package is drawn using graphic elements using TikZ, it is important to include a trailing % on the end of RUBIKCUBE package commands when used *outside* a TikZ picture environment, and also on the end of the `\end{tikzpicture}` environment command itself. In particular it is important to use a trailing % on the end of lines which break before the terminal curly bracket of a `\newcommand`.

This is to prevent accumulating spurious spaces which may otherwise appear in figures and diagrams as a strange or unexpected horizontal shift or white-space. That this can occur is because in TEX every newline character is automatically converted to a white space—unless you have an empty line (Feuersänger 2016).

The LATEX fbox is a useful aid for visualising unwanted white space which may have accumulated, and for identifying the cause. See Section 6 on the `\ShowCubeF` command for more details regarding this approach.

Although this effect is mostly small, and is generally only observed in situations when centering a graphic is critical, it is, however, cumulative and can be surprisingly large. In these situations, the cure is the addition of terminal % characters to preceding code guided by careful detective use of the fbox technique mentioned above.

## 3.6 Cubies, cubicles, faces and facelets

The sub-cubes which make up the Rubik cube are known as 'cubies'; the small coloured face of a cubie is known as a 'facelet'. The cubies are named either according to the colours of their two or three facelets, or according to their physical position.

We distinguish three types of cubie: centre-cubies (single colour), edge-cubies (two colours) and corner-cubies (three colours). For example, the red/white edge-cubie is called the RW cubie, and the red/white/green corner-cubie is called the RWG cubie etc. Note that the colour of a particular face of a 3x3x3 Rubik cube is determined by the colour of its centre-cubie.

Similarly, the positions (known as 'cubicles') occupied by cubies are defined using either a two or three letter face code. For example, the right edge position

in the Up-layer is termed the Up/Right position, or just the UR position, and the corner joining the DOWN FRONT and RIGHT faces is the DFR position.

# 4 Rubik cube coordinates

The coordinate origin of all 2D cube images is located at the bottom-left corner of the FRONT face, as shown in Figure 1. Note also that the bottom left extent of this particular 2D rendering of the 3x3x3 cube is actually at $(-1, -1)$, and hence the default height and width of all oblique-view cubes is 4 units (i.e., equivalent to 4cm if the TikZ scale-factor = 1).

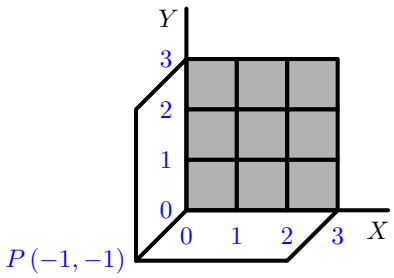



Figure 1: Origin of coordinates is at the bottom left corner of the grey FRONT face. Since $P$ is at $(-1, -1)$ then the default height and width of the 2D cube image is 4 units.

Arranging for $P$ to be at $(-1, -1)$, as well as using the bottom-left corner of the FRONT face as the origin, is a useful design feature which make it easy to figure-out the coordinates of any point on the image plane (either on the cube or outside the cube), and hence facilitates the use of TikZ commands (e.g., `\draw` and `\node` commands) to superimpose lines, arrows and text etc., onto the Rubik cube (see Section 13).

## 4.1 Size of cube `\minipage`

Since the the default height and width of the oblique 2D cube image is 4 units (see Section 4 above), it follows that the width of the `\minipage` required for a cube in a `tikzpicture` environment can be easily calculated. For example, if the `tikzpicture` scale factor used is 0.5, then the minimum width of the required minipage for the `\DrawRubikCubeLD` view (shown above) is therefore $0.5 \times 4\text{cm} = 2\text{cm}$.

Note that the width of the semi-flat (SF) cube representation is therefore 10 units (= 3+3+1+3), and that of the flat (F) cube is 12 units (= 3+3+3+3) — see Section **??** for images of these forms. If in doubt check the horizontal extent of an image using the `\ShowCubeF` command, which places an fbox around the image.

# 5 TikZ picture environment

All the Rubik bundle \Draw.. commands are designed to be used with the TikZ picture environment, and are compatible with standard TikZ. For a basic introduction to the use of TikZ see the following manuals (from CTAN or from http://altermundus.com/).

- https://en.wikipedia.org/wiki/PGF/TikZ
- pgfmanual.pdf, version 3.0.1a (2015) (1161 pages)
- pgfplot.pdf, version 1.14 (2016) (561 pages)
- tkz-base-screen.pdf, version 1.16c (2011) (91 pages)

An example of the TikZ picture environment for use with the Rubik bundle is as follows:

```
\begin{tikzpicture}[scale=0.5]
...
\end{tikzpicture}%
```

If no scale-factor is used (default scale-factor = 1), then each of the small cubie sides will have a length of 1 cm.

USEFUL COMMANDS: Probably the most useful TikZ commands for use with regard to the Rubik bundle are the \draw command (for drawing lines, arrows, circles), and the \node command (for writing text at specific coordinate locations). The basic structure of these commands is as follows, where (x,y) represent grid coordinates of start or end points of lines or arrows, or of a circle centre, or of text position (see Sections 10.4 and 13 for examples).

```
\draw[->,thick,color=blue] (4.5, 2.5) -- (3.5,2.5);
\draw[->,ultra thick,color=red] (4.5, 2.5) -- (3.5,2.5);
\draw [color=blue, thick] (0.3, 0.3)  circle (1.3);
\node (B) at (7.5, 1.5)   [black]{\small\textsf{B}};
```

Remember that all TikZ commands which are valid inside a tikzpicture environment require a terminal semicolon (see Section 13 for examples).

COLOURS: The following colors are predefined by TikZ: red, green, blue, cyan, magenta, yellow, black, gray, darkgray, lightgray, brown, lime, olive, orange, pink, purple, teal, violet and white (see https://en.wikipedia.org/wiki/PGF/TikZ).

LINE WIDTH: TikZ allows line width to be specified directly (e.g., [line width=<dimension>]), or by using the following abbreviations: 'ultra thin' for 0.1pt, 'very thin' for 0.2pt, 'thin' for 0.4pt (the default width), 'semi thick' for 0.6pt, 'thick' for 0.8pt, 'very thick' for 1.2pt, 'ultra thick' for 1.6pt (see https://en.wikipedia.org/wiki/PGF/TikZ).

WHITE SPACE: A particularly useful feature of TikZ is that it automatically minimises any horizontal white-space. However, it is good practice to place a % symbol after the \end{tikzpicture} command to avoid additional white space inadvertently being added by LaTeX (see Section 3.5).

When making images it can be helpful to place them inside a minipage (e.g., using the \ShowCube command / environment below). A convenient approach is to first adjust the value of the tikzpicture scale-factor (to obtain the appropriate size), and then adjust the minipage-width as necessary, using the fbox associated with the \ShowCubeF command (see Section 4.1 for a useful guide on this).

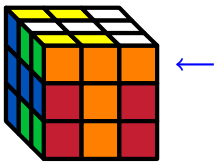
The main 'display' tool for drawing cubes is the \ShowCube command (see below), and this incorporates a TikZ picture environment inside a minipage. The equivalent tool for displaying rotation sequences is the \ShowSequence command.

# 6 \ShowCube command

\ShowCube This command \ShowCube{⟨*width*⟩}{⟨*scale-factor*⟩}{⟨*commands*⟩} is a convenient tool for placing one or more commands inside a tikzpicture environment which is also inside a minipage (see Section 20.4 for the code). This command takes three arguments: the first (#1) is the minipage width, the second (#2) is the tikzpicture scale factor, and the third (#3) is a series of any RUBIKCUBE package \Draw.. and other commands, as well as any TikZ commands which are valid in a tikzpicture environment (e.g., \draw or \node etc.).

USAGE: The following \ShowCube command displays a Rubik cube (the 'SixT's configuration[4]) and a blue arrow in a minipage of width 3cm, using a tikzpicture scale factor of 0.5. Note that the TikZ \draw command requires a terminal semicolon (see Section 5).

```
\RubikCubeSolved
\RubikRotation{\sixts}
\ShowCube{3cm}{0.5}{%
    \DrawRubikCubeLU
    \draw[->,thick,color=blue] (4.5, 2.5) -- (3.5,2.5);
     }
```



The action of the \ShowCube command is illustrated below; the \ShowCube command on the left is equivalent to the bunch of commands on the right (see Section 20.4 for the complete code).

```
\ShowCube{3cm}{0.5}{...}  ⟨
    \begin{minipage}{3cm}%
    \centering%
    \begin{tikzpicture}[scale=0.5]
    ...
    \end{tikzpicture}%
    \end{minipage}%
```

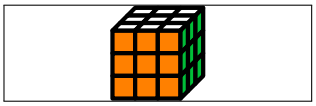
\ShowCubeF The \ShowCubeF command is similar in all respects except that it places an fbox around the minipage in order to enable users to see the extent of any associated white space. For example, unexpected spacing between two adjacent images, or between an image and adjacent text, is usually related to 'hidden' white-space associated with the image itself or excessive width of the associated \minipage

[4] The \sixts macro is from the RUBIKPATTERNS package.

(see also Section 3.5). Consequently, a temporary `fbox` around the minipage can be a useful aid when trying to visualise the full extent of the minipage (and its associated white-space). Use the `\ShowCubeF` command for this.

For example, the following use of the `\ShowCubeF` command reveals a significant white-space problem:

```
\ShowCubeF{4cm}{0.3}{\DrawRubikCubeRU}
```



In this example, clearly either the minipage is too wide (4cm) or the `tikzpicture` scale factor is too small (0.3). Once the figure/code has been corrected, then the `F` in the `\ShowCubeF` command can be removed.

Note that while the `\ShowCube` command centres the image inside the minipage, LaTeX positions the minipage in the `\textwidth`, and hence it is generally best to minimise the horizontal white-space as revealed by the `\ShowCubeF` command. The relationship between the required width of the minipage and the TikZ scale factor for the various Rubik cube images is detailed in Section 4.1.

# 7 Optimum strategy

We suggest that the most convenient (and intuitive) approach for drawing cubes or particular faces is to do it in stages, as follows (all these steps are well illustrated in the examples file `RubikExamples.pdf`):

- first, start by setting the colour state of the cube. This can be done using either (a) a `\RubikCubeSolved..` or `\RubikCubeGrey..` command (for defining the whole cube), or (b) using one or more `\RubikFace..` commands (for defining parts of faces), or (c) by imputting a file containing a previously saved colour state[5].

- second, use the `\RubikRotation` command to process a sequence of rotations (remembering that this requires use of the `--shell-escape` command-line option). The RUBIKPATTERNS package is a small library of named rotation sequences.

- third, draw the image(s) using `\DrawRubikCube..` or `\DrawRubikFace..` commands, plus any TikZ commands (e.g., `\draw` and/or `\node`) in conjunction with the `\ShowCube` command. Use the `\ShowCube` scale factor to adjust the size, and use the `\ShowCubeF` command to reveal the extent of any `minipage` whitespace.

- fourth, spacing between graphic elements can be influenced by adjusting either (a) horizontal whitespace as set by the `\ShowCube` command, or (b) using standard TeX spacing commands, e.g., `\quad`, `\qquad`, `\hspace..` etc.

[5] See the RUBIKROTATION package documentation for details of the `\SaveRubikState` command; see also the 'SaveRubikState' example in the file `RubikExamples.pdf`.

- finally, give some thought to using a trailing % in commands which are broken across multiple lines (see Section 3.5).

With this approach the internal colour state will be updated and processed correctly by all subsequent \Draw.. or \RubikRotation commands. Note that exchanging the word 'Rubik' for the word 'Two' in a command will generate the equivalent TwoCube version of the command (see Section 3.1).

# 8 Colour commands

The Rubik bundle of packages uses seven colours which are defined as follows: R (red), O (orange), Y (yellow), G (green), B (blue), W (white), and X (grey). Now according to the following webpage [6]

http://The-Rubiks-Cube.deviantart.com/journal/Using-Official-Rubik-s-Cube-Colors-268760351 (Nov 2011)

the official Rubik cube colours are defined as

```
... colours which are red (PMS 200C*), green (PMS 347C*),
blue (PMS 293C*), orange (PMS 021C*), yellow (PMS 012C*)
and white.
...
Pantone colors can not be accurately converted to RGB colors,
the colors the web runs on. But they can be approximated.
Through some research, I have found some estimations which
may help you which I have listed below. Remember, these are
just approximate RGB equivalents to the official Rubik's Cube
 colors.

Red: 200C #C41E3A  (www.perbang.dk/rgb/c41e3a/)
Green: 347C #009E60 (www.perbang.dk/rgb/009e60/)
Blue: 293C #0051BA (www.perbang.dk/rgb/0051ba/)
Orange: 021C "Pantone Orange" #FF5800 (www.perbang.dk/rgb/ff5800/)
Yellow: 012C "Pantone Yellow" #FFD500 (www.perbang.dk/rgb/ffd500/)
White: N/A #FFFFFF

Red    {HTML}{C41E3A}
green  {HTML}{009E60}
Blue   {HTML}{0051BA}
Yellow {HTML}{FFD500}
Orange {HTML}{FF5800}
White  {HTML}{FFFFFF}
```

The following RGB specifications are given by Sher (2014):

```
White  {RGB}{255,255,255}
Red    {RGB}{137,18,20}
```

[6] We thank Peter Bartal for bringing this webpage to our attention.

```
Blue   {RGB}{13,72,172}
Orange {RGB}{255,85,37}
Green  {RGB}{25,155,76}
Yellow {RGB}{254,213,47}
```

However, we have tried to optimise these prescribed colours very slightly for screen & print use (for example, the yellow was made very slightly brighter), and so the actual colours implemented by the RUBIKCUBE package are as follows[7] (see Section 20.2):

```
\definecolor{R}{HTML}{C41E33}
\definecolor{G}{HTML}{00BE38}
\definecolor{B}{HTML}{0051BA}
\definecolor{Y}{HTML}{FFFF00}
\colorlet{O}{orange}
\colorlet{W}{white}
\colorlet{X}{black!30}%
```

Different colours can be allocated to the ROYGBWX letters (using the standard LaTeX `\colorlet` command) as required. For example, the standard 'red' colour could be allocated to the letter R using the command

```
\colorlet{R}{red}
```

However, it is important to appreciate that the letter codes ROYGBWX are 'hard-wired' into many of the macros in the RUBIKCUBE package, so don't change these.

## 8.1 Colour state of the cube

A given cubie facelet on a given face is denoted using an ordered sequence of three letters, as follows: first the face code (U,D,L,R,F,B), second the X-position of the column (l,m,r), and third the Y-position of the row (t,m,b). For example, the 'right-bottom' facelet of the FRONT face is denoted as `Frb`, and consequently the curent colour-code (R,O,Y,G,B,W,X) of this facelet is held as the variable `\Frb` etc. (see Section 20.7 for details and code).

Initially, when LaTeX reads the file `rubikcube.sty` all facelets are allocated the colour-code X, which can be regarded as a zero-colour state. Until a facelet is allocated one of the six cube colours (using a suitable command) it will be rendered as grey by a `\Draw...` command, since these commands simply implement the current colour state of the cube (e.g., `\DrawRubikCubeRU`). Facelets retain their colour allocation even if they are moved using the `\RubikRotation` command (see RUBIKROTATION package), unless they are overwritten by a subsequent colour allocation command.

`\RubikFace..`
`\RubikSlice..`
`\RubikCubeSolved..`
`\RubikCubeGrey..`
`\RubikCubeGreyAll`

Colours are allocated to facelets using using `\Rubik..` commands. For example, the commands `\RubikCubeSolvedWY` and `\RubikCubeSolvedWB` allocate prescribed colour states for the whole 'solved' cube, and are a very useful starting

[7] Although the Pantone colours cannot be converted to RGB, there is a subset of of Pantone colours which can be be converted using CMYK (see `https://en.wikipedia.org/wiki/Pantone`).

point (configuration) for subsequent rotations. The commands \RubikCubeGreyWY and \RubikCubeGreyAll allocate different colour states for the whole cube, and are designed to be useful starting points when illustrating aspects of how to solve the cube. These two commands accept both 'grey' and 'gray' (to be consistent with TikZ).

Colours can also be allocated to subsets of facelets (eg faces, slices etc); for example, using the commands \RubikFace... and \RubikSlice... commands (see Sections 8.2 and 8.6).

To visualise the current state of the cube one has to use a \Draw... command. \Draw.. commands never influence the internal colour state of the cube.[8]

The current colour state / configuration of a cube can also be saved and written to a named file, which can then be \input and processed later when required, using the \SaveRubikState command (3x3x3 cube) or \SaveTwoState command (2x2x2 cube).

## 8.2 RubikFace commands

\RubikFaceUp
\RubikFaceDown
\RubikFaceLeft
\RubikFaceRight
\RubikFaceFront
\RubikFaceBack

These commands allocate colours to the individual cubies of a 3x3x3 cube face; they take nine colour arguments (see Section 20.7 for the code). The ordering is isomorphic to the sequence 1–9, i.e., numbering the small squares 1-3 (top row, left to right), 4-6 (middle row, left to right), 7-9 (bottom row, left to right), as follows:

| | | |
|---|---|---|
| #1 | #2 | #3 |
| #4 | #5 | #6 |
| #7 | #8 | #9 |

Conveniently, LaTeX allows the colour arguments to be separated by spaces (e.g., in groups of three), or even spread across several lines (e.g., in a square block to resemble a 9-face). This is fortunate, as it allows the command to be written in several visually intuitive ways, as follows:

```
\RubikFaceUp{G}{B}{G}   {G}{W}{O}   {G}{O}{G}

\RubikFaceFront{O}{W}{R}
               {W}{W}{W}
               {G}{W}{G}
```

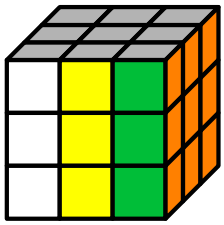
Failure to include a valid colour argument will generate a 'missing parameter' error, and no colour will be allocated (i.e., you will see a black-hole) when it is rendered.

\RubikFaceUpAll
\RubikFaceDownAll
\RubikFaceLeftAll
\RubikFaceRightAll
\RubikFaceFrontAll
\RubikFaceBackAll

Each of the above commands has an associated 'All' version which allocates the same colour to all the cubies on a 9-face (i.e., only a single colour argument is required). For example, if you want the RIGHT face to be all orange, then use

[8] That said, the now deprecated \DrawRubikLayerFace... and \DrawRubikLayerSice... commands (see Section 15) did, confusingly, allow you to specify colours as arguments, but they only 'painted' colours onto facelet positions (on the page, so to speak), and for this reason they are now deprecated, and will be phased out in due course.

the command \RubikFaceRightAll{O}. Use of these commands is shown in the following example.



```
\RubikCubeGreyAll
\RubikFaceRightAll{O}
\RubikFaceFront{W}{Y}{G}
               {W}{Y}{G}
               {W}{Y}{G}
\ShowCube{3cm}{0.7}{\DrawRubikCubeRU}
```

Note that instead of using \RubikCubeGreyAll we could have used the command \RubikFaceUpAll{X} to allocate grey to the whole of the UP face. However, the \RubikCubeGreyAll command can be a useful starting point when dealing with a new cube, since it resets all the faces to their initial default colour.

Finally, it is important to bear in mind that when allocating colours using the \RubikFace.. commands it is very easy to inadvertently create a non-valid cube (ie a cube with either the wrong number of facelets with particular colours, or one which has a non-sovable configuration). However, some basic error checking of this sort is done whenever the \RubikRotation command is used (see the RUBIKROTATION package documentation).

## 8.3 RubikSolvedConfig command

\RubikSolvedConfig

This command allocates the six face colours according to the following ordered XYZ+− argument rule, namely X+, X−, Y+, Y−, Z+, Z−; i.e., the order of the six colour arguments follows the face order RIGHT, LEFT, UP, DOWN, FRONT, BACK (for notation see Section 3.4 and Figure 2).
USAGE: \RubikSolvedConfig{G}{B}{W}{Y}{O}{R}
Examples of its use are shown in the next section.

## 8.4 RubikCubeSolved commands

\RubikCubeSolved
\RubikCubeSolvedWY

The action of both of these commands is identical: they both set all the face colours to the following standard 'solved' cube configuration, namely Up=white, Down=yellow, Right=green, Left=blue, Front=orange, Back=red, by invoking the above \RubikSolvedConfig command, as follows:

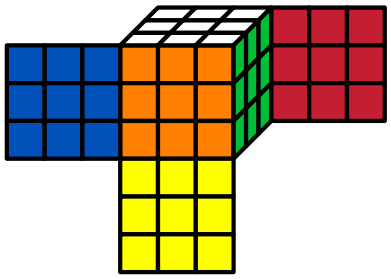
```
\newcommand{\RubikCubeSolved}{\RubikSolvedConfig{G}{B}{W}{Y}{O}{R}}
```

Note that this is in fact just a convenient short-hand for the following:

```
\newcommand{\RubikCubeSolved}{%
  \RubikFaceRightAll{G}%
  \RubikFaceLeftAll{B}%
  \RubikFaceUpAll{W}%
  \RubikFaceDownAll{Y}%
  \RubikFaceFrontAll{O}%
```

```
  \RubikFaceBackAll{R}%
}
```

Note that for convenience, this configuration is also available using the command `\RubikCubeSolvedWY` (WY denoting White opposite Yellow). This solved configuration is shown in the following semi-flat (SF) image.



```
\RubikCubeSolvedWY
\ShowCube{5cm}{0.5}{\DrawRubikCubeSF}
```

Note that the width of the minipage used in the `\ShowCube` command above is set to 5cm. This value is derived from the fact that the *unscaled* width of the semi-flat image is 10cm (9 + 1 squares), and hence if the TikZ scale factor is set to 0.5 (as in the above example) then the minimum minipage width = $10 \times 0.5 = 5$cm (see Section 4 for details).

**Other orientations:** If other orientations of the solved cube are required, this can be easily achieved using the `\RubikRotation` command (from the `RubikRotation` package) to rotate the cube as required. For example, we could make a command to show the above solved cube upside-down and rotated slightly, as follows:

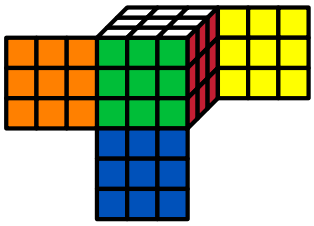
```
\newcommand{\CubeUpSideDown}{\RubikCubeSolved\RubikRotation{x2,y}}
```

—this uses the rotations **x**, **x**, to invert, and then **y** to turn the cube 90 deg:

```
\CubeUpSideDown
\ShowCube{2cm}{0.5}\DrawRubikCubeRU}
```

`\RubikCubeSolvedWB`

**Other configurations:** While the 'solved' WY colour configuration described above (White face opposite Yellow) is that of the most commonly occurring Rubik cube, another 'solved' colour configuration which is also commercially available has the White opposite Blue configuration, which is available using the command `\RubikCubeSolvedWB`. Its colour configuration is as follows:



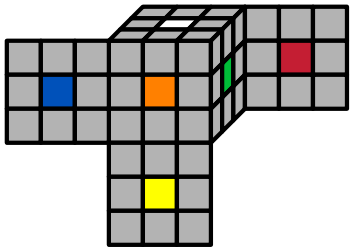
Note that users can easily create their own alternative 'solved' face/colour versions. For example, the above mentioned white opposite blue (WB) solved configuration command `\RubikCubeSolvedWB` (white opposite blue, red opposite orange, and green opposite yellow), was created using `\RubikSolvedConfig{R}{O}{W}{B}{G}{Y}` (for the code see 20.11).

## 8.5 RubikCubeGrey.. commands

`\RubikCubeGreyWY`
`\RubikCubeGreyWB`
`\RubikCubeGreyAll`

The command `\RubikCubeGreyWY` generates a 3x3x3 cube with no colours allocated *except* for the central cubie of each face, which takes the same colour configuration as defined for the `\RubikCubeSolvedWY` command. The command `\RubikCubeGreyAll` generates a cube with *all* the faces completely grey; this is useful as it can be used to reset all the facelets to the initial default state. These commands will accept either 'grey' or 'gray' (to be consistent with TikZ).

These commands, are designed to be useful starting points when wanting to describe the movement of particular cubies. We can see the effect of the `\RubikCubeGreyWY` command by viewing the cube in a semi-flat (SF) format, as follows:



```
\RubikCubeGreyWY
\ShowCube{4.5cm}{0.45}{\DrawRubikCubeSF}
```

Users can of course set up their own alternative face/colour configuration by creating a new 'variant' command altogether.

## 8.6 RubikSlice commands

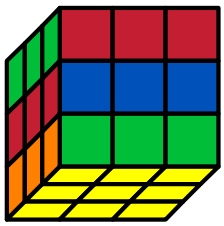
`\RubikSliceTopX`
`\RubikSliceMiddleX`
`\RubikSliceBottomX`

These three commands allocate the six visible cubie colours associated with a *horizontal* slice of a Rubik cube. There are three pairs of Slice commands; one pair (Left view & Right view) for each of the horizontal slices Top, Middle, Bottom. The six colour arguments associated with a given slice run in sequence from left to right irrespective of the viewpoint, e.g., #1 #2 #3 #4 #5 #6.

Since the viewpoint of the Rubik cube (from the Right or from the Left) influences which face the colours are associated with, it is necessary to have the view (R or L) specified in the command name.

The format of the 'slice' command is shown in the following example. The Rubik cube is shown from the LeftDown (LD) view and consequently each of the 'slice' commands in this particular example ends in L, consistent with the final `\DrawRubikCubeLD` command.

Note that the two legacy 'Equator' versions (now replaced by 'Middle') are retained to allow backward compatibility.



```
\RubikFaceDownAll{Y}
\RubikSliceTopL     {G}{G}{G} {R}{R}{R}
\RubikSliceMiddleL  {R}{R}{R} {B}{B}{B}
\RubikSliceBottomL  {O}{O}{O} {G}{G}{G}
\ShowCube{3cm}{0.7}{\DrawRubikCubeLD}
```

# 9 Rotation commands

The Rubik bundle implements not only the standard Rubik cube notation of the World Cube Association (see WCA website), but also the main variant notations used by the Rubik interest groups and websites.

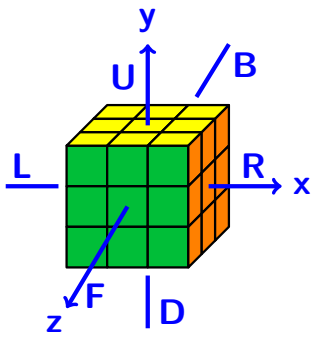



Figure 2: Face rotations

To avoid confusion the Rubik bundle uses a trailing 'p' (lower case) in rotation-codes to denote a 'prime' (reversed direction); we also recommend that commas are used to separate sequential Rubik rotations (moves). While these are mainly to avoid ambiguity, they also greatly facilitate computer searching and copy-and-pasting of rotation sequences.

Unfortunately, obtaining a good balance between an intuitive notation for defining rotations and the need for flexibility is difficult, and consequently some notation is more intuitive than others. A good compromise seems to be the World Cube Association's FADN structure; i.e., Face (L,R,U,D,F,B), Action (m,w,s,a,c), Direction (p), N (n); for example, codes like `R, R2, Rc, Rm, Rwp, Rwp2` etc.

The RUBIKCUBE package includes commands for typesetting a wide range of rotation-codes (e.g., **R**, **y**, **Bw**) and equivalent hieroglyphs (e.g., ||↑, [**y**], **Bw**), as well as commands for typesetting 3x3x3 cubes[9] and single cubies. All the rotation-codes and hieroglyphs are typeset using one particular font & size which we call the 'rubikfont' for convenience (see Section 9.10 for details). All of the

[9] See the RUBIKTWOCUBE package documentation for 2x2x2 cube commands.

rotation-codes described here are recognised by the RUBIKROTATION package (see Section 1.2).

Note that there are some rotation codes which are not represented by arrow hieroglyphs, since their rotation is not visible from the FRONT face, and hence cannot easily be rendered as an arrow hieroglyph. Consequently these rotations have a simple 'letter' hieroglyph in the form of the rotation-code in a square; for example, **Bw**, **Fm**.

## 9.1 Typesetting

We now describe the four commands used for typesetting the various rotation-codes.

`\rr` The text version of a rotation-code is typeset using the rubik-rotation command `\rr{`⟨*rotation-code*⟩`}`, i.e., **R** is typeset using the command `\rr{R}`. The

`\rrh` hieroglyph of a rotation is generated (in text) by using instead the command `\rrh{`⟨*rotation-code*⟩`}`. For example, the command `\rrh{R}` generates ||↑ which is the hieroglyph associated with the rotation **R**.

`\Rubik` A vertically combined rotation-code and its hieroglyph is generated using the command `\Rubik{`⟨*rotation-code*⟩`}`. For example, ||↑ / **R** is generated by the command `\Rubik{R}`, with the square hieroglyph sitting on the baseline. For some hieroglyphs (e.g., [**x**], [**y**], [**z**] denoting 90 degree cube-axis rotations) the only difference between the `\rrh{}` and `\Rubik{}` form is that the `\Rubik{}` form is elevated to sit on the baseline just like the other `\Rubik{}` hieroglyphs. For example, `\rrh{yp}` generates [**y'**], while `\Rubik{yp}` generates [**y'**].

`\textRubik` A horizontally combined rotation-code and its hieroglyph (in sequence as in text) is generated using the command `\textRubik{`⟨*rotation-code*⟩`}`. For example, **R** ||↑ is typeset using the command `\textRubik{R}`. A list of all rotation-code commands and their associated hieroglyphs is given in Section 9.9.

## 9.2 Face rotations

`U` The six main faces of the cube are denoted as FRONT (towards the observer),
`D` BACK, LEFT, RIGHT, UP, DOWN. The upper-case initial letter of each face-name
`L` (**F**, **B**, **L**, **R**, **U**, **D**) denotes a clockwise 90-degree rotation of the face as shown in
`R` Figure 2. For example, **D** is generated by the 'rubik rotation' command `\rr{D}`.
`F` An appended prime ′ indicates an anticlockwise rotation; e.g., **F'**. This is some-
`B` times written as $\mathbf{F}^{-1}$. The 'prime' notation is achieved by appending a lower-case
`Up` 'p' to the face rotation command. For example, **R'** is generated by `\rr{Rp}`. More
`Dp` formally, **R'** is the 'inverse' of **R**.
`Lp` The superscript $^2$, or sometimes just an ordinary 2, indicates that the rotation
`Rp` is applied twice. For example, $\mathbf{R}^2$ or **R2** denote *two* successive 90 degree clockwise
`Fp` rotations of the RIGHT face; $\mathbf{R}^3$ is equivalent to **R'** etc.
`Bp`

## 9.3 Inner-slice rotations

The Rubik cube (3x3x3) has three orthogonal so-called 'inner' slices (middle layers, middle slices), whose +ve rotation direction follows that of a named face. For example, the inner-slice rotation between the RIGHT and LEFT faces whose rotation direction follows the rotation **R** (i.e., its rotation is isomorphic to **R**). The inner-slice rotations form a group (the Slice group), originally described by John Conway (Frey and Singmaster, 1982, p 105).

### The 'm' notation

Um
Dm
Lm
Rm
Fm
Bm

Here 'm' stands for the 'middle' slice, namely that parallel to the designated FACE; its rotation mirrors that of the FACE. The `m` must be in lower case. Each of these rotation-codes has a complementary 'prime' version, formed by appending a 'p'; for example, **Rm** (`\rr{Rm}`) is a middle layer rotation between the RIGHT and LEFT faces, and is in the same direction as **R**. The code **Rm'** (`\rr{Rmp}`) refers to the same middle slice, but rotated in the opposite direction .

This notation, which was probably invented by Singmaster, was originally used on the Cube Lovers usenet group (1981–1997). It is now much used on the Jaap Puzzles website (see Scherphius J) —see also Section 20.22.

### The 'M' notation

MU
MD
ML
MR
MF
MB

This variant of the above 'middle' slice notation (e.g., `MR` ≡ `Rm`) is part of the 'superset' notation of Randelshofer. As before, the rotation direction follows that of the designated FACE. Each has a complementary 'prime' version formed by appending a 'p'. The `M` must be in upper case.

### The MES notation

M
E
S
Mp
Ep
Sp

An alternative but very confusing inner-slice notation (e.g., `Ep` ≡ `Um`) which is occasionally used is the so-called MES notation as used in the Waterman algorithm (Treep and Waterman 1987), and the Roux method (Giles Roux).

- **M** (MIDDLE , between the LEFT and RIGHT faces; direction follows **L**),
- **E** (EQUATOR , between the UP and DOWN faces; direction follows **D**),
- **S** (STANDING , between the FRONT and BACK faces; direction follows **F**).

Each of these also has an inverse (prime) version.

### The 'S' notation

Su Sd Sl Sr Sf Sb

In this equally confusing inner-slice notation, 'S' stands for 'inner-slice'; the face letter must be in lower case (e.g., `Sr` ≡ `Rm`). For example, the inner-slice rotation between the RIGHT and LEFT faces whose rotation direction follows the rotation **R** is denoted as **Sr**, which is typeset using the command `\rr{Sr}`. Each has an inverse (prime) p-form.

## 9.4 Outer-slice rotations

### The 's' (slice) notation

Us Ds Ls Rs Fs Bs

This is a 'paired' form of notation (two rotations at once), which can be thought of as complementing the inner-slice (middle layer) rotations. Each of these 'slice' commands denotes a rotation of two opposite faces in the *same* direction. For example, **Us** ≡ **U** + **D'**; i.e., both face-rotations are in the *same* direction as **U**. Each of these rotation-codes has a complementary 'anti-slice' version (see below).

This notation was originally described by Singmaster (Frey and Singmaster, 1982), and is much used on the 'Pretty patterns' page of the Fridrich website (this page also has a useful link to 'notation').

SU SD SL SR SF SB

This variant of the above 'slice' notation (e.g., `SU` ≡ `Us`) is part of the 'superset' notation of Randelshofer. As before, the rotation direction follows that of the designated FACE. Each has a complementary 'prime' version formed by appending a 'p'.

### The 'a' (anti-slice) notation

Ua Da La Ra Fa Ba

Each of these commands denotes a rotation of two opposite faces in *opposite* directions. For example, **Ua** ≡ **U** + **D**. This notation is much used on the 'Pretty patterns' page of the Fridrich website (see the note above re: 'slice notation').

## 9.5 Wide rotations

### The 'w' notation

Uw Dw Lw Rw Fw Bw

The clockwise *combined* rotation of an outer face AND its adjacent inner-slice (officially known as a 'double block', or 'double outer slice' move) is denoted by appending a lower-case **w** (denoting 'wide') to a rotation-code (endorsed by the WCA). For example, a RIGHT double outer slice rotation (`\rrh{Rw}`) is denoted as **Rw** (`\rr{Rw}`). The 'prime' version is formed by appending a 'p' to the rotation-code. For example, **Rw'** is generated by `\rr{Rwp}`.

### The 'T' notation

TU
TD
TL
TR
TF
TB

This confusing variant of the above 'w' notation (e.g., TR ≡ Rw) is part of the 'superset' notation of Randelshofer. As before, the rotation direction follows that of the designated FACE. Each has a complementary 'prime' version formed by appending a 'p'.

## 9.6 Axis rotations

### The x, y, z notation

x
y
z

Whole-cube clockwise rotations of 90-degrees about about the orthogonal axes centred on the RIGHT, UP, FRONT faces are denoted as **x**, **y**, **z** (the \rr{} forms) respectively (see Figure 2), with their hieroglyphs (the \rrh{} forms) being denoted as [**x**], [**y**], [**z**] in order to distinguish them from square layer-rotation hieroglyphs. Note that since **x**, **y**, **z** rotations are always expressed in lower case; this practice is also extended to the commands.

For example, an **x2** rotation (two **x** rotations one after the other, i.e., [**x**] [**x**]) denotes rotating the cube 180 degrees about its x axis so as to bring the DOWN face into the UP position.

An appended prime ′ indicates an anticlockwise rotation; for example, **x'** (which is generated by appending a 'p' to the rotation-code, i.e., \rr{xp}).

The \Rubik{} forms (and their prime 'p' versions) generate the same hieroglyphs as their \rrh{} versions, except that their spacing is similar to that associated with the 'square box' \Rubik{} hieroglyphs. Consequently when typesetting an axis command in a sequence of 'square-box' \Rubik{} commands, it is better to use the \Rubik{} form rather than the equivalent \rrh{} form (see the examples in Section 9.7). There are no \textRubik{} forms for the axis commands (since they are not necessary).

### The u, d, l, r, f, b notation

u
d
l
r
f
b

These are a commonly used alternative for the **x**, **y**, **z** notation (and also endorsed by the WCA), and denote a 90 degree whole-cube rotation in the same directional sense as that of the associated face rotation.

Thus **d** ≡ **u'** ≡ **y'** etc. For example, **d** and **d'** are generated by the commands \rr{d} and \rr{dp} respectively. Note that **u** is the opposite of **d**, **l** is the opposite of **r**, and **f** is the opposite of **b**, etc.

As with the [**x**], [**y**], [**z**] forms (described above) there are also equivalent \rrh{} and \Rubik{} forms. For example, [**d**] is generated by the command \rrh{d}.

### The 'c' notation

Uc Dc Lc Rc Fc Bc

This slightly more intuitive notation (the 'c' stands for 'cube') also associates the rotation direction with that of the designated FACE (e.g., `Rc` ≡ `x`). Each has a complementary 'prime' version formed by appending a 'p'. For example, **Rc** (`\rr{Rc}`) is equivalent to **x**; **Rc'** (`\rr{Rcp}`) is equivalent to **x'**.

This notation, which was probably invented by Singmaster, was originally used on the Cube Lovers usenet group (1981–1997). It is now much used on the Jaap Puzzles website (see Scherphius J) —see also Section 20.22.

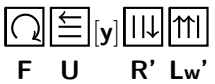
### The 'C' notation

CU CD CL CR CF CB

This variant of the whole cube rotation notation (e.g., `CR` ≡ `Rc` ≡ `x`) is part of the 'superset' notation of Randelshofer. As before, the rotation direction follows that of the designated FACE. Each has a complementary 'prime' version formed by appending a 'p'.

## 9.7 Examples

**R** is generated by the command `\rr{R}`
**Fw** is generated by the command `\rr{Fw}`
**L**$^2$ is generated by `\rr{L}$^2$`
**L**2 is generated by `\rr{L}2`
**R'** is generated by `\rr{Rp}`
**Fw'** is generated by `\rr{Fwp}`
**x** and [**y**] and [**z'**] are generated by `\rr{x}` and `\rrh{y}` and `\Rubik{zp}`
**Fc** and [**Bc**] are generated by `\rr{Fc}` and `\rrh{Bc}`
**UURR** is generated by `\rr{U}\rr{U}\rr{R}\rr{R}`



F U R' Lw' `\Rubik{F}\Rubik{U}\Rubik{y}\Rubik{Rp}\Rubik{Lwp}`

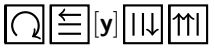
**F** **U**  `\textRubik{F}\ \textRubik{U}`

Commas can be important in avoiding ambiguity; for example,

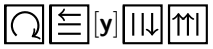
**D**,**U**2,**F**2,**Ds**2,**B**, `\rr{U}2,\rr{F}2,\rr{Ds}2,\rr{B},`

2, 2, 2, `\rrh{U}2,\,\rrh{F}2,\,\rrh{Ds}2`

Finally, if each rotation element uses the *same* font or encoding, for example

 `\rrh{F}\rrh{U}\rrh{y}\rrh{Rp}\rrh{Lwp}`

then typesetting such a rotation sequence can be achieved more easily using the `\ShowSequence` command (see Section 11). For example, we can typeset the last sequence much more conveniently, as follows:

`\ShowSequence{}{\rrh}{F,U,y,Rp,Lwp}` → 

## 9.8 Backwards compatibility

Note that in keeping with 'backwards compatibility' all rotation commands (see below) can still be written without the usual curly braces `{}`. For example, the hieroglyph (`\rrh{D}`) can also be generated using the command `\rrhD`.

## 9.9 Listing of all rotation commands

Note that all the commands presented here also have a `\Rubik{}` equivalent form which typesets both the hieroglyph and its lettercode in a vertical format, as shown in the 'Examples' section above. These have been omitted here owing to the difficulty of including this form easily in the following table.

Note also that some `\rrh{}` commands (e.g., the `\rrh{B}` command) show only the lettercode in a square box, e.g., . This is because these rotations do not have a 'true' visual representation as seen from the FRONT face, and hence can be somewhat ambiguous unless typeset with their associated lettercode.

| | | |
|---|---|---|
| **U** `\rr{U}` | `\rrh{U}` | **U**  `\textRubik{U}` |
| **U'** `\rr{Up}` | `\rrh{Up}` | **U'**  `\textRubik{Up}` |
| **Uw** `\rr{Uw}` | `\rrh{Uw}` | **Uw**  `\textRubik{Uw}` |
| **Uw'** `\rr{Uwp}` | `\rrh{Uwp}` | **Uw'**  `\textRubik{Uwp}` |
| **Us** `\rr{Us}` | `\rrh{Us}` | **Us**  `\textRubik{Us}` |
| **Us'** `\rr{Usp}` | `\rrh{Usp}` | **Us'**  `\textRubik{Usp}` |
| **Ua** `\rr{Ua}` | `\rrh{Ua}` | **Ua**  `\textRubik{Ua}` |
| **Ua'** `\rr{Uap}` | `\rrh{Uap}` | **Ua'**  `\textRubik{Uap}` |
| **Um** `\rr{Um}` | `\rrh{Um}` | **Um**  `\textRubik{Um}` |
| **Um'** `\rr{Ump}` | `\rrh{Ump}` | **Um'**  `\textRubik{Ump}` |
| **Uc** `\rr{Uc}` | [**Uc**] `\rrh{Uc}` | [**Uc**] `\Rubik{Uc}` |
| **Uc'** `\rr{Ucp}` | [**Uc'**] `\rrh{Ucp}` | [**Uc'**] `\Rubik{Ucp}` |
| **D** `\rr{D}` | `\rrh{D}` | **D**  `\textRubik{D}` |
| **D'** `\rr{Dp}` | `\rrh{Dp}` | **D'**  `\textRubik{Dp}` |
| **Dw** `\rr{Dw}` |  `\rrh{Dw}` | **Dw**  `\textRubik{Dw}` |

| | | |
|---|---|---|
| **Dw'** \rr{Dwp} | \rrh{Dwp} | **Dw'** \textRubik{Dwp} |
| **Ds** \rr{Ds} | \rrh{Ds} | **Ds** \textRubik{Ds} |
| **Ds'** \rr{Dsp} | \rrh{Dsp} | **Ds'** \textRubik{Dsp} |
| **Da** \rr{Da} | \rrh{Da} | **Da** \textRubik{Da} |
| **Da'** \rr{Dap} | \rrh{Dap} | **Da'** \textRubik{Dap} |
| **Dm** \rr{Dm} | \rrh{Dm} | **Dm** \textRubik{Dm} |
| **Dm'** \rr{Dmp} | \rrh{Dmp} | **Dm'** \textRubik{Dmp} |
| **Dc** \rr{Dc} | [**Dc**] \rrh{Dc} | [**Dc**] \Rubik{Dc} |
| **Dc'** \rr{Dcp} | [**Dc'**] \rrh{Dcp} | [**Dc'**] \Rubik{Dcp} |
| **L** \rr{L} | \rrh{L} | **L** \textRubik{L} |
| **L'** \rr{Lp} | \rrh{Lp} | **L'** \textRubik{Lp} |
| **Lw** \rr{Lw} | \rrh{Lw} | **Lw** \textRubik{Lw} |
| **Lw'** \rr{Lwp} | \rrh{Lwp} | **Lw'** \textRubik{Lwp} |
| **Ls** \rr{Ls} | \rrh{Ls} | **Ls** \textRubik{Ls} |
| **Ls'** \rr{Lsp} | \rrh{Lsp} | **Ls'** \textRubik{Lsp} |
| **La** \rr{La} | \rrh{La} | **La** \textRubik{La} |
| **La'** \rr{Lap} | \rrh{Lap} | **La'** \textRubik{Lap} |
| **Lm** \rr{Lm} | \rrh{Lm} | **Lm** \textRubik{Lm} |
| **Lm'** \rr{Lmp} | \rrh{Lmp} | **Lm'** \textRubik{Lmp} |
| **Lc** \rr{Lc} | [**Lc**] \rrh{Lc} | [**Lc**] \Rubik{Lc} |
| **Lc'** \rr{Lcp} | [**Lc'**] \rrh{Lcp} | [**Lc'**] \Rubik{Lcp} |
| **R** \rr{R} | \rrh{R} | **R** \textRubik{R} |
| **R'** \rr{Rp} | \rrh{Rp} | **R'** \textRubik{Rp} |
| **Rw** \rr{Rw} | \rrh{Rw} | **Rw** \textRubik{Rw} |

| | | |
|---|---|---|
| **Rw'** \rr{Rwp} | \rrh{Rwp} | **Rw'** \textRubik{Rwp} |
| **Rs** \rr{Rs} | \rrh{Rs} | **Rs** \textRubik{Rs} |
| **Rs'** \rr{Rsp} | \rrh{Rsp} | **Rs'** \textRubik{Rsp} |
| **Ra** \rr{Ra} | \rrh{Ra} | **Ra** \textRubik{Ra} |
| **Ra'** \rr{Rap} | \rrh{Rap} | **Ra'** \textRubik{Rap} |
| **Rm** \rr{Rm} | \rrh{Rm} | **Rm** \textRubik{Rm} |
| **Rm'** \rr{Rmp} | \rrh{Rmp} | **Rm'** \textRubik{Rmp} |
| **Rc** \rr{Rc} | [**Rc**] \rrh{Rc} | [**Rc**] \Rubik{Rc} |
| **Rc'** \rr{Rcp} | [**Rc'**] \rrh{Rcp} | [**Rc'**] \Rubik{Rcp} |
| **F** \rr{F} | \rrh{F} | **F** \textRubik{F} |
| **F'** \rr{Fp} | \rrh{Fp} | **F'** \textRubik{Fp} |
| **Fw** \rr{Fw} | \rrh{Fw} | **Fw** \textRubik{Fw} |
| **Fw'** \rr{Fwp} | \rrh{Fwp} | **Fw'** \textRubik{Fwp} |
| **Fs** \rr{Fs} | **Fs** \rrh{Fs} | **Fs** \textRubik{Fs} |
| **Fs'** \rr{Fsp} | **Fs'** \rrh{Fsp} | **Fs'** \textRubik{Fsp} |
| **Fa** \rr{Fa} | **Fa** \rrh{Fa} | **Fa** \textRubik{Fa} |
| **Fa'** \rr{Fap} | **Fa'** \rrh{Fap} | **Fa'** \textRubik{Fap} |
| **Fm** \rr{Fm} | **Fm** \rrh{Fm} | **Fm** \textRubik{Fm} |
| **Fm'** \rr{Fmp} | **Fm'** \rrh{Fmp} | **Fm'** \textRubik{Fmp} |
| **Fc** \rr{Fc} | [**Fc**] \rrh{Fc} | [**Fc**] \Rubik{Fc} |
| **Fc'** \rr{Fcp} | [**Fc'**] \rrh{Fcp} | [**Fc'**] \Rubik{Fcp} |
| **B** \rr{B} | **B** \rrh{B} | **B** \textRubik{B} |
| **B'** \rr{Bp} | **B'** \rrh{Bp} | **B'** \textRubik{Bp} |
| **Bw** \rr{Bw} | **Bw** \rrh{Bw} | **Bw** \textRubik{Bw} |

| | | |
|---|---|---|
| Bw' \rr{Bwp} | Bw' \rrh{Bwp} | Bw' \textRubik{Bwp} |
| Bs \rr{Bs} | Bs \rrh{Bs} | Bs \textRubik{Bs} |
| Bs' \rr{Bsp} | Bs' \rrh{Bsp} | Bs' \textRubik{Bsp} |
| Ba \rr{Ba} | Ba \rrh{Ba} | Ba \textRubik{Ba} |
| Ba' \rr{Bap} | Ba' \rrh{Bap} | Ba' \textRubik{Bap} |
| Bm \rr{Bm} | Bm \rrh{Bm} | Bm \textRubik{Bm} |
| Bm' \rr{Bmp} | Bm' \rrh{Bmp} | Bm' \textRubik{Bmp} |
| Bc \rr{Bc} | [Bc] \rrh{Bc} | [Bc] \Rubik{Bc} |
| Bc' \rr{Bcp} | [Bc'] \rrh{Bcp} | [Bc'] \Rubik{Bcp} |
| Su \rr{Su} | ⇇ \rrh{Su} | Su ⇇ \textRubik{Su} |
| Su' \rr{Sup} | ⇉ \rrh{Sup} | Su' ⇉ \textRubik{Sup} |
| Sd \rr{Sd} | ⇉ \rrh{Sd} | Sd ⇉ \textRubik{Sd} |
| Sd' \rr{Sdp} | ⇇ \rrh{Sdp} | Sd' ⇇ \textRubik{Sdp} |
| Sl \rr{Sl} | ⇊ \rrh{Sl} | Sl ⇊ \textRubik{Sl} |
| Sl' \rr{Slp} | ⇈ \rrh{Slp} | Sl' ⇈ \textRubik{Slp} |
| Sr \rr{Sr} | ⇈ \rrh{Sr} | Sr ⇈ \textRubik{Sr} |
| Sr' \rr{Srp} | ⇊ \rrh{Srp} | Sr' ⇊ \textRubik{Srp} |
| Sf \rr{Sf} | Sf \rrh{Sf} | Sf \textRubik{Sf} |
| Sf' \rr{Sfp} | Sf' \rrh{Sfp} | Sf' \textRubik{Sfp} |
| Sb \rr{Sb} | Sb \rrh{Sb} | Sb \textRubik{Sb} |
| Sb' \rr{Sbp} | Sb' \rrh{Sbp} | Sb' \textRubik{Sbp} |
| E \rr{E} | ⇉ \rrh{E} | E ⇉ \textRubik{E} |
| E' \rr{Ep} | ⇇ \rrh{Ep} | E' ⇇ \textRubik{Ep} |
| M \rr{M} | ⇊ \rrh{M} | M ⇊ \textRubik{M} |

| | | |
|---|---|---|
| **M'** \rr{Mp} | $\boxed{|\uparrow\uparrow|}$ \rrh{Mp} | **M'** $\boxed{|\uparrow\uparrow|}$ \textRubik{Mp} |
| **S** \rr{S} | $\boxed{\mathsf{S}}$ \rrh{S} | $\boxed{\mathsf{S}}$ \textRubik{S} |
| **S'** \rr{Sp} | $\boxed{\mathsf{S'}}$ \rrh{Sp} | $\boxed{\mathsf{S'}}$ \textRubik{Sp} |
| **x** \rr{x} | [**x**] \rrh{x} | [**x**] \Rubik{x} |
| **x'** \rr{xp} | [**x'**] \rrh{xp} | [**x'**] \Rubik{xp} |
| **y** \rr{y} | [**y**] \rrh{y} | [**y**] \Rubik{y} |
| **y'** \rr{yp} | [**y'**] \rrh{yp} | [**y'**] \Rubik{yp} |
| **z** \rr{z} | [**z**] \rrh{z} | [**z**] \Rubik{z} |
| **z'** \rr{zp} | [**z'**] \rrh{zp} | [**z'**] \Rubik{zp} |
| **u** \rr{u} | [**u**] \rrh{u} | [**u**] \Rubik{u} |
| **u'** \rr{up} | [**u'**] \rrh{up} | [**u'**] \Rubik{up} |
| **d** \rr{d} | [**d**] \rrh{d} | [**d**] \Rubik{d} |
| **d'** \rr{dp} | [**d'**] \rrh{dp} | [**d'**] \Rubik{dp} |
| **l** \rr{l} | [**l**] \rrh{l} | [**l**] \Rubik{l} |
| **l'** \rr{lp} | [**l'**] \rrh{lp} | [**l'**] \Rubik{lp} |
| **r** \rr{r} | [**r**] \rrh{r} | [**r**] \Rubik{r} |
| **r'** \rr{rp} | [**r'**] \rrh{rp} | [**r'**] \Rubik{rp} |
| **f** \rr{f} | [**f**] \rrh{f} | [**f**] \Rubik{f} |
| **f'** \rr{fp} | [**f'**] \rrh{fp} | [**f'**] \Rubik{fp} |
| **b** \rr{b} | [**b**] \rrh{b} | [**b**] \Rubik{b} |
| **b'** \rr{bp} | [**b'**] \rrh{bp} | [**b'**] \Rubik{bp} |

### 9.9.1 Randelshofer notation

| | | |
|---|---|---|
| **CR** \rr{CR} | [**CR**] \rrh{CR} | [**CR**] \Rubik{CR} |
| **CR'** \rr{CRp} | [**CR'**] \rrh{CRp} | [**CR'**] \Rubik{CRp} |

| | | |
|---|---|---|
| **CL** \rr{CL} | [**CL**] \rrh{CL} | [**CL**] \Rubik{CL} |
| **CL'** \rr{CLp} | [**CL'**] \rrh{CLp} | [**CL'**] \Rubik{CLp} |
| **CU** \rr{CU} | [**CU**] \rrh{CU} | [**CU**] \Rubik{CU} |
| **CU'** \rr{CUp} | [**CU'**] \rrh{CUp} | [**CU'**] \Rubik{CUp} |
| **CD** \rr{CD} | [**CD**] \rrh{CD} | [**CD**] \Rubik{CD} |
| **CD'** \rr{CDp} | [**CD'**] \rrh{CDp} | [**CD'**] \Rubik{CDp} |
| **CF** \rr{CF} | [**CF**] \rrh{CF} | [**CF**] \Rubik{CF} |
| **CF'** \rr{CFp} | [**CF'**] \rrh{CFp} | [**CF'**] \Rubik{CFp} |
| **CB** \rr{CB} | [**CB**] \rrh{CB} | [**CB**] \Rubik{CB} |
| **CB'** \rr{CBp} | [**CB'**] \rrh{CBp} | [**CB'**] \Rubik{CBp} |
| **MR** \rr{MR} | ⇕ \rrh{MR} | **MR** ⇕ \textRubik{MR} |
| **MR'** \rr{MRp} | ⇕ \rrh{MRp} | **MR'** ⇕ \textRubik{MRp} |
| **ML** \rr{ML} | ⇕ \rrh{ML} | **ML** ⇕ \textRubik{ML} |
| **ML'** \rr{MLp} | ⇕ \rrh{MLp} | **ML'** ⇕ \textRubik{MLp} |
| **MU** \rr{MU} | ⇐ \rrh{MU} | **MU** ⇐ \textRubik{MU} |
| **MU'** \rr{MUp} | ⇒ \rrh{MUp} | **MU'** ⇒ \textRubik{MUp} |
| **MD** \rr{MD} | ⇒ \rrh{MD} | **MD** ⇒ \textRubik{MD} |
| **MD'** \rr{MDp} | ⇐ \rrh{MDp} | **MD'** ⇐ \textRubik{MDp} |
| **MF** \rr{MF} | **MF** \rrh{MF} | **MF** \textRubik{MF} |
| **MF'** \rr{MFp} | **MF'** \rrh{MFp} | **MF'** \textRubik{MFp} |
| **MB** \rr{MB} | **MB** \rrh{MB} | **MB** \textRubik{MB} |
| **MB'** \rr{MBp} | **MB'** \rrh{MBp} | **MB'** \textRubik{MBp} |
| **SR** \rr{SR} | ⇑ \rrh{SR} | **SR** ⇑ \textRubik{SR} |
| **SR'** \rr{SRp} | ⇓ \rrh{SRp} | **SR'** ⇓ \textRubik{SRp} |
| **SL** \rr{SL} | ⇓ \rrh{SL} | **SL** ⇓ \textRubik{SL} |

| | | |
|---|---|---|
| SL' \rr{SLp} | \rrh{SLp} | SL' \textRubik{SLp} |
| SU \rr{SU} | \rrh{SU} | SU \textRubik{SU} |
| SU' \rr{SUp} | \rrh{SUp} | SU' \textRubik{SUp} |
| SD \rr{SD} | \rrh{SD} | SD \textRubik{SD} |
| SD' \rr{SDp} | \rrh{SDp} | SD' \textRubik{SDp} |
| SF \rr{SF} | \rrh{SF} | \textRubik{SF} |
| SF' \rr{SFp} | \rrh{SFp} | \textRubik{SFp} |
| SB \rr{SB} | \rrh{SB} | \textRubik{SB} |
| SB' \rr{SBp} | \rrh{SBp} | \textRubik{SBp} |
| TR \rr{TR} | \rrh{TR} | TR \textRubik{TR} |
| TR' \rr{TRp} | \rrh{TRp} | TR' \textRubik{TRp} |
| TL \rr{TL} | \rrh{TL} | TL \textRubik{TL} |
| TL' \rr{TLp} | \rrh{TLp} | TL' \textRubik{TLp} |
| TU \rr{TU} | \rrh{TU} | TU \textRubik{TU} |
| TU' \rr{TUp} | \rrh{TUp} | TU' \textRubik{TUp} |
| TD \rr{TD} | \rrh{TD} | TD \textRubik{TD} |
| TD' \rr{TDp} | \rrh{TDp} | TD' \textRubik{TDp} |
| TF \rr{TF} | \rrh{TF} | TF \textRubik{TF} |
| TF' \rr{TFp} | \rrh{TFp} | TF' \textRubik{TFp} |
| TB \rr{TB} | \rrh{TB} | \textRubik{TB} |
| TB' \rr{TBp} | \rrh{TBp} | \textRubik{TBp} |

## 9.10 The rubikfont

For hieroglyph-related text we use the standard Computer Modern Sans (cmss) bold extended (bx) 10pt font for upper-case letters, and the 8pt footnote size for lower-case letters (see Section 20.3 for details). This font (rubikfont) and the 'prime' symbol (rubikprime) can be easily changed by 'renewing' the three

commands there.

For example, to change to the somewhat 'lighter' semi-bold extended (sbx) CM Sans (cmss) form one can simply include the following in the preamble (the FNS suffix stands for 'footnotesize'):

```
\makeatletter
\renewcommand{\@rubikfont}{\fontsize{10}{12pt}\usefont{T1}{cmss}{sbx}{n}}
\renewcommand{\@rubikfontFNS}{\fontsize{8}{12pt}\usefont{T1}{cmss}{sbx}{n}}
\makeatother
```

### The 'rubikprime' symbol

We currently use the apostrophe for the prime symbol (see Section 20.3), since the maths `\prime` seems to be a bit too faint (especially since we need to use the 'scriptstyle' size in this setting). However, users can easily make the Rubik bundle use the maths prime instead, by loading the following in the preamble.

```
\makeatletter
\renewcommand{\@rubikprime}{\raisebox{1.2pt}{\ensuremath{\scriptstyle{^\prime}}}}
\makeatother
```

## 10 Draw commands

A `\Draw..` command typesets either a cubie, cube or face using parameters set or defined via previous parameter-allocation commands (e.g., colours, dimensions etc).

It is important to distinguish between the RUBIKCUBE package `\Draw..` commands (with an upper-case D) and TikZ `\draw..` commands (with a lowercase d). Rubik `\Draw..` commands are implemented by the TikZ `\draw..` commands, and consequently `\Draw..` commands can only be used *inside* a TikZ picture environment—and hence they can also be used safely in conjunction with the `\ShowCube` command, which itself uses a TikZ picture environment. See also Section 10.1 below.

There are six types of `\Draw..` commands, as follows:

```
\DrawCubie..
\DrawRubikCube..
\DrawRubikCubeSidebar..
\DrawRubikFace..
\DrawRubikFlat..
\DrawNCube..
```

Note that the former `\DrawRubikLayer..`, `\DrawCube..`, `\DrawFace..` commands are now deprecated, since they have been superseded by more versatile and intuitive commands (see Section 15).

## 10.1 \Draw error message

If a \Draw.. command is used *outside* a TikZ picture environment, then LaTeX issues an "Undefined control sequence" error message, indicating that it is trying to draw something using an undefined TikZ \draw command [10].

This is because all Rubik \Draw.. commands achieve their effects by implementing a series of TikZ \draw.. and other commands, all of which need to be inside a tikzpicture environment.

For example, if the command \DrawRubikCubeF is used without a surrounding TikZ picture environment, then something similar to the following error message will be generated.

```
! Undefined control sequence.
\DrawRubikFlatUp ... }{#1}\pgfmathsetmacro {\uy }{#2}\draw
                                                  [line join=round,...
l.56 \DrawRubikCubeF
```

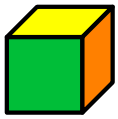
## 10.2 DrawCubie commands

\DrawCubieXY This command draws a single cubie in one of four orientations as denoted by the terminal XY viewing-direction codes. Since a single cubie has only three visible faces this command takes three xyz-ordered colour parameter arguments. Consequently the \DrawCubie command has the format

```
\DrawCubieXY{x}{y}{z}
```

where the XY pair denotes the viewing direction as before, and the xyz parameters denote the face colours associated with each of the three axes.

For example, the command \DrawCubieRU{O}{Y}{G} draws a single cubie as viewed from the RightUp direction, with face colours Orange (x-axis), Yellow (y-axis), Green (z-axis), as follows.



```
\ShowCube{1.33cm}{1}{\DrawCubieRU{O}{Y}{G}}
```

Since the front face is 1 unit wide and the 2D width of the side approx 1/3 unit, and the scale-factor = 1, then the minipage width required for the cubie image = $(1.33 \times 1) = 1.33$cm.

\Cubiedy \Cubiedx Minor cubie configuration changes can be effected by adjusting the Cubiedy and Cubiedx values ($> 0$; no units) shown in Figure 3 via the two commands

```
\Cubiedy{}
\Cubiedx{}
```

as shown in the following example.

[10] Note that the TikZ \draw command uses a lower-case 'd', while all RUBIKCUBE commands start with an upper-case letter.

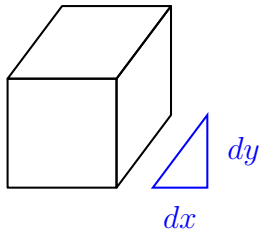
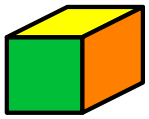



Figure 3: Cubiedy and Cubiedx parameters



```
\ShowCube{1.7cm}{1}{%
  \Cubiedy{0.4}
  \Cubiedx{0.8}
  \DrawCubieRU{O}{Y}{G}
 }
```

Note that the FRONT face of the cubie is a unit square, and the graphic origin of the cubie image is at the bottom left corner of the FRONT face (see also the section on Arrows: Section 13). The default values of `\Cubiedy` and `\Cubiedx` are 0.4.

## 10.3 textCubie commands

`\textCubieRU`
`\textCubieRD`
`\textCubieLU`
`\textCubieLD`

For convenience, there are also four (smaller) 'text' versions of the four `\DrawCubie` commands for use in ordinary text, as follows:

`\textCubieRU{O}{Y}{G}`

`\textCubieRD{O}{Y}{G}`

`\textCubieLU{O}{Y}{G}`

`\textCubieLD{O}{Y}{G}`

Note that these `\textCubieXY` commands are not influenced by the `\Cubiedy`, `\Cubiedx` commands as their size is pre-set for text use.

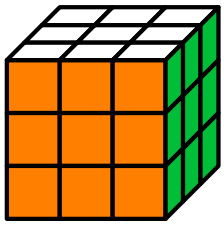
## 10.4 DrawRubikCube commands

`\DrawRubikCubeXY`
`\DrawRubikCubeF`
`\DrawRubikCubeSF`

This command draws Rubik cubes in one of four oblique orientations or configurations as denoted by the following terminal XY viewing-direction codes: RU (RightUp), RD (RightDown), LU (LeftUp), LD (LeftDown); two additional terminal codes are F (Flat) and SF (Semi-Flat). For example, the command

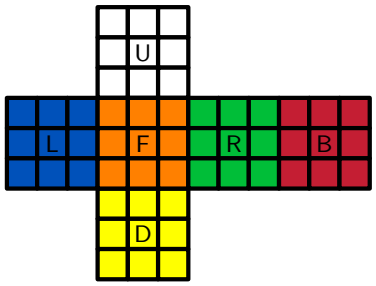
```
\DrawRubikCubeRU
```

will draw a Rubik cube as viewed from the RightUp direction (RU), as shown in the following figure.



```
\RubikCubeSolvedWY
\ShowCube{3cm}{0.7}{\DrawRubikCubeRU}
```

`\DrawRubikCubeF` This command draws the completely flat (F) format of the cube, as shown in the following example.

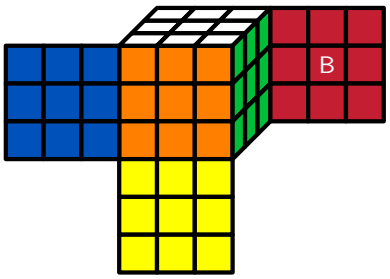



```
\RubikCubeSolvedWY
\ShowCube{5cm}{0.4}{\DrawRubikCubeF}
```

The addition of text (numbers or letters) in the faces is straightforward—the origin of the 1-unit grid is located at the bottom left corner of the FRONT face (orange here). The letters were placed using the following TikZ code inside the TikZ picture environment (remember TikZ commands require a terminal semicolon ;).

```
\RubikCubeSolved
\ShowCube{5cm}{0.4}{%
  \DrawRubikCubeF
  \node (U) at (1.5, 4.5)   [black]{\small\textsf{U}};
  \node (D) at (1.5, -1.5)  [black]{\small\textsf{D}};
  \node (L) at (-1.5, 1.5)  [black]{\small\textsf{L}};
  \node (R) at (4.5, 1.5)   [black]{\small\textsf{R}};
  \node (F) at (1.5, 1.5)   [black]{\small\textsf{F}};
  \node (B) at (7.5, 1.5)   [black]{\small\textsf{B}};
  }
```

`\DrawRubikCubeSF` A useful 'semi-flat' (SF) alternative format, which uses the standard RU view of the cube and appends the three hidden sides (cf., Rokicki *et al.*, 2013), is generated by the command `\DrawRubikCubeSF` as follows.




```
\RubikCubeSolvedWY
\ShowCube{5cm}{0.5}{%
   \DrawRubikCubeSF
   \node (B) at (5.5, 2.5)
            [white]{\small\textsf{B}};
   }
```

Note that even in this configuration it is straight-forward to write text on the graphic, since the 2D width (on the page) of the green RIGHT face is exactly 1-unit, and the bottom right-hand corner of the green face is raised exactly 1-unit (see Figure 1). Consequently, since the origin of the coordinate-grid is at the bottom left corner of the FRONT face (the orange face here), the $(x, y)$ coordinates of the centre of the red BACK face are easily determined to be (5.5, 2.5).

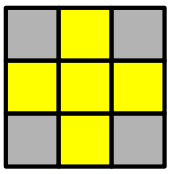
## 10.5 DrawRubikFace.. commands

\DrawRubikFaceUp
\DrawRubikFaceDown
\DrawRubikFaceLeft
\DrawRubikFaceRight
\DrawRubikFaceFront
\DrawRubikFaceBack
\DrawRubikFaceUpSide
\DrawRubikFaceDownSide
\DrawRubikFaceLeftSide
\DrawRubikFaceRightSide
\DrawRubikFaceFrontSide
\DrawRubikFaceBackSide

These commands draw the current state of a specified face (e.g., `\DrawRubikFaceUp`), or the face and all the associated sidebars (e.g., `\DrawRubikFaceUpSide`). These commands do NOT take any arguments—for code see Section 20.15.

NOTE: These commands replace the earlier `\DrawFace...` commands (see Section 15).

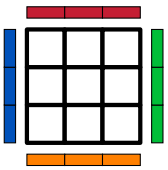
For example, a simple way to show the yellow-cross configuration in the UP face would be to first define the colours using the `\RubikFaceUp` command, and then draw the UP face using the `\DrawRubikFaceUp` command, as follows:



```
\RubikFaceUp{X}{Y}{X}
            {Y}{Y}{Y}
            {X}{Y}{X}
\ShowCube{2.1cm}{0.7}{\DrawRubikFaceUp}
```

## 10.6 Sidebars & DrawRubikFaceXSide commands

In the next example we use the `\DrawRubikFaceUpSide` command to draw the UP face and all its sidebars in a cube having a 'solved' WY (White opposite Yellow) configuration.



```
\RubikCubeSolvedWY
\ShowCube{1.6cm}{0.5}{\DrawRubikFaceUpSide}
```

**Short-hand versions**: For convenience each of these commands has an equivalent short-hand version generated by using just the first letter of the face name and

(where appropriate) the first letter of the word Side. For example,
\DrawRubikFaceR ≡ \DrawRubikFaceRight,
\DrawRubikFaceRS ≡ \DrawRubikFaceRightSide, etc.

## 10.7 Sidebar parameters

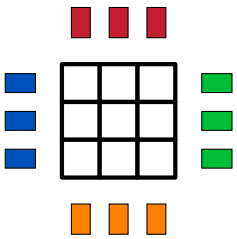
\RubikSidebarWidth
\RubikSidebarLength
\RubikSidebarSep

The default values (size) of the sidebars are as follows: width (0.3), length(1) and separation from the square face (0.3) —see Section 20.16 for the code. Note that the default value of the length of a cubie side is 1. These sidebar values (decimal values ≥ 0; no units) can be changed from their default values using the three commands.

| | |
|---|---|
| \RubikSidebarWidth{} | (default = 0.3) |
| \RubikSidebarLength{} | (default = 1.0) |
| \RubikSidebarSep{} | (default = 0.3) |

Values set in the document preamble will apply globally, while values set within a TikZ picture environment will apply only locally to that particular environment. Alternatively, one can keep the effect local using braces (see below).

In the following example, we show the effect on the UP face and sidebars of a normally solved (WY) cube after dramatically changing the sidebar width, length and separation from the default values—compare with the previous image. For convenience, we have used a pair of braces to keep the effect local to this example.



```
{
\RubikCubeSolvedWY
\RubikSidebarWidth{0.8}
\RubikSidebarLength{0.5}
\RubikSidebarSep{0.7}
\ShowCube{2cm}{0.5}{\DrawRubikFaceUpSide}
}
```

Note also that changing the sidebar-width or sidebar-separation values may well also change the surrounding white-space (use \fbox to visualise this) and may therefore require some fine-tuning of the minipage width setting in order to optimise appearance.

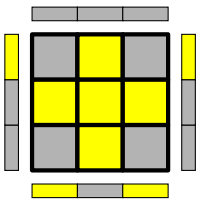
## 10.8 \NoSidebar command

\NoSidebar

The \NoSidebar{⟨*colour-code*⟩} command (which takes a single colour code argument) allows the user to disable the drawing of sidebars having a particular colour (for code see Section 20.16.1). Its action can be localised by placing the command inside an environment (e.g., inside the \ShowCube environment). Alternatively, the action of this command can be disabled simply by writing it with an empty argument, e.g., \NoSidebar{}.

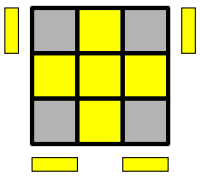
This command is designed to facilitate the drawing of so-called OLL (Orientate Last Layer) configurations, which are typically rendered using the yellow face.

For example, the following figure uses the `\DrawRubikFaceUpSide` command to draw the commonly encountered OLL configuration known as the 'yellow cross' (the remaining four yellow facelets associated with this layer are shown as sidebars). In this example, we first define the colours for the whole cube (grey), and then redefine the colours for the UP face and its four adjacent faces. Finally we draw the UP face and sidebars; we also show an alternative way of writing the facelet colour codes (ie without using the curly brackets).



```
\RubikCubeGreyAll
\RubikFaceUp XYX
             YYY
             XYX
\RubikFaceFront YXY XXXXXX
\RubikFaceRight XXY XXXXXX
\RubikFaceBack  XXX XXXXXX
\RubikFaceLeft  YXX XXXXXX
\ShowCube{2.6cm}{0.6}{\DrawRubikFaceUpSide}
```

However, we can greatly improve the OLL image by disabling the drawing of all the grey (X) sidebars by using the `\NoSidebar{X}` command as follows (here we have placed the `\NoSidebar{X}` command inside the `\ShowCube` environment in order to limit its action locally). Note also that this time we have used the short-hand US (UpSide) version of the `\DrawRubikFaceUpSide` command.



```
\RubikCubeGreyAll
...
...
\ShowCube{2.6cm}{0.6}{\NoSidebar{X}%
                       \DrawRubikFaceUS%
                      }
```

## 10.9 Cube sidebars & DrawRubikCubeSidebar commands

Cube sidebars are drawn adjacent to cube edges which are defined by the two faces forming the edge. Thus the BR (Back-Right) sidebar is placed adjacent to the edge formed by the BACK and RIGHT faces.

Since the cube orientation (view direction; RU, RD, LU, LD) determines which sidebars are visible, the command for drawing the sidebar (`DrawRubikCubeSidebar..`) also needs to incorporate the view direction. The command takes two mandatory arguments: the first is the pair of face codes defining the edge (XX); the second is the view direction, as follows:

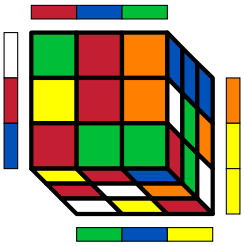
`\DrawRubikCubeSidebarXX{`⟨*view direction*⟩`}`

EXAMPLE: `\DrawRubikCubeSidebarRB{LD}`

Note that the pair of face codes are order *independent*, and hence can be written in any order, which makes remembering the commands very easy.

Note also that at present commands are only available for the eight sidebars which are parallel to X,Y axes, as these seem to be the most useful.

In the following example, we input a previously saved cube state (in the file `cubestate-A.tex`) and draw the cube and the four main sidebars (BR, BD, FL, FU) visible from the RD view direction.



```
\input{cubestate-A.tex}
\ShowCube{3cm}{0.6}{
        \DrawRubikCubeRD
        \DrawRubikCubeSidebarBR{RD}
        \DrawRubikCubeSidebarBD{RD}
        \DrawRubikCubeSidebarFL{RD}
        \DrawRubikCubeSidebarFU{RD}
        }
```

## 10.10 DrawRubikFlat commands

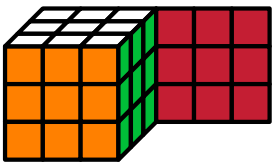
`\DrawRubikFlatUp`
`\DrawRubikFlatDown`
`\DrawRubikFlatLeft`
`\DrawRubikFlatRight`
`\DrawRubikFlatFront`
`\DrawRubikFlatBack`

The `\DrawRubikFlat..{`$\langle x \rangle$`}{`$\langle y \rangle$`}` commands draw a 'flat' square representation of a specified face, located such that its bottom left corner is positioned at $(x, y)$. Each command (except `\DrawRubikFlatFront`) takes two arguments, namely the X-coordinate and Y-coordinate of the bottom left corner of the face. This $(x, y)$ pair allows the user to position the face (see Section 20.14 for the code).

These commands are designed to supplement the `\DrawRubikCube...` commands and allow hidden faces to be represented.

Note also that the `\DrawRubikFlatFront` command currently takes *no* arguments, since by definition the bottom left corner of this face is always at (0,0), and there seems to be no reason (just now) for this face to have the $(x, y)$ facility.

USAGE: The following example uses the command `\DrawRubikFlatBack{4}{1}` to append the BACK face to the side of a 3D cube. Note that since the coordinates of the bottom/back/right corner of the cube rendered by the command `\DrawRubikCubeRU` is (4,1) (see Section 4), we can position the lower/left corner of the BACK face at this point using the command `\DrawRubikFlatBack{4}{1}` as follows:



```
\RubikCubeSolvedWY
\ShowCube{3cm}{0.5}{%
   \DrawRubikCubeRU
   \DrawRubikFlatBack{4}{1}
}
```

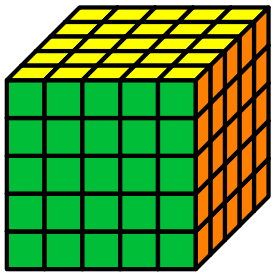
## 10.11 DrawNCube (NxNxN)

`\DrawNCubeAll`

An 'NCube' is a solved NxNxN cube drawn from the RU direction; (i.e., only shows faces UP, FRONT, RIGHT). The cubie colours of each face are All the same.

`\DrawNCubeAll{N}{Xcolour}{Ycolour}{Zcolour}`.

This command takes four ordered parameters (N, X, Y, Z)—the number (integer; $N > 0$) of cubies along an edge, followed by three face colours in XYZ order. Since the viewpoint is only from the RU direction, the three colour parameters are: X(Right), Y(Up), Z(Front).



```
\ShowCube{3.5cm}{0.5}{\DrawNCubeAll{5}{O}{Y}{G}}
```

## 11 \ShowSequence command

`\ShowSequence`
`\ShowSequenceF`
`\ShowSequencef`

The `\ShowSequence{`⟨*separator*⟩`}{`⟨*font-code*⟩`}{`⟨*sequence*⟩`}` command typesets a comma-separated sequence of rotation codes—for code see Section 20.5. Appending an 'F' to the command-name (`\ShowSequenceF`) results in an fbox around the whole output (cf. the `\ShowCubeF` command). Appending a lower-case 'f' to the command-name (`\ShowSequencef`) generates fboxs around *each element* in the output (these two forms can be helpful when checking white space).

The `\ShowSequence` command takes three mandatory arguments: the first is the separator (`#1`), the second is the font or style code (`#2`), and third is a comma-separated sequence of Rubik rotation codes (`#3`).

The separator (e.g., comma) used for typesetting the sequence can be specified (or just omitted—i.e., an empty bracket); the sequence can be either a named sequence (i.e., encoded as a macro) or just a comma separated sequence of rotation codes.

USAGE: The following command displays the rotation sequence F, R, U, Rp, Up, using the `\rr` encoding and comma & space separated.

`\ShowSequence{,\ }{\rr}{F,R,U,Rp,Up}` → **F**, **R**, **U**, **R'**, **U'**

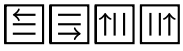
Remember that if you want a very long sequence to break automatically at the line-ends, then you need to include at least one space between the sequence elements, either with or without a comma; for this the separator argument needs to include a space, i.e., something like `{,\ }` or `{\ }` or just `{ }` perhaps.

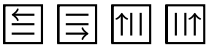
For example, the following commands typeset the comma-separated rotation sequence U,D,Lp,R using different separators and fonts. For convenience, we start by encoding the sequence as the macro `\myseq`.

```
\newcommand{\myseq}{U,D,Lp,R}
```

`\ShowSequence{,}{\texttt}{\myseq}` ⟶ U,D,Lp,R

`\ShowSequence{,}{\rr}{\myseq}` ⟶ **U**,**D**,**L'**,**R**

`\ShowSequence{}{\rrh}{\myseq}` ⟶ 

`\ShowSequence{\ }{\rrh}{\myseq}` ⟶ 

`\ShowSequence{,\ }{\textRubik}{\myseq}` ⟶ **U** , **D** , **L'** , **R** 

## 11.1 Trailing digits

Note that the commands `\rr`, `\rrh`, `\Rubik` and `\textRubik` will fail for rotation codes which have a trailing digit, e.g., R2, Dp3 (so-called 'short' codes), since the `\ShowSequence` macro currently only reads the whole string between pairs of commas (i.e., it does not *interpret* strings like R2 → R,R etc.). However, if you just want to 'see' the text, then the standard LaTeX typewriter command `\texttt` will typeset the text correctly, as you would expect (but using the `tt` font of course).

A useful 'work-around' for this limitation is provided by the `\RubikRotation{}` command [11] (part of the RUBIKROTATION package), since this returns (via the Perl program `rubikrotation.pl`) an extended ('long') form of a given sequence in which any 'short' rotation codes are converted into their separate (atomic) long-form rotation codes; the associated string is `\SequenceLong`. In fact three different forms of the sequence are returned, as shown below.

For example, consider the following short-form sequence L,R,D2,L3, and denote it as 'seqA' as follows: `\newcommand{\seqA}{[seqA],L,R,D2,L3,<test>}`. In this case D2,L3, and the name 'seqA' fail to be typeset when using the `\rr` font with the `\ShowSequence` command (notice there are lots of commas with nothing between them), while `\texttt` does work, as follows:

```
\newcommand{\seqA}{[seqA],L,R,D2,L3,<test>}
```

`\ShowSequence{,}{\rr}{\seqA}` ⟶ ,**L**,**R**,,,

`\ShowSequence{,}{\texttt}{\seqA}` ⟶ `[seqA],L,R,D2,L3,<test>`

However, if we now run the command `\RubikRotation{\seqA}` we shall then have at our disposal the following four strings:

| | | |
|---|---|---|
| `\SequenceInfo` | = | test |
| `\SequenceName` | = | seqA |
| `\SequenceShort` | = | L,R,D2,L3 |
| `\SequenceLong` | = | L,R,D,D,L,L,L |

and therefore have full control over typesetting rotation sequences. This approach therefore offers sufficient flexibility for most purposes. For example, if we now write the command `\ShowSequence{,}{\rr}{\SequenceLong}` we obtain

`\ShowSequence{,}{\rr}{\SequenceLong}` ⟶ **L**,**R**,**D**,**D**,**L**,**L**,**L**

In practice, any trailing digit is converted (modulo 4) and expanded accordingly, the details being copied to the log-file. For example, if one writes the

[11] Using the `--shell-escape` command-line option with the LaTeX engine—see Section 1.2.

command \RubikRotation{[test],R,D28978}, then since 28978 ≡ 2 (modulo 4) then \SequenceLong will be returned containing the string R,D,D. The associated entries in the log-file are as follows:

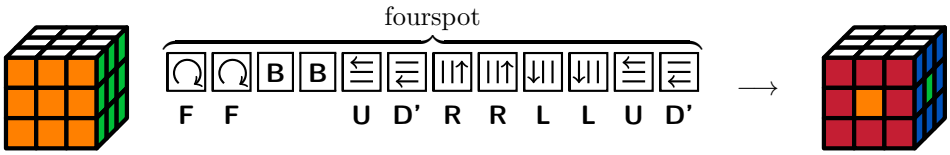
```
...rotation R OK
...Expanding D28978 (28978 = 2 mod 4) ...
...rotation D OK (= Dp3)
...rotation D OK (= Dp3)
...writing new Rubik state to file rubikstateNEW.dat
...SequenceName = test
...SequenceShort = R,D28978
...SequenceLong = R,D,D
```

## 12 SequenceBrace commands

\SequenceBraceA
\SequenceBraceB
\SequenceBraceAF
\SequenceBraceBF

The \SequenceBraceX{⟨*name*⟩}{⟨*sequence*⟩} command is a tool for displaying a named sequence using a brace. The trailing A denotes that the brace is placed Above the sequence; the trailing B denotes the brace is Below the sequence. Appending an 'F' to the command-name (e.g., \SequenceBraceAF) results in a surrounding fbox (cf. the \ShowCubeF command). For the code see Section 20.6. Thus \SequenceBraceB{myseq}{U,D,L,R} generates $\underbrace{\text{U,D,L,R}}_{\text{myseq}}$.

A typical example of its use might be:




which was generated by the following code, showing that the \ShowSequence command can be used as an argument for the \SequenceBraceA command:

```
\newcommand{\fourspot}{[fourspot],F2,B2,U,Dp,R2,L2,U,Dp}%
\noindent%
\RubikCubeSolvedWY%
\ShowCube{1.6cm}{0.4}{\DrawRubikCubeRU}%
\RubikRotation{\fourspot}%
\quad\SequenceBraceA{\SequenceName}{%
                     \ShowSequence{}{\Rubik}{\SequenceLong}}%
\quad$\longrightarrow$\quad%
\ShowCube{1.6cm}{0.4}{\DrawRubikCubeRU}%
```

Note (1) that the 'fourspot' listing is included in the RUBIKPATTERNS package, and (2) the \RubikRotation command requires using the --shell-escape command-line option with the LaTeX engine—see Section 1.2 for details.

# 13 Arrows

The RUBIKCUBE package does not offer any special commands for drawing arrows since it is straightforward just to include the appropriate TikZ 'draw' commands in the `tikzpicture` environment (i.e., in our own `\ShowCube` environment).

In order to facilitate using the standard TikZ 'draw' commands the graphic grid origin of Rubik cube images is located at the bottom left corner of the FRONT face (see also Section 4 for details of the cube's coordinate system). Similarly, single face images (e.g., drawn using the `\DrawRubikFace..` or `\DrawRubikFlat..` commands etc.) and also Sidebars, have their grid origin at the bottom left corner of the face. Consequently the start and finish coordinates for any arrow or line are easy to determine.

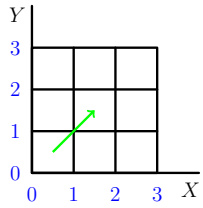
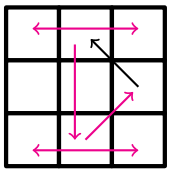



Figure 4: Grid showing the positive coordinates associated with FRONT face of a cube image, or of a face-only image.

For example, Figure 4 shows a green arrow drawn from the centre of the lower-left cubie $(0.5, 0.5)$ to the centre of middle cubie $(1.5, 1.5)$. To do this we just include the following TikZ command (remember that TikZ commands require a terminal semicolon):

```
\draw[->,color=green] (0.5,0.5) -- (1.5, 1.5);
```

The following example shows the cubie movement in the UP face generated by the rotation sequence **FRUR'U'F'**. The magenta arrows indicate movement *with* cubie rotations, while the black arrow indicates movement *without* rotation. This example also highlights the fact that when there are several arrows, the start and end positions often need to be offset slightly away from cubie centres.

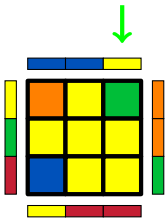


```
\RubikCubeSolvedWY
\ShowCube{2.5cm}{0.7}{%
  \DrawRubikFaceUp
  \draw[->,thick,color=magenta] (1.5,0.5) -- (2.4, 1.4);
  \draw[->,thick] (2.5,1.5) -- (1.6, 2.4);
  \draw[->,thick,color=magenta] (1.3, 2.3) -- (1.3, 0.5);
  \draw[<->,thick,color=magenta] (0.5,2.6) -- (2.5, 2.6);
  \draw[<->,thick,color=magenta] (0.5,0.3) -- (2.5, 0.3);
  }
```

Since the coordinates shown in Figure 4 extend outwards in all directions, they

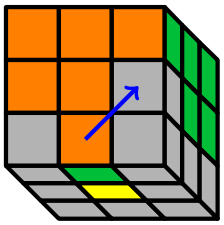
can also be used as a guide for drawing arrows (or other structures) outside this 3x3 'face' square. The origin is at lower left corner of the face.

In the following example, we input a Rubik cube configuration (previously saved as the file `CubeFour.tex` (see the RUBIKROTATION package documentation for details)[12], and draw an arrow to highlight a yellow side facelet.



```
\input{CubeFour.tex}
\ShowCube{2.2cm}{0.5}{\DrawRubikFaceUpSide%
     \draw[->,ultra thick,color=green] (2.5,5) -- (2.5, 4);
     }
```

The following example shows an arrow on the Rubik cube. The origin of coordinates is at the bottom left corner of the FRONT face (see Section 4).



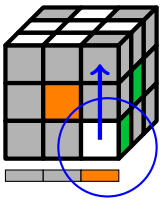
```
\RubikFaceFront{O}{O}{O}
               {O}{O}{X}
               {X}{O}{X}

\RubikFaceRight{G}{G}{G}
               {X}{G}{G}
               {X}{X}{X}

\RubikFaceDown {X}{G}{X}
               {X}{Y}{X}
               {X}{X}{X}
\ShowCube{3cm}{0.7}{%
   \DrawRubikCubeRD
   \draw[ultra thick,->,color=blue]
               (1.5,0.5) -- (2.5, 1.5);
   }
```

In the following example we use a blue circle to highlight a corner cubie to be rotated into the top layer. Note we also use the command `\DrawRubikCubeSidebarFD{RU}` to draw the sidebar along the FD (Front-Down) edge, and show the colour of the hidden facelet of the corner-cubie.

[12] See also the 'SaveRubikState' example in the file `RubikExamples.pdf`.



```
\RubikCubeGreyWY
\RubikFaceUp{X}{W}{X}
            {W}{W}{W}
            {W}{W}{X}
\RubikFaceDown{X}{X}{O}{X}{Y}{X}{X}{X}{X}%
\RubikSliceBottomR{X}{X}{W}{G}{X}{X}
\ShowCube{2.5cm}{0.5}{%
  \DrawRubikCubeRU
  \DrawRubikCubeSidebarFD{RU}
  \draw[ultra thick,->,color=blue]
                              (2.5,0.5) -- (2.5, 2.5);
  \draw [color=blue, thick] (2.7, 0.3)  circle (1.3);
  }%
```

## 14 Final example

We now present, as a final example, the code used to draw the front page figure [13]. This code uses the `\RubikRotation` command (from the RUBIKROTATION package), and therefore the LaTeX engine needs to be run using the `--shell-escape` command-line switch (see Section 1.2 for details).

```
\noindent\hfil%
\RubikCubeSolvedWY%
\ShowCube{1.6cm}{0.4}{\DrawRubikCubeRU}%
\quad\ShowCube{1.6cm}{0.4}{%
   \DrawRubikFaceUpSide%
   \draw[thick,->,color=magenta] (1.5,0.5) -- (2.4, 1.4);
   \draw[thick,->] (2.5,1.5) -- (1.6, 2.4);
   \draw[thick,->,color=magenta] (1.3, 2.3) -- (1.3, 0.5);
   \draw[thick,<->,  color=blue] (0.5,2.6) -- (2.5, 2.6);
   \draw[thick,<->,  color=blue] (0.5,0.3) -- (2.5, 0.3);
   }%
\RubikRotation{F,R,U,Rp,Up,Fp}%
\quad\ShowSequence{}{\Rubik}{\SequenceLong}\quad$\longrightarrow$%
\ShowCube{1.6cm}{0.4}{\DrawRubikFaceUpSide}%
\hfil
```

[13] This is a well-known sequence of order 6 used to cycle three edge cubies; it is used to generate the 'cross' configuration in the final layer when solving the cube. Here we are performing the sequence on a 'solved' cube, since this allows you to see how the three edge cubies move, and either flip (magenta arrows) or do not flip (black arrow).

### 14.1 Without using \RubikRotation

If you really need to draw the above figure *without* using the RUBIKROTATION package (as we had to in order to write this particular document) then you would need to replace the commands

```
\RubikRotation{F,R,U,Rp,Up,Fp}
\ShowSequence{,}{\Rubik}{\SequenceLong} \ \ \ $\longrightarrow$
\ShowCube{2cm}{0.4}{\DrawRubikFaceUpSide}
```

with the following commands:

```
\Rubik{F}\Rubik{R}\Rubik{U}\Rubik{Rp}\Rubik{Up}\Rubik{Fp}
\ \ \ $\longrightarrow$
 \RubikFaceUp   WWB WWOWRB
 \RubikFaceBack RGG XXXXXX
 \RubikFaceLeft RBO XXXXXX
\RubikFaceFront GWO XXXXXX
\RubikFaceRight WWW XXXXXX
\quad\ShowCube{1.6cm}{0.4}{\DrawRubikFaceUpSide}
\hfil
```

## 15 Deprecated commands

The \DrawRubikLayerFace.. and \DrawRubikLayerSide.. are now deprecated; they were found to be confusing since (a) they both drew faces *and* took colour arguments for facelets, and (b) they did not update the internal colour state of a cube or face—i.e., their colouring was simply a local 'painting' action without memory. They have both been superseded by the more versatile \DrawRubikFace.. commands (see Section 10.5).

The \RubikSide.. commands are also deprecated, since they duplicated some of the function of the \RubikFace.. commands.

The earlier \DrawFace.. (v4) and \DrawFlat.. (v3) commands are also deprecated since they lacked the Rubik or Two keyword (see Sections 3.1 and 10.5).

Summary of all deprecated commands (and their current versions) since v3.

| | | |
|---|---|---|
| \DrawRubikCubeFlat | → | \DrawRubikCubeSF (Semi-Flat) |
| \DrawRubikFlat | → | \DrawRubikCubeF (Flat) |
| \DrawFace.. | → | \DrawRubikFace.. |
| \DrawFlat.. | → | \DrawRubikFlat.. |
| \DrawFlat..Side | → | \DrawRubikFace..Side |
| \DrawRubikLayerFace.. | → | \DrawRubikFace.. |
| \DrawRubikLayerSide.. | → | \DrawRubikFace.. |

## 16 Known limitations

Please contact the authors regarding any ideas for improvement, errors, problems or shortcomings etc.

- Note that the rotation hieroglyphs are optimised for a 10pt font and do not scale with document font size. However, they do work well in conjunction with the standard 11pt and 12pt document fonts sizes. Nevertheless, the font size can of course be changed by renewing the font command (see Section 9.10 for details).
- The sidebars cannot be arbitrarily positioned (note: hidden faces can be arbitrarily positioned).

# 17 Change history

- Version 5 (February 2018)

  – Removed some now unnecessary commands (see Section 15) and made some of the internal code a bit more efficient.

  —Added two new 'grey' commands: `\RubikCubeGreyWY` and `\RubikCubeGreyWB` (so as to complement the related `\RubikCubeSolvedXX` commands).

  —Bugfix: added a pair of containing braces around the `\ifthenelse..` command used by the `\ShowSequence` command (for code see Section 20.5). This fixed an occasional problem of a font not being contained.

  —Added missing lowercase cube 'prime' versions of whole cube rotation notation (**u'**, **d'**, **l'**, **r'**, **f'**, **b'**); see Section 9.6.

  —New `\NoSidebar` command for disabling the drawing of sidebars of a particular colour (see Section 10.8).

  —Implemented the terminal $x, y$ position parameters for the `\DrawRubikFlatLeft` and `\DrawRubikFlatRight` commands; all the `\DrawRubikFlat..` commands work correctly now (see Section 20.14)

- Version 4.0 (March 2017)

  —Improved documentation.

  —Improved inter-hieroglyph spacing and vertical position. The Computer Modern sans bold font (10/12pt) is used for the hieroglyphs and rotation codes (see Section 20.3 for details).

  —Improved the `\ShowCube` and `\ShowCubeF` macros (see Sections 6 and 20.4).

  —Additional notation for middle slice rotations ('m' notation), e.g., **Rm**, **Rm'** etc (see Sections 9.3 and 20.22).

  —Additional notation for whole cube rotations ('c' notation), e.g., **Rc**, **Rc'** etc (see Sections 9.6 and 20.22).

  —Added Randelshofer notation (the 'CMST' rotations), e.g., **CR**, **MR** etc (see Sections 9.9.1 and 20.23).

  —Six new commands for showing and annotating rotation sequences; the versions with a terminal 'F' also surround the object with an fbox to allow users to see the extent of any associated white space (see Sections 11 & 12):

```
\ShowSequence
\ShowSequenceF
\ShowSequencef
\SequenceBraceA
\SequenceBraceAF
\SequenceBraceB
\SequenceBraceBF
```

—A new command for setting up or allocating a 'solved' colour configuration. (see Section 8.3):

```
\RubikSolvedConfig
```

—A new command for setting up a 'starter cube' for which the *whole* cube is allocated the default 'grey' colour (see Section 8.5):

```
\RubikCubeGreyAll
```

—A new supporting RUBIKPATTERNS package has been added to the Rubik bundle. It is a small macro database of well-known named Rubik patterns and associated sequences (see Section 1.4).

- Version 3.0 (September 2015)

  —All rotation commands can now use the rotation-code as an argument; for example, the rotation **D** can now be typeset using the command `\rr{D}` etc (see Section 9). The new rotation commands are:

  ```
  \rr{⟨rotation-code⟩}
  \rrh{⟨rotation-code⟩}
  \Rubik{⟨rotation-code⟩}
  \textRubik{⟨rotation-code⟩}
  ```

  The original rotation command formats (e.g., `\rrD`) are still supported for backwards compatibility.

  — `\ShowCube` and `\ShowCubeF` are new commands for displaying a cube inside a minipage (see Sections 6 and 20.4).

  — `\RubikCubeGrey` is a new command for setting up a 'starter cube' for which the only allocated colours are those for the centre cubies (see Section 8.5). The colour configuration matches that of the `\RubikCubeSolved`.

- Version 2.2 (January 2015)

  —Fixed typos and minor errors in the documentation.

  —Added the following commands to facilitate typesetting a face (but see Section 15).

  ```
  \DrawFlatUp
  \DrawFlatDown
  \DrawFlatLeft
  ```

```
\DrawFlatRight
\DrawFlatFront
\DrawFlatBack
\DrawFlatUpSide
\DrawFlatDownSide
\DrawFlatLeftSide
\DrawFlatRightSide
\DrawFlatFrontSide
\DrawFlatBackSide
```

—Changed 'Equator' → 'Middle' in all `\DrawLayer..` commands (for consistency). Hence 'E' → 'M' in all Flat commands and Slice commands. Note that although the former use of 'Equator' is retained for backward compatibility (for the moment) it is now deprecated.

—Fixed a conflict with the TEX `\sb` command as used by the **url** package which resulted in reference chaos when the **url** package was used with the RUBIKCUBE package (internalised `\sb` to `\@sb`). Also internalised, for convenience, `\sd` to `\@sd`; `\sh` to `\@sh`; `\sc` to `\@sc`; `\sq` to `\@sq`.

- Version 2.0 (February 5, 2014)

  —First release.

## 18 Acknowledgements

We thank Peter Bartal and Peter Grill for useful ideas and suggestions; we have built on some of their ideas and have acknowledged these instances in the documentation. We also thank Christian Tellechea for the `\@join{}{}` command (see Section 20.21.3), Christian Schröppel for help regarding the forarray package (see Section 20.5), Herbert Kociemba for helpful comments, and Robert Mařík for suggesting the `\NoSidebar{}` command (see Section 10.8).

## 19 References

- Bartal P (2011). http://tex.stackexchange.com/questions/34482/
- Chen JJ (2004). Group theory and the Rubik's cube. http://www.math.harvard.edu/~jjchen/docs/rubik.pdf
- Davis T (2006). Group theory via Rubik's cube. http://www.geometer.org/rubik/group.pdf
- Demaine ED, Demaine ML, Eisenstat S, Lubiw A and Winslow A (2011). Algorithms for solving Rubik's cubes. http://www.arxiv.org/abs/1106.5736/

- Duvoid T (2010). Méthode simple pour remonter le Rubik's cube. http://duvoid.fr/rubik/rubik-debutant-couleurs.pdf http://duvoid.fr/rubik/sources/notation_en.eps http://duvoid.fr/rubik/sources/rubik-debutant-couleurs.tex
- Duvoid T (2011). Méthode avancée pour remonter le Rubik's cube. http://duvoid.fr/rubik/rubik-friddrich-couleurs.pdf http://duvoid.fr/rubik/sources/rubik-friddrich-couleurs.tex
- Feuersänger (2016). Manual for package PGFPLOTS. v 1.13 (2016/01/06), § 3.2.3, page 21. (pgfplots.pdf) http://www.ctan.org/pkg/pgfplots. [re: preventing extra white space]
- Fridrich website (Fridrich J). http://www.ws.binghamton.edu/fridrich/. See the useful 'notation' section on the 'Pretty patterns' webpage at http://www.ws.binghamton.edu/fridrich/ptrns.html.
- Frey AH and Singmaster D (1982). *Handbook of cubik math*, (Enslow Publishers, Inc.) (republished: 2010, Lutterworth Press, UK)
- Fung website (Fung A). Solving the Rubik's cube systematically. http://alexfung.info/favorite/cube/cube.htm
- Garfath-Cox, A (1981). *The cube*, (Bolden Publishing Co., East Molesey, Surrey) pp.32. [copy in British Library]
- Golomb SW (1981). Rubik's cube and a model of quark confinement. *Am. J. Phys.*; vol 49, pp 1030–1031.
- Golomb SW (1982). Rubik's cube and quarks: twists on the eight corner cells of Rubik's cube provide a model for many aspects of quark behaviour. *American Scientist*; 70, No 3 (May–June 1982), pp. 257–259. http://www.jstor.org/stable/27851433
- Gymrek M (2009). The mathematics of the Rubik's cube. http://web.mit.edu/sp.268/www/rubik.pdf
- Harris D (2008). Speedsolving the cube. (Sterling Publishing Co. Inc., New York, USA.) pp. 166. [covers 2x2x2, 3x3x3, 4x4x4, 5x5x5 cubes]
- Harris website (Harris D). http://www.cubestation.co.uk
- Heise website (Heise R). Rubik's cube theory. http://www.ryanheise.com/cube/theory.html
- Hofstadter D (1981). Rubik cube. *Scientific American*; March issue.
- Hutchings M (2011). The mathematics of Rubik's cube (slide presentation). http://math.berkeley.edu/~hutching/rubik.pdf

- Jelinek website (Jelinek J). Rubik's cube solution methods. http://www.rubikscube.info/
- Joyner D (2008). *Adventures in group theory: Rubik's cube, Merlin's machine and other mathematical toys*; pp 322. http://www.mike.verdone.ca/media/rubiks.pdf
- Kociemba website (Kociemba H). http://www.kociemba.org/cube.htm —for superflip see: http://www.kociemba.org/math/oh.htm
- Kriz I and Siegel P (2008). Rubik's cube-inspired puzzles demonstrate math's simple groups. *Scientific American*; July 2008
- Longridge website (Longridge M). The cube pattern archive. http://www.cubeman.org
- Randelshofer website (Randelshofer W). Pretty patterns. http://www.randelshofer.ch/rubik/patterns/
- Reid website (Reid M). http://www.cflmath.com/Rubik, for patterns see http://www.cflmath.com/Rubik/patterns.html
- Reid M. (1995). Superflip requires 20 face turns. (January 1995) http://www.math.ucf.edu/~reid/Rubik/CubeLovers/
  [cited from Rokicki *et al.*, 2013].
  (Note: easier to use is the following html indexed version of the archive of the Cube-Lovers usenet group (1982–1997) http://www.math.rwth-aachen.de/~Martin.Schoenert/Cube-Lovers/)
- Rokicki T, Kociemba H, Davidson M and Dethridge J (2013). The diameter of the Rubik's cube is twenty. *SIAM. J. Discrete Math.*, 27, 1082–1105. (http://tomas.rokicki.com/rubik20.pdf)
- Roux website (Roux G). http://www.grroux.free.fr
- Rubik's cube. See Section on notation.
  http://en.wikipedia.org/wiki/Rubik's_Cube
- RuWix website (Ferenec D). http://www.ruwix.com. See the online Rubik's cube solver http://www.ruwix.com/online-rubiks-cube-solver-program.
- Scherphius website (Scherphius J). Jaap Puzzles website http://www.jaapsch.net/puzzles/symmetr1.htm
- Sher S. T-H. (2014). The new durable Rubik's cube (technical description). http://www.scf.usc.edu/~tsher/files/Rubiks_Cube.pdf [includes RGB colour specification]
- Singmaster D (1981). *Notes on Rubik's magic cube* (Harmondsworth, Eng., Penguin Books)

- Speedsolving website. http://www.speedsolving.com/
- Storer website (Storer JA). http://www.cs.brandeis.edu/~storer/JimPuzzles/ For Rubik cube, see: http://www.cs.brandeis.edu/~storer/JimPuzzles/RUBIK/Rubik3x3x3.pdf For puzzle book, see: http://www.cs.brandeis.edu/~storer/zzzJimPuzzles/JimPuzzlesBook.pdf
- Tran R (2005). A mathematical approach to solving Rubik's cube. http://www.math.ubc.ca/~cass/courses/m308/projects/rtran/rtran.pdf
- Treep A and Waterman M (1987). Marc Waterman's Algorithm, Part 2. *Cubism For Fun 15*, p. 10 (Nederlandse Kubus Club) [cited from *Wikipedia* (Rubik's cube)]
- Vandenbergh website (Vandenbergh L). CUBEZONE http://www.cubezone.be
- WCA (2016). World Cube Association Regulations. See § 12 for notation. http://www.worldcubeassociation.org/regulations.htm


# 20 The code (rubikcube.sty)

The conventions we adopt regarding capital letters and the XYZ argument ordering are detailed in Section 3.

Note that it is important when using a graphics package to use a trailing % on the end of lines which break before the terminal curly bracket of a `\newcommand`. This is to prevent accumulating spurious spaces which may otherwise appear in figures and diagrams as a strange or unexpected horizontal shift or white-space.

## 20.1 Package heading

```
⟨*rubikcube⟩
\def\RCfileversion{5.0}%
\def\RCfiledate{2018/02/25}% February 25, 2018
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{rubikcube}[\RCfiledate\space (v\RCfileversion)]
```

The package requires TikZ—so we load it if not already loaded.

```
\@ifpackageloaded{tikz}{}{%
  \typeout{---rubikcube requires the TikZ package.}%
  \RequirePackage{tikz}}%
```

The package requires the Forarray package (see Section 20.5)—so we load it if not already loaded.

```
\@ifpackageloaded{forarray}{}{%
  \typeout{---rubikcube requires the Forarray package.}%
  \RequirePackage{forarray}}%
```

The package requires the IfThen package (see Section 20.5)—so we load it if not already loaded.

```
\@ifpackageloaded{ifthen}{}{%
  \typeout{---rubikcube requires the IfThen package.}%
  \RequirePackage{ifthen}}%
```

\rubikcube First we create a suitable logo

```
\newcommand{\rubikcube}{\textsc{rubikcube}}%
\newcommand{\Rubikcube}{\textsc{Rubikcube}}%
```

## 20.2 Colours

We have adopted the following colour allocations—see Section 8 for details.

```
\definecolor{R}{HTML}{C41E33}%
\definecolor{G}{HTML}{00BE38}%
\definecolor{B}{HTML}{0051BA}%
\definecolor{Y}{HTML}{FFFF00}%
\colorlet{X}{black!30}% grey
\colorlet{O}{orange}%
\colorlet{W}{white}%
```

## 20.3 The rubikfont

\@rubikfont \@rubikfontFNS \@rubikprime

We define two fonts for text associated with the Rubik glyphs (both the 'arrow' glyphs and the 'letter' glyphs), namely, (1) Computer Modern Sans (cmss), bold extended (bx), normal shape (n) at 10/12pt, and (2) a footnotesize (FNS) version (8pt) for the lower-case letters [for cmss see Latex Companion (2004), p. 417 & p. 354 [14]]. This has the effect of keeping the size of Rubik glyphs constant in the face of any changes in the document fonts. We make the baseline-skip values the same, since the 'arrow' glyphs generated by the \Rubik commands involve a single baseline-skip (for example, as with \Rubik{D}; see Section 20.21.13). We use the cmss font apostrophe as the 'prime' symbol (the user has the opportunity to use the maths \prime instead—see Section 9.10).

```
\newcommand{\@rubikfont}{\fontsize{10}{12pt}\usefont{T1}{cmss}{bx}{n}}
\newcommand{\@rubikfontFNS}{\fontsize{8}{12pt}\usefont{T1}{cmss}{bx}{n}}
\newcommand{\@rubikprime}{'}
```

## 20.4 ShowCube command

\ShowCube \ShowCubeF

The macro \ShowCube{⟨*minipage width*⟩}{⟨*TikZ scale factor*⟩}{⟨*Draw.. cmd*⟩} displays the cube inside a minipage, so that we can easily tailor the minipage width (#1) and also the TikZ scale factor (#2). The \ShowCubeF command places an fbox around the minipage so users can see the extent of any white space.

USAGE: \ShowCube{2cm}{0.5}{\DrawRubikCubeRU}

[14] Note the typo in Table 7.5 (p. 354): the font-series code for the Sans semi-bold condensed form is 'sbx' (not sbc).

February 2017 (RWDN): We first require a new length variable (which will become the minipage-width), so we can add the length 1.6pt to it (this is the width of the TikZ ultra-thick line which is used to draw the Rubik cubes). In order for a width of an image made up of $x$ units to be equal to $x \times$ (scale-factor) we need to add an extra line-width (i.e., to include the right-hand edge).

```
\newlength{\@showcubewidth}%
```

We can now build the two macros. We set the `\fboxsep` value to zero.

```
\newcommand{\ShowCube}[3]{%
  \setlength{\fboxsep}{0cm}%
  \setlength{\@showcubewidth}{#1}%
  \advance\@showcubewidth by 1.6pt\relax%
  \begin{minipage}{\the\@showcubewidth}%
  \centering%
  \begin{tikzpicture}[scale=#2]%
  #3%
  \end{tikzpicture}%
  \end{minipage}%
}%
\newcommand{\ShowCubeF}[3]{%
  \setlength{\fboxsep}{0cm}%
  \setlength{\fboxrule}{0.4pt}%
  \setlength{\@showcubewidth}{#1}%
  \advance\@showcubewidth by 1.6pt\relax%
  \framebox{%
  \begin{minipage}{\the\@showcubewidth}%
  \centering%
  \begin{tikzpicture}[scale=#2]%
  #3%
  \end{tikzpicture}%
  \end{minipage}%
}}%
```

## 20.5 ShowSequence command

`\ShowSequence`
`\ShowSequenceF`
`\ShowSequencef`

The `\ShowSequence{`⟨*separator*⟩`}{`⟨*font-code*⟩`}{`⟨*sequence*⟩`}` command typesets a comma separated sequence of rotation commands. (See Section 11). This command takes three mandatory arguments: the first is the separator (`#1`), the second is the font or style code (`#2`), and third is a comma-separated sequence of Rubik rotation commands (`#3`).

This command requires the `forarray` package—by Christian Schröppel—(for the `\ForEachX` command) and the `ifthen` package—by David Carlisle—(for the `\ifthenelse` command). These two packages are loaded at startup if not already loaded. We first need to define two variables for use by the command; these are derived from the `forarray` package.

```
\newcommand{\x}{\thislevelitem}
\newcommand{\xcount}{\thislevelcount}
```

EXAMPLE USAGE: `\ForEachX{;}{\texttt{\x}}{L;R;U;D}`

An important feature of the `\ForEachX` command is that it expands its third argument (the list of elements), since this allows the list of elements to be presented as a macro, which is extremely convenient for the user.

The `\ShowSequence` command typesets a sequence of elements (`#3`), and places an optional separator (`#1`) between them. For each element (`\x`, see above) of the sequence `#3` this command forms the construction `#2{element of #3}`. For example, if `#2 = \rr`, and D is an element of `#3`, then it will form the command `\rr{D}` etc.

Note that it is not straightforward to place the separator (`#1`) *only between* the derived elements (i.e., without the separator being either before the first element, or following the last element) using only the `ForEachX` command. This is because the `ForEachX` command processes each element in exactly the same way—i.e., a comma after the first element (good) means there will be a comma after the final element (bad).

We solve this problem by using the `\ifthenelse` command to allow the first element to be processed differently from all the remaining elements. This is because it is easy for TEX to identify the first element of a sequence, but very difficult for it to identify the final element since we generally don't know the number of elements beforehand. Consequently we identify the first element (using the `\xcount`) variable (see above), and then process this first element *without* any comma; we then place a comma in front of each of the remaining elements.

We also create two fbox versions of the command: the 'F' version places an fbox about the whole output; the 'f' version places an fbox about each element in the output (these two versions can be helpful when checking white space).

Note: bugfix 22 October 2017 (RWDN): if the user implemented tt output using `\tt` instead of the standard `\texttt` as the `#2` argument, then the action would not of course remain local, and consequently we have added a leading brace and complementary trailing brace around the the `\ifthenelse...` command in each of the following three macros to limit the action.

USAGE: `\ShowSequence{,}{\rr}{R,L,Up,Dp.....}`

```
\newcommand{\ShowSequence}[3]{%
   \ForEachX{,}{%
    {\ifthenelse{\xcount=1}{#2{\x}}{#1#2{\x}}}%
    }{#3}%
}%
\newcommand{\ShowSequenceF}[3]{%
\fbox{%
   \ForEachX{,}{%
    {\ifthenelse{\xcount=1}{#2{\x}}{#1#2{\x}}}%
    }{#3}%
}}%
\newcommand{\ShowSequencef}[3]{%
   \ForEachX{,}{%
    {\ifthenelse{\xcount=1}{\fbox{#2{\x}}}{#1{\fbox{#2{\x}}}}}%
    }{#3}%
}%
```

\SequenceInfo
\SequenceName
\SequenceShort
\SequenceLong

SEQUENCE HOLDERS: providing none of the Rubik rotation-codes has a trailing integer (e.g., R3) then the Rubik macros (\rr, \rrh, \Rubik, \textRubik) will work as expected when used as the second argument in the \ShowSequence command (described above). However, a problem arises when trying to process in this way any Rubik rotation-codes having a terminal integer (for example, short-codes e.g., R2, D3,...), since the \ShowSequence macro cannot expand short-codes into their long-code elements (e.g., R,R,D,D,D,...).

Accommodating such codes when using the \ShowSequence command is currently solved by using separate 'holders' for four derived strings, namely: \SequenceInfo, \SequenceName,\SequenceShort and \SequenceLong (for details see Section 11). These are generated automatically by the Perl RUBIKROTATION program, which returns a so-called 'long' version of the 'short' string (the argument of the \RubikRotation command). For example, the Perl program converts any short codes (e.g., R2, D3,... ) → long form, e.g., R,R,D,D,D,... (see the RUBIKROTATION documentation for details). In order for the four 'holders' of these derived strings generated by the Perl program (written to the file rubikstateNEW.dat) to be accessible to the user they need to defined here so that they can then be 'redefined' (by the Perl program) in the file rubikstateNEW.dat:

```
\newcommand{\SequenceInfo}{{}}%   %% INFO only
\newcommand{\SequenceName}{{}}%   %% NAME only
\newcommand{\SequenceShort}{{}}%  %% original SHORT seq but with NO NAME
\newcommand{\SequenceLong}{{}}%   %% just the LONG string \& no name
```

## 20.6 SequenceBrace commands

\SequenceBraceA
\SequenceBraceB

The \SequenceBraceX{⟨*name*⟩}{⟨*sequence*⟩} command is a tool for displaying a named sequence using a brace. The trailing A denotes that the brace is placed Above the sequence; B denotes the brace is Below the sequence. For usage see Section 12.

```
\newcommand{\SequenceBraceA}[2]{$\overbrace{\mbox{#2}}^{\mbox{#1}}$}%
\newcommand{\SequenceBraceB}[2]{$\underbrace{\mbox{#2}}_{\mbox{#1}}$}%
\newcommand{\SequenceBraceAF}[2]{\fbox{$\overbrace{\mbox{#2}}^{\mbox{#1}}$}}%
\newcommand{\SequenceBraceBF}[2]{\fbox{$\underbrace{\mbox{#2}}_{\mbox{#1}}$}}%
```

## 20.7 RubikFace commands

Cube face notation: U, D, L, R, F, B (Singmaster)
Cubie-facelet notation: t, m, b, l, m, r = top, middle, bottom, left, middle, right. We use this lower case notation in XY-pairs to denote individual cubie facelets on a given face (to avoid confusion with cube Face notation), as follows:

top row (1,2,3) = lt, mt, rt
middle row (4,5,6) = lm, mm, rm
bottom row (7,8,9) = lb, mb, rb

For example, the current colour of the **r**ight **b**ottom facelet on the FRONT face is held as the variable \Frb etc.

The cubie-facelets (squares) on a face are also parameterized numerically `#1`–`#9` reading from left-to-right, starting top-left & ending bottom-right, when used as arguments for specifying particular colours (as in the `\RubikFace..` commands—see below).

`\RubikFaceUp` `\RubikFaceDown` `\RubikFaceLeft` `\RubikFaceRight` `\RubikFaceFront` `\RubikFaceBack`

These 5 commands allocate a colour to each of the 9 cubie-squares in the specified face (Up, Down, Left, Right, Front, Back). Each command takes 9 arguments (colour codes) in the order 1–9 as specified above.

EXAMPLE: `\RubikFaceUp{R}{O}{Y} {G}{B}{W} {X}{R}{G}`

Each of the 9 `\def{}` commands below allocates one colour to a specific cubie-square (facelet), using a simple three-letter encoding. Each letter is an initial letter of the words Up, Down, Left, Right, Front, Back, left, middle, right, top, middle, bottom.

For example, in the command `\def\Urt{#1}` the U denotes the Up face of the cube, while the `rt` denotes the "right-top" facelet on this face. Note that the order of the two lower-case letters (`rt`) is in the $x, y$ order; i.e., the first of the two lower-case letters relates to the $x$ direction (either left, middle, or right), while the second lower-case letter relates to the $y$ direction (either top, middle, or bottom)—this rule makes it easy to remember the order.

```
\newcommand{\RubikFaceUp}[9]{%
\def\Ult{#1}\def\Umt{#2}\def\Urt{#3}%
\def\Ulm{#4}\def\Umm{#5}\def\Urm{#6}%
\def\Ulb{#7}\def\Umb{#8}\def\Urb{#9}%
}
\newcommand{\RubikFaceFront}[9]{%
\def\Flt{#1}\def\Fmt{#2}\def\Frt{#3}%
\def\Flm{#4}\def\Fmm{#5}\def\Frm{#6}%
\def\Flb{#7}\def\Fmb{#8}\def\Frb{#9}%
}
\newcommand{\RubikFaceRight}[9]{%
\def\Rlt{#1}\def\Rmt{#2}\def\Rrt{#3}%
\def\Rlm{#4}\def\Rmm{#5}\def\Rrm{#6}%
\def\Rlb{#7}\def\Rmb{#8}\def\Rrb{#9}%
}
\newcommand{\RubikFaceDown}[9]{%
\def\Dlt{#1}\def\Dmt{#2}\def\Drt{#3}%
\def\Dlm{#4}\def\Dmm{#5}\def\Drm{#6}%
\def\Dlb{#7}\def\Dmb{#8}\def\Drb{#9}%
}
\newcommand{\RubikFaceLeft}[9]{%
\def\Llt{#1}\def\Lmt{#2}\def\Lrt{#3}%
\def\Llm{#4}\def\Lmm{#5}\def\Lrm{#6}%
\def\Llb{#7}\def\Lmb{#8}\def\Lrb{#9}%
}
\newcommand{\RubikFaceBack}[9]{%
\def\Blt{#1}\def\Bmt{#2}\def\Brt{#3}%
\def\Blm{#4}\def\Bmm{#5}\def\Brm{#6}%
\def\Blb{#7}\def\Bmb{#8}\def\Brb{#9}%
```

```
}
```

\RubikFaceUpAll
\RubikFaceDownAll
\RubikFaceLeftAll
\RubikFaceRightAll
\RubikFaceFrontAll
\RubikFaceBackAll

These 5 commands allocate the same colour to all 9 cubie-squares in the specified face (Up, Down, Left, Right, Front). Each command therefore takes only 1 argument (one of the colour codes).
For example, \RubikFaceUpAll{R}

```
\newcommand{\RubikFaceUpAll}[1]{%
\def\Ult{#1}\def\Umt{#1}\def\Urt{#1}%
\def\Ulm{#1}\def\Umm{#1}\def\Urm{#1}%
\def\Ulb{#1}\def\Umb{#1}\def\Urb{#1}%
}
\newcommand{\RubikFaceFrontAll}[1]{%
\def\Flt{#1}\def\Fmt{#1}\def\Frt{#1}%
\def\Flm{#1}\def\Fmm{#1}\def\Frm{#1}%
\def\Flb{#1}\def\Fmb{#1}\def\Frb{#1}%
}
\newcommand{\RubikFaceRightAll}[1]{%
\def\Rlt{#1}\def\Rmt{#1}\def\Rrt{#1}%
\def\Rlm{#1}\def\Rmm{#1}\def\Rrm{#1}%
\def\Rlb{#1}\def\Rmb{#1}\def\Rrb{#1}%
}
\newcommand{\RubikFaceLeftAll}[1]{%
\def\Llt{#1}\def\Lmt{#1}\def\Lrt{#1}%
\def\Llm{#1}\def\Lmm{#1}\def\Lrm{#1}%
\def\Llb{#1}\def\Lmb{#1}\def\Lrb{#1}%
}
\newcommand{\RubikFaceDownAll}[1]{%
\def\Dlt{#1}\def\Dmt{#1}\def\Drt{#1}%
\def\Dlm{#1}\def\Dmm{#1}\def\Drm{#1}%
\def\Dlb{#1}\def\Dmb{#1}\def\Drb{#1}%
}
\newcommand{\RubikFaceBackAll}[1]{%
\def\Blt{#1}\def\Bmt{#1}\def\Brt{#1}%
\def\Blm{#1}\def\Bmm{#1}\def\Brm{#1}%
\def\Blb{#1}\def\Bmb{#1}\def\Brb{#1}%
}
```

Finally, we now use these commands to initialise all visible faces to the default colour grey (X)

```
\RubikFaceUpAll{X}%
\RubikFaceDownAll{X}%
\RubikFaceLeftAll{X}%
\RubikFaceRightAll{X}%
\RubikFaceFrontAll{X}%
\RubikFaceBackAll{X}%
```

## 20.8 RubikCubeGrey command

\RubikCubeGrey
\RubikCubeGreyWY
\RubikCubeGreyWB

This command sets the face/colour configuration (state) of a 3x3x3 Rubik cube with no colours allocated except for the central cubie of each face. The colour configuration of the central cubies matches those defined for the RubikCubeSolved command (i.e., white opposite yellow etc). We also make WY and WB versions, and implement equivalent 'gray' versions (to be consistent with TikZ).

```
\newcommand{\RubikCubeGrey}{%
  \RubikFaceRight{X}{X}{X}{X}{G}{X}{X}{X}{X}%
  \RubikFaceLeft {X}{X}{X}{X}{B}{X}{X}{X}{X}%
  \RubikFaceUp   {X}{X}{X}{X}{W}{X}{X}{X}{X}%
  \RubikFaceDown {X}{X}{X}{X}{Y}{X}{X}{X}{X}%
  \RubikFaceFront{X}{X}{X}{X}{O}{X}{X}{X}{X}%
  \RubikFaceBack {X}{X}{X}{X}{R}{X}{X}{X}{X}%
}
\newcommand{\RubikCubeGray}{\RubikCubeGrey}
\newcommand{\RubikCubeGreyWY}{\RubikCubeGrey}
\newcommand{\RubikCubeGrayWY}{\RubikCubeGreyWY}
%%
\newcommand{\RubikCubeGreyWB}{%
  \RubikFaceRight{X}{X}{X}{X}{R}{X}{X}{X}{X}%
  \RubikFaceLeft {X}{X}{X}{X}{O}{X}{X}{X}{X}%
  \RubikFaceUp   {X}{X}{X}{X}{W}{X}{X}{X}{X}%
  \RubikFaceDown {X}{X}{X}{X}{B}{X}{X}{X}{X}%
  \RubikFaceFront{X}{X}{X}{X}{G}{X}{X}{X}{X}%
  \RubikFaceBack {X}{X}{X}{X}{Y}{X}{X}{X}{X}%
}
\newcommand{\RubikCubeGrayWB}{\RubikCubeGreyWB}
```

## 20.9 SolvedConfig command

\RubikSolvedConfig

This command sets the face/colour configuration (state) of a typical solved Rubik cube. Note that the order is Right, Left, Up, Down, Front, Back (i.e., X+, X−, Y+, Y−, Z+, Z−, order). We shall use this command to define solved cube configurations.

```
\newcommand{\RubikSolvedConfig}[6]{%
  \RubikFaceRightAll{#1}%
  \RubikFaceLeftAll{#2}%
  \RubikFaceUpAll{#3}%
  \RubikFaceDownAll{#4}%
  \RubikFaceFrontAll{#5}%
  \RubikFaceBackAll{#6}%
}
```

## 20.10 RubikCubeGreyAll command

\RubikCubeGreyAll

This command sets the face/colour configuration (state) of a 3x3x3 Rubik cube with no colours allocated. This colour configuration is the same as the

startup default state—all cubies will appear grey. We implement it using the \RubikSolvedConfig command (above). We also implement an equivalent 'gray' version (to be consistent with TikZ).

```
\newcommand{\RubikCubeGreyAll}{\RubikSolvedConfig{X}{X}{X}{X}{X}{X}}%
\newcommand{\RubikCubeGrayAll}{\RubikCubeGreyAll}
```

## 20.11 RubikCubeSolved command

\RubikCubeSolved
\RubikCubeSolvedWY
\RubikCubeSolvedWB

The first (default) command sets the face/colour configuration (state) one of the standard commercially available solved Rubik cube (white opposite yellow). The argument order follows the XYZ notation. For convenience we make a copy named \RubikCubeSolvedWY (denoting the White opposite Yellow configuration), and also a different version named \RubikCubeSolvedWB (denoting the White opposite Blue configuration). These represent the two standard versions of the Rubik Cube.

```
\newcommand{\RubikCubeSolved}{\RubikSolvedConfig{G}{B}{W}{Y}{O}{R}}%
\newcommand{\RubikCubeSolvedWY}{\RubikCubeSolved}%
\newcommand{\RubikCubeSolvedWB}{\RubikSolvedConfig{R}{O}{W}{B}{G}{Y}}%
```

## 20.12 Slice commands

\RubikSliceTopR
\RubikSliceTopL
\RubikSliceMiddleR
\RubikSliceMiddleL
\RubikSliceBottomR
\RubikSliceBottomL

These 6 commands allocate the colour arguments for the 6 visible ordered facelets along a horizontal slice. There are three horizontal slices to consider (Top, Middle, Bottom) and each has two viewpoints. The colour-code arguments are ordered 1–6 from left to right. The terminal L and R denote the Left (L) viewpoint and Right (R) viewpoint versions. Note that the two legacy 'Equator' versions (now replaced by 'Middle') are retained (below) to allow backward compatibility.

```
\newcommand{\RubikSliceTopR}[6]{%
\def\Flt{#1}\def\Fmt{#2}\def\Frt{#3}%
\def\Rlt{#4}\def\Rmt{#5}\def\Rrt{#6}%
}
\newcommand{\RubikSliceTopL}[6]{%
\def\Llt{#1}\def\Lmt{#2}\def\Lrt{#3}%
\def\Flt{#4}\def\Fmt{#5}\def\Frt{#6}%
}
\newcommand{\RubikSliceMiddleR}[6]{%
\def\Flm{#1}\def\Fmm{#2}\def\Frm{#3}%
\def\Rlm{#4}\def\Rmm{#5}\def\Rrm{#6}%
}
\newcommand{\RubikSliceMiddleL}[6]{%
\def\Llm{#1}\def\Lmm{#2}\def\Lrm{#3}%
\def\Flm{#4}\def\Fmm{#5}\def\Frm{#6}%
}
\newcommand{\RubikSliceEquatorR}[6]{%
\def\Flm{#1}\def\Fmm{#2}\def\Frm{#3}%
\def\Rlm{#4}\def\Rmm{#5}\def\Rrm{#6}%
}
\newcommand{\RubikSliceEquatorL}[6]{%
```

```
\def\Llm{#1}\def\Lmm{#2}\def\Lrm{#3}%
\def\Flm{#4}\def\Fmm{#5}\def\Frm{#6}%
}
\newcommand{\RubikSliceBottomR}[6]{%
\def\Flb{#1}\def\Fmb{#2}\def\Frb{#3}%
\def\Rlb{#4}\def\Rmb{#5}\def\Rrb{#6}%
}
\newcommand{\RubikSliceBottomL}[6]{%
\def\Llb{#1}\def\Lmb{#2}\def\Lrb{#3}%
\def\Flb{#4}\def\Fmb{#5}\def\Frb{#6}%
}
```

## 20.13 Cube drawing macros

Since the three visible sides of a Rubik cube have up to 27 non-grey colours, and TEX has only 9 macro parameters available, we are forced to draw Rubik cubes by first specifying the colours on each of the three faces, and then using a 'DrawRubikCubeXY' command, where the trailing XY code defines the viewing direction (X = either R or L; Y = either U or D). The order of the XY code is important: X first, Y second (so it is easy to remember).

On each face the facelets are drawn in the following order: Top row (left to right), Middle row (left to right), Bottom row (left to right).

The TikZ draw cycle for each facelet square on a Rubik cube face cycles through the four corners of the facelet in the following order: lb → lt → rt → rb; the code being lb (LeftBottom), lt (LeftTop), rt (RightTop), rb (RightBottom) (only need four coords); the (x,y) grid origin is at the bottom-left corner of the FRONT face.

`\DrawRubikCubeFrontFace` This 'FrontFace' command is an 'internal' command which draws and paints all the facelets on the FRONT face of a Rubik cube. It is used by all of the cube drawing macros which display the FRONT face. The 9 colours are allocated by an earlier `\RubikFaceFront` command. These Face macros are based, in part, on those of Peter Bartal (2011).

```
\newcommand{\DrawRubikCubeFrontFace}{%
%  ---top row left to right
\draw[line join=round,line cap=round,ultra thick,fill=\Flt]%
(0,2) -- (0, 3) -- (1,3) -- (1,2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Fmt]%
(1,2) -- (1, 3) -- (2,3) -- (2,2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Frt]%
(2,2) -- (2, 3) -- (3,3) -- (3,2) -- cycle;
% -----middle row left to right
\draw[line join=round,line cap=round,ultra thick,fill=\Flm]%
(0,1) -- (0, 2) -- (1,2) -- (1,1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Fmm]%
(1,1) -- (1, 2) -- (2,2) -- (2,1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Frm]%
(2,1) -- (2, 2) -- (3,2) -- (3,1) -- cycle;
% ----bottom row left to right
```

```
\draw[line join=round,line cap=round,ultra thick,fill=\Flb]%
(0,0) -- (0, 1) -- (1,1) -- (1,0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Fmb]%
(1,0) -- (1, 1) -- (2,1) -- (2,0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Frb]%
(2,0) -- (2, 1) -- (3,1) -- (3,0) -- cycle;
}
```

### 20.13.1 Viewing direction

The command 'DrawRubikCubeXY' command uses the trailing XY code to specify the view direction (X = either R or L; Y = either U or D). The order of the XY code is important: X first, Y second (so it is easy to remember).

`\DrawRubikCubeRU` This command draws and paints a Rubik cube as viewed from the Right Upper (RU) viewpoint. It starts by using the internal command `\DrawRubikCubeFrontFace` to draw the FRONT face, and then draws the UP face followed by the RIGHT face. The colours are allocated to particular facelets using the `\RubikFaceUp`, `\RubikFaceRight` and `\RubikFaceFront` commands.

The (x,y) grid origin is at the bottom-left corner of the FRONT face (see Section 4). The perspective is designed so that the 2D graphic image of the side face (RIGHT in this particular case) has its 'horizontal' lines running at 45 degrees. This has the useful advantage that the 2D width of the side is exactly 1-unit, and so makes it easy to determine the 2D (x,y) coordinates of any position, and hence facilitates typesetting text onto the image of the cube using TikZ commands.

```
\newcommand{\DrawRubikCubeRU}{%
\DrawRubikCubeFrontFace %% frontface
%%-----------Up face----------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Ult]%
(0.66,3.66) -- (1,4) -- (2,4) -- (1.66,3.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umt]%
(1.66,3.66) -- (2,4) -- (3,4) -- (2.66,3.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urt]%
(2.66,3.66) -- (3,4) -- (4,4) -- (3.66,3.66)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulm]%
(0.33,3.33) -- (0.66,3.66) -- (1.66,3.66) -- (1.33,3.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umm]%
(1.33,3.33) -- (1.66,3.66) -- (2.66,3.66) -- (2.33,3.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urm]%
(2.33,3.33) -- (2.66,3.66) -- (3.66,3.66) -- (3.33,3.33)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulb]%
(0,3) -- (0.33,3.33) -- (1.33,3.33) -- (1,3)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umb]%
(1,3) -- (1.33,3.33) -- (2.33,3.33) -- (2,3)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urb]%
```

```
(2,3) -- (2.33,3.33) -- (3.33,3.33) -- (3,3)  -- cycle;
%%-----------Right face----------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlt]%
(3,2) -- (3, 3) -- (3.33,3.33) -- (3.33,2.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmt]%
(3.33,2.33) -- (3.33, 3.33) -- (3.66,3.66) -- (3.66,2.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrt]%
(3.66,2.66) -- (3.66, 3.66) -- (4,4) -- (4,3)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlm]%
(3,1) -- (3, 2) -- (3.33,2.33) -- (3.33,1.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmm]%
(3.33,1.33) -- (3.33, 2.33) -- (3.66,2.66) -- (3.66,1.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrm]%
(3.66,1.66) -- (3.66, 2.66) -- (4,3) -- (4,2)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlb]%
(3,0) -- (3, 1) -- (3.33,1.33) -- (3.33,0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmb]%
(3.33,0.33) -- (3.33, 1.33) -- (3.66,1.66) -- (3.66,0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrb]%
(3.66,0.66) -- (3.66, 1.66) -- (4,2) -- (4,1)  -- cycle;
}
```

`\DrawRubikCube` This command is equivalent to the previous `\DrawRubikCubeRU` and hence is the default form (i.e., if the trailing XY viewpoint code is accidentally omitted).

```
\newcommand{\DrawRubikCube}{\DrawRubikCubeRU}
```

`\DrawRubikCubeRD` This command draws and paints a Rubik cube as viewed from the Right Down (RD) viewpoint.

```
\newcommand{\DrawRubikCubeRD}{%
\DrawRubikCubeFrontFace %% frontface
%%----------Right face--------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlt]%
(3,2) -- (3, 3) -- (3.33,2.66) -- (3.33,1.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmt]%
(3.33,1.66) -- (3.33, 2.66) -- (3.66,2.33) -- (3.66,1.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrt]%
(3.66,1.33) -- (3.66, 2.33) -- (4,2) -- (4,1)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlm]%
(3,1) -- (3, 2) -- (3.33,1.66) -- (3.33,0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmm]%
(3.33,0.66) -- (3.33, 1.66) -- (3.66,1.33) -- (3.66,0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrm]%
(3.66,0.33) -- (3.66, 1.33) -- (4,1) -- (4,0)  -- cycle;
```

```
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlb]%
(3,0) -- (3, 1) -- (3.33,0.66) -- (3.33,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmb]%
(3.33,-0.33) -- (3.33, 0.66) -- (3.66,0.33) -- (3.66,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrb]%
(3.66,-0.66) -- (3.66, 0.33) -- (4,0) -- (4,-1)  -- cycle;
%%----------Down face---------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlt]%
(0.33,-0.33) -- (0, 0) -- (1,0) -- (1.33,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmt]%
(1.33,-0.33) -- (1, 0) -- (2,0) -- (2.33,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drt]%
(2.33,-0.33) -- (2, 0) -- (3,0) -- (3.33,-0.33)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlm]%
(0.66,-0.66) -- (0.33, -0.33) -- (1.33,-0.33) -- (1.66,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmm]%
(1.66,-0.66) -- (1.33, -0.33) -- (2.33,-0.33) -- (2.66,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drm]%
(2.66,-0.66) -- (2.33, -0.33) -- (3.33,-0.33) -- (3.66,-0.66)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlb]%
(1,-1) -- (0.66, -0.66) -- (1.66,-0.66) -- (2,-1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmb]%
(2,-1) -- (1.66, -0.66) -- (2.66,-0.66) -- (3,-1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drb]%
(3,-1) -- (2.66, -0.66) -- (3.66,-0.66) -- (4,-1)  -- cycle;
}
```

\DrawRubikCubeLD This command draws and paints a Rubik cube as viewed from the Left Down (LD) viewpoint.

```
\newcommand{\DrawRubikCubeLD}{%
\DrawRubikCubeFrontFace %% frontface
%%------------Left face--------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Llt]%
(-1,1) -- (-1, 2) -- (-0.66,2.33) -- (-0.66,1.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmt]%
(-0.66,1.33) -- (-0.66, 2.33) -- (-0.33,2.66) -- (-0.33,1.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrt]%
(-0.33,1.66) -- (-0.33, 2.66) -- (0,3) -- (0,2)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Llm]%
(-1,0) -- (-1, 1) -- (-0.66,1.33) -- (-0.66,0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmm]%
(-0.66,0.33) -- (-0.66, 1.33) -- (-0.33,1.66) -- (-0.33,0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrm]%
```

```
(-0.33,0.66) -- (-0.33, 1.66) -- (0,2) -- (0,1)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Llb]%
(-1,-1) -- (-1, 0) -- (-0.66,0.33) -- (-0.66,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmb]%
(-0.66,-0.66) -- (-0.66, 0.33) -- (-0.33,0.66) -- (-0.33,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrb]%
(-0.33,-0.33) -- (-0.33, 0.66) -- (0,1) -- (0,0)  -- cycle;
%%------------Down face----------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlt]%
(-0.33,-0.33) -- (0, 0) -- (1,0) -- (0.66,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmt]%
(0.66,-0.33) -- (1, 0) -- (2,0) -- (1.66,-0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drt]%
(1.66,-0.33) -- (2, 0) -- (3,0) -- (2.66,-0.33)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlm]%
(-0.66,-0.66) -- (-0.33, -0.33) -- (0.66,-0.33) -- (0.33,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmm]%
(0.33,-0.66) -- (0.66, -0.33) -- (1.66,-0.33) -- (1.33,-0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drm]%
(1.33,-0.66) -- (1.66, -0.33) -- (2.66,-0.33) -- (2.33,-0.66)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlb]%
(-1,-1) -- (-0.66, -0.66) -- (0.33,-0.66) -- (0,-1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmb]%
(0,-1) -- (0.33, -0.66) -- (1.33,-0.66) -- (1,-1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drb]%
(1,-1) -- (1.33, -0.66) -- (2.33,-0.66) -- (2,-1)  -- cycle;
}
```

\DrawRubikCubeLU This command draws and paints a Rubik cube as viewed from the Left Up (LU) viewpoint.

```
\newcommand{\DrawRubikCubeLU}{%
\DrawRubikCubeFrontFace %% frontface
%%-----------Left face-----------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Llt]%
(-1,3) -- (-1, 4) -- (-0.66,3.66) -- (-0.66,2.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmt]%
(-0.66,2.66) -- (-0.66, 3.66) -- (-0.33,3.33) -- (-0.33,2.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrt]%
(-0.33,2.33) -- (-0.33, 3.33) -- (0,3) -- (0,2)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Llm]%
(-1,2) -- (-1, 3) -- (-0.66,2.66) -- (-0.66,1.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmm]%
(-0.66,1.66) -- (-0.66, 2.66) -- (-0.33,2.33) -- (-0.33,1.33)  -- cycle;
```

```
\draw[line join=round,line cap=round,ultra thick,fill=\Lrm]%
(-0.33,1.33) -- (-0.33, 2.33) -- (0,2) -- (0,1)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Llb]%
(-1,1) -- (-1, 2) -- (-0.66,1.66) -- (-0.66,0.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmb]%
(-0.66,0.66) -- (-0.66, 1.66) -- (-0.33,1.33) -- (-0.33,0.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrb]%
(-0.33,0.33) -- (-0.33, 1.33) -- (0,1) -- (0,0)  -- cycle;
%%----------Up face---------
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Ult]%
(-0.66,3.66) -- (-1, 4) -- (0,4) -- (0.33,3.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umt]%
(0.33,3.66) -- (0, 4) -- (1,4) -- (1.33,3.66)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urt]%
(1.33,3.66) -- (1, 4) -- (2,4) -- (2.33,3.66)  -- cycle;
%%---middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulm]%
(-0.33,3.33) -- (-0.66, 3.66) -- (0.33,3.66) -- (0.66,3.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umm]%
(0.66,3.33) -- (0.33, 3.66) -- (1.33,3.66) -- (1.66,3.33)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urm]%
(1.66,3.33) -- (1.33, 3.66) -- (2.33,3.66) -- (2.66,3.33)  -- cycle;
%%---bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulb]%
(0,3) -- (-0.33, 3.33) -- (0.66,3.33) -- (1,3)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umb]%
(1,3) -- (0.66, 3.33) -- (1.66,3.33) -- (2,3)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urb]%
(2,3) -- (1.66, 3.33) -- (2.66,3.33) -- (3,3)  -- cycle;%
\ %%trailing space
}
```

RWDN19D removed DrawRubikLayerFace commands Feb 19 2018

## 20.14 DrawRubikFlatX commands

[Background: These commands (new in version 3.0) were modified from the earlier `\FlatUp`, `\FlatDown` etc., commands; i.e., they were renamed as a set of `\Draw...` commands so as to make this notation consistent with the other `\Draw..` commands. Note also that the `\DrawRubikFace..` commands are essentially these same commands but with their two coordinate arguments X,Y set to $x = 0$, $y = 0$—see Section 20.15]

`\DrawRubikFlatUp`
`\DrawRubikFlatDown`
`\DrawRubikFlatLeft`
`\DrawRubikFlatRight`
`\DrawRubikFlatFront`
`\DrawRubikFlatBack`

Each of these commands draws a separate (flat) face (9 facelets). Each command (except `\DrawRubikFlatFront`) takes two arguments, namely the X-coordinate and Y-coordinate of the bottom-left corner of the face. This (X,Y) pair of coordinates therefore allows the user to position the face in relation to the cube itself.

These commands were motivated by a need to be able to show hidden faces under certain circumstances.

Note also that the \DrawRubikFlatFront command takes no arguments, since by definition the bottom left corner of this face is at (0,0), and there seems to be no reason (just now) for this face to have the facility to be positioned otherwise.

EXAMPLE: The following command positions the Up face so that its bottom left corner is located at (0,3):

```
\DrawRubikFlatUp{0}{3}
```

These new commands are also used by the commands \DrawRubikCubeF and \DrawRubikCubeSF to draw various 'flat' representations of a Rubik cube.

The (x,y) variables are here encoded as (\ux, \uy) where the 'u' stands for Up etc. However, since we are unable to use a 'dx, dy' notation with the \DrawRubikFlatDown command (since dx and dy are already used by the \cube@dxdydz... command), we encode these instead as (\ddx, \ddy).

```
\newcommand{\DrawRubikFlatUp}[2]{%
\pgfmathsetmacro{\ux}{#1}%
\pgfmathsetmacro{\uy}{#2}%
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Ult]%
(\ux + 0,\uy + 2) -- (\ux + 0,\uy + 3) -- (\ux + 1,\uy + 3)%
 -- (\ux + 1,\uy + 2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umt]%
(\ux + 1,\uy + 2) -- (\ux + 1,\uy + 3) -- (\ux + 2,\uy + 3)%
 -- (\ux + 2,\uy + 2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urt]%
(\ux + 2,\uy + 2) -- (\ux + 2,\uy + 3) -- (\ux + 3,\uy + 3)%
 -- (\ux + 3,\uy + 2) -- cycle;
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulm]%
(\ux + 0,\uy + 1) -- (\ux + 0,\uy + 2) -- (\ux + 1,\uy + 2)%
 -- (\ux + 1,\uy + 1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umm]%
(\ux + 1,\uy + 1) -- (\ux + 1,\uy + 2) -- (\ux + 2,\uy + 2)%
 -- (\ux + 2,\uy + 1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urm]%
(\ux + 2,\uy + 1) -- (\ux + 2,\uy + 2) -- (\ux + 3,\uy + 2)%
 -- (\ux + 3,\uy + 1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Ulb]%
(\ux + 0,\uy + 0) -- (\ux + 0,\uy + 1) -- (\ux + 1,\uy + 1)%
 -- (\ux + 1,\uy + 0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Umb]%
(\ux + 1,\uy + 0) -- (\ux + 1,\uy + 1) -- (\ux + 2,\uy + 1)%
 -- (\ux + 2,\uy + 0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Urb]%
(\ux + 2,\uy + 0) -- (\ux + 2,\uy + 1) -- (\ux + 3,\uy + 1)%
 -- (\ux + 3,\uy + 0) -- cycle;
```

```
}
%%-------------------------
\newcommand{\DrawRubikFlatDown}[2]{%
\pgfmathsetmacro{\ddx}{#1}%
\pgfmathsetmacro{\ddy}{#2}%
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlt]%
(\ddx + 0,\ddy + 2) -- (\ddx + 0,\ddy + 3) -- (\ddx + 1,\ddy + 3)%
 -- (\ddx + 1,\ddy + 2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmt]%
(\ddx + 1,\ddy + 2) -- (\ddx + 1,\ddy + 3) -- (\ddx + 2,\ddy + 3)%
 -- (\ddx + 2,\ddy + 2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drt]%
(\ddx + 2,\ddy + 2) -- (\ddx + 2,\ddy + 3) -- (\ddx + 3,\ddy + 3)%
 -- (\ddx + 3,\ddy + 2) -- cycle;
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlm]%
(\ddx + 0,\ddy + 1) -- (\ddx + 0,\ddy + 2) -- (\ddx + 1,\ddy + 2)%
 -- (\ddx + 1,\ddy + 1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmm]%
(\ddx + 1,\ddy + 1) -- (\ddx + 1,\ddy + 2) -- (\ddx + 2,\ddy + 2)%
 -- (\ddx + 2,\ddy + 1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drm]%
(\ddx + 2,\ddy + 1) -- (\ddx + 2,\ddy + 2) -- (\ddx + 3,\ddy + 2)%
 -- (\ddx + 3,\ddy + 1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Dlb]%
(\ddx + 0,\ddy + 0) -- (\ddx + 0,\ddy + 1) -- (\ddx + 1,\ddy + 1)%
 -- (\ddx + 1,\ddy + 0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Dmb]%
(\ddx + 1,\ddy + 0) -- (\ddx + 1,\ddy + 1) -- (\ddx + 2,\ddy + 1)%
 -- (\ddx + 2,\ddy + 0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Drb]%
(\ddx + 2,\ddy + 0) -- (\ddx + 2,\ddy + 1) -- (\ddx + 3,\ddy + 1)%
 -- (\ddx + 3,\ddy + 0) -- cycle;
}
%%-------------------------
\newcommand{\DrawRubikFlatLeft}[2]{%
\pgfmathsetmacro{\lx}{#1}%
\pgfmathsetmacro{\ly}{#2}%
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Llt]%
(\lx + 0, \ly + 2) -- (\lx + 0, \ly + 3) -- (\lx + 1, \ly + 3)%
 -- (\lx + 1, \ly + 2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmt]%
(\lx + 1, \ly + 2) -- (\lx + 1, \ly + 3) -- (\lx + 2, \ly + 3)%
 -- (\lx + 2, \ly + 2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrt]%
(\lx + 2, \ly + 2) -- (\lx + 2, \ly + 3) -- (\lx + 3, \ly + 3)%
 -- (\lx + 3, \ly + 2) -- cycle;
```

```
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Llm]%
(\lx + 0, \ly + 1) -- (\lx + 0, \ly + 2) -- (\lx + 1, \ly + 2)%
 -- (\lx + 1, \ly + 1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmm]%
(\lx + 1, \ly + 1) -- (\lx + 1, \ly + 2) -- (\lx + 2, \ly + 2)%
 -- (\lx + 2, \ly + 1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrm]%
(\lx + 2, \ly + 1) -- (\lx + 2, \ly + 2) -- (\lx + 3, \ly + 2)%
 -- (\lx + 3, \ly + 1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Llb]%
(\lx + 0, \ly + 0) -- (\lx + 0, \ly + 1) -- (\lx + 1, \ly + 1)%
 -- (\lx + 1, \ly + 0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lmb]%
(\lx + 1, \ly + 0) -- (\lx + 1, \ly + 1) -- (\lx + 2, \ly + 1)%
 -- (\lx + 2, \ly + 0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Lrb]%
(\lx + 2, \ly + 0) -- (\lx + 2, \ly + 1) -- (\lx + 3, \ly + 1)%
 -- (\lx + 3, \ly + 0) -- cycle;
}
%%-------------------------
\newcommand{\DrawRubikFlatRight}[2]{%
\pgfmathsetmacro{\rx}{#1}%
\pgfmathsetmacro{\ry}{#2}%
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlt]%
(\rx + 0, \ry + 2) -- (\rx + 0,  \ry + 3) -- (\rx + 1, \ry + 3)%
 -- (\rx + 1, \ry + 2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmt]%
(\rx + 1, \ry + 2) -- (\rx + 1, \ry + 3) -- (\rx + 2, \ry + 3)%
 -- (\rx + 2, \ry + 2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrt]%
(\rx + 2, \ry + 2) -- (\rx + 2, \ry + 3) -- (\rx + 3, \ry + 3)%
 -- (\rx + 3, \ry + 2) -- cycle;
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlm]%
(\rx + 0, \ry + 1) -- (\rx + 0, \ry + 2) -- (\rx + 1, \ry + 2)%
 -- (\rx + 1, \ry + 1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmm]%
(\rx + 1, \ry + 1) -- (\rx + 1, \ry + 2) -- (\rx + 2, \ry + 2)%
 -- (\rx + 2, \ry + 1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrm]%
(\rx + 2, \ry + 1) -- (\rx + 2, \ry + 2) -- (\rx + 3, \ry + 2)%
 -- (\rx + 3, \ry + 1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Rlb]%
(\rx + 0, \ry + 0) -- (\rx + 0, \ry + 1) -- (\rx + 1, \ry + 1)%
 -- (\rx + 1, \ry + 0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rmb]%
```

```
(\rx + 1, \ry + 0) -- (\rx + 1, \ry + 1) -- (\rx + 2, \ry + 1)%
 -- (\rx + 2, \ry + 0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Rrb]%
(\rx + 2, \ry + 0) -- (\rx + 2, \ry + 1) -- (\rx + 3, \ry + 1)%
 -- (\rx + 3, \ry + 0) -- cycle;
}
%%-----------------------
\newcommand{\DrawRubikFlatFront}{%
%% This command is used /only/ by  the \cmd{\DrawRubikCubeF} command.
%% NOTE: x, y variables not implemented as not required here
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Flt]%
(0,2) -- (0, 3) -- (1,3) -- (1,2)  -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Fmt]%
(1,2) -- (1, 3) -- (2,3) -- (2,2) -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Frt]%
(2,2) -- (2, 3) -- (3,3) -- (3,2) -- cycle;
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Flm]%
(0,1) -- (0, 2) -- (1,2) -- (1,1)  -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Fmm]%
(1,1) -- (1, 2) -- (2,2) -- (2,1) -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Frm]%
(2,1) -- (2, 2) -- (3,2) -- (3,1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Flb]%
(0,0) -- (0, 1) -- (1,1) -- (1,0)  -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Fmb]%
(1,0) -- (1, 1) -- (2,1) -- (2,0) -- cycle;
%%
\draw[line join=round,line cap=round,ultra thick,fill=\Frb]%
(2,0) -- (2, 1) -- (3,1) -- (3,0) -- cycle;
}
%%-------------------------
\newcommand{\DrawRubikFlatBack}[2]{%
\pgfmathsetmacro{\bx}{#1}%
\pgfmathsetmacro{\by}{#2}%
%%---top row
\draw[line join=round,line cap=round,ultra thick,fill=\Blt]%
(\bx + 0,\by + 2) -- (\bx + 0,\by + 3) -- (\bx + 1,\by + 3)%
 -- (\bx + 1,\by + 2)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Bmt]%
(\bx + 1,\by + 2) -- (\bx + 1,\by + 3) -- (\bx + 2,\by + 3)%
 -- (\bx + 2,\by + 2) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Brt]%
```

```
(\bx + 2,\by + 2) -- (\bx + 2,\by + 3) -- (\bx + 3,\by + 3)%
 -- (\bx + 3,\by + 2) -- cycle;
%%-----middle row
\draw[line join=round,line cap=round,ultra thick,fill=\Blm]%
(\bx + 0,\by + 1) -- (\bx + 0,\by + 2) -- (\bx + 1,\by + 2)%
 -- (\bx + 1,\by + 1)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Bmm]%
(\bx + 1,\by + 1) -- (\bx + 1,\by + 2) -- (\bx + 2,\by + 2)%
 -- (\bx + 2,\by + 1) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Brm]%
(\bx + 2,\by + 1) -- (\bx + 2,\by + 2) -- (\bx + 3,\by + 2)%
 -- (\bx + 3,\by + 1) -- cycle;
%%----bottom row
\draw[line join=round,line cap=round,ultra thick,fill=\Blb]%
(\bx + 0,\by + 0) -- (\bx + 0,\by + 1) -- (\bx + 1,\by + 1)%
 -- (\bx + 1,\by + 0)  -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Bmb]%
(\bx + 1,\by + 0) -- (\bx + 1,\by + 1) -- (\bx + 2,\by + 1)%
 -- (\bx + 2,\by + 0) -- cycle;
\draw[line join=round,line cap=round,ultra thick,fill=\Brb]%
(\bx + 2,\by + 0) -- (\bx + 2,\by + 1) -- (\bx + 3,\by + 1)%
 -- (\bx + 3,\by + 0) -- cycle;
}
```

`\DrawRubikCubeF` Draws a standard flat (F) representation of the Rubik cube (colours only). Note that `\DrawRubikFlatFront` (below) does not take any arguments (x,y).

```
\newcommand{\DrawRubikCubeF}{%
  \DrawRubikFlatUp{0}{3}%
  \DrawRubikFlatDown{0}{-3}%
  \DrawRubikFlatLeft{-3}{0}%
  \DrawRubikFlatFront%
  \DrawRubikFlatRight{3}{0}%
  \DrawRubikFlatBack{6}{0}%
}
```

`\DrawRubikCubeSF` Draws a Rubik cube together with the three hidden faces (colours only) in a semi-flat (SF) representation. The (x,y) arguments (below) are for the bottom-left corner of the face.

```
\newcommand{\DrawRubikCubeSF}{%
  \DrawRubikCubeRU%
  \DrawRubikFlatDown{0}{-3}%
  \DrawRubikFlatLeft{-3}{0}%
  \DrawRubikFlatBack{4}{1}%
}
```

### 20.14.1 DrawRubikFaceXSide commands

These six commands draw a face together with all four sidebars (colours only). They use the `\DrawRubikFlatX` commands to draw the face, and the `\side@bar..`

commands to draw the sidebars. Since each of the four sides of a face has three small (side) bars, we use a total of 12 side@bar commands for each command.

The parameter codes for the `\side@bar..` commands are as follows (see Section 20.16.1 for full details). Top (T) and Bottom (B) = left to right; Left (L) and Right (R) = top to bottom;

Note that since these commands are quite long, thay all have a slightly more convenient short-hand version whereby the terminal two words are contracted to their two initial letters (see Section 20.15). RWDN19B

`\DrawRubikFaceUpSide` Draws the UP face together with all four sidebars.

```
\newcommand{\DrawRubikFaceUpSide}{%
\DrawRubikFlatUp{0}{0}%
%Top is a horizontal sidebar, so 1 = r, 3=l
\side@barT{1}{\Brt}
\side@barT{2}{\Bmt}
\side@barT{3}{\Blt}
%Left is a vertical sidebar, so 1 = r, 3=l
\side@barL{1}{\Lrt}
\side@barL{2}{\Lmt}
\side@barL{3}{\Llt}
%Right is a vertical sidebar, so 1 = l, 3=r
\side@barR{1}{\Rlt}
\side@barR{2}{\Rmt}
\side@barR{3}{\Rrt}
%Bottom is a horizontal sidebar, so 1 = l, 3=r
\side@barB{1}{\Flt}
\side@barB{2}{\Fmt}
\side@barB{3}{\Frt}
}
```

`\DrawRubikFaceFrontSide` Draws the FRONT face together with all four sidebars.

```
\newcommand{\DrawRubikFaceFrontSide}{%
\DrawRubikFlatFront{0}{0}%
%Top
\side@barT{1}{\Ulb}
\side@barT{2}{\Umb}
\side@barT{3}{\Urb}
%Left
\side@barL{1}{\Lrb}
\side@barL{2}{\Lrm}
\side@barL{3}{\Lrt}
%Right
\side@barR{1}{\Rlb}
\side@barR{2}{\Rlm}
\side@barR{3}{\Rlt}
%Bottom
\side@barB{1}{\Dlt}
\side@barB{2}{\Dmt}
\side@barB{3}{\Drt}
```

```
}
```

\DrawRubikFaceRightSide Draws the RIGHT face together with all four sidebars.

```
\newcommand{\DrawRubikFaceRightSide}{%
\DrawRubikFlatRight{0}{0}%
%Top
\side@barT{1}{\Urb}
\side@barT{2}{\Urm}
\side@barT{3}{\Urt}
%Left
\side@barL{1}{\Frb}
\side@barL{2}{\Frm}
\side@barL{3}{\Frt}
%Right
\side@barR{1}{\Blb}
\side@barR{2}{\Blm}
\side@barR{3}{\Blt}
%Bottom
\side@barB{1}{\Drt}
\side@barB{2}{\Drm}
\side@barB{3}{\Drb}
}
```

\DrawRubikFaceLeftSide Draws the LEFT face together with all four sidebars.

```
\newcommand{\DrawRubikFaceLeftSide}{%
\DrawRubikFlatLeft{0}{0}%
%Top
\side@barT{1}{\Ult}
\side@barT{2}{\Ulm}
\side@barT{3}{\Ulb}
%Left
\side@barL{1}{\Brb}
\side@barL{2}{\Brm}
\side@barL{3}{\Brt}
%Right
\side@barR{1}{\Flb}
\side@barR{2}{\Flm}
\side@barR{3}{\Flt}
%Bottom
\side@barB{1}{\Dlb}
\side@barB{2}{\Dlm}
\side@barB{3}{\Dlt}
}
```

\DrawRubikFaceBackSide Draws the BACK face together with all four sidebars.

```
\newcommand{\DrawRubikFaceBackSide}{%
\DrawRubikFlatBack{0}{0}%
%Top
\side@barT{1}{\Urt}
```

```
\side@barT{2}{\Umt}
\side@barT{3}{\Ult}
%Left
\side@barL{1}{\Rrb}
\side@barL{2}{\Rrm}
\side@barL{3}{\Rrt}
%Right
\side@barR{1}{\Llb}
\side@barR{2}{\Llm}
\side@barR{3}{\Llt}
%Bottom
\side@barB{1}{\Drb}
\side@barB{2}{\Dmb}
\side@barB{3}{\Dlb}
}
```

`\DrawRubikFaceDownSide` Draws the DOWN face together with all four sidebars.

```
\newcommand{\DrawRubikFaceDownSide}{%
\DrawRubikFlatDown{0}{0}%
%Top
\side@barT{1}{\Flb}
\side@barT{2}{\Fmb}
\side@barT{3}{\Frb}
%Left
\side@barL{1}{\Llb}
\side@barL{2}{\Lmb}
\side@barL{3}{\Lrb}
%Right
\side@barR{1}{\Rrb}
\side@barR{2}{\Rmb}
\side@barR{3}{\Rlb}
%Bottom
\side@barB{1}{\Brb}
\side@barB{2}{\Bmb}
\side@barB{3}{\Blb}
}
```

## 20.15 DrawRubikFace commands

BACKGROUND: The above `\DrawRubikFlat..` commands were originally just `\DrawFlat..` commands. They were subsequently copied & renamed (here) as the more intuitive `\DrawFace..` commands, since most of the time the user wants just to draw a particular face with or without sidebars.

In v5 it was necessary to rename them as `\DrawRubikFace...` commands since they relate to a 3x3x3 cube. At the same time we also included their more convenient short-hand versions (eg `\DrawRubikFaceUpSide` → `\DrawRubikFaceUS` etc), as the commands were getting a bit too long.

We therefore now deprecate the use of the earlier `\DrawFace...` commands,

although they will be maintained for the moment at least.

```
\newcommand{\DrawRubikFaceUp}{\DrawRubikFlatUp{0}{0}}
\newcommand{\DrawRubikFaceDown}{\DrawRubikFlatDown{0}{0}}
\newcommand{\DrawRubikFaceLeft}{\DrawRubikFlatLeft{0}{0}}
\newcommand{\DrawRubikFaceRight}{\DrawRubikFlatRight{0}{0}}
\newcommand{\DrawRubikFaceFront}{\DrawRubikFlatFront{0}{0}}
\newcommand{\DrawRubikFaceBack}{\DrawRubikFlatBack{0}{0}}
```

RWDN19A Finally, we create the short-hand versions.

```
\newcommand{\DrawRubikFaceU}{\DrawRubikFaceUp}
\newcommand{\DrawRubikFaceD}{\DrawRubikFaceDown}
\newcommand{\DrawRubikFaceL}{\DrawRubikFaceLeft}
\newcommand{\DrawRubikFaceR}{\DrawRubikFaceRight}
\newcommand{\DrawRubikFaceF}{\DrawRubikFaceFront}
\newcommand{\DrawRubikFaceB}{\DrawRubikFaceBack}
\newcommand{\DrawRubikFaceUS}{\DrawRubikFaceUpSide}
\newcommand{\DrawRubikFaceDS}{\DrawRubikFaceDownSide}
\newcommand{\DrawRubikFaceLS}{\DrawRubikFaceLeftSide}
\newcommand{\DrawRubikFaceRS}{\DrawRubikFaceRightSide}
\newcommand{\DrawRubikFaceFS}{\DrawRubikFaceFrontSide}
\newcommand{\DrawRubikFaceBS}{\DrawRubikFaceBackSide}
```

## 20.16 Sidebars (Face)

Sidebar commands draw narrow bars of colour indicating the side colours of each of the facelets forming the side of a given layer (face). Each Sidebar is the length of a single facelet (see Section 10.6).

`\RubikSidebarWidth` `\RubikSidebarLength` `\RubikSidebarSep`

These three commands allow the user to set the Width, Length and Separation parameters for the sidebar (in decimal values, where 1 is equivalent to the length of the side of a facelet).

```
\newcommand{\RubikSidebarWidth}[1]{\pgfmathsetmacro{\bw}{#1}}
\newcommand{\RubikSidebarLength}[1]{\pgfmathsetmacro{\bl}{#1}}
\newcommand{\RubikSidebarSep}[1]{\pgfmathsetmacro{\bs}{#1}}
```

We first set some default values

```
\RubikSidebarWidth{0.3}%
\RubikSidebarLength{1}%
\RubikSidebarSep{0.3}%
```

In order to avoid conflicting nomenclature (between bar and Bar) we recommend using the lowercase 'bar' and deprecate the use of 'Bar' in commands. For backwards compatibility, however, we will retain the three original 'Bar' commands (for the moment at least), as follows:

```
\newcommand{\RubikSideBarWidth}[1]{\pgfmathsetmacro{\bw}{#1}}
\newcommand{\RubikSideBarLength}[1]{\pgfmathsetmacro{\bl}{#1}}
\newcommand{\RubikSideBarSep}[1]{\pgfmathsetmacro{\bs}{#1}}
```

### 20.16.1 Drawing and allocating a colour to a single facelet sidebar

Full length face sidebars are really multiple instances of small single facelet bars, each of which is drawn using one of the internal sidebar commands.

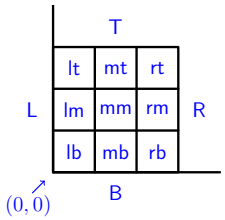



Figure 5: Face-edge and facelet-position codes (see text)

\side@barT
\side@barB
\side@barL
\side@barR

There are four \side@barX commands, each with a trailing letter code X, which indicates the position of the sidebar relative to the square face displayed, namely either T (Top), B (Bottom), L (Left) or R (Right) —see Figure 5.

\side@barL{⟨*position-number*⟩}{⟨*facelet location-code*⟩}. Each \side@barX command takes two arguments: The first argument is a number (distance) {⟨1 | 2 | 3⟩} from the relevant axis depending on whether it is a vertical or horizontal sidebar (e.g., 1 = first bar (nearest the origin); 2 = second bar, 3 = third bar).

The second argument is the facelet location-code expressed as a command (e.g., \Lrt, \Dlb etc. Note that the facelet location uses a three letter code: the first (capital) letter (U, D, L, R, F, B) denotes the FACE; the second (lower-case) letter (l,m,r) is the 'x' position in the 3x3 matrix; the third (lower-case) letter (t,m,b) is the 'y' position —see Figure 5.

EXAMPLE: the following command draws a small single Right sidebar, in the middle position (no. 2), with the colour allocated to the Rlt facelet (left top facelet in the RIGHT face.

```
\side@barR{2}{\Rlt}
```

Notice that we use the command \Rlt as the argument; this is because the command is defined as: \def{\Rlt}{#1}, and hence the command gets replaced by the colour-code currently allocated to this particular facelet.

There are three small rectangular sidebars on each of the four sides of a 3x3 square face, and these are embedded in a coordinate system with origin at the bottom left corner (0,0) of the square face (see Figure 5).

The \side@bar.. command also implements the set (or default) RubikSidebarLength \bl, RubikSidebarWidth \bw and RubikSidebarSep \bs (separation) values mentioned above. \blh = Half \bl = \bl/2. Note that the TikZ \pgfmathsetmacro commands (which do the maths) must be inside the \side@bar.. command in order to work. The start point of the TikZ \draw command for each rectangular sidebar is the bottom Left corner of the sidebar = (\dx,\dy).

\no@sidebar The \NoSidebar{⟨*colour code*⟩} command defines a (single) colour for which sidebars should *not* be drawn (particularly useful when drawing OLL configurations). This idea was suggested by Robert Mařík (May 2017) —see Section 10.8.
The principle is that we let the command \NoSidebar define a face colour, and then we use the \ifthenelse{\equal{#2}{\no@sidebar}}{}{...} structure inside the \side@bar.. commands (see below) to either (a) draw all sidebars as usual (if \NoSidebar is undefined), or (b) draw all sidebars *except* those having the \NoSidebar colour (if \NoSidebar colour = #2).
USAGE: \NoSidebar{X} If this command in *not* inside an environment, then its action will continue until it is disabled (undefined) as follows: \NoSidebar{}.

```
\def\no@sidebar{}%
\newcommand{\NoSidebar}[1]{\def\no@sidebar{#1}}

\newcommand{\side@barL}[2]{%
  %% #1 = cubie possn no, #2 = colour
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{0 - \bs - \bw}%
  \pgfmathsetmacro{\dy}{#1-1+0.5-\blh}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
\newcommand{\side@barR}[2]{%
  %% #1 = cubie possn no, #2 = colour
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{3 + \bs}%
  \pgfmathsetmacro{\dy}{#1 -1+0.5-\blh}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
\newcommand{\side@barT}[2]{%
  %% #1 = cubie possn no, #2 = colour
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh}%
  \pgfmathsetmacro{\dy}{3 +\bs}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
\newcommand{\side@barB}[2]{%
  %% #1 = cubie possn no, #2 = colour
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh}%
  \pgfmathsetmacro{\dy}{0 -\bs-\bw}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
```

RWDN19C removed 20:17:2 and 20:17:3 19 Feb 2018

## 20.17 Sidebars (Cube)

In order to position sidebars adjacent to a Rubik Cube (ie in 3D) requires that we first make some new `\side@bar..` commands for drawing sidebars adjacent to the BACK face of the cube (we have already made the macros for the front face sidebars—see Section 20.16). Furthermore, these new macros need to be tailored to each of the four standard cube viewing directions RU, LU, RD, LD.

Finally, the USER commands for drawing these sidebars need to accommodate (a) some code for identifying each set of sidebars, and (b) the viewing direction. So, for example, a USER command for drawing the sidebars associated with the cube edge formed by the RIGHT face and the BACK face (lets define this as the RB sidebar) as viewed from the RU direction, might be something like `\DrawRubikCubeSidebarRBRU`. Since this is not particularly user-friendly, we can improve on this slightly for the USER by (a) defining the sidebar as `SidebarRB`, and (b) appending the view direction in a curly bracket, say as `{RU}`. This allows a more intuitive command structure for the USER, as follows: `\DrawRubikCubeSidebarRB{RU}`. We then use the `\@join` command to append the string `RU` to the string `DrawRubicCubeSidebarRB` forming the (internal) command `\DrawRubicCubeSidebarRBRU`.

In the following we will group the code according to to the view direction (RU, LU, RD, LD).

### 20.17.1 Sidebars: RU view

#### Right-Back vert sidebar (RU view)

Need to write a new command for this position modified from `\side@barR` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikRbackRU}[2]{%
  %% #1 = cubie possn no, #2 = colour
  %% dx --> dx+1
  %% dy --> dy+1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{3 + \bs +1}%
  \pgfmathsetmacro{\dy}{#1 -1+0.5-\blh +1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
```

Make the RB (RightBack) version; bar 1 is at the bottom

```
\newcommand{\DrawRubikCubeSidebarRBRU}{%
\side@barRubikRbackRU{3}{\Blt}%
\side@barRubikRbackRU{2}{\Blm}%
\side@barRubikRbackRU{1}{\Blb}%
```

```
}
```

Now do the reverse (BR) = RB

```
\newcommand{\DrawRubikCubeSidebarBRRU}{\DrawRubikCubeSidebarRBRU}
```

Make the join commands

```
\newcommand{\DrawRubikCubeSidebarRB}[1]{\@join{\DrawRubikCubeSidebarRB}{#1}}
\newcommand{\DrawRubikCubeSidebarBR}[1]{\@join{\DrawRubikCubeSidebarBR}{#1}}
```

### Up-Back horiz sidebar (RU view)

Need to write a new command for this position modified from `\side@barT` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikTbackRU}[2]{%
  %% #1 = cubie possn no; #2 = colour
  %% dx --> dx+1
  %% dy --> dy+1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh +1}%
  \pgfmathsetmacro{\dy}{3 +\bs +1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
```

Make the UB (Up-Back) version; bar 1 is at the left, 3 on the rhs (as we look at the image)

```
\newcommand{\DrawRubikCubeSidebarUBRU}{%
\side@barRubikTbackRU{1}{\Brt}%
\side@barRubikTbackRU{2}{\Bmt}%
\side@barRubikTbackRU{3}{\Blt}%
}
```

Now do the reverse (BU) = UB

```
\newcommand{\DrawRubikCubeSidebarBURU}{\DrawRubikCubeSidebarUBRU}
```

Make the join commands

```
\newcommand{\DrawRubikCubeSidebarUB}[1]{\@join{\DrawRubikCubeSidebarUB}{#1}}
\newcommand{\DrawRubikCubeSidebarBU}[1]{\@join{\DrawRubikCubeSidebarBU}{#1}}
```

### Front-Left vert sidebar (RU view)

For the front face we can use the regular `\ide@barL` commands since it is the same as for an ordinary face sidebar

```
\newcommand{\DrawRubikCubeSidebarFLRU}{%
\side@barL{3}{\Lrt}%
\side@barL{2}{\Lrm}%
\side@barL{1}{\Lrb}%
}
```

Now do the reverse (LF)

```
\newcommand{\DrawRubikCubeSidebarLFRU}{\DrawRubikCubeSidebarFLRU}
```

Now do the two join commands

```
\newcommand{\DrawRubikCubeSidebarFL}[1]{\@join{\DrawRubikCubeSidebarFL}{#1}}
\newcommand{\DrawRubikCubeSidebarLF}[1]{\@join{\DrawRubikCubeSidebarLF}{#1}}
```

#### Front-Down horizontal sidebar (RU view)

Horiz sidebar, so 1 at the left, 2=middle, 3= rhs) here we have to use the B for bottom (of front face) and the facelets of the top row of the Down face

```
\newcommand{\DrawRubikCubeSidebarFDRU}{%
\side@barB{1}{\Dlt}%
\side@barB{2}{\Dmt}%
\side@barB{3}{\Drt}%
}
```

Now do the reverse (DF) = FD

```
\newcommand{\DrawRubikCubeSidebarDFRU}{\DrawRubikCubeSidebarFDRU}
```

Now do the two join commands

```
\newcommand{\DrawRubikCubeSidebarFD}[1]{\@join{\DrawRubikCubeSidebarFD}{#1}}
\newcommand{\DrawRubikCubeSidebarDF}[1]{\@join{\DrawRubikCubeSidebarDF}{#1}}
```

But FD-LU is the same as FD-RU, so need to make copies of each

```
\newcommand{\DrawRubikCubeSidebarDFLU}{\DrawRubikCubeSidebarDFRU}
\newcommand{\DrawRubikCubeSidebarFDLU}{\DrawRubikCubeSidebarFDRU}
```

### 20.17.2 Sidebars: LU view

#### Left-Back vert sidebar (LU view)

Need to write a new command for this position modified from `\side@barL` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikLbackLU}[2]{%
  %% #1 = cubie possn no, #2 = colour
  %% dx --> dx-1
  %% dy --> dy+1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{0 - \bs -\bw -1}%
  \pgfmathsetmacro{\dy}{#1 -1+0.5-\blh +1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
```

Make the LB (LeftBack) version; bar 1 is at the bottom

```
\newcommand{\DrawRubikCubeSidebarLBLU}{%
\side@barRubikLbackLU{3}{\Brt}%
\side@barRubikLbackLU{2}{\Brm}%
\side@barRubikLbackLU{1}{\Brb}%
}
```

Now do the reverse (BL) = LB

```
\newcommand{\DrawRubikCubeSidebarBLLU}{\DrawRubikCubeSidebarLBLU}
```

Make the join commands

```
\newcommand{\DrawRubikCubeSidebarLB}[1]{\@join{\DrawRubikCubeSidebarLB}{#1}}
\newcommand{\DrawRubikCubeSidebarBL}[1]{\@join{\DrawRubikCubeSidebarBL}{#1}}
```

### Up-Back horizontal sidebar (LU view)

Modified from `\side@barT` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikTbackLU}[2]{%
  %% #1 = cubie possn no; #2 = colour
  %% dx --> dx-1
  %% dy --> dy+1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh -1}%
  \pgfmathsetmacro{\dy}{3 +\bs +1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
```

Make the UB (Up-Back) version; bar 1 is at the left, 3 on the rhs (as we look at the image)

```
\newcommand{\DrawRubikCubeSidebarUBLU}{%
\side@barRubikTbackLU{1}{\Brt}%
\side@barRubikTbackLU{2}{\Bmt}%
\side@barRubikTbackLU{3}{\Blt}%
}
```

Now do the reverse (BU) = UB

```
\newcommand{\DrawRubikCubeSidebarBULU}{\DrawRubikCubeSidebarUBLU}
```

Do not need to make the join commands as the USER commands for BU and UB are the same as for the RU.

### Front-Right vertical sidebar (LU view)

Only needed for the LU view and LD. for the front face we can use the regular side@barR commands since it is the same as for an ordinary face sidebar RHS

```
\newcommand{\DrawRubikCubeSidebarFRLU}{%
\side@barR{3}{\Rlt}%
\side@barR{2}{\Rlm}%
\side@barR{1}{\Rlb}%
}
```

Now do the reverse (RF)

```
\newcommand{\DrawRubikCubeSidebarRFLU}{\DrawRubikCubeSidebarFRLU}
```

Now do the two join commands

```
\newcommand{\DrawRubikCubeSidebarFR}[1]{\@join{\DrawRubikCubeSidebarFR}{#1}}
\newcommand{\DrawRubikCubeSidebarRF}[1]{\@join{\DrawRubikCubeSidebarRF}{#1}}
```

### 20.17.3 Sidebars: RD view

#### Front-Up horizontal sidebar (RD view)

Horiz sidebar, so 1 at the left, 2=middle, 3= rhs here we have to use the T for bottom (of front face) and the facelets of the top row of the Down face

```
\newcommand{\DrawRubikCubeSidebarFURD}{%
\side@barT{1}{\Ulb}%
\side@barT{2}{\Umb}%
\side@barT{3}{\Urb}%
}
```

Now do the reverse (UF) = FU

```
\newcommand{\DrawRubikCubeSidebarUFRD}{\DrawRubikCubeSidebarFURD}
```

Now do the two join commands

```
\newcommand{\DrawRubikCubeSidebarFU}[1]{\@join{\DrawRubikCubeSidebarFU}{#1}}
\newcommand{\DrawRubikCubeSidebarUF}[1]{\@join{\DrawRubikCubeSidebarUF}{#1}}
```

#### Front-Left vertical sidebar (RD view)

Front LEFT (RD view = same as for RU)

```
\newcommand{\DrawRubikCubeSidebarFLRD}{\DrawRubikCubeSidebarFLRU}
\newcommand{\DrawRubikCubeSidebarLFRD}{\DrawRubikCubeSidebarLFRU}
```

#### Right-Back vertical sidebar (RD view)

Modified from `\side@barR` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikRbackRD}[2]{%
  %% #1 = cubie possn no, #2 = colour
  %% dx --> dx+1
  %% dy --> dy-1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
```

```
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{3 + \bs +1}%
  \pgfmathsetmacro{\dy}{#1 -1+0.5-\blh -1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
```

Make the RB (RightBack) version; bar 1 is at the bottom

```
\newcommand{\DrawRubikCubeSidebarRBRD}{%
\side@barRubikRbackRD{3}{\Blt}%
\side@barRubikRbackRD{2}{\Blm}%
\side@barRubikRbackRD{1}{\Blb}%
}
```

Now do the reverse (BR) = RB

```
\newcommand{\DrawRubikCubeSidebarBRRD}{\DrawRubikCubeSidebarRBRD}
```

Do NOT need to make the join commands (same as for the RU view)

### Down-Back horizotal sidebar (RD view)

Modified from `\side@barB` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikBbackRD}[2]{%
  %% #1 = cubie possn no; #2 = colour
  %% dx --> dx+1
  %% dy --> dy-1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh +1}%
  \pgfmathsetmacro{\dy}{0 -\bs - \bw -1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
```

Make the DB (Down-Back) version; bar 1 is at the left, 3 on the rhs (as we look at the image)

```
\newcommand{\DrawRubikCubeSidebarDBRD}{%
\side@barRubikBbackRD{1}{\Brb}%
\side@barRubikBbackRD{2}{\Bmb}%
\side@barRubikBbackRD{3}{\Blb}%
}
```

Now do the reverse (BD) = DB

```
\newcommand{\DrawRubikCubeSidebarBDRD}{\DrawRubikCubeSidebarDBRD}
```

Make the join commands

```
\newcommand{\DrawRubikCubeSidebarDB}[1]{\@join{\DrawRubikCubeSidebarDB}{#1}}
\newcommand{\DrawRubikCubeSidebarBD}[1]{\@join{\DrawRubikCubeSidebarBD}{#1}}
```

### 20.17.4 Sidebars: LD view

#### Front-Up horizontal sidebar (LD view)

But FR (LD view) is the same as for (RU view), (see above)

```
\newcommand{\DrawRubikCubeSidebarFULD}{\DrawRubikCubeSidebarFURD}
\newcommand{\DrawRubikCubeSidebarUFLD}{\DrawRubikCubeSidebarUFRD}
```

#### Front-Right vertical sidebar (LD view)

Front Right (LDview) = same as for (LU view), (see above)

```
\newcommand{\DrawRubikCubeSidebarFRLD}{\DrawRubikCubeSidebarFRLU}
\newcommand{\DrawRubikCubeSidebarRFLD}{\DrawRubikCubeSidebarRFLU}
```

#### Left-Back vertical sidebar (LD view)

Modified from `\side@barL` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikLbackLD}[2]{%
  %% #1 = cubie possn no, #2 = colour
  %% dx --> dx-1
  %% dy --> dy-1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{0 - \bs -\bw -1}%
  \pgfmathsetmacro{\dy}{#1 -1+0.5-\blh -1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bl)
  -- (\dx+\bw,\dy+\bl) -- (\dx+\bw,\dy) -- cycle;
}}
```

Make the LB (LeftBack) version; bar 1 is at the bottom

```
\newcommand{\DrawRubikCubeSidebarLBLD}{%
\side@barRubikLbackLD{3}{\Brt}%
\side@barRubikLbackLD{2}{\Brm}%
\side@barRubikLbackLD{1}{\Brb}%
}
```

Now do the reverse (BL) = LB

```
\newcommand{\DrawRubikCubeSidebarBLLD}{\DrawRubikCubeSidebarLBLD}
```

Do NOT need to make the join commands (same as for the LU view)

#### Down-Back horizontal sidebar (LD view)

Modified from `\side@barB` (in Section 20.16.1) draws only a single small bar each of the three small bars has a numbered position (1,2,3); (dx,dy) = bottom Left corner of single facelet bar

```
\newcommand{\side@barRubikBbackLD}[2]{%
  %% #1 = cubie possn no; #2 = colour
  %% dx --> dx-1
  %% dy --> dy-1
  \ifthenelse{\equal{#2}{\no@sidebar}}{}{%
  \pgfmathsetmacro{\blh}{\bl*(0.5)}%
  \pgfmathsetmacro{\dx}{#1 -1+0.5-\blh -1}%
  \pgfmathsetmacro{\dy}{0 -\bs - \bw -1}%
  \draw[fill=#2] (\dx,\dy) -- (\dx,\dy + \bw)
  -- (\dx+\bl,\dy+\bw) -- (\dx+\bl,\dy) -- cycle;
}}
```

Make the DB (Down-Back) version; bar 1 is at the left, 3 on the rhs (as we look at the image)

```
\newcommand{\DrawRubikCubeSidebarDBLD}{%
\side@barRubikBbackLD{1}{\Brb}%
\side@barRubikBbackLD{2}{\Bmb}%
\side@barRubikBbackLD{3}{\Blb}%
}
```

Now do the reverse (BD) = DB

```
\newcommand{\DrawRubikCubeSidebarBDLD}{\DrawRubikCubeSidebarDBLD}
```

Do NOT need to make the join commands (same as for the RD view

## 20.18 DrawNCube command

HISTORY: The essence of this command was originally developed by Peter Bartal as his command `\rubikcube` (see Bartal, 2011). We have modified it, as follows (June 2012):

(1) adjusted to use the TikZ `\pgfmathsetmacro{}{}` command (suggested by Peter Grill),

(2) renamed to `\DrawNCubeAll`.

`\DrawNCubeAll` This command draws a solved NxNxN Rubik's cube from the RightUp viewpoint. All cubies on a given face have the same colour. The command takes four ordered arguments, as follows:

> #1 = number of cubies ($n > 0$) along each side,
> #2, #3, #4 = colours of the visible faces (in X,Y,Z order); X=Right face colour, Y=Up face colour, Z=Front face colour.

We use the `\pgfmathsetmacro{`⟨*variable-name*⟩`}{`⟨*numeric value or maths*⟩`}` command. Note that the second argument must not involve any units—just numeric values or mathematics.

```
\newcommand{\DrawNCubeAll}[4]{%
   \pgfmathsetmacro{\ncubes}{#1-1}%
%% need to subtract 1 from the given number of cubies per side
%% to avoid the origin of the initial cube to be displaced
    \foreach \x in {0,...,\ncubes}{%
```

```
      \foreach \y in {0,...,\ncubes}{%
        \foreach \z in {0,...,\ncubes}{%
          \cube@dxdydz{1}{#2}{#3}{#4}{\x}{\y}{\z}%
 }}}}
```

\cube@dxdydz This internal command is used only by the \DrawNCubeAll command (see above). The original version of this command was developed by Peter Bartal (see Bartal, 2011). It was later modified (2012) by RWD Nickalls (to implement a more intuitive X, Y, Z ordering of the face colour parameters).

The cube need not be in the origin, the distances of the DOWN-behind [L] corner from the origin are taken as parameters 5,6,7. The command takes 7 ordered arguments:

1 - length of an edge
2 - X-face colour (RIGHT face)
3 - Y-face colour (UP face)
4 - Z-face colour (FRONT face)
5 - x-position in space
6 - y-position in space
7 - z-position in space

USAGE: \cube@dxdydz{1}{X}{Y}{Z}{x}{y}{z}
The original code \pgfmathparse{#1+#5}\let\dy\pgfmathresult
was changed to the more intuitive \pgfmathsetmacro{\dx}{#1+#5} (suggested by Peter Grill 2011).

CHANGES: RWD Nickalls (2012): (1) added the [line join=round,line cap=round] options to each of the TikZ \draw commands, in order to improve the line joining (first two options); (2) adjusted the \cube@dxdydz macro to adopt the ordered XYZ face colour notation (by reassigning #2, #3, #4 to the X, Y, Z face colours, as shown above).

```
\newcommand{\cube@dxdydz}[7]{%
   \pgfmathsetmacro{\dx}{#1+#5}%
%% calculates the 'displacement' (distance from the origin) of the
%% far corners of the cube along the x axis from the arguments
   \pgfmathsetmacro{\dy}{#1+#6}%
%% calculates the 'displacement' (distance from the origin) of the
%% far corners of the cube along the y axis from the arguments
   \pgfmathsetmacro{\dz}{#1+#7}%
%% calculates the 'displacement' (distance from the origin) of the
%% far corners of the cube along the z axis from the arguments
%% Draw FRONT face   (using the X colour = #4)
    \draw[line join=round,line cap=round,ultra thick,fill=#4]%
 (#5,#6,\dz) -- (\dx,#6,\dz) -- (\dx,\dy,\dz) -- (#5,\dy,\dz) -- cycle;
%% The 'rectangle' command does not work with 3D coordinates,
%% so this is the way to draw the squares with space coordinates
%% Draw UP face (using the Y colour = #3)
   \draw[line join=round,line cap=round,ultra thick,fill=#3]%
 (#5,\dy,\dz) -- (\dx,\dy,\dz) -- (\dx,\dy,#7) -- (#5,\dy,#7) -- cycle;
```

```
%% Draw RIGHT face   (using the X colour = #2)
 \draw[line join=round,line cap=round,ultra thick,fill=#2]%
 (\dx,#6,\dz) -- (\dx,#6,#7) -- (\dx,\dy,#7) -- (\dx,\dy,\dz) -- cycle;
    }
```

## 20.19 Drawing single cubies

\Cubiedx
\Cubiedy

These two commands set the value of the two length parameters cx and cy, and allow the user to vary the size (adjust cy) and horizontal viewpoint (adjust cx) of a single cubie (described in more detail in the RUBIKCUBE package documentation). Note that we cannot use the names dx, dy for the variables here since these names have been allocated already (see above). However, we can use dx, dy in the command names as these will be more readily understood by the user.

```
\newcommand{\Cubiedx}[1]{\pgfmathsetmacro{\cx}{#1}}
\newcommand{\Cubiedy}[1]{\pgfmathsetmacro{\cy}{#1}}
```

We now set the default values (cx=cy=0.4)

```
\Cubiedx{0.4}
\Cubiedy{0.4}
```

\DrawCubieRU
\DrawCubieRD
\DrawCubieLU
\DrawCubieLD

These four commands draw a single cubie from the RightUp, RightDown, LeftUp, LeftDown viewpoint. The viewpoint is specified using an appended two-letter XY ordered viewpoint code: either RU, RD, LU, LD. These commands take three arguments, namely three different XYZ ordered colour codes (R,O,Y,G,B,W,X).
FORMAT: \DrawCubieRU{⟨*Xcolour*⟩}{⟨*Ycolour*⟩}{⟨*Zcolour*⟩}
USAGE: \DrawCubieRU{G}{B}{W}

```
\newcommand{\DrawCubieRU}[3]{%
%% Front face (z)
\draw[line join=round,line cap=round,ultra thick,fill=#3]%
 (0,0) -- (0, 1) -- (1, 1) -- (1,0) -- cycle;
%% Up face(y)
\draw[line join=round,line cap=round,ultra thick,fill=#2]%
 (0,1) -- (\cx, 1+\cy) -- (1+\cx,1+\cy) -- (1,1) -- cycle;
%% Right face(x)
\draw[line join=round,line cap=round,ultra thick,fill=#1]%
 (1,0) -- (1,1) -- (1+\cx,1+\cy) -- (1+\cx, \cy) -- cycle;
}
\newcommand{\DrawCubieRD}[3]{%
%% Front face (z)
\draw[line join=round,line cap=round,ultra thick,fill=#3]%
 (0,0) -- (0, 1) -- (1, 1) -- (1,0) -- cycle;
%% Down face (y)
\draw[line join=round,line cap=round,ultra thick,fill=#2]%
 (\cx,-\cy) -- (0, 0) -- (1,0) -- (1+\cx,-\cy) -- cycle;
%% Right face (x)
\draw[line join=round,line cap=round,ultra thick,fill=#1]%
 (1,0) -- (1,1) -- (1+\cx,-\cy+1) -- (1+\cx, -\cy) -- cycle;
}
```

```
\newcommand{\DrawCubieLD}[3]{%
%% Front face (z)
\draw[line join=round,line cap=round,ultra thick,fill=#3]%
 (0,0) -- (0, 1) -- (1, 1) -- (1,0) -- cycle;
%% Down face (y)
\draw[line join=round,line cap=round,ultra thick,fill=#2]%
 (-\cx,-\cy) -- (0, 0) -- (1,0) -- (1-\cx,-\cy) -- cycle;
%% Left face (x)
\draw[line join=round,line cap=round,ultra thick,fill=#1]%
 (-\cx,-\cy) -- (-\cx,-\cy+1) -- (0,1) -- (0,0) -- cycle;
}
\newcommand{\DrawCubieLU}[3]{%
%% Front face (z)
\draw[line join=round,line cap=round,ultra thick,fill=#3]%
 (0,0) -- (0, 1) -- (1, 1) -- (1,0) -- cycle;
%% Up face (y)
\draw[line join=round,line cap=round,ultra thick,fill=#2]%
 (-\cx,1+\cy) -- (1-\cx, 1+\cy) -- (1,1) -- (0,1) -- cycle;
%% Left face (x)
\draw[line join=round,line cap=round,ultra thick,fill=#1]%
 (-\cx, \cy) -- (-\cx,\cy+1) -- (0,1) -- (0,0) -- cycle;
}
```

## 20.20 Text cubies

\textCubieRU
\textCubieRD
\textCubieLU
\textCubieLD

These four commands draw a single 'text' cubie from the RightUp, RightDown, LeftUp, LeftDown viewpoint. They are 'text' forms of the `\DrawCubie` commands described above. Their size was chosen to be suitable for use with 10–12 point fonts.

As before, the viewpoint is specified using an appended two-letter XY ordered viewpoint code: either RU, RD, LU, LD. These commands take three arguments (since just three faces are visible with this cube format), namely three different XYZ ordered colour codes (R,O,Y,G,B,W,X).

FORMAT: `\textCubieRU{`⟨*Xcolour*⟩`}{`⟨*Ycolour*⟩`}{`⟨*Zcolour*⟩`}`

USAGE: `\textCubieRU{G}{B}{W}`

```
\newcommand{\textCubieRU}[3]{%
\begin{minipage}{0.66cm}
\centering
\begin{tikzpicture}[scale=0.5]
\Cubiedx{0.4}\Cubiedy{0.4}
\DrawCubieRU{#1}{#2}{#3}
\end{tikzpicture}%
\end{minipage}
}
\newcommand{\textCubieRD}[3]{%
\begin{minipage}{0.66cm}
\centering
\begin{tikzpicture}[scale=0.5]
```

```
\Cubiedx{0.4}\Cubiedy{0.4}
\DrawCubieRD{#1}{#2}{#3}
\end{tikzpicture}%
\end{minipage}
}
\newcommand{\textCubieLD}[3]{%
\begin{minipage}{0.66cm}
\centering
\begin{tikzpicture}[scale=0.5]
\Cubiedx{0.4}\Cubiedy{0.4}
\DrawCubieLD{#1}{#2}{#3}
\end{tikzpicture}%
\end{minipage}
}
\newcommand{\textCubieLU}[3]{%
\begin{minipage}{0.66cm}
\centering
\begin{tikzpicture}[scale=0.5]
\Cubiedx{0.4}\Cubiedy{0.4}
\DrawCubieLU{#1}{#2}{#3}
\end{tikzpicture}%
\end{minipage}
}
```

## 20.21 Rotation commands

### 20.21.1 Introduction

We use a special prefix notation to denote each of four different representations of the various Rubik cube rotations as follows: the name of the Rubik rotation (rr), its associated hieroglyph (rrh), and combinations of name and hieroglyph both vertical (Rubik) and horizontal (textRubik). A rotation command is a combination of a rotation-code appended to one of the four prefixes.

For example, the command `\rrhD` generates the hieroglyph (rrh) associated with the rotation-code D. In this form it is used internally, but it is also available for the user.

In version 3.0, however, all the rotation commands were also made available to the user in the much more intuitive form stem{argument} form, for example, `\rrh{D}`. In practice, this 'argument' form actually generates the original non-argument form by the use of the internal macro `\@join`. For example, `\rrh{D}` → join(`\rrh` + `D`) → `\rrhD` (see Section 20.21.4 for details).

The hieroglyphs are of two types: 'arrow' glyphs (all exactly square), and 'letter' glyphs (mostly square, but many are rectangular); however both types are designed to have the same height so they sit nicely when arranged side-by-side. A lot of special macros for generating these glyphs are described below in Section 20.21.3 (and also in Section 20.21.2).

The 'arrow' hieroglyphs are built up in stages using TikZ. We first create a command for drawing the square (`\DrawNotationBox`; see Section 20.21.2) and

then draw the contents (lines, arrows, arcs of circles). For an example, see the D form  constructed in Section 20.21.13.

The 'letter' hieroglyphs (glyphs for which the rotations cannot be seen from the front, and hence cannot have arrows) just give a letter representation of the rotation (say, Bw for 'back wide'). These glyphs are therefore made using an fbox (for convenience), and therefore these are sometimes not square. Some vertical fine-tuning using the `\raisebox` command is often required to force these 'letter' glyphs to have the same vertical position as their 'arrow' cousins. A typical example is the form  which is detailed in Section 20.21.7.

The presence of small overfilled `\hbox`es associated with these squares were originally checked for using the `ltugboat.cls`, and all fixed mainly by setting their associated small minipages → width = 0.6cm, and using TikZ scale=0.5.

### 20.21.2 DrawNotationBox

`\DrawNotationBox` This internal command draws the surrounding square box of all the hieroglyphs. Note that we start at (0,0) and draw to the final point in order to make a nice corner join.

TODO: ? make this a proper internal command using `@` sometime.

```
\newcommand{\DrawNotationBox}{%
  \draw [thick] (0,0) -- (0,1) -- (1,1) -- (1,0) -- (0,0) -- (0,1)%
}
```

We now define a number of points and line-segments inside the square 'notationbox' (e.g., `\@sd`, `\@sh` ... etc.) which will be required for use in drawing the various lines and arrows. Some hieroglyphs contain either one circular arc, or two concentric arcs, and these arcs require both a centre and a start point. Note that the final argument does not use any units. For the TikZ ARC command see TikZ pgfmanual (2012) page 146 (§14.8).

TODO: make a small diagram to illustrate the position of these parameters and make things a bit clearer sometime.

```
\pgfmathsetmacro{\@sd}{0.25}    % a small horiz space
\pgfmathsetmacro{\@sdd}{2*\@sd}  % 2x horiz space
\pgfmathsetmacro{\@sddd}{3*\@sd} % 3x horiz space
\pgfmathsetmacro{\@sh}{0.6} % height
\pgfmathsetmacro{\@sb}{0.2} % base
\pgfmathsetmacro{\@sbh}{\@sb + \@sh} % UP
\pgfmathsetmacro{\@scx}{\@sdd+0.2} % Start of CircleX arc
\pgfmathsetmacro{\@scy}{\@sd*2/3}  % Start of CircleY arc
\pgfmathsetmacro{\@sqcx}{\@scx-0.13} %% SQuare CenterX coord
\pgfmathsetmacro{\@sqcy}{\@scy+0.25} %% SQuare CenterY cpprd
```

### 20.21.3 Some useful internal commands

`\@rr` `\@rrp` `\@rrw` `\@rrwp` `\@rrs` `\@rrsp` `\@rra` `\@rrap` `\@xyzh` `\@xyzhp` `\@xyzRubik` `\@xyzRubikp` `\@SquareLetter`

These internal commands are used to generate the prime, w, w-prime, s, s-prime, a, a-prime rotation commands. They attach a letter or a prime to the associated

argument; for example, the command `\@rrwp{B}` appends a 'w' and a prime (p) to the argument 'B', i.e. → **Bw'** (see Section 20.21.8). Users are then able to access this glyph by typing the command `\rrBwp`, or, more intuitively, `\rr{Bwp}` (see also `\@join` detailed in Section 20.21.4).

The `\@xyz..` commands are used to generate the x, y, z, u, d, l, r, f, b commands and their associated prime rotation commands. The commands `\@xyzhbdfl` and `\@xyzbdflRubik` relate to the axis rotations denoted as b, d, f, l; since these four letters have long upstrokes they require special fine-tuning for vertical position.

The `\@SquareLetter` command is used to form the separate square hieroglyph form used for rotations with no visible representation from the front (e.g., B.., Fs, Fsp, Fa, Fap, S, Sp, Sf, Sfp, Sb, Sbp). Note that the TikZ 'thick' line code = 0.8pt (used in `\@SquareLetter`). The `\@hRubik` is the vertical shift used to raise the box carrying the rotation rotation-code in `\Rubik..` commands not visible from the front.

The idea is that by using these internal tools taking parameters we are able to more easily standardise the size and position of all the various glyphs. For details of the rubikfont and rubikprime see Section 20.3).

```
\newcommand{\@rr}[1]{{\@rubikfont #1}}
\newcommand{\@rrp}[1]{{\@rubikfont #1\@rubikprime}}
\newcommand{\@rrw}[1]{{\@rubikfont #1{\@rubikfontFNS w}}}
\newcommand{\@rrwp}[1]{{\@rubikfont #1{\@rubikfontFNS w}\@rubikprime}}
\newcommand{\@rrs}[1]{{\@rubikfont #1{\@rubikfontFNS s}}}
\newcommand{\@rrsp}[1]{{\@rubikfont #1{\@rubikfontFNS s}\@rubikprime}}
\newcommand{\@rra}[1]{{\@rubikfont #1{\@rubikfontFNS a}}}
\newcommand{\@rrap}[1]{{\@rubikfont #1{\@rubikfontFNS a}\@rubikprime}}
\newcommand{\@rru}[1]{{\@rubikfont #1{\@rubikfontFNS u}}}
\newcommand{\@rrup}[1]{{\@rubikfont #1{\@rubikfontFNS u}\@rubikprime}}
\newcommand{\@rrd}[1]{{\@rubikfont #1{\@rubikfontFNS d}}}
\newcommand{\@rrdp}[1]{{\@rubikfont #1{\@rubikfontFNS d}\@rubikprime}}
\newcommand{\@rrl}[1]{{\@rubikfont #1{\@rubikfontFNS l}}}
\newcommand{\@rrlp}[1]{{\@rubikfont #1{\@rubikfontFNS l}\@rubikprime}}
\newcommand{\@rrr}[1]{{\@rubikfont #1{\@rubikfontFNS r}}}
\newcommand{\@rrrp}[1]{{\@rubikfont #1{\@rubikfontFNS r}\@rubikprime}}
\newcommand{\@rrf}[1]{{\@rubikfont #1{\@rubikfontFNS f}}}
\newcommand{\@rrfp}[1]{{\@rubikfont #1{\@rubikfontFNS f}\@rubikprime}}
\newcommand{\@rrb}[1]{{\@rubikfont #1{\@rubikfontFNS b}}}
\newcommand{\@rrbp}[1]{{\@rubikfont #1{\@rubikfontFNS b}\@rubikprime}}
\newcommand{\@rrc}[1]{{\@rubikfont #1{\@rubikfontFNS c}}}
\newcommand{\@rrcp}[1]{{\@rubikfont #1{\@rubikfontFNS c}\@rubikprime}}
\newcommand{\@rrm}[1]{{\@rubikfont #1{\@rubikfontFNS m}}}
\newcommand{\@rrmp}[1]{{\@rubikfont #1{\@rubikfontFNS m}\@rubikprime}}
\newcommand{\@xyzh}[1]{[{\@rubikfont #1}]}
\newcommand{\@xyzhp}[1]{[{\@rubikfont #1\raisebox{-0.6pt}{\@rubikprime}}]}
\newcommand{\@xyzRubik}[1]%
   {\raisebox{3.45pt}{[{\@rubikfont #1}]}}
\newcommand{\@xyzRubikp}[1]%
   {\raisebox{3.45pt}{[{\@rubikfont #1\raisebox{-0.6pt}{\@rubikprime}}]}}
```

```
\newcommand{\@xyzhbdfl}[1]%
   {[\raisebox{-0.6pt}{{\@rubikfont #1}}]}
\newcommand{\@xyzhbdflp}[1]%
   {[\raisebox{-0.6pt}{{\@rubikfont #1\@rubikprime}}]}
\newcommand{\@xyzbdflRubik}[1]%
   {\raisebox{3.45pt}{[\raisebox{-0.6pt}{{\@rubikfont #1}}]}}
\newcommand{\@xyzbdflRubikp}[1]%
   {\raisebox{3.45pt}{[\raisebox{-0.6pt}{{\@rubikfont #1\@rubikprime}}]}}
\newcommand{\@SquareLetter}[1]{\setlength{\fboxsep}{2.5pt}%
   \setlength{\fboxrule}{0.8pt}%
   \fbox{\rule[-1pt]{0pt}{8.5pt}\raisebox{-0.5pt}{#1}}}
\newlength\@hRubik%
\setlength{\@hRubik}{0.185cm}%
```

`\@tlen` Feb 2017 (RWDN): We also need to define a small length for fine-tuning the default horizontal space between a pair of 'letter' hieroglyphs, eg B (i.e., when no additional space has been added by the user), so that this matches that between a pair of 'arrow' hieroglyphs. This length is inserted on both sides of the square frame. This length is used in two settings: (a) in 'letter' hieroglyphs (for an example, see the definition of the macro `\SquareB` in Section 20.21.5), and in (b) in 'arrow' hieroglyphs (for an example, see the definition of the macro `\rrhD` in Section 20.21.13).

```
\newcommand{\@tlen}{\hspace{1pt}}%
```

### 20.21.4 Using `\@join`

`\@join` We also require a macro for joining two strings so we can convert a rotation-code, say U, into a macro (say, `\rrhU`) which typesets it in some form. The following `\@join{}{}` command is by Christian Tellechea (many thanks!).
USAGE: `\@join{`⟨*command-stem*⟩`}{`⟨*rotation-code*⟩`}`. For example, to create the command `\rrhU` we would write `\@join{\rrh}{U}`, and hence the intuitive command `\rrh{U}` is equivalent to `\rrhU`.
Since this macro is also useful for processing rotation-codes in a list, which may also include macros, it is important that `#2` is not detokenized.

```
\newcommand*\@join[2]{%
    \csname\expandafter\@gobble\string#1#2\endcsname}
```

The following section shows how this command is used in practice.

`\textRubik` `\Rubik` `\rr` `\rrh` The following four commands typeset a single rotation, where a rotation-code (e.g., U) is the argument (see Section 9.1). As an example, the format for the `\rrh{}` command is `\rrh{`⟨*rotation-code*⟩`}`. In practice, these four commands are really a sort of front-end for all the commands which follow this section. For example, the command `\rrh{U}` generates the command `\rrhU` which itself typesets the rotation hieroglyph for the rotation U, etc.

These four commands, which use the internal `\@join` command (see above), are especially useful when typesetting a list of rotation-codes. Furthermore, it is

more intuitive for the user to specify a rotation command using the rotation-code as an argument.

```
\newcommand*{\Rubik}[1]{\@join{\Rubik}{#1}}
\newcommand*{\textRubik}[1]{\@join{\textRubik}{#1}}
\newcommand*{\rr}[1]{\@join{\rr}{#1}}
\newcommand*{\rrh}[1]{\@join{\rrh}{#1}}
```

### 20.21.5 Rotation B

\rrB
\SquareB
\rrhB
\RubikB
\textRubikB

These commands all draw forms which denote the B (BACK-face) rotation. Not visible from the front.

Feb 2017 (RWDN): added the `\@tlen` length (= 1pt; defined above) to the `\SquareB` command, and removed the terminal `\,` space from the rrhB, RubikB, textRubikB commands, and copied this action with all the subsequent Letter hieroglyphs (e.g., B, Bw,..). These minor changes were to improve the spacing between two Letter hieroglyphs, and make it match that between two square 'arrow' hieroglyphs. The same changes were made to all the 'letter' hieroglyphs.

```
\newcommand{\rrB}{\@rr{B}}
\newcommand{\SquareB}{\@tlen\@SquareLetter{\rrB}\@tlen}
\newcommand{\rrhB}{\raisebox{-0.25mm}{\SquareB}}
\newcommand{\RubikB}{\raisebox{\@hRubik}{\SquareB}}
\newcommand{\textRubikB}{\rrhB}
```

### 20.21.6 Rotation Bp

\rrBp
\SquareBp
\rrhBp
\RubikBp
\textRubikBp

These commands all draw forms which denote the Bp rotation. Not visible from the front.

```
\newcommand{\rrBp}{\@rrp{B}}
\newcommand{\SquareBp}{\@tlen\@SquareLetter{\rrBp}\@tlen}
\newcommand{\rrhBp}{\raisebox{-0.25mm}{\SquareBp}}
\newcommand{\RubikBp}{\raisebox{\@hRubik}{\SquareBp}}
\newcommand{\textRubikBp}{\rrhBp}
```

### 20.21.7 Rotation Bw

\rrBw
\SquareBw
\rrhBw
\RubikBw
\textRubikBw

These commands all draw forms which denote the Bw rotation. Not visible from the front.

```
\newcommand{\rrBw}{\@rrw{B}}
\newcommand{\SquareBw}{\@tlen\@SquareLetter{\rrBw}\@tlen}
\newcommand{\rrhBw}{\raisebox{-0.25mm}{\SquareBw}}
\newcommand{\RubikBw}{\raisebox{\@hRubik}{\SquareBw}}
\newcommand{\textRubikBw}{\rrhBw}
```

### 20.21.8 Rotation Bwp

\rrBwp
\SquareBwp
\rrhBwp
\RubikBwp
\textRubikBwp

These commands all draw forms which denote the Bwp rotation. Not visible from the front.

```
\newcommand{\rrBwp}{\@rrwp{B}}
\newcommand{\SquareBwp}{\@tlen\@SquareLetter{\rrBwp}\@tlen}
\newcommand{\rrhBwp}{\raisebox{-0.25mm}{\SquareBwp}}
\newcommand{\RubikBwp}{\raisebox{\@hRubik}{\SquareBwp}}
\newcommand{\textRubikBwp}{\rrhBwp}
```

### 20.21.9 Rotation Bs

\rrBs
\SquareBs
\rrhBs
\RubikBs
\textRubikBs

These commands all draw forms which denote the Bs rotation. Not visible from the front.

```
\newcommand{\rrBs}{\@rrs{B}}
\newcommand{\SquareBs}{\@tlen\@SquareLetter{\rrBs}\@tlen}
\newcommand{\rrhBs}{\raisebox{-0.25mm}{\SquareBs}}
\newcommand{\RubikBs}{\raisebox{\@hRubik}{\SquareBs}}
\newcommand{\textRubikBs}{\rrhBs}
```

### 20.21.10 Rotation Bsp

\rrBsp
\SquareBsp
\rrhBsp
\RubikBsp
\textRubikBsp

These commands all draw forms which denote the Bsp rotation. Not visible from the front.

```
\newcommand{\rrBsp}{\@rrsp{B}}
\newcommand{\SquareBsp}{\@tlen\@SquareLetter{\rrBsp}\@tlen}
\newcommand{\rrhBsp}{\raisebox{-0.25mm}{\SquareBsp}}
\newcommand{\RubikBsp}{\raisebox{\@hRubik}{\SquareBsp}}
\newcommand{\textRubikBsp}{\rrhBsp}
```

### 20.21.11 Rotation Ba

\rrBa
\SquareBa
\rrhBa
\RubikBa
\textRubikBa

These commands all draw forms which denote the Ba rotation. Not visible from the front.

```
\newcommand{\rrBa}{\@rra{B}}
\newcommand{\SquareBa}{\@tlen\@SquareLetter{\rrBa}\@tlen}
\newcommand{\rrhBa}{\raisebox{-0.25mm}{\SquareBa}}
\newcommand{\RubikBa}{\raisebox{\@hRubik}{\SquareBa}}
\newcommand{\textRubikBa}{\rrhBa}
```

### 20.21.12 Rotation Bap

\rrBap
\SquareBap
\rrhBap
\RubikBap
\textRubikBap

These commands all draw forms which denote the Bap rotation. Not visible from the front.

```
\newcommand{\rrBap}{\@rrap{B}}
\newcommand{\SquareBap}{\@tlen\@SquareLetter{\rrBap}\@tlen}
\newcommand{\rrhBap}{\raisebox{-0.25mm}{\SquareBap}}
\newcommand{\RubikBap}{\raisebox{\@hRubik}{\SquareBap}}
\newcommand{\textRubikBap}{\rrhBap}
```

### 20.21.13 Rotation D

\rrD
\SquareD
\rrhD
\RubikD
\textRubikD

These commands all draw forms which denote the D rotation.

Feb 2017 (RWDN): added the \@tlen length to the \rrhD command to improve the spacing between two 'arrow' square hieroglyphs; and also removed the terminal \, space. The same changes were made to all the 'arrow' hieroglyphs.

```
\newcommand{\rrD}{\@rr{D}}
%%
\newcommand{\SquareD}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhD}{\raisebox{-0.333\height}{\@tlen\SquareD\@tlen}}
%%
\newcommand{\RubikD}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareD\\
\rrD%
\end{minipage}%
}}
\newcommand{\textRubikD}{\rrD\,\rrhD}
```

### 20.21.14 Rotation Dp

\rrDp
\SquareDp
\rrhDp
\RubikDp
\textRubikDp

These commands all draw forms which denote the Dp rotation.

```
\newcommand{\rrDp}{\@rrp{D}}
%%
\newcommand{\SquareDp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick]     (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd)  -- (\@sbh, \@sdd);
\draw [thick, <-] (\@sb,\@sd)   -- (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDp}{\raisebox{-0.333\height}{\@tlen\SquareDp\@tlen}}
%%
\newcommand{\RubikDp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDp\\
\rrDp%
```

```
\end{minipage}%
}}
\newcommand{\textRubikDp}{\rrDp\,\rrhDp}
```

### 20.21.15 Rotation Dw

\rrDw
\SquareDw
\rrhDw
\RubikDw
\textRubikDw

These commands all draw forms which denote the Dw rotation.

```
\newcommand{\rrDw}{\@rrw{D}}
%%
\newcommand{\SquareDw}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick]     (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick, ->] (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->] (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDw}{\raisebox{-0.333\height}{\@tlen\SquareDw\@tlen}}
%%
\newcommand{\RubikDw}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDw\\
\rrDw%
\end{minipage}%
}}
\newcommand{\textRubikDw}{\rrDw\,\rrhDw}
```

### 20.21.16 Rotation Dwp

\rrDwp
\SquareDwp
\rrhDwp
\RubikDwp
\textRubikDwp

These commands all draw forms which denote the Dwp rotation.

```
\newcommand{\rrDwp}{\@rrwp{D}}
%%
\newcommand{\SquareDwp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick]     (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick, <-] (\@sb,\@sdd)  -- (\@sbh, \@sdd);
\draw [thick, <-] (\@sb,\@sd)   -- (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDwp}{\raisebox{-0.333\height}{\@tlen\SquareDwp\@tlen}}
%%
\newcommand{\RubikDwp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDwp\\
```

```
\rrDwp%
\end{minipage}%
}}
\newcommand{\textRubikDwp}{\rrDwp\,\rrhDwp}
```

### 20.21.17 Rotation Ds

\rrDs
\SquareDs
\rrhDs
\RubikDs
\textRubikDs

These commands all draw forms which denote the Ds rotation.

```
\newcommand{\rrDs}{\@rrs{D}}
%%
\newcommand{\SquareDs}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDs}{\raisebox{-0.333\height}{\@tlen\SquareDs\@tlen}}
%%
\newcommand{\RubikDs}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDs\\
\rrDs%
\end{minipage}%
}}
\newcommand{\textRubikDs}{\rrDs\,\rrhDs}
```

### 20.21.18 Rotation Dsp

\rrDsp
\SquareDsp
\rrhDsp
\RubikDsp
\textRubikDsp

These commands all draw forms which denote the Dsp rotation.

```
\newcommand{\rrDsp}{\@rrsp{D}}
%%
\newcommand{\SquareDsp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, <-]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDsp}{\raisebox{-0.333\height}{\@tlen\SquareDsp\@tlen}}
%%
\newcommand{\RubikDsp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
```

```
\SquareDsp\\
\rrDsp%
\end{minipage}%
}}
\newcommand{\textRubikDsp}{\rrDsp\,\rrhDsp}
```

### 20.21.19 Rotation Da

\rrDa
\SquareDa
\rrhDa
\RubikDa
\textRubikDa

These commands all draw forms which denote the Da rotation.

```
\newcommand{\rrDa}{\@rra{D}}
%%
\newcommand{\SquareDa}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDa}{\raisebox{-0.333\height}{\@tlen\SquareDa\@tlen}}
%%
\newcommand{\RubikDa}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDa\\
\rrDa%
\end{minipage}%
}}
\newcommand{\textRubikDa}{\rrDa\,\rrhDa}
```

### 20.21.20 Rotation Dap

\rrDap
\SquareDap
\rrhDap
\RubikDap
\textRubikDap

These commands all draw forms which denote the Dap rotation.

```
\newcommand{\rrDap}{\@rrap{D}}
%%
\newcommand{\SquareDap}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, <-]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhDap}{\raisebox{-0.333\height}{\@tlen\SquareDap\@tlen}}
%%
\newcommand{\RubikDap}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
```

```
\centering%
\SquareDap\\
\rrDap%
\end{minipage}%
}}
\newcommand{\textRubikDap}{\rrDap\,\rrhDap}
```

### 20.21.21 Rotation E

\rrE
\SquareE
\rrhE
\RubikE
\textRubikE

These commands all draw forms which denote the E rotation.

```
\newcommand{\rrE}{\@rr{E}}
%%
\newcommand{\SquareE}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick, ->]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhE}{\raisebox{-0.333\height}{\@tlen\SquareE\@tlen}}
%%
\newcommand{\RubikE}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrE%
\end{minipage}%
}}
\newcommand{\textRubikE}{\rrE\,\rrhE}
```

### 20.21.22 Rotation Ep

\rrEp
\SquareEp
\rrhEp
\RubikEp
\textRubikEp

These commands all draw forms which denote the Ep rotation.

```
\newcommand{\rrEp}{\@rrp{E}}
%%
\newcommand{\SquareEp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick, <-]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhEp}{\raisebox{-0.333\height}{\@tlen\SquareEp\@tlen}}
%%
\newcommand{\RubikEp}{%
{\@rubikfont%
```

```
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrEp%
\end{minipage}%
}}
\newcommand{\textRubikEp}{\rrEp\,\rrhEp}
```

### 20.21.23 Rotation F

\rrF \SquareF \rrhF \RubikF \textRubikF

These commands all draw forms which denote the F rotation.

```
\newcommand{\rrF}{\@rr{F}}
%%
\newcommand{\SquareF}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@scx, \@scy) arc[radius=0.35, start angle=-60, delta angle=290];
\end{tikzpicture}%
}
\newcommand{\rrhF}{\raisebox{-0.333\height}{\@tlen\SquareF\@tlen}}
%%
\newcommand{\RubikF}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareF\\
\rrF%
\end{minipage}%
}}
\newcommand{\textRubikF}{\rrF\,\rrhF}
```

### 20.21.24 Rotation Fp

\rrFp \SquareFp \rrhFp \RubikFp \textRubikFp

These commands all draw forms which denote the Fp rotation.

```
\newcommand{\rrFp}{\@rrp{F}}
%%
\newcommand{\SquareFp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@scx, \@scy) arc[radius=0.35, start angle=-60, delta angle=290];
\end{tikzpicture}%
}
\newcommand{\rrhFp}{\raisebox{-0.333\height}{\@tlen\SquareFp\@tlen}}
%%
\newcommand{\RubikFp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareFp\\
```

```
\rrFp%
\end{minipage}%
}}
\newcommand{\textRubikFp}{\rrFp\,\rrhFp}
```

### 20.21.25 Rotation Fw

\rrFw
\SquareFw
\rrhFw
\RubikFw
\textRubikFw

These commands all draw forms which denote the Fw rotation.

```
\newcommand{\rrFw}{\@rrw{F}}
%%
\newcommand{\SquareFw}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@scx, \@scy) arc[radius=0.35, start angle=-60, delta angle=290];
\draw [thick] (\@sqcx,\@sqcy) arc[radius=0.1, start angle=-60, delta angle=360];
%\node (squareLab) at (0.5,0.5)  {$o$};
\end{tikzpicture}%
}
\newcommand{\rrhFw}{\raisebox{-0.333\height}{\@tlen\SquareFw\@tlen}}
%%
\newcommand{\RubikFw}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareFw\\
\rrFw%
\end{minipage}%
}}
\newcommand{\textRubikFw}{\rrFw\,\rrhFw}
```

### 20.21.26 Rotation Fwp

\rrFwp
\SquareFwp
\rrhFwp
\RubikFwp
\textRubikFwp

These commands all draw forms which denote the Fwp rotation.

```
\newcommand{\rrFwp}{\@rrwp{F}}
%%
\newcommand{\SquareFwp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@scx, \@scy) arc[radius=0.35, start angle=-60, delta angle=290];
\draw [thick] (\@sqcx,\@sqcy) arc[radius=0.1, start angle=-60, delta angle=360];
\end{tikzpicture}%
}
\newcommand{\rrhFwp}{\raisebox{-0.333\height}{\@tlen\SquareFwp\@tlen}}
%%
\newcommand{\RubikFwp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareFwp\\
```

```
\rrFwp%
\end{minipage}%
}}
\newcommand{\textRubikFwp}{\rrFwp\,\rrhFwp}
%%
```

### 20.21.27 Rotation Fs

\rrFs
\rrhFs
\RubikFs
\textRubikFs

These commands draw forms of the Singmaster Fs slice rotation. We need to just make square with Fs in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFs}{\@rrs{F}}
\newcommand{\SquareFs}{\@tlen\@SquareLetter{\rrFs}\@tlen}
\newcommand{\rrhFs}{\raisebox{-0.25mm}{\SquareFs}}
\newcommand{\RubikFs}{\raisebox{\@hRubik}{\SquareFs}}
\newcommand{\textRubikFs}{\rrhFs}
```

### 20.21.28 Rotation Fsp

\rrFsp
\rrhFsp
\RubikFsp
\textRubikFsp

These commands draw forms of the Singmaster Fsp slice rotation. We need to just make square with Fsp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFsp}{\@rrsp{F}}
\newcommand{\SquareFsp}{\@tlen\@SquareLetter{\rrFsp}\@tlen}
\newcommand{\rrhFsp}{\raisebox{-0.25mm}{\SquareFsp}}
\newcommand{\RubikFsp}{\raisebox{\@hRubik}{\SquareFsp}}
\newcommand{\textRubikFsp}{\rrhFsp}
```

### 20.21.29 Rotation Fa

\rrFa
\rrhFa
\RubikFa
\textRubikFa

These commands draw forms of the Singmaster Fa slice rotation. We need to just make square with Fa in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFa}{\@rra{F}}
\newcommand{\SquareFa}{\@tlen\@SquareLetter{\rrFa}\@tlen}
\newcommand{\rrhFa}{\raisebox{-0.25mm}{\SquareFa}}
\newcommand{\RubikFa}{\raisebox{\@hRubik}{\SquareFa}}
\newcommand{\textRubikFa}{\rrhFa}
```

### 20.21.30 Rotation Fap

\rrFap
\rrhFap
\RubikFap
\textRubikFap

These commands draw forms of the Singmaster Fap slice rotation. We need to just make square with Fap in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFap}{\@rrap{F}}
\newcommand{\SquareFap}{\@tlen\@SquareLetter{\rrFap}\@tlen}
\newcommand{\rrhFap}{\raisebox{-0.25mm}{\SquareFap}}
\newcommand{\RubikFap}{\raisebox{\@hRubik}{\SquareFap}}
\newcommand{\textRubikFap}{\rrhFap}
```

### 20.21.31 Rotation L

\rrL
\SquareL
\rrhL
\RubikL
\textRubikL

These commands all draw forms which denote the L rotation.

```
\newcommand{\rrL}{\@rr{L}}
%%
\newcommand{\SquareL}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhL}{\raisebox{-0.333\height}{\@tlen\SquareL\@tlen}}
%%
\newcommand{\RubikL}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareL\\
\rrL%
\end{minipage}%
}}
\newcommand{\textRubikL}{\rrL\,\rrhL}
```

### 20.21.32 Rotation Lp

\rrLp
\SquareLp
\rrhLp
\RubikLp
\textRubikLp

These commands all draw forms which denote the Lp rotation.

```
\newcommand{\rrLp}{\@rrp{L}}
%%
\newcommand{\SquareLp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,->] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLp}{\raisebox{-0.333\height}{\@tlen\SquareLp\@tlen}}
%%
\newcommand{\RubikLp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
```

```
\centering%
\SquareLp\\
\rrLp%
\end{minipage}%
}}
\newcommand{\textRubikLp}{\rrLp\,\rrhLp}
```

### 20.21.33 Rotation Lw

\rrLw
\SquareLw
\rrhLw
\RubikLw
\textRubikLw

These commands all draw forms which denote the Lw rotation.

```
\newcommand{\rrLw}{\@rrw{L}}
%%
\newcommand{\SquareLw}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick, <-] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLw}{\raisebox{-0.333\height}{\@tlen\SquareLw\@tlen}}
%%
\newcommand{\RubikLw}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLw\\
\rrLw%
\end{minipage}%
}}
\newcommand{\textRubikLw}{\rrLw\,\rrhLw}
```

### 20.21.34 Rotation Lwp

\rrLwp
\SquareLwp
\rrhLwp
\RubikLwp
\textRubikLwp

These commands all draw forms which denote the Lwp rotation.

```
\newcommand{\rrLwp}{\@rrwp{L}}
%%
\newcommand{\SquareLwp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,->] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick,->] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLwp}{\raisebox{-0.333\height}{\@tlen\SquareLwp\@tlen}}
%%
\newcommand{\RubikLwp}{%
{\@rubikfont%
```

```
\begin{minipage}{0.6cm}
\centering%
\SquareLwp\\
\rrLwp%
\end{minipage}%
}}
\newcommand{\textRubikLwp}{\rrLwp\,\rrhLwp}
```

### 20.21.35 Rotation Ls

\rrLs
\SquareLs
\rrhLs
\RubikLs
\textRubikLs

These commands all draw forms which denote the Ls rotation.

```
\newcommand{\rrLs}{\@rrs{L}}
%%
\newcommand{\SquareLs}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, <-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLs}{\raisebox{-0.333\height}{\@tlen\SquareLs\@tlen}}
%%
\newcommand{\RubikLs}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLs\\
\rrLs%
\end{minipage}%
}}
\newcommand{\textRubikLs}{\rrLs\,\rrhLs}
```

### 20.21.36 Rotation Lsp

\rrLsp
\SquareLsp
\rrhLsp
\RubikLsp
\textRubikLsp

These commands all draw forms which denote the Lsp rotation.

```
\newcommand{\rrLsp}{\@rrsp{L}}
%%
\newcommand{\SquareLsp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, ->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLsp}{\raisebox{-0.333\height}{\@tlen\SquareLsp\@tlen}}
%%
\newcommand{\RubikLsp}{%
```

```
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLsp\\
\rrLsp%
\end{minipage}%
}}
\newcommand{\textRubikLsp}{\rrLsp\,\rrhLsp}
```

### 20.21.37 Rotation La

\rrLa
\SquareLa
\rrhLa
\RubikLa
\textRubikLa

These commands all draw forms which denote the La rotation.

```
\newcommand{\rrLa}{\@rra{L}}
%%
\newcommand{\SquareLa}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, ->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLa}{\raisebox{-0.333\height}{\@tlen\SquareLa\@tlen}}
%%
\newcommand{\RubikLa}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLa\\
\rrLa%
\end{minipage}%
}}
\newcommand{\textRubikLa}{\rrLa\,\rrhLa}
```

### 20.21.38 Rotation Lap

\rrLap
\SquareLap
\rrhLap
\RubikLap
\textRubikLap

These commands all draw forms which denote the Lap rotation.

```
\newcommand{\rrLap}{\@rrap{L}}
%%
\newcommand{\SquareLap}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, <-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhLap}{\raisebox{-0.333\height}{\@tlen\SquareLap\@tlen}}
%%
```

```
\newcommand{\RubikLap}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLap\\
\rrLap%
\end{minipage}%
}}
\newcommand{\textRubikLap}{\rrLap\,\rrhLap}
```

### 20.21.39 Rotation M

\rrM
\SquareM
\rrhM
\RubikM
\textRubikM

These commands all draw forms which denote the M rotation.

```
\newcommand{\rrM}{\@rr{M}}
%%
\newcommand{\SquareM}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick, <-] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhM}{\raisebox{-0.333\height}{\@tlen\SquareM\@tlen}}
%%
\newcommand{\RubikM}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrM%
\end{minipage}%
}}
\newcommand{\textRubikM}{\rrM\,\rrhM}
```

### 20.21.40 Rotation Mp

\rrMp
\SquareMp
\rrhMp
\RubikMp
\textRubikMp

These commands all draw forms which denote the Mp rotation.

```
\newcommand{\rrMp}{\@rrp{M}}
%%
\newcommand{\SquareMp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick,->] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhMp}{\raisebox{-0.333\height}{\@tlen\SquareMp\@tlen}}
```

```
%%
\newcommand{\RubikMp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrMp%
\end{minipage}%
}}
\newcommand{\textRubikMp}{\rrMp\,\rrhMp}
```

### 20.21.41 Rotation R

\rrR
\SquareR
\rrhR
\RubikR
\textRubikR

These commands all draw forms which denote the R rotation.

```
\newcommand{\rrR}{\@rr{R}}
%%
\newcommand{\SquareR}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
%% draw three lines in the square, one with an arrow
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, ->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhR}{\raisebox{-0.333\height}{\@tlen\SquareR\@tlen}}
%%
\newcommand{\RubikR}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareR\\
\rrR%
\end{minipage}%
}}
\newcommand{\textRubikR}{\rrR\,\rrhR}
```

### 20.21.42 Rotation Rp

\rrRp
\SquareRp
\rrhRp
\RubikRp
\textRubikRp

These commands all draw forms which denote the Rp rotation.

```
\newcommand{\rrRp}{\@rrp{R}}
%%
\newcommand{\SquareRp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, <-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
```

```
}
\newcommand{\rrhRp}{\raisebox{-0.333\height}{\@tlen\SquareRp\@tlen}}
%%
\newcommand{\RubikRp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRp\\
\rrRp%
\end{minipage}%
}}
\newcommand{\textRubikRp}{\rrRp\,\rrhRp}
```

### 20.21.43 Rotation Rw

\rrRw
\SquareRw
\rrhRw
\RubikRw
\textRubikRw

These commands all draw forms which denote the Rw rotation.

```
\newcommand{\rrRw}{\@rrw{R}}
%%
\newcommand{\SquareRw}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick, ->] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, ->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhRw}{\raisebox{-0.333\height}{\@tlen\SquareRw\@tlen}}
%%
\newcommand{\RubikRw}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRw\\
\rrRw%
\end{minipage}%
}}
\newcommand{\textRubikRw}{\rrRw\,\rrhRw}
```

### 20.21.44 Rotation Rwp

\rrRwp
\SquareRwp
\rrhRwp
\RubikRwp
\textRubikRwp

These commands all draw forms which denote the Rwp rotation.

```
\newcommand{\rrRwp}{\@rrwp{R}}
%%
\newcommand{\SquareRwp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick, <-] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick, <-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
```

```
\end{tikzpicture}%
}
\newcommand{\rrhRwp}{\raisebox{-0.333\height}{\@tlen\SquareRwp\@tlen}}
%%
\newcommand{\RubikRwp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRwp\\
\rrRwp%
\end{minipage}%
}}
\newcommand{\textRubikRwp}{\rrRwp\,\rrhRwp}
```

### 20.21.45 Rotation Rs

\rrRs
\SquareRs
\rrhRs
\RubikRs
\textRubikRs

These commands all draw forms which denote the Rs rotation.

```
\newcommand{\rrRs}{\@rrs{R}}
%%
\newcommand{\SquareRs}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,->] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick,->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhRs}{\raisebox{-0.333\height}{\@tlen\SquareRs\@tlen}}
%%
\newcommand{\RubikRs}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRs\\
\rrRs%
\end{minipage}%
}}
\newcommand{\textRubikRs}{\rrRs\,\rrhRs}
```

### 20.21.46 Rotation Rsp

\rrRsp
\SquareRsp
\rrhRsp
\RubikRsp
\textRubikRsp

These commands all draw forms which denote the Rsp rotation.

```
\newcommand{\rrRsp}{\@rrsp{R}}
%%
\newcommand{\SquareRsp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,<-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
```

```
\draw [thick,<-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhRsp}{\raisebox{-0.333\height}{\@tlen\SquareRsp\@tlen}}
%%
\newcommand{\RubikRsp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRsp\\
\rrRsp%
\end{minipage}%
}}
\newcommand{\textRubikRsp}{\rrRsp\,\rrhRsp}
```

### 20.21.47 Rotation Ra

\rrRa
\SquareRa
\rrhRa
\RubikRa
\textRubikRa

These commands all draw forms which denote the Ra rotation.

```
\newcommand{\rrRa}{\@rra{R}}
%%
\newcommand{\SquareRa}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,<-] (\@sd, \@sb) -- (\@sd, \@sbh);
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick,->] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhRa}{\raisebox{-0.333\height}{\@tlen\SquareRa\@tlen}}
%%
\newcommand{\RubikRa}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRa\\
\rrRa%
\end{minipage}%
}}
\newcommand{\textRubikRa}{\rrRa\,\rrhRa}
```

### 20.21.48 Rotation Rap

\rrRap
\SquareRap
\rrhRap
\RubikRap
\textRubikRap

These commands all draw forms which denote the Rap rotation.

```
\newcommand{\rrRap}{\@rrap{R}}
%%
\newcommand{\SquareRap}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick,->] (\@sd, \@sb) -- (\@sd, \@sbh);
```

```
\draw [thick] (\@sdd,\@sb) -- (\@sdd, \@sbh);
\draw [thick,<-] (\@sddd, \@sb) -- (\@sddd, \@sbh);
\end{tikzpicture}%
}
\newcommand{\rrhRap}{\raisebox{-0.333\height}{\@tlen\SquareRap\@tlen}}
%%
\newcommand{\RubikRap}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRap\\
\rrRap%
\end{minipage}%
}}
\newcommand{\textRubikRap}{\rrRap\,\rrhRap}
```

### 20.21.49 Rotation S

\rrS \SquareS \rrhS \RubikS \textRubikS

These commands all draw forms which denote the S rotation. Not visible from the front.

```
\newcommand{\rrS}{\@rr{S}}
\newcommand{\SquareS}{\@tlen\@SquareLetter{\rrS}\@tlen}
\newcommand{\rrhS}{\raisebox{-0.25mm}{\SquareS}}
\newcommand{\RubikS}{\raisebox{\@hRubik}{\SquareS}}
\newcommand{\textRubikS}{\rrhS}
```

### 20.21.50 Rotation Sp

\rrSp \SquareSp \rrhSp \RubikSp \textRubikSp

These commands all draw forms which denote the Sp rotation. Not visible from the front.

```
\newcommand{\rrSp}{\@rrp{S}}
\newcommand{\SquareSp}{\@tlen\@SquareLetter{\rrSp}\@tlen}
\newcommand{\rrhSp}{\raisebox{-0.25mm}{\SquareSp}}
\newcommand{\RubikSp}{\raisebox{\@hRubik}{\SquareSp}}
\newcommand{\textRubikSp}{\rrhSp}
```

### 20.21.51 Rotation Su

\rrSu \rrhSu \RubikSu \textRubikSu

These commands draw forms of the Singmaster Su slice rotation. We also need to fine-tune the spacing between these 'slice' hieroglyphs (especially Fs and Bs).

```
\newcommand{\rrSu}{\@rru{S}}
\newcommand{\rrhSu}{\rrhEp}%
\newcommand{\RubikSu}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrSu%
\end{minipage}%
```

```
}}
\newcommand{\textRubikSu}{\rrSu\,\rrhEp}
```

### 20.21.52 Rotation Sup

\rrSup
\rrhSup
\RubikSup
\textRubikSup

These commands draw forms of the Singmaster Sup slice rotation. We also need to fine-tune the spacing between these 'slice' hieroglyphs (especially Fs and Bs).

```
\newcommand{\rrSup}{\@rrup{S}}
\newcommand{\rrhSup}{\rrhE}%
\newcommand{\RubikSup}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrSup%
\end{minipage}%
}}
\newcommand{\textRubikSup}{\rrSup\,\rrhE}
```

### 20.21.53 Rotation Sd

\rrSd
\rrhSd
\RubikSd
\textRubikSd

These commands draw forms of the Singmaster Sd slice rotation.

```
\newcommand{\rrSd}{\@rrd{S}}
\newcommand{\rrhSd}{\rrhE}%
\newcommand{\RubikSd}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrSd%
\end{minipage}%
}}
\newcommand{\textRubikSd}{\rrSd\,\rrhE}
```

### 20.21.54 Rotation Sdp

\rrSdp
\rrhSdp
\RubikSdp
\textRubikSdp

These commands draw forms of the Singmaster Sdp slice rotation.

```
\newcommand{\rrSdp}{\@rrdp{S}}
\newcommand{\rrhSdp}{\rrhEp}%
\newcommand{\RubikSdp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrSdp%
\end{minipage}%
}}
\newcommand{\textRubikSdp}{\rrSdp\,\rrhEp}
```

### 20.21.55 Rotation Sl

\rrSl
\rrhSl
\RubikSl
\textRubikSl

These commands draw forms of the Singmaster Sl slice rotation.

```
\newcommand{\rrSl}{\@rrl{S}}
\newcommand{\rrhSl}{\rrhM}%
\newcommand{\RubikSl}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrSl%
\end{minipage}%
}}
\newcommand{\textRubikSl}{\rrSl\,\rrhM}
```

### 20.21.56 Rotation Slp

\rrSlp
\rrhSlp
\RubikSlp
\textRubikSlp

These commands draw forms of the Singmaster Slp slice rotation.

```
\newcommand{\rrSlp}{\@rrlp{S}}
\newcommand{\rrhSlp}{\rrhMp}%
\newcommand{\RubikSlp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrSlp%
\end{minipage}%
}}
\newcommand{\textRubikSlp}{\rrSlp\,\rrhMp}
```

### 20.21.57 Rotation Sr

\rrSr
\rrhSr
\RubikSr
\textRubikSr

These commands draw forms of the Singmaster Sr slice rotation.

```
\newcommand{\rrSr}{\@rrr{S}}
\newcommand{\rrhSr}{\rrhMp}%
\newcommand{\RubikSr}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrSr%
\end{minipage}%
}}
\newcommand{\textRubikSr}{\rrSr\,\rrhMp}
```

### 20.21.58 Rotation Srp

\rrSrp
\rrhSrp
\RubikSrp
\textRubikSrp

These commands draw forms of the Singmaster Srp slice rotation.

```
\newcommand{\rrSrp}{\@rrrp{S}}
```

```
\newcommand{\rrhSrp}{\rrhM}%
\newcommand{\RubikSrp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrSrp%
\end{minipage}%
}}
\newcommand{\textRubikSrp}{\rrSrp\,\rrhM}
```

### 20.21.59 Rotation Sf

\rrSf
\rrhSf
\RubikSf
\textRubikSf

These commands draw forms of the Singmaster Sf slice rotation. We need to just make square with Sf in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSf}{\@rrf{S}}
\newcommand{\SquareSf}{\@tlen\@SquareLetter{\rrSf}\@tlen}
\newcommand{\rrhSf}{\raisebox{-0.25mm}{\SquareSf}}
\newcommand{\RubikSf}{\raisebox{\@hRubik}{\SquareSf}}
\newcommand{\textRubikSf}{\rrhSf}
```

### 20.21.60 Rotation Sfp

\rrSfp
\rrhSfp
\RubikSfp
\textRubikSfp

These commands draw forms of the Singmaster Sfp slice rotation. We need to just make square with Sfp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSfp}{\@rrfp{S}}
\newcommand{\SquareSfp}{\@tlen\@SquareLetter{\rrSfp}\@tlen}
\newcommand{\rrhSfp}{\raisebox{-0.25mm}{\SquareSfp}}
\newcommand{\RubikSfp}{\raisebox{\@hRubik}{\SquareSfp}}
\newcommand{\textRubikSfp}{\rrhSfp}
```

### 20.21.61 Rotation Sb

\rrSb
\rrhSb
\RubikSb
\textRubikSb

These commands draw forms of the Singmaster Sb slice rotation. We need to just make square with Sb in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSb}{\@rrb{S}}
\newcommand{\SquareSb}{\@tlen\@SquareLetter{\rrSb}\@tlen}
\newcommand{\rrhSb}{\raisebox{-0.25mm}{\SquareSb}}
\newcommand{\RubikSb}{\raisebox{\@hRubik}{\SquareSb}}
\newcommand{\textRubikSb}{\rrhSb}
```

### 20.21.62 Rotation Sbp

\rrSbp
\rrhSbp
\RubikSbp
\textRubikSbp

These commands draw forms of the Singmaster Sbp slice rotation. We need to just make square with Sbp in square; adjust box height using a `\rule`; adjust `\fboxsep` (default=3pt); adjust `\fboxrule` (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from th front.

```
\newcommand{\rrSbp}{\@rrbp{S}}
\newcommand{\SquareSbp}{\@tlen\@SquareLetter{\rrSbp}\@tlen}
\newcommand{\rrhSbp}{\raisebox{-0.25mm}{\SquareSbp}}
\newcommand{\RubikSbp}{\raisebox{\@hRubik}{\SquareSbp}}
\newcommand{\textRubikSbp}{\rrhSbp}
```

### 20.21.63 Rotation U

\rrU
\SquareU
\rrhU
\RubikU
\textRubikU

These commands all draw forms which denote the U rotation.

```
\newcommand{\rrU}{\@rr{U}}
%%
\newcommand{\SquareU}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhU}{\raisebox{-0.333\height}{\@tlen\SquareU\@tlen}}
%%
\newcommand{\RubikU}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareU\\
\rrU%
\end{minipage}%%
}}
\newcommand{\textRubikU}{\rrU\,\rrhU}
```

### 20.21.64 Rotation Uw

\rrUw
\SquareUw
\rrhUw
\RubikUw
\textRubikUw

These commands all draw forms which denote the Uw rotation.

```
\newcommand{\rrUw}{\@rrw{U}}
%%
\newcommand{\SquareUw}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick, <-] (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
```

```
}
%
\newcommand{\rrhUw}{\raisebox{-0.333\height}{\@tlen\SquareUw\@tlen}}
%%
\newcommand{\RubikUw}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUw\\
\rrUw%
\end{minipage}%%
}}
%%
\newcommand{\textRubikUw}{\rrUw\,\rrhUw}
```

### 20.21.65 Rotation Up

\rrUp
\SquareUp
\rrhUp
\RubikUp
\textRubikUp

These commands all draw forms which denote the Up rotation.

```
\newcommand{\rrUp}{\@rrp{U}}
%%
\newcommand{\SquareUp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUp}{\raisebox{-0.333\height}{\@tlen\SquareUp\@tlen}}
%%
\newcommand{\RubikUp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUp\\
\rrUp%
\end{minipage}%%
}}
\newcommand{\textRubikUp}{\rrUp\,\rrhUp}
```

### 20.21.66 Rotation Uwp

\rrUwp
\SquareUwp
\rrhUwp
\RubikUwp
\textRubikUwp

These commands all draw forms which denote the Uwp rotation.

```
\newcommand{\rrUwp}{\@rrwp{U}}
%%
\newcommand{\SquareUwp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
```

```
\draw [thick, ->] (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUwp}{\raisebox{-0.333\height}{\@tlen\SquareUwp\@tlen}}
%%
\newcommand{\RubikUwp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUwp\\
\rrUwp%
\end{minipage}%%
}}
\newcommand{\textRubikUwp}{\rrUwp\,\rrhUwp}
```

### 20.21.67 Rotation Us

\rrUs
\SquareUs
\rrhUs
\RubikUs
\textRubikUs

These commands all draw forms which denote the Us rotation.

```
\newcommand{\rrUs}{\@rrs{U}}
%%
\newcommand{\SquareUs}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, <-]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUs}{\raisebox{-0.333\height}{\@tlen\SquareUs\@tlen}}
%%
\newcommand{\RubikUs}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUs\\
\rrUs%
\end{minipage}%
}}
\newcommand{\textRubikUs}{\rrUs\,\rrhUs}
```

### 20.21.68 Rotation Usp

\rrUsp
\SquareUsp
\rrhUs
\RubikUs
\textRubikUsp

These commands all draw forms which denote the Usp rotation.

```
\newcommand{\rrUsp}{\@rrsp{U}}
%%
\newcommand{\SquareUsp}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
```

```
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUsp}{\raisebox{-0.333\height}{\@tlen\SquareUsp\@tlen}}
%%
\newcommand{\RubikUsp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUsp\\
\rrUsp%
\end{minipage}%
}}
\newcommand{\textRubikUsp}{\rrUsp\,\rrhUsp}
```

### 20.21.69 Rotation Ua

\rrUa
\SquareUa
\rrhUa
\RubikUa
\textRubikUa

These commands all draw forms which denote the Ua rotation.

```
\newcommand{\rrUa}{\@rra{U}}
%%
\newcommand{\SquareUa}{%
\begin{tikzpicture}[scale=0.5]
\DrawNotationBox;
\draw [thick, <-] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, ->]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUa}{\raisebox{-0.333\height}{\@tlen\SquareUa\@tlen}}
%%
\newcommand{\RubikUa}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUa\\
\rrUa%
\end{minipage}%
}}
\newcommand{\textRubikUa}{\rrUa\,\rrhUa}
```

### 20.21.70 Rotation Uap

\rrUap
\SquareUap
\rrhUap
\RubikUap
\textRubikUap

These commands all draw forms which denote the Uap rotation.

```
\newcommand{\rrUap}{\@rrap{U}}
%%
\newcommand{\SquareUap}{%
\begin{tikzpicture}[scale=0.5]
```

```
\DrawNotationBox;
\draw [thick, ->] (\@sb,\@sddd) -- (\@sbh, \@sddd);
\draw [thick]     (\@sb,\@sdd) --  (\@sbh, \@sdd);
\draw [thick, <-]     (\@sb,\@sd) --   (\@sbh, \@sd);
\end{tikzpicture}%
}
\newcommand{\rrhUap}{\raisebox{-0.333\height}{\@tlen\SquareUap\@tlen}}
%%
\newcommand{\RubikUap}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUap\\
\rrUap%
\end{minipage}%
}}
\newcommand{\textRubikUap}{\rrUap\,\rrhUap}
```

### 20.21.71 Rotations x and xp

\rrx
\rrhx
\Rubikx

These commands all draw forms which denote the x rotation.

```
\newcommand{\rrx}{\@rr{x}}
\newcommand{\Rubikx}{\@xyzRubik{x}}
\newcommand{\rrhx}{\@xyzh{x}}
```

\rrxp
\rrhxp
\Rubikxp

These commands all draw forms which denote the xp rotation.

```
\newcommand{\rrxp}{\@rrp{x}}
\newcommand{\Rubikxp}{\@xyzRubikp{x}}
\newcommand{\rrhxp}{\@xyzhp{x}}
```

### 20.21.72 Rotations y and yp

\rry
\rrhy
\Rubiky

These commands all draw forms which denote the y rotation.

```
\newcommand{\rry}{\@rr{y}}
\newcommand{\Rubiky}{\@xyzRubik{y}}
\newcommand{\rrhy}{\@xyzh{y}}
```

\rryp
\rrhyp
\Rubikyp

These commands all draw forms which denote the yp rotation.

```
\newcommand{\rryp}{\@rrp{y}}
\newcommand{\Rubikyp}{\@xyzRubikp{y}}
\newcommand{\rrhyp}{\@xyzhp{y}}
```

### 20.21.73 Rotations z and zp

\rrz
\rrhz
\Rubikz

These commands all draw forms which denote the z rotation.

```
\newcommand{\rrz}{\@rr{z}}
\newcommand{\Rubikz}{\@xyzRubik{z}}
\newcommand{\rrhz}{\@xyzh{z}}
```

\rrzp
\rrhzp
\Rubikzp

These commands all draw forms which denote the zp rotation.

```
\newcommand{\rrzp}{\@rrp{z}}
\newcommand{\Rubikzp}{\@xyzRubikp{z}}
\newcommand{\rrhzp}{\@xyzhp{z}}
```

### 20.21.74 Rotations u and d

\rru
\rrhu
\Rubiku

These commands all draw forms which denote the u and up cube rotation.

```
\newcommand{\rru}{\@rr{u}}
\newcommand{\rrup}{\@rrp{u}}
\newcommand{\rrhu}{\@xyzh{u}}
\newcommand{\rrhup}{\@xyzhp{u}}
\newcommand{\Rubiku}{\@xyzRubik{u}}
\newcommand{\Rubikup}{\@xyzRubikp{u}}
```

\rrd
\rrhd
\Rubikd

These commands all draw forms which denote the d and dp cube rotation.

```
\newcommand{\rrd}{\@rr{d}}
\newcommand{\rrdp}{\@rrp{d}}
\newcommand{\rrhd}{\@xyzhbdfl{d}}
\newcommand{\rrhdp}{\@xyzhbdflp{d}}
\newcommand{\Rubikd}{\@xyzbdflRubik{d}}
\newcommand{\Rubikdp}{\@xyzbdflRubikp{d}}
```

### 20.21.75 Rotations l and r

\rrl
\rrhl
\Rubikl

These commands all draw forms which denote the l and lp cube rotation.

```
\newcommand{\rrl}{\@rr{l}}
\newcommand{\rrlp}{\@rrp{l}}
\newcommand{\rrhl}{\@xyzhbdfl{l}}
\newcommand{\rrhlp}{\@xyzhbdflp{l}}
\newcommand{\Rubikl}{\@xyzbdflRubik{l}}
\newcommand{\Rubiklp}{\@xyzbdflRubikp{l}}
```

\rrr
\rrhr
\Rubikr

These commands all draw forms which denote the r and rp cube rotation.

```
\newcommand{\rrr}{\@rr{r}}
\newcommand{\rrrp}{\@rrp{r}}
\newcommand{\rrhr}{\@xyzh{r}}
\newcommand{\rrhrp}{\@xyzhp{r}}
\newcommand{\Rubikr}{\@xyzRubik{r}}
\newcommand{\Rubikrp}{\@xyzRubikp{r}}
```

### 20.21.76 Rotations f and b

\rrf
\rrhf
\Rubikf

These commands all draw forms which denote the f and fp cube rotation.

```
\newcommand{\rrf}{\@rr{f}}
\newcommand{\rrfp}{\@rrp{f}}
\newcommand{\rrhf}{\@xyzhbdfl{f}}
\newcommand{\rrhfp}{\@xyzhbdflp{f}}
```

```
\newcommand{\Rubikf}{\@xyzbdflRubik{f}}
\newcommand{\Rubikfp}{\@xyzbdflRubikp{f}}
```

\rrb
\rrhb
\Rubikb

These commands all draw forms which denote the b and bp cube rotation.

```
\newcommand{\rrb}{\@rr{b}}
\newcommand{\rrbp}{\@rrp{b}}
\newcommand{\rrhb}{\@xyzhbdfl{b}}
\newcommand{\rrhbp}{\@xyzhbdflp{b}}
\newcommand{\Rubikb}{\@xyzbdflRubik{b}}
\newcommand{\Rubikbp}{\@xyzbdflRubikp{b}}
```

## 20.22 Face(c) and Face(m) rotations

This subsection (added Nov 2016) is to accommodate some additional notation, much used on the Jaap Puzzles website (Scherphius J), which, although quite convenient, is technically 'non-standard'. This additional notation makes available the (middle slice) rotations of the form Rm, Rmp, Lm, Lmp, ... and also the (whole cube) rotations Rc, Lc, ... (referenced to a face) as used on the Cube Lovers usenet group (1981–1997). This notation was probably invented by Singmaster (see Scherphius J).

\@xyzhc
\@xyzhcp
\@xyzRubikc
\@xyzRubikcp

First we have some useful facilitating commands we shall make use of in conjunction with the Face(c) and Face(cp) notation.

```
\newcommand{\@xyzhc}[1]{[\raisebox{-1.2pt}%
   {{\@rubikfont #1\@rubikfontFNS c}}]}
\newcommand{\@xyzhcp}[1]{[\raisebox{-1.2pt}%
   {{\@rubikfont #1{\@rubikfontFNS c}\@rubikprime}}]}
\newcommand{\@xyzRubikc}[1]{\raisebox{3.45pt}%
   {[\raisebox{-1.2pt}{{\@rubikfont #1\@rubikfontFNS c}}]}}
\newcommand{\@xyzRubikcp}[1]{\raisebox{3.45pt}%
   {[\raisebox{-1.2pt}{{\@rubikfont #1{\@rubikfontFNS c}\@rubikprime}}]}}
```

### 20.22.1 Rotations Rc and Rcp

Whole cube rotations Rc = x, Rcp = xp.

\rrRc
\rrhRc
\RubikRc

These commands all draw forms which denote the Rc rotation.

```
\newcommand{\rrRc}{\@rrc{R}}
\newcommand{\RubikRc}{\@xyzRubikc{R}}
\newcommand{\rrhRc}{\@xyzhc{R}}
```

\rrRcp
\rrhRcp
\RubikRcp

These commands all draw forms which denote the Rcp rotation.

```
\newcommand{\rrRcp}{\@rrcp{R}}
\newcommand{\RubikRcp}{\@xyzRubikcp{R}}
\newcommand{\rrhRcp}{\@xyzhcp{R}}
```

### 20.22.2 Rotations Lc and Lcp

Whole cube rotations Lc = xp, Lcp = x.

\rrLc These commands all draw forms which denote the Lc rotation.
\rrhLc
\RubikLc
```
\newcommand{\rrLc}{\@rrc{L}}
\newcommand{\RubikLc}{\@xyzRubikc{L}}
\newcommand{\rrhLc}{\@xyzhc{L}}
```

\rrLcp These commands all draw forms which denote the Lcp rotation.
\rrhLcp
\RubikLcp
```
\newcommand{\rrLcp}{\@rrcp{L}}
\newcommand{\RubikLcp}{\@xyzRubikcp{L}}
\newcommand{\rrhLcp}{\@xyzhcp{L}}
```

### 20.22.3 Rotations Uc and Ucp

Whole cube rotation Uc = y, Ucp=yp.

\rrUc These commands all draw forms which denote the Uc rotation.
\rrhUc
\RubikUc
```
\newcommand{\rrUc}{\@rrc{U}}
\newcommand{\RubikUc}{\@xyzRubikc{U}}
\newcommand{\rrhUc}{\@xyzhc{U}}
```

\rrUcp These commands all draw forms which denote the Ucp rotation.
\rrhUcp
\RubikUcp
```
\newcommand{\rrUcp}{\@rrcp{U}}
\newcommand{\RubikUcp}{\@xyzRubikcp{U}}
\newcommand{\rrhUcp}{\@xyzhcp{U}}
```

### 20.22.4 Rotations Dc and Dcp

Whole cube rotations Dc = yp, Dcp = y.

\rrDc These commands all draw forms which denote the Dc rotation.
\rrhDc
\RubikDc
```
\newcommand{\rrDc}{\@rrc{D}}
\newcommand{\RubikDc}{\@xyzRubikc{D}}
\newcommand{\rrhDc}{\@xyzhc{D}}
```

\rrDcp These commands all draw forms which denote the Dcp rotation.
\rrhDcp
\RubikDcp
```
\newcommand{\rrDcp}{\@rrcp{D}}
\newcommand{\RubikDcp}{\@xyzRubikcp{D}}
\newcommand{\rrhDcp}{\@xyzhcp{D}}
```

### 20.22.5 Rotations Fc and Fcp

Whole cube rotations Fc = z, Fcp = zp.

\rrFc These commands all draw forms which denote the Fc rotation.
\rrhFc
\RubikFc
```
\newcommand{\rrFc}{\@rrc{F}}
\newcommand{\RubikFc}{\@xyzRubikc{F}}
\newcommand{\rrhFc}{\@xyzhc{F}}
```

\rrFcp
\rrhFcp
\RubikFcp

These commands all draw forms which denote the Fcp rotation.

```
\newcommand{\rrFcp}{\@rrcp{F}}
\newcommand{\RubikFcp}{\@xyzRubikcp{F}}
\newcommand{\rrhFcp}{\@xyzhcp{F}}
```

### 20.22.6 Rotations Bc and Bcp

Whole cube rotation Bc = zp, Bcp = z.

\rrBc
\rrhBc
\RubikBc

These commands all draw forms which denote the Bc rotation.

```
\newcommand{\rrBc}{\@rrc{B}}
\newcommand{\RubikBc}{\@xyzRubikc{B}}
\newcommand{\rrhBc}{\@xyzhc{B}}
```

\rrBcp
\rrhBcp
\RubikBcp

These commands all draw forms which denote the Bcp rotation.

```
\newcommand{\rrBcp}{\@rrcp{B}}
\newcommand{\RubikBcp}{\@xyzRubikcp{B}}
\newcommand{\rrhBcp}{\@xyzhcp{B}}
```

### 20.22.7 Rotations Rm and Rmp

Rm = Mp= Sr (M follows Left). We use Sr and Srp as the templates.

\rrRm
\rrhRm
\RubikRm
\textRubikRm

These commands draw forms of the Rm slice rotation.

```
\newcommand{\rrRm}{\@rrm{R}}
\newcommand{\rrhRm}{\rrhMp}%
\newcommand{\RubikRm}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrRm%
\end{minipage}%
}}
\newcommand{\textRubikRm}{\rrRm\,\rrhMp}
```

\rrRmp
\rrhRmp
\RubikRmp
\textRubikRmp

These commands draw forms of the Rmp slice rotation.

```
\newcommand{\rrRmp}{\@rrmp{R}}
\newcommand{\rrhRmp}{\rrhM}%
\newcommand{\RubikRmp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrRmp%
\end{minipage}%
}}
\newcommand{\textRubikRmp}{\rrRmp\,\rrhM}
```

### 20.22.8 Rotations Lm and Lmp

Lm = M = Sl (M follows Left). We use Sl and Slp as the templates.

\rrLm
\rrhLm
\RubikLm
\textRubikLm

These commands draw forms of the Lm slice rotation.

```
\newcommand{\rrLm}{\@rrm{L}}
\newcommand{\rrhLm}{\rrhM}%
\newcommand{\RubikLm}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrLm%
\end{minipage}%
}}
\newcommand{\textRubikLm}{\rrLm\,\rrhM}
```

\rrLmp
\rrhLmp
\RubikLmp
\textRubikLmp

These commands draw forms of the Lmp slice rotation.

```
\newcommand{\rrLmp}{\@rrmp{L}}
\newcommand{\rrhLmp}{\rrhMp}%
\newcommand{\RubikLmp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrLmp%
\end{minipage}%
}}
\newcommand{\textRubikLmp}{\rrLmp\,\rrhMp}
```

### 20.22.9 Rotations Um and Ump

Um = Ep = Su (E follows Down). We use Su and Sup as the templates.

\rrUm
\rrhUm
\RubikUm
\textRubikUm

These commands draw forms of the Um slice rotation. We also need to fine-tune the spacing between these 'slice' hieroglyphs (especially Fs and Bs).

```
\newcommand{\rrUm}{\@rrm{U}}
\newcommand{\rrhUm}{\rrhEp}%
\newcommand{\RubikUm}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrUm%
\end{minipage}%
}}
\newcommand{\textRubikUm}{\rrUm\,\rrhEp}
```

\rrUmp
\rrhUmp
\RubikUmp
\textRubikUmp

These commands draw forms of the Ump slice rotation. We also need to fine-tune the spacing between these 'slice' hieroglyphs (especially Fs and Bs).

```
\newcommand{\rrUmp}{\@rrmp{U}}
\newcommand{\rrhUmp}{\rrhE}%
\newcommand{\RubikUmp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrUmp%
\end{minipage}%
}}
\newcommand{\textRubikUmp}{\rrUmp\,\rrhE}
```

### 20.22.10 Rotations Dm and Dmp

Dm = E = Sd (E follows Down). We use Sd and Sdp as the templates.

\rrDm
\rrhDm
\RubikDm
\textRubikDm

These commands draw forms of the Singmaster Dm slice rotation.

```
\newcommand{\rrDm}{\@rrm{D}}
\newcommand{\rrhDm}{\rrhE}%
\newcommand{\RubikDm}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrDm%
\end{minipage}%
}}
\newcommand{\textRubikDm}{\rrDm\,\rrhE}
```

\rrDmp
\rrhDmp
\RubikDmp
\textRubikDmp

These commands draw forms of the Singmaster Dmp slice rotation.

```
\newcommand{\rrDmp}{\@rrmp{D}}
\newcommand{\rrhDmp}{\rrhEp}%
\newcommand{\RubikDmp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrDmp%
\end{minipage}%
}}
\newcommand{\textRubikDmp}{\rrDmp\,\rrhEp}
```

### 20.22.11 Rotations Fm and Fmp

Fm = S = Sf (S follows Front). S is not visible from the front, so is represented in a square box. We use Sf and Sfp as the templates.

\rrFm
\rrhFm
\RubikFm
\textRubikFm

These commands draw forms of the Fm slice rotation. We need to just make square with Fm in square; adjust box height using a `\rule`; adjust `\fboxsep`

(default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFm}{\@rrm{F}}
\newcommand{\SquareFm}{\@tlen\@SquareLetter{\rrFm}\@tlen}
\newcommand{\rrhFm}{\raisebox{-0.25mm}{\SquareFm}}
\newcommand{\RubikFm}{\raisebox{\@hRubik}{\SquareFm}}
\newcommand{\textRubikFm}{\rrhFm}
```

\rrFmp
\rrhFmp
\RubikFmp
\textRubikFmp

These commands draw forms of the Fmp slice rotation. We need to just make square with Fmp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrFmp}{\@rrmp{F}}
\newcommand{\SquareFmp}{\@tlen\@SquareLetter{\rrFmp}\@tlen}
\newcommand{\rrhFmp}{\raisebox{-0.25mm}{\SquareFmp}}
\newcommand{\RubikFmp}{\raisebox{\@hRubik}{\SquareFmp}}
\newcommand{\textRubikFmp}{\rrhFmp}
```

### 20.22.12 Rotations Bm and Bmp

Bm = Sp = Sb. We use Sb and Sbp as the templates.

\rrBm
\rrhBm
\RubikBm
\textRubikBm

These commands draw forms of the Bm slice rotation. We need to just make square with Bm in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrBm}{\@rrm{B}}
\newcommand{\SquareBm}{\@tlen\@SquareLetter{\rrBm}\@tlen}
\newcommand{\rrhBm}{\raisebox{-0.25mm}{\SquareBm}}
\newcommand{\RubikBm}{\raisebox{\@hRubik}{\SquareBm}}
\newcommand{\textRubikBm}{\rrhBm}
```

\rrBmp
\rrhBmp
\RubikBmp
\textRubikBmp

These commands draw forms of the Bmp slice rotation. We need to just make square with Bmp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrBmp}{\@rrmp{B}}
\newcommand{\SquareBmp}{\@tlen\@SquareLetter{\rrBmp}\@tlen}
\newcommand{\rrhBmp}{\raisebox{-0.25mm}{\SquareBmp}}
\newcommand{\RubikBmp}{\raisebox{\@hRubik}{\SquareBmp}}
\newcommand{\textRubikBmp}{\rrhBmp}
```

## 20.23 Randelshofer superset ENG rotations (CMST)

(see webpage: Randelshofer.ch/rubik/patterns/doc/supersetENG_3x3.html)

This section relates to the recent Rubik notation developed by Randelshofer (see URL above), known as the "superset ENG rotations" (see Sections 9 and 9.9.1 for details).

I denote this notation as "CMST" (i.e., his C, M, S and T notation). Although this notation replicates previously existing World Cube Association notation, we include it here for completeness. Fortunately the Randelshofer notation does not conflict with existing notation.

Briefly, the CMST letters denote whole Cube, Middle, outer-Slice and wide (T) rotations associated with named faces. For example, the rotation-code MR indicates a rotation of the middle slice parallel to the RIGHT face, and in the same sense (direction) as **R** (see Section 9 for details).

First we make some useful facilitating commands for the CX1 and CX1p forms as these are handled with raised square brackets. We model these four macros on the equivalent `\@xyzh`, `\@xyzhp`,`\@xyzRubik`,`\@xyzRubikp` macros defined above.

```
\newcommand{\@randhc}[1]{[\raisebox{-1.2pt}{{\@rubikfont C#1}}]}
\newcommand{\@randhcp}[1]{[\raisebox{-1.2pt}{{\@rubikfont C#1\@rubikprime}}]}
\newcommand{\@randRubikc}[1]%
   {\raisebox{3.45pt}{[\raisebox{-1.2pt}{{\@rubikfont C#1}}]}}
\newcommand{\@randRubikcp}[1]%
   {\raisebox{3.45pt}{[\raisebox{-1.2pt}{{\@rubikfont C#1\@rubikprime}}]}}
```

In the following we create the base form of a rotation code, eg [**CR**] and then the user typesets this rotation using CR as the argument for `\rrh{}`. This works because the macro `\@join{}{}` is used to join `\rrh` and CR → `\rrhCR` etc.

The = signs at the beginning of each new section indicate the equivalence between the Randelshofer notation and the current (World Cube Association) notation.

### 20.23.1 Rotations CR and CRp

Whole cube rotations CR = Rc = x, CRp = Rcp = xp.

`\rrCR` `\rrhCR` `\RubikCR` These commands all draw forms which denote the CR = Rc rotation.

```
\newcommand{\rrCR}{\@rr{CR}}
\newcommand{\RubikCR}{\@randRubikc{R}}
\newcommand{\rrhCR}{\@randhc{R}}
```

`\rrCRp` `\rrhCRp` `\RubikCRp` These commands all draw forms which denote the CRp = Rcp rotation.

```
\newcommand{\rrCRp}{\@rrp{CR}}
\newcommand{\RubikCRp}{\@randRubikcp{R}}
\newcommand{\rrhCRp}{\@randhcp{R}}
```

### 20.23.2 Rotations CL and CLp

Whole cube rotations CL = Lc = xp, CLp = Lcp = x.

`\rrCL` `\rrhCL` `\RubikCL` These commands all draw forms which denote the CL rotation.

```
\newcommand{\rrCL}{\@rr{CL}}
\newcommand{\RubikCL}{\@randRubikc{L}}
\newcommand{\rrhCL}{\@randhc{L}}
```

\rrCLp
\rrhCLp
\RubikCLp
These commands all draw forms which denote the CLp rotation.

```
\newcommand{\rrCLp}{\@rrp{CL}}
\newcommand{\RubikCLp}{\@randRubikcp{L}}
\newcommand{\rrhCLp}{\@randhcp{L}}
```

### 20.23.3 Rotations CU and CUp

Whole cube rotation CU = Uc = y, CUp = Ucp=yp.

\rrCU
\rrhCU
\RubikCU
These commands all draw forms which denote the CU rotation.

```
\newcommand{\rrCU}{\@rr{CU}}
\newcommand{\RubikCU}{\@randRubikc{U}}
\newcommand{\rrhCU}{\@randhc{U}}
```

\rrCUp
\rrhCUp
\RubikCUp
These commands all draw forms which denote the CUp rotation.

```
\newcommand{\rrCUp}{\@rrp{CU}}
\newcommand{\RubikCUp}{\@randRubikcp{U}}
\newcommand{\rrhCUp}{\@randhcp{U}}
```

### 20.23.4 Rotations CD and CDp

Whole cube rotations CD = Dc = yp, CDp = Dcp = y.

\rrCD
\rrhCD
\RubikCD
These commands all draw forms which denote the CD rotation.

```
\newcommand{\rrCD}{\@rr{CD}}
\newcommand{\RubikCD}{\@randRubikc{D}}
\newcommand{\rrhCD}{\@randhc{D}}
```

\rrCDp
\rrhCDp
\RubikCDp
These commands all draw forms which denote the CDp rotation.

```
\newcommand{\rrCDp}{\@rrp{CD}}
\newcommand{\RubikCDp}{\@randRubikcp{D}}
\newcommand{\rrhCDp}{\@randhcp{D}}
```

### 20.23.5 Rotations CF and CFp

Whole cube rotations CF = Fc = z, CFp = Fcp = zp.

\rrCF
\rrhCF
\RubikCF
These commands all draw forms which denote the CF rotation.

```
\newcommand{\rrCF}{\@rr{CF}}
\newcommand{\RubikCF}{\@randRubikc{F}}
\newcommand{\rrhCF}{\@randhc{F}}
```

\rrCFp
\rrhCFp
\RubikCFp
These commands all draw forms which denote the CFp rotation.

```
\newcommand{\rrCFp}{\@rrp{CF}}
\newcommand{\RubikCFp}{\@randRubikcp{F}}
\newcommand{\rrhCFp}{\@randhcp{F}}
```

### 20.23.6 Rotations CB and CBp

Whole cube rotation CB = Bc = zp, CBp = Bcp = z.

\rrCB
\rrhCB
\RubikCB
These commands all draw forms which denote the CB rotation.

```
\newcommand{\rrCB}{\@rr{CB}}
\newcommand{\RubikCB}{\@randRubikc{B}}
\newcommand{\rrhCB}{\@randhc{B}}
```

\rrCBp
\rrhCBp
\RubikCBp
These commands all draw forms which denote the CBp rotation.

```
\newcommand{\rrCBp}{\@rrp{CB}}
\newcommand{\RubikCBp}{\@randRubikcp{B}}
\newcommand{\rrhCBp}{\@randhcp{B}}
```

### 20.23.7 Rotations MR and MRp

MR = Rm = Mp= Sr (M follows Left). We use Sr and Srp as the templates.

\rrMR
\rrhMR
\RubikMR
\textRubikMR
These commands draw forms of the MR middle slice rotation.

```
\newcommand{\rrMR}{\@rr{MR}}
\newcommand{\rrhMR}{\rrhMp}%
\newcommand{\RubikMR}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrMR%
\end{minipage}%
}}
\newcommand{\textRubikMR}{\rrMR\,\rrhMp}
```

\rrMRp
\rrhMRp
\RubikMRp
\textRubikMRp
These commands draw forms of the MRp slice rotation.

```
\newcommand{\rrMRp}{\@rrp{MR}}
\newcommand{\rrhMRp}{\rrhM}%
\newcommand{\RubikMRp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrMRp%
\end{minipage}%
}}
\newcommand{\textRubikMRp}{\rrMRp\,\rrhM}
```

### 20.23.8 Rotations ML and MLp

ML = Lm = M = Sl (M follows Left). We use Sl and Slp as the templates.

\rrML
\rrhML
\RubikML
\textRubikML

These commands draw forms of the ML slice rotation.

```
\newcommand{\rrML}{\@rr{ML}}
\newcommand{\rrhML}{\rrhM}%
\newcommand{\RubikML}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareM\\
\rrML%
\end{minipage}%
}}
\newcommand{\textRubikML}{\rrML\,\rrhM}
```

\rrMLp
\rrhMLp
\RubikMLp
\textRubikMLp

These commands draw forms of the MLp slice rotation.

```
\newcommand{\rrMLp}{\@rrp{ML}}
\newcommand{\rrhMLp}{\rrhMp}%
\newcommand{\RubikMLp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareMp\\
\rrMLp%
\end{minipage}%
}}
\newcommand{\textRubikMLp}{\rrMLp\,\rrhMp}
```

### 20.23.9 Rotations MU and MUp

MU = Um = Ep = Su (E follows Down). We use Su and Sup as the templates.

\rrMU
\rrhMU
\RubikMU
\textRubikMU

These commands draw forms of the MU slice rotation.

```
\newcommand{\rrMU}{\@rr{MU}}
\newcommand{\rrhMU}{\rrhEp}%
\newcommand{\RubikMU}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrMU%
\end{minipage}%
}}
\newcommand{\textRubikMU}{\rrMU\,\rrhEp}
```

\rrMUp
\rrhMUp
\RubikMUp
\textRubikMUp

These commands draw forms of the MUp slice rotation.

```
\newcommand{\rrMUp}{\@rrp{MU}}
\newcommand{\rrhMUp}{\rrhE}%
\newcommand{\RubikMUp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
```

```
\centering%
\SquareE\\
\rrMUp%
\end{minipage}%
}}
\newcommand{\textRubikMUp}{\rrMUp\,\rrhE}
```

### 20.23.10 Rotations MD and MDp

MD = Dm = E = Sd (E follows Down). We use Sd and Sdp as the templates.

\rrMD
\rrhMD
\RubikMD
\textRubikMD

These commands draw forms of the Randelshofer MD slice rotation.

```
\newcommand{\rrMD}{\@rr{MD}}
\newcommand{\rrhMD}{\rrhE}%
\newcommand{\RubikMD}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareE\\
\rrMD%
\end{minipage}%
}}
\newcommand{\textRubikMD}{\rrMD\,\rrhE}
```

\rrMDp
\rrhMDp
\RubikMDp
\textRubikMDp

These commands draw forms of the Randelshofer MDp slice rotation.

```
\newcommand{\rrMDp}{\@rrp{MD}}
\newcommand{\rrhMDp}{\rrhEp}%
\newcommand{\RubikMDp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareEp\\
\rrMDp%
\end{minipage}%
}}
\newcommand{\textRubikMDp}{\rrMDp\,\rrhEp}
```

### 20.23.11 Rotations MF and MFp

MF = Fm = S = Sf (S follows Front). MF = S is not visible from the front, so is represented in a square box. We use Sf and Sfp as the templates.

\rrMF
\rrhMF
\RubikMF
\textRubikMF

These commands draw forms of the MF slice rotation. We need to just make square with MF in square; adjust box height using a `\rule`; adjust `\fboxsep` (default=3pt); adjust `\fboxrule` (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrMF}{\@rr{MF}}
\newcommand{\SquareMF}{\@tlen\@SquareLetter{\rrMF}\@tlen}
\newcommand{\rrhMF}{\raisebox{-0.25mm}{\SquareMF}}
```

```
\newcommand{\RubikMF}{\raisebox{\@hRubik}{\SquareMF}}
\newcommand{\textRubikMF}{\rrhMF}
```

\rrMFp
\rrhMFp
\RubikMFp
\textRubikMFp

These commands draw forms of the MFp slice rotation. We need to just make square with MFp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrMFp}{\@rrp{MF}}
\newcommand{\SquareMFp}{\@tlen\@SquareLetter{\rrMFp}\@tlen}
\newcommand{\rrhMFp}{\raisebox{-0.25mm}{\SquareMFp}}
\newcommand{\RubikMFp}{\raisebox{\@hRubik}{\SquareMFp}}
\newcommand{\textRubikMFp}{\rrhMFp}
```

### 20.23.12 Rotations MB and MBp

MB = Bm = Sp = Sb. We use Sb and Sbp as the templates.

\rrMB
\rrhMB
\RubikMB
\textRubikMB

These commands draw forms of the MB slice rotation. We need to just make square with MB in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrMB}{\@rr{MB}}
\newcommand{\SquareMB}{\@tlen\@SquareLetter{\rrMB}\@tlen}
\newcommand{\rrhMB}{\raisebox{-0.25mm}{\SquareMB}}
\newcommand{\RubikMB}{\raisebox{\@hRubik}{\SquareMB}}
\newcommand{\textRubikMB}{\rrhMB}
```

\rrMBp
\rrhMBp
\RubikMBp
\textRubikMBp

These commands draw forms of the MBp slice rotation. We need to just make square with MBp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrMBp}{\@rrp{MB}}
\newcommand{\SquareMBp}{\@tlen\@SquareLetter{\rrMBp}\@tlen}
\newcommand{\rrhMBp}{\raisebox{-0.25mm}{\SquareMBp}}
\newcommand{\RubikMBp}{\raisebox{\@hRubik}{\SquareMBp}}
\newcommand{\textRubikMBp}{\rrhMBp}
```

### 20.23.13 Rotations T (wide)

TR = Rw (We use Rw and Rwp as the templates).
We first set up some useful T and Tp macros for use with (L,R,U,D,F,B) slightly more general than before. As before the rubikfont command needs to be delimited using two curly brackets to contain the font expression.

```
\newcommand{\@rrT}[1]{{\@rubikfont T#1}}
\newcommand{\@rrhT}[1]{\@join{\rrh#1}{w}}%
\newcommand{\@textRubikT}[1]{\@rrT{#1}\,\@join{\rrh#1}{w}}
%
\newcommand{\@RubikT}[1]{%
```

```
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%%
\@join{\Square#1}{w}\\%
\@rrT{#1}%%
\end{minipage}%
}}
%%
\newcommand{\@rrTp}[1]{{\@rubikfont T#1\@rubikprime}}
\newcommand{\@rrhTp}[1]{\@join{\rrh#1}{wp}}%
\newcommand{\@textRubikTp}[1]{\@rrTp{#1}\,\@join{\rrh#1}{wp}}
%%
\newcommand{\@RubikTp}[1]{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%%
\@join{\Square#1}{wp}\\%
\@rrTp{#1}%%
\end{minipage}%
}}
```

\rrTL \rrhTL \RubikTL \textRubikTL

These commands draw forms of the TL = Lw wide slice rotation.

```
\newcommand{\rrTL}{\@rrT{L}}% = rrLw
\newcommand{\rrhTL}{\@rrhT{L}}
\newcommand{\textRubikTL}{\@textRubikT{L}}
\newcommand{\RubikTL}{\@RubikT{L}}
```

\rrTLp \rrhTLp \RubikTLp \textRubikTLp

These commands draw forms of the TLp = Lwp wide slice rotation.

```
\newcommand{\rrTLp}{\@rrTp{L}}
\newcommand{\rrhTLp}{\@rrhTp{L}}
\newcommand{\textRubikTLp}{\@textRubikTp{L}}
\newcommand{\RubikTLp}{\@RubikTp{L}}
```

\rrTR \rrhTR \RubikTR \textRubikTR

These commands draw forms of the TR = Rw wide slice rotation.

```
\newcommand{\rrTR}{\@rrT{R}}
\newcommand{\rrhTR}{\@rrhT{R}}
\newcommand{\textRubikTR}{\@textRubikT{R}}
\newcommand{\RubikTR}{\@RubikT{R}}
```

\rrTRp \rrhTRp \RubikTRp \textRubikTRp

These commands draw forms of the TRp = Rmp wide slice rotation.

```
\newcommand{\rrTRp}{\@rrTp{R}}
\newcommand{\rrhTRp}{\@rrhTp{R}}
\newcommand{\textRubikTRp}{\@textRubikTp{R}}
\newcommand{\RubikTRp}{\@RubikTp{R}}
```

\rrTU \rrhTU \RubikTU \textRubikTU

These commands draw forms of the TU = Uw wide slice rotation.

```
\newcommand{\rrTU}{\@rrT{U}}
\newcommand{\rrhTU}{\@rrhT{U}}
\newcommand{\textRubikTU}{\@textRubikT{U}}
\newcommand{\RubikTU}{\@RubikT{U}}
```

\rrTUp
\rrhTUp
\RubikTUp
\textRubikTUp

These commands draw forms of the TUp = Uwp wide slice rotation.

```
\newcommand{\rrTUp}{\@rrTp{U}}
\newcommand{\rrhTUp}{\@rrhTp{U}}
\newcommand{\textRubikTUp}{\@textRubikTp{U}}
\newcommand{\RubikTUp}{\@RubikTp{U}}
```

\rrTD
\rrhTD
\RubikTD
\textRubikTD

These commands draw forms of the TD = Dw wide slice rotation.

```
\newcommand{\rrTD}{\@rrT{D}}
\newcommand{\rrhTD}{\@rrhT{D}}
\newcommand{\textRubikTD}{\@textRubikT{D}}
\newcommand{\RubikTD}{\@RubikT{D}}
```

\rrTDp
\rrhTDp
\RubikTDp
\textRubikTDp

These commands draw forms of the TDp = Dwp wide slice rotation.

```
\newcommand{\rrTDp}{\@rrTp{D}}
\newcommand{\rrhTDp}{\@rrhTp{D}}
\newcommand{\textRubikTDp}{\@textRubikTp{D}}
\newcommand{\RubikTDp}{\@RubikTp{D}}
```

\rrTF
\rrhTF
\RubikTF
\textRubikTF

These commands draw forms of the TF = Fw wide slice rotation.

```
\newcommand{\rrTF}{\@rrT{F}}
\newcommand{\rrhTF}{\@rrhT{F}}
\newcommand{\textRubikTF}{\@textRubikT{F}}
\newcommand{\RubikTF}{\@RubikT{F}}
```

\rrTFp
\rrhTFp
\RubikTFp
\textRubikTFp

These commands draw forms of the TFp = Fwp wide slice rotation.

```
\newcommand{\rrTFp}{\@rrTp{F}}
\newcommand{\rrhTFp}{\@rrhTp{F}}
\newcommand{\textRubikTFp}{\@textRubikTp{F}}
\newcommand{\RubikTFp}{\@RubikTp{F}}
```

\rrTB
\rrhTB
\RubikTB
\textRubikTB

These commands draw forms of the TB = Bw wide slice rotation. NOTE we just use the letters TB in a square

```
\newcommand{\rrTB}{\@rrT{B}}
\newcommand{\SquareTB}{\@tlen\@SquareLetter{\rrTB}\@tlen}
\newcommand{\rrhTB}{\raisebox{-0.25mm}{\SquareTB}}
\newcommand{\textRubikTB}{\rrhTB}
\newcommand{\RubikTB}{\raisebox{\@hRubik}{\SquareTB}}
```

\rrTBp
\rrhTBp
\RubikTBp
\textRubikTBp

These commands draw forms of the TBp = Bwp wide slice rotation. NOTE we just use the letters TB' in a square

```
\newcommand{\rrTBp}{\@rrTp{B}}
\newcommand{\SquareTBp}{\@tlen\@SquareLetter{\rrTBp}\@tlen}
\newcommand{\rrhTBp}{\raisebox{-0.25mm}{\SquareTBp}}
\newcommand{\textRubikTBp}{\rrhTBp}
\newcommand{\RubikTBp}{\raisebox{\@hRubik}{\SquareTBp}}
```

### 20.23.14 Rotations SR and SRp (opposite slices)

opposite slices in same direction SR = Rs (both rotating in R direction)

\rrSR
\rrhSR
\RubikSR
\textRubikSR

These commands draw forms of the SR opposite slice rotation.

```
\newcommand{\rrSR}{\@rr{SR}}
\newcommand{\rrhSR}{\rrhRs}%
\newcommand{\RubikSR}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRs\\
\rrSR%
\end{minipage}%
}}
\newcommand{\textRubikSR}{\rrSR\,\rrhRs}
```

\rrSRp
\rrhSRp
\RubikSRp
\textRubikSRp

These commands draw forms of the SRp opposite slice rotation.

```
\newcommand{\rrSRp}{\@rrp{SR}}
\newcommand{\rrhSRp}{\rrhRsp}%
\newcommand{\RubikSRp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareRsp\\
\rrSRp%
\end{minipage}%
}}
\newcommand{\textRubikSRp}{\rrSRp\,\rrhRsp}
```

### 20.23.15 Rotations SL and SLp

SL = Ls (both rotating in L direction)

\rrSL
\rrhSL
\RubikSL
\textRubikSL

These commands draw forms of the SL opposite slice rotation.

```
\newcommand{\rrSL}{\@rr{SL}}
\newcommand{\rrhSL}{\rrhLs}%
\newcommand{\RubikSL}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLs\\
\rrSL%
\end{minipage}%
}}
\newcommand{\textRubikSL}{\rrSL\,\rrhLs}
```

\rrSLp
\rrhSLp
\RubikSLp
\textRubikSLp

These commands draw forms of the SLp opposite slice rotation.

```
\newcommand{\rrSLp}{\@rrp{SL}}
\newcommand{\rrhSLp}{\rrhLsp}%
```

```
\newcommand{\RubikSLp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareLsp\\
\rrSLp%
\end{minipage}%
}}
\newcommand{\textRubikSLp}{\rrSLp\,\rrhLsp}
```

### 20.23.16 Rotations SU and SUp

SU = Us (both rotating in U direction)

\rrSU
\rrhSU
\RubikSU
\textRubikSU

These commands draw forms of the SU opposite slice rotation.

```
\newcommand{\rrSU}{\@rr{SU}}
\newcommand{\rrhSU}{\rrhUs}%
\newcommand{\RubikSU}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUs\\
\rrSU%
\end{minipage}%
}}
\newcommand{\textRubikSU}{\rrSU\,\rrhUs}
```

\rrSUp
\rrhSUp
\RubikSUp
\textRubikSUp

These commands draw forms of the SUp opposite slice rotation.

```
\newcommand{\rrSUp}{\@rrp{SU}}
\newcommand{\rrhSUp}{\rrhUsp}%
\newcommand{\RubikSUp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareUsp\\
\rrSUp%
\end{minipage}%
}}
\newcommand{\textRubikSUp}{\rrSUp\,\rrhUsp}
```

### 20.23.17 Rotations SD and SDp

SD = Ds (both rotating in D direction)

\rrSD
\rrhSD
\RubikSD
\textRubikSD

These commands draw forms of the SD opposite slice rotation.

```
\newcommand{\rrSD}{\@rr{SD}}
\newcommand{\rrhSD}{\rrhDs}%
\newcommand{\RubikSD}{%
{\@rubikfont%
```

```
\begin{minipage}{0.6cm}
\centering%
\SquareDs\\
\rrSD%
\end{minipage}%
}}
\newcommand{\textRubikSD}{\rrSD\,\rrhDs}
```

\rrSDp
\rrhSDp
\RubikSDp
\textRubikSDp

These commands draw forms of the SDp opposite slice rotation.

```
\newcommand{\rrSDp}{\@rrp{SD}}
\newcommand{\rrhSDp}{\rrhDsp}%
\newcommand{\RubikSDp}{%
{\@rubikfont%
\begin{minipage}{0.6cm}
\centering%
\SquareDsp\\
\rrSDp%
\end{minipage}%
}}
\newcommand{\textRubikSDp}{\rrSDp\,\rrhDsp}
```

### 20.23.18 Rotations SF and SFp

SF = Fs is not visible from the front, so is represented in a square box. Both rotating in the F direction We use Fs and Fsp as the templates.

\rrSF
\rrhSF
\RubikSF
\textRubikSF

These commands draw forms of the SF opposite slice rotation. We need to just make square with SF in square; adjust box height using a `\rule`; adjust `\fboxsep` (default=3pt); adjust `\fboxrule` (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSF}{\@rr{SF}}
\newcommand{\SquareSF}{\@tlen\@SquareLetter{\rrSF}\@tlen}
\newcommand{\rrhSF}{\raisebox{-0.25mm}{\SquareSF}}
\newcommand{\RubikSF}{\raisebox{\@hRubik}{\SquareSF}}
\newcommand{\textRubikSF}{\rrhSF}
```

\rrSFp
\rrhSFp
\RubikSFp
\textRubikSFp

These commands draw forms of the SFp opposite slice rotation. We need to just make square with SFp in square; adjust box height using a `\rule`; adjust `\fboxsep` (default=3pt); adjust `\fboxrule` (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSFp}{\@rrp{SF}}
\newcommand{\SquareSFp}{\@tlen\@SquareLetter{\rrSFp}\@tlen}
\newcommand{\rrhSFp}{\raisebox{-0.25mm}{\SquareSFp}}
\newcommand{\RubikSFp}{\raisebox{\@hRubik}{\SquareSFp}}
\newcommand{\textRubikSFp}{\rrhSFp}
```

### 20.23.19 Rotations SB and SBp

SB = Bs is not visible from the front, so is represented in a square box. Both rotating in the B direction We use Bs and Bsp as the templates.

\rrSB
\rrhSB
\RubikSB
\textRubikSB

These commands draw forms of the SB opposite slice rotation. We need to just make square with SB in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSB}{\@rr{SB}}
\newcommand{\SquareSB}{\@tlen\@SquareLetter{\rrSB}\@tlen}
\newcommand{\rrhSB}{\raisebox{-0.25mm}{\SquareSB}}
\newcommand{\RubikSB}{\raisebox{\@hRubik}{\SquareSB}}
\newcommand{\textRubikSB}{\rrhSB}
```

\rrSBp
\rrhSBp
\RubikSBp
\textRubikSBp

These commands draw forms of the SBp opposite slice rotation. We need to just make square with SBp in square; adjust box height using a \rule; adjust \fboxsep (default=3pt); adjust \fboxrule (default=0.4pt); bounded by {} so no need to reset to defaults. Not visible from the front.

```
\newcommand{\rrSBp}{\@rrp{SB}}
\newcommand{\SquareSBp}{\@tlen\@SquareLetter{\rrSBp}\@tlen}
\newcommand{\rrhSBp}{\raisebox{-0.25mm}{\SquareSBp}}
\newcommand{\RubikSBp}{\raisebox{\@hRubik}{\SquareSBp}}
\newcommand{\textRubikSBp}{\rrhSBp}
```

## 20.24 Axis rotations (textRubik versions)

For completeness we include a \textRubik version of all the axis rotation codes (making them equal to their hieroglyphic \rrh version). Obviously this list must go at the end of this file. While these commands are perhaps strictly unnecessary, the motivation is to allow users to include them in a \ShowSequence command when using the \textRubik font argument.

```
\newcommand{\textRubikx}{\rrhx}
\newcommand{\textRubikxp}{\rrhxp}
\newcommand{\textRubiky}{\rrhy}
\newcommand{\textRubikyp}{\rrhyp}
\newcommand{\textRubikz}{\rrhz}
\newcommand{\textRubikzp}{\rrhzp}
\newcommand{\textRubikl}{\rrhl}
\newcommand{\textRubiklp}{\rrhlp}
\newcommand{\textRubikr}{\rrhr}
\newcommand{\textRubikrp}{\rrhrp}
\newcommand{\textRubiku}{\rrhu}
\newcommand{\textRubikup}{\rrhup}
\newcommand{\textRubikd}{\rrhd}
\newcommand{\textRubikdp}{\rrhdp}
\newcommand{\textRubikf}{\rrhf}
\newcommand{\textRubikfp}{\rrhfp}
```

```
\newcommand{\textRubikb}{\rrhb}
\newcommand{\textRubikbp}{\rrhbp}
\newcommand{\textRubikLc}{\rrhLc}
\newcommand{\textRubikLcp}{\rrhLcp}
\newcommand{\textRubikRc}{\rrhRc}
\newcommand{\textRubikRcp}{\rrhRcp}
\newcommand{\textRubikUc}{\rrhUc}
\newcommand{\textRubikUcp}{\rrhUcp}
\newcommand{\textRubikDc}{\rrhDc}
\newcommand{\textRubikDcp}{\rrhDcp}
\newcommand{\textRubikFc}{\rrhFc}
\newcommand{\textRubikFcp}{\rrhFcp}
\newcommand{\textRubikBc}{\rrhBc}
\newcommand{\textRubikBcp}{\rrhBcp}
\newcommand{\textRubikCL}{\rrhCL}
\newcommand{\textRubikCLp}{\rrhCLp}
\newcommand{\textRubikCR}{\rrhCR}
\newcommand{\textRubikCRp}{\rrhCRp}
\newcommand{\textRubikCU}{\rrhCU}
\newcommand{\textRubikCUp}{\rrhCUp}
\newcommand{\textRubikCD}{\rrhCD}
\newcommand{\textRubikCDp}{\rrhCDp}
\newcommand{\textRubikCF}{\rrhCF}
\newcommand{\textRubikCFp}{\rrhCFp}
\newcommand{\textRubikCB}{\rrhCB}
\newcommand{\textRubikCBp}{\rrhCBp}
```

End of this package

⟨/rubikcube⟩

# Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.


**Symbols**

\& . . . . . 73

\, . 1244, 1265, 1286, 1307, 1328, 1349, 1370, 1391, 1412, 1433, 1452, 1471, 1492, 1512, 1554, 1575, 1596, 1617, 1638, 1659, 1680, 1701, 1722, 1743, 1765, 1786, 1807, 1828, 1849, 1870, 1891, 1912, 1933, 1944, 1955, 1966, 1977, 1988, 1999, 2010, 2051, 2074, 2095, 2116, 2137, 2158, 2179, 2200, 2309, 2320, 2331, 2342, 2353, 2364, 2375, 2386, 2459, 2470, 2481, 2492, 2503, 2514, 2525, 2536, 2559, 2572, 2642, 2653, 2664,

2675, 2686, 2697, 2708, 2719
\@RubikT . . . . . 2561, 2585, 2593, 2601, 2609, 2617
\@RubikTp . . . . 2574, 2589, 2597, 2605, 2613, 2621
\@SquareLetter . . . . . 1134, 1185, 1190, 1195, 1200, 1205, 1210, 1215, 1220, 1515, 1520, 1525, 1530, 1914, 1919, 2012, 2017, 2022, 2027, 2388, 2393, 2398, 2403, 2538, 2543, 2548, 2553, 2623, 2628, 2721, 2726, 2731, 2736
\@gobble . . . . . 1179
\@hRubik 1134, 1187, 1192, 1197, 1202, 1207, 1212, 1217, 1222, 1517, 1522, 1527, 1532, 1916, 1921, 2014, 2019, 2024, 2029, 2390, 2395, 2400, 2405, 2540, 2545, 2550, 2555, 2626, 2631, 2723, 2728, 2733, 2738
\@ifpackageloaded . . . . . 6, 9, 12
\@join 839, 840, 858, 859, 866, 867, 874, 875, 895, 896, 920, 921, 928, 929, 966, 967, 1178, 1180, 1181, 1182, 1183, 2558, 2559, 2565, 2571, 2572, 2578
\@randRubikc . . . . . 2409, 2414, 2420, 2426, 2432, 2438, 2444
\@randRubikcp . . . . . 2411, 2417, 2423, 2429, 2435, 2441, 2447
\@randhc 2407, 2415, 2421, 2427, 2433, 2439, 2445
\@randhcp 2408, 2418, 2424, 2430, 2436, 2442, 2448
\@rr . . . . 1134, 1184, 1224, 1392, 1434, 1534, 1702, 1744, 1913, 2031, 2201, 2207, 2213, 2219, 2225, 2231, 2237, 2243, 2249, 2413, 2419, 2425, 2431, 2437, 2443, 2449, 2471, 2493, 2515, 2537, 2547, 2632, 2654, 2676, 2698, 2720, 2730
\@rrT . . . 2557, 2559, 2566, 2582, 2590, 2598, 2606, 2614, 2622
\@rrTp . . 2570, 2572, 2579, 2586, 2594, 2602, 2610, 2618, 2627
\@rra . . . 1134, 1214, 1350, 1524, 1660, 1871, 2159
\@rrap . . 1134, 1219, 1371, 1529, 1681, 1892, 2180
\@rrb . . . . . 1152, 2021
\@rrbp . . . . . 1153, 2026
\@rrc . . . 1154, 2263, 2269, 2275, 2281, 2287, 2293
\@rrcp . . 1155, 2266, 2272, 2278, 2284, 2290, 2296
\@rrd . . . . . 1144, 1945
\@rrdp . . . . . 1145, 1956
\@rrf . . . . . 1150, 2011
\@rrfp . . . . . 1151, 2016
\@rrhT . . . . . 2558, 2583, 2591, 2599, 2607, 2615
\@rrhTp . . . . . 2571, 2587, 2595, 2603, 2611, 2619
\@rrl . . . . . 1146, 1967
\@rrlp . . . . . 1147, 1978
\@rrm . . . 1156, 2299, 2321, 2343, 2365, 2387, 2397
\@rrmp . . 1157, 2310, 2332, 2354, 2376, 2392, 2402
\@rrp . . . 1134, 1189, 1245, 1413, 1453, 1555, 1723, 1766, 1918, 2075, 2204, 2210, 2216, 2220, 2226, 2232, 2238, 2244, 2250, 2416, 2422, 2428, 2434, 2440, 2446, 2460, 2482, 2504, 2526, 2542, 2552, 2643, 2665, 2687, 2709, 2725, 2735
\@rrr . . . . . 1148, 1989
\@rrrp . . . . . 1149, 2000
\@rrs . . . 1134, 1204, 1308, 1514, 1618, 1829, 2117

\@rrsp . . 1134, 1209, 1329, 1519, 1639, 1850, 2138
\@rru . . . . . 1142, 1923
\@rrup . . . . . 1143, 1934
\@rrw . . . 1134, 1194, 1266, 1472, 1576, 1787, 2052
\@rrwp . . 1134, 1199, 1287, 1493, 1597, 1808, 2096
\@rubikfont 24, 1134, 1135, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152, 1153, 1154, 1155, 1156, 1157, 1158, 1159, 1161, 1163, 1165, 1167, 1169, 1171, 1237, 1258, 1279, 1300, 1321, 1342, 1363, 1384, 1405, 1426, 1445, 1464, 1485, 1505, 1547, 1568, 1589, 1610, 1631, 1652, 1673, 1694, 1715, 1736, 1758, 1779, 1800, 1821, 1842, 1863, 1884, 1905, 1926, 1937, 1948, 1959, 1970, 1981, 1992, 2003, 2044, 2066, 2088, 2109, 2130, 2151, 2172, 2193, 2256, 2258, 2260, 2262, 2302, 2313, 2324, 2335, 2346, 2357, 2368, 2379, 2407, 2408, 2410, 2412, 2452, 2463, 2474, 2485, 2496, 2507, 2518, 2529, 2557, 2562, 2570, 2575, 2635, 2646, 2657, 2668, 2679, 2690, 2701, 2712
\@rubikfontFNS . . . . . 24, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152, 1153, 1154, 1155, 1156, 1157, 2256, 2258, 2260, 2262
\@rubikprime . . . 24, 1135, 1137, 1139, 1141, 1143, 1145, 1147, 1149, 1151, 1153, 1155, 1157, 1159, 1163, 1167, 1171, 2258, 2262, 2408, 2412, 2570
\@sb 1128, 1129, 1229, 1230, 1231, 1250, 1251, 1252, 1271, 1272, 1273, 1292, 1293, 1294, 1313, 1314, 1315, 1334, 1335, 1336, 1355, 1356, 1357, 1376, 1377, 1378, 1397, 1398, 1399, 1418, 1419, 1420, 1539, 1540, 1541, 1560, 1561, 1562, 1581, 1582, 1583, 1602, 1603, 1604, 1623, 1624, 1625, 1644, 1645, 1646, 1665, 1666, 1667, 1686, 1687, 1688, 1707, 1708, 1709, 1728, 1729, 1730, 1750, 1751, 1752, 1771, 1772, 1773, 1792, 1793, 1794, 1813, 1814, 1815, 1834, 1835, 1836, 1855, 1856, 1857, 1876, 1877, 1878, 1897, 1898, 1899, 2036, 2037, 2038, 2057, 2058, 2059, 2080, 2081, 2082, 2101, 2102, 2103, 2122, 2123, 2124, 2143, 2144, 2145, 2164, 2165, 2166, 2185, 2186, 2187
\@sbh . . . 1129, 1229, 1230, 1231,

1250, 1251, 1252, 1271, 1272, 1273, 1292, 1293, 1294, 1313, 1314, 1315, 1334, 1335, 1336, 1355, 1356, 1357, 1376, 1377, 1378, 1397, 1398, 1399, 1418, 1419, 1420, 1539, 1540, 1541, 1560, 1561, 1562, 1581, 1582, 1583, 1602, 1603, 1604, 1623, 1624, 1625, 1644, 1645, 1646, 1665, 1666, 1667, 1686, 1687, 1688, 1707, 1708, 1709, 1728, 1729, 1730, 1750, 1751, 1752, 1771, 1772, 1773, 1792, 1793, 1794, 1813, 1814, 1815, 1834, 1835, 1836, 1855, 1856, 1857, 1876, 1877, 1878, 1897, 1898, 1899, 2036, 2037, 2038, 2057, 2058, 2059, 2080, 2081, 2082, 2101, 2102, 2103, 2122, 2123, 2124, 2143, 2144, 2145, 2164, 2165, 2166, 2185, 2186, 2187

`\@scx` 1130, 1132, 1439, 1458, 1477, 1498

`\@scy` 1131, 1133, 1439, 1458, 1477, 1498

`\@sd` 1124, 1125, 1126, 1131, 1231, 1252, 1273, 1294, 1315, 1336, 1357, 1378, 1399, 1420, 1539, 1560, 1581, 1602, 1623, 1644, 1665, 1686, 1707, 1728, 1750, 1771, 1792, 1813, 1834, 1855, 1876, 1897, 2038, 2059, 2082, 2103, 2124, 2145, 2166, 2187

`\@sdd` 1125, 1130, 1230, 1251, 1272, 1293, 1314, 1335, 1356, 1377, 1398, 1419, 1540, 1561, 1582, 1603, 1624, 1645, 1666, 1687, 1708, 1729, 1751, 1772, 1793, 1814, 1835, 1856, 1877, 1898, 2037, 2058, 2081, 2102, 2123, 2144, 2165, 2186

`\@sddd` . . 1126, 1229, 1250, 1271, 1292, 1313, 1334, 1355, 1376, 1397, 1418, 1541, 1562, 1583, 1604, 1625, 1646, 1667, 1688, 1709, 1730, 1752, 1773, 1794, 1815, 1836, 1857, 1878, 1899, 2036, 2057, 2080, 2101, 2122, 2143, 2164, 2185

`\@sh` . . . . . 1127, 1129

`\@showcubewidth` . . . . . 27, 30, 31, 32, 42, 43, 45

`\@sqcx` . 1132, 1478, 1499

`\@sqcy` . 1133, 1478, 1499

`\@textRubikT` . . 2559, 2584, 2592, 2600, 2608, 2616

`\@textRubikTp` . 2572, 2588, 2596, 2604, 2612, 2620

`\@tlen` . . 1177, 1185, 1190, 1195, 1200, 1205, 1210, 1215, 1220, 1234, 1255, 1276, 1297, 1318, 1339, 1360, 1381, 1402, 1423, 1442, 1461, 1482, 1502, 1515, 1520, 1525, 1530, 1544, 1565, 1586, 1607, 1628, 1649, 1670, 1691, 1712, 1733, 1755, 1776, 1797, 1818, 1839, 1860, 1881, 1902, 1914, 1919, 2012, 2017, 2022, 2027, 2041, 2063, 2085,

2106, 2127, 2148, 2169, 2190, 2388, 2393, 2398, 2403, 2538, 2543, 2548, 2553, 2623, 2628, 2721, 2726, 2731, 2736
\@xyzRubik . . . 1134, 2202, 2208, 2214, 2223, 2241
\@xyzRubikc . . . . . 2255, 2264, 2270, 2276, 2282, 2288, 2294
\@xyzRubikcp . . . . . 2255, 2267, 2273, 2279, 2285, 2291, 2297
\@xyzRubikp . . . 1134, 2205, 2211, 2217, 2224, 2242
\@xyzbdflRubik . . . . 1168, 2229, 2235, 2247, 2253
\@xyzbdflRubikp . . . 1170, 2230, 2236, 2248, 2254
\@xyzh . . . . . 1134, 2203, 2209, 2215, 2221, 2239
\@xyzhbdfl 1164, 2227, 2233, 2245, 2251
\@xyzhbdflp . . . . . 1166, 2228, 2234, 2246, 2252
\@xyzhc . 2255, 2265, 2271, 2277, 2283, 2289, 2295
\@xyzhcp 2255, 2268, 2274, 2280, 2286, 2292, 2298
\@xyzhp . . . . . 1134, 2206, 2212, 2218, 2222, 2240
\\ . 1240, 1261, 1282, 1303, 1324, 1345, 1366, 1387, 1408, 1429, 1448, 1467, 1488, 1508, 1550, 1571, 1592, 1613, 1634, 1655, 1676, 1697, 1718, 1739, 1761, 1782, 1803, 1824, 1845, 1866, 1887, 1908, 1929, 1940, 1951, 1962, 1973, 1984, 1995, 2006, 2047, 2069, 2091, 2112, 2133, 2154, 2175, 2196, 2305, 2316, 2327, 2338, 2349, 2360, 2371, 2382, 2455, 2466, 2477, 2488, 2499, 2510, 2521, 2532, 2565, 2578, 2638, 2649, 2660, 2671, 2682, 2693, 2704, 2715
\␣ . . . . . 421

**A**

\advance . . . . . 31, 43

**B**

\B . . . . . *25*
\b . . . . . *28*
\Ba . . . . . *27*
\Bc . . . . . *29*
\begin . . . . . 32, 34, 45, 47, 1086, 1088, 1095, 1097, 1104, 1106, 1113, 1115, 1227, 1238, 1248, 1259, 1269, 1280, 1290, 1301, 1311, 1322, 1332, 1343, 1353, 1364, 1374, 1385, 1395, 1406, 1416, 1427, 1437, 1446, 1456, 1465, 1475, 1486, 1496, 1506, 1537, 1548, 1558, 1569, 1579, 1590, 1600, 1611, 1621, 1632, 1642, 1653, 1663, 1674, 1684, 1695, 1705, 1716, 1726, 1737, 1747, 1759, 1769, 1780, 1790, 1801, 1811, 1822, 1832, 1843, 1853, 1864, 1874, 1885, 1895, 1906, 1927, 1938, 1949, 1960, 1971, 1982, 1993, 2004, 2034, 2045, 2055, 2067, 2078, 2089, 2099, 2110, 2120, 2131, 2141, 2152, 2162, 2173, 2183, 2194, 2303, 2314, 2325, 2336, 2347, 2358, 2369, 2380, 2453, 2464, 2475, 2486, 2497,

2508, 2519, 2530, 2563, 2576, 2636, 2647, 2658, 2669, 2680, 2691, 2702, 2713
\bl 776, 782, 789, 792, 793, 798, 801, 802, 807, 811, 816, 820, 827, 830, 831, 846, 850, 883, 886, 887, 902, 906, 937, 940, 941, 954, 958, 977, 980, 981, 994, 998
\Blb . . . . . 106, 136, 619, 692, 755, 836, 946, 963, 1003
\blh . . . . . 789, 791, 798, 800, 807, 808, 816, 817, 827, 829, 846, 847, 883, 885, 902, 903, 937, 939, 954, 955, 977, 979, 994, 995
\Blm . . . . . 105, 135, 609, 693, 835, 945
\Blt . . . . . 104, 134, 599, 648, 694, 834, 855, 911, 944
\Bm . . . . . *26*
\Bmb . . . . . 106, 136, 622, 754, 962, 1002
\Bmm . . . . . 105, 135, 612
\Bmt . . . . . 104, 134, 602, 647, 854, 910
\Bp . . . . . *25*
\Brb . . . . . 106, 136, 625, 707, 753, 892, 961, 986, 1001
\Brm . . . . . 105, 135, 615, 708, 891, 985
\Brt . . . . . 104, 134, 605, 646, 709, 853, 890, 909, 984
\Bs . . . . . *27*
\bs 777, 783, 790, 799, 809, 818, 828, 848, 884, 904, 938, 956, 978, 996
\Bw . . . . . *27*
\bw 775, 781, 790, 793, 802, 810, 811, 818, 819, 820, 831, 849, 850, 884, 887, 905, 906, 941, 956, 957, 958, 978, 981, 996, 997, 998
\bx . . 596, 600, 601, 603, 604, 606, 607, 610, 611, 613, 614, 616, 617, 620, 621, 623, 624, 626, 627
\by . . 597, 600, 601, 603, 604, 606, 607, 610, 611, 613, 614, 616, 617, 620, 621, 623, 624, 626, 627

## C

\CB . . . . . *29*
\CD . . . . . *29*
\centering . . . . . 33, 46, 1087, 1096, 1105, 1114, 1239, 1260, 1281, 1302, 1323, 1344, 1365, 1386, 1407, 1428, 1447, 1466, 1487, 1507, 1549, 1570, 1591, 1612, 1633, 1654, 1675, 1696, 1717, 1738, 1760, 1781, 1802, 1823, 1844, 1865, 1886, 1907, 1928, 1939, 1950, 1961, 1972, 1983, 1994, 2005, 2046, 2068, 2090, 2111, 2132, 2153, 2174, 2195, 2304, 2315, 2326, 2337, 2348, 2359, 2370, 2381, 2454, 2465, 2476, 2487, 2498, 2509, 2520, 2531, 2564, 2577, 2637, 2648, 2659, 2670, 2681, 2692, 2703, 2714
\CF . . . . . *29*
\CL . . . . . *29*
\cmd . . . . . 564
\colorlet . . . 21, 22, 23
\CR . . . . . *29*
\csname . . . . . 1179
\CU . . . . . *29*
\cube@dxdydz 1013, 1015
\Cubiedx . . *38*, 1037, 1039, 1089, 1098, 1107, 1116
\Cubiedy . . *38*, 1037, 1040, 1089, 1098, 1107, 1116
\cx . . . . 1037, 1047, 1050, 1058, 1061, 1069, 1072, 1080, 1083
\cy . . . . 1038, 1047, 1050, 1058, 1061, 1069, 1072, 1080, 1083

## D

\D . . . . . *25*
\d . . . . . *28*
\Da . . . . . *27*
\Dc . . . . . *29*
\ddx . . 459, 463, 464, 466, 467, 469, 470, 473, 474, 476, 477, 479,

480, 483, 484, 486, 487, 489, 490
\ddy . . 460, 463, 464, 466, 467, 469, 470, 473, 474, 476, 477, 479, 480, 483, 484, 486, 487, 489, 490
\def . . . . 2, 3, 79, 80, 81, 84, 85, 86, 89, 90, 91, 94, 95, 96, 99, 100, 101, 104, 105, 106, 109, 110, 111, 114, 115, 116, 119, 120, 121, 124, 125, 126, 129, 130, 131, 134, 135, 136, 179, 180, 183, 184, 187, 188, 191, 192, 195, 196, 199, 200, 203, 204, 207, 208, 784, 785
\definecolor . . . . . . . . . . 17, 18, 19, 20
\Dlb . . . 96, 131, 321, 368, 482, 715, 736
\Dlm . . . . . 95, 130, 314, 361, 472, 716
\Dlt 94, 129, 307, 354, 462, 677, 717, 869
\Dm . . . . . *26*
\Dmb . . . . . 96, 131, 323, 370, 485, 735
\Dmm 95, 130, 316, 363, 475
\Dmt . . . 94, 129, 309, 356, 465, 678, 870
\Dp . . . . . *25*
\draw . . . . . 212, 214, 216, 219, 221, 223, 226, 228, 230, 237, 239, 241, 244, 246, 248, 251, 253, 255, 259, 261, 263, 266, 268, 270, 273, 275, 277, 285, 287, 289, 292, 294, 296, 299, 301, 303, 307, 309, 311, 314, 316, 318, 321, 323, 325, 332, 334, 336, 339, 341, 343, 346, 348, 350, 354, 356, 358, 361, 363, 365, 368, 370, 372, 379, 381, 383, 386, 388, 390, 393, 395, 397, 401, 403, 405, 408, 410, 412, 415, 417, 419, 427, 430, 433, 437, 440, 443, 447, 450, 453, 462, 465, 468, 472, 475, 478, 482, 485, 488, 497, 500, 503, 507, 510, 513, 517, 520, 523, 532, 535, 538, 542, 545, 548, 552, 555, 558, 567, 570, 573, 576, 579, 582, 585, 588, 591, 599, 602, 605, 609, 612, 615, 619, 622, 625, 792, 801, 810, 819, 830, 849, 886, 905, 940, 957, 980, 997, 1026, 1031, 1034, 1043, 1046, 1049, 1054, 1057, 1060, 1065, 1068, 1071, 1076, 1079, 1082, 1122, 1229, 1230, 1231, 1250, 1251, 1252, 1271, 1272, 1273, 1292, 1293, 1294, 1313, 1314, 1315, 1334, 1335, 1336, 1355, 1356, 1357, 1376, 1377, 1378, 1397, 1398, 1399, 1418, 1419, 1420, 1439, 1458, 1477, 1478, 1498, 1499, 1539, 1540, 1541, 1560, 1561, 1562, 1581, 1582, 1583, 1602, 1603, 1604, 1623, 1624, 1625, 1644, 1645, 1646, 1665, 1666, 1667, 1686, 1687, 1688, 1707, 1708, 1709, 1728, 1729, 1730, 1750, 1751, 1752, 1771, 1772, 1773, 1792, 1793, 1794, 1813, 1814, 1815, 1834, 1835, 1836, 1855, 1856, 1857, 1876, 1877, 1878, 1897, 1898, 1899, 2036, 2037, 2038, 2057, 2058, 2059, 2080, 2081, 2082, 2101, 2102, 2103, 2122, 2123, 2124,

2143, 2144, 2145, 2164, 2165, 2166, 2185, 2186, 2187
\DrawCubieLD . . . . . 1041, 1063, 1108
\DrawCubieLU . . . . . 1041, 1074, 1117
\DrawCubieRD . . . . . 1041, 1052, 1099
\DrawCubieRU . . . . . 1041, 1041, 1090
\DrawCubieXY . . . . . *38*
\DrawNCubeAll . *44*, 1006
\DrawNotationBox . . . . 1121, 1228, 1249, 1270, 1291, 1312, 1333, 1354, 1375, 1396, 1417, 1438, 1457, 1476, 1497, 1538, 1559, 1580, 1601, 1622, 1643, 1664, 1685, 1706, 1727, 1748, 1770, 1791, 1812, 1833, 1854, 1875, 1896, 2035, 2056, 2079, 2100, 2121, 2142, 2163, 2184
\DrawRubikCube 280, 280
\DrawRubikCubeF . . . . *39*, *40*, 564, 629
\DrawRubikCubeFrontFace . . . . . 210, 234, 282, 329, 376
\DrawRubikCubeLD . . . . . 328, 328
\DrawRubikCubeLU . . . . . 375, 375
\DrawRubikCubeRD . . . . . 281, 281
\DrawRubikCubeRU . . . 233, 233, 280, 638
\DrawRubikCubeSF . . . . . *39*, *40*, 637
\DrawRubikCubeSidebarBD . . . . . 967
\DrawRubikCubeSidebarBDLD . . . . . 1005
\DrawRubikCubeSidebarBDRD . . . . . 965
\DrawRubikCubeSidebarBL . . . . . 896
\DrawRubikCubeSidebarBLLD . . . . . 988
\DrawRubikCubeSidebarBLLU . . . . . 894
\DrawRubikCubeSidebarBR . . . . . 840
\DrawRubikCubeSidebarBRRD . . . . . 948
\DrawRubikCubeSidebarBRRU . . . . . 838
\DrawRubikCubeSidebarBU . . . . . 859
\DrawRubikCubeSidebarBULU . . . . . 913
\DrawRubikCubeSidebarBURU . . . . . 857
\DrawRubikCubeSidebarDB . . . . . 966
\DrawRubikCubeSidebarDBLD . . . . . 1000, 1005
\DrawRubikCubeSidebarDBRD . . . . . 960, 965
\DrawRubikCubeSidebarDF . . . . . 875
\DrawRubikCubeSidebarDFLU . . . . . 876
\DrawRubikCubeSidebarDFRU . . . . . 873, 876
\DrawRubikCubeSidebarFD . . . . . 874
\DrawRubikCubeSidebarFDLU . . . . . 877
\DrawRubikCubeSidebarFDRU . . . . 868, 873, 877
\DrawRubikCubeSidebarFL . . . . . 866
\DrawRubikCubeSidebarFLRD . . . . . 930
\DrawRubikCubeSidebarFLRU . . . . 860, 865, 930
\DrawRubikCubeSidebarFR . . . . . 920
\DrawRubikCubeSidebarFRLD . . . . . 970
\DrawRubikCubeSidebarFRLU . . . . 914, 919, 970
\DrawRubikCubeSidebarFU . . . . . 928
\DrawRubikCubeSidebarFULD . . . . . 968
\DrawRubikCubeSidebarFURD . . . . 922, 927, 968
\DrawRubikCubeSidebarLB . . . . . 895
\DrawRubikCubeSidebarLBLD . . . . . 983, 988
\DrawRubikCubeSidebarLBLU . . . . . 889, 894
\DrawRubikCubeSidebarLF . . . . . 867
\DrawRubikCubeSidebarLFRD . . . . . 931
\DrawRubikCubeSidebarLFRU . . . . . 865, 931
\DrawRubikCubeSidebarRB . . . . . 839
\DrawRubikCubeSidebarRBRD . . . . . 943, 948
\DrawRubikCubeSidebarRBRU . . . . . 833, 838
\DrawRubikCubeSidebarRF . . . . . 921
\DrawRubikCubeSidebarRFLD . . . . . 971
\DrawRubikCubeSidebarRFLU . . . . . 919, 971
\DrawRubikCubeSidebarUB . . . . . 858
\DrawRubikCubeSidebarUBLU . . . . . 908, 913
\DrawRubikCubeSidebarUBRU . . . . . 852, 857
\DrawRubikCubeSidebarUF . . . . . 929
\DrawRubikCubeSidebarUFLD . . . . . 969
\DrawRubikCubeSidebarUFRD . . . . . 927, 969
\DrawRubikCubeXY . . *39*
\DrawRubikFaceB . . . 768

\DrawRubikFaceBack . . . . . *41*, 762, 768
\DrawRubikFaceBackSide . . . . . *41*, 719, 774
\DrawRubikFaceBS . . 774
\DrawRubikFaceD . . . 764
\DrawRubikFaceDown . . . . . *41*, 758, 764
\DrawRubikFaceDownSide . . . . . *41*, 738, 770
\DrawRubikFaceDS . . 770
\DrawRubikFaceF . . . 767
\DrawRubikFaceFront . . . . . *41*, 761, 767
\DrawRubikFaceFrontSide . . . . . *41*, 662, 773
\DrawRubikFaceFS . . 773
\DrawRubikFaceL . . . 765
\DrawRubikFaceLeft . . . . . *41*, 759, 765
\DrawRubikFaceLeftSide . . . . . *41*, 700, 771
\DrawRubikFaceLS . . 771
\DrawRubikFaceR . . . 766
\DrawRubikFaceRight . . . . . *41*, 760, 766
\DrawRubikFaceRightSide . . . . . *41*, 681, 772
\DrawRubikFaceRS . . 772
\DrawRubikFaceU . . . 763
\DrawRubikFaceUp . . . . . *41*, 757, 763
\DrawRubikFaceUpSide . . . . . *41*, 643, 769
\DrawRubikFaceUS . . 769
\DrawRubikFlatBack . . . *44*, 423, 595, 635, 641, 720, 762
\DrawRubikFlatDown . . . *44*, 423, 458, 631, 639, 739, 758
\DrawRubikFlatFront . . . . . *44*, 423, 563, 633, 663, 761
\DrawRubikFlatLeft . . . *44*, 423, 493, 632, 640, 701, 759
\DrawRubikFlatRight . . . . . *44*, 423, 528, 634, 682, 760
\DrawRubikFlatUp . . . . . *44*, 423, 423, 630, 644, 757
\Drb . . . 96, 131, 325, 372, 488, 698, 734
\Drm . . . . . 95, 130, 318, 365, 478, 697
\Drt 94, 129, 311, 358, 468, 679, 696, 871
\Ds . . . . . *27*
\Dw . . . . . *27*
\dx 790, 792, 793, 799, 801, 802, 808, 810, 811, 817, 819, 820, 828, 830, 831, 847, 849, 850, 884, 886, 887, 903, 905, 906, 938, 940, 941, 955, 957, 958, 978, 980, 981, 995, 997, 998, 1016, 1027, 1032, 1035
\dy 791, 792, 793, 800, 801, 802, 809, 810, 811, 818, 819, 820, 829, 830, 831, 848, 849, 850, 885, 886, 887, 904, 905, 906, 939, 940, 941, 956, 957, 958, 979, 980, 981, 996, 997, 998, 1019, 1027, 1032, 1035
\dz 1022, 1027, 1032, 1035

**E**

\E . . . . . *26*
\end . . . . . 36, 37, 49, 50, 1091, 1092, 1100, 1101, 1109, 1110, 1118, 1119, 1232, 1242, 1253, 1263, 1274, 1284, 1295, 1305, 1316, 1326, 1337, 1347, 1358, 1368, 1379, 1389, 1400, 1410, 1421, 1431, 1440, 1450, 1459, 1469, 1480, 1490, 1500, 1510, 1542, 1552, 1563, 1573, 1584, 1594, 1605, 1615, 1626, 1636, 1647, 1657, 1668, 1678, 1689, 1699, 1710, 1720, 1731, 1741, 1753, 1763, 1774, 1784, 1795, 1805, 1816, 1826, 1837, 1847, 1858, 1868, 1879, 1889, 1900, 1910, 1931, 1942, 1953, 1964, 1975, 1986, 1997, 2008, 2039, 2049, 2060, 2071, 2083, 2093, 2104, 2114, 2125, 2135, 2146, 2156, 2167, 2177, 2188, 2198, 2307, 2318, 2329, 2340, 2351, 2362, 2373, 2384, 2457, 2468, 2479, 2490, 2501, 2512, 2523, 2534, 2567, 2580, 2640,

2651, 2662, 2673, 2684, 2695, 2706, 2717
\endcsname . . . . . 1179
\Ep . . . . . *26*
\equal . . . . 788, 797, 806, 815, 826, 845, 882, 901, 936, 953, 976, 993
\expandafter . . . . . 1179

**F**

\F . . . . . *25*
\f . . . . . *28*
\Fa . . . . . *27*
\fbox 60, 67, 76, 77, 1174
\fboxrule . . . . 41, 1173
\fboxsep . . 29, 40, 1172
\Fc . . . . . *29*
\Flb 86, 116, 203, 208, 226, 585, 711, 741
\Flm . . . . . 85, 115, 187, 192, 195, 200, 219, 576, 712
\Flt 84, 114, 179, 184, 212, 567, 658, 713
\Fm . . . . . *26*
\Fmb . . . 86, 116, 203, 208, 228, 588, 742
\Fmm 85, 115, 187, 192, 195, 200, 221, 579
\Fmt . . . 84, 114, 179, 184, 214, 570, 659
\fontsize . . . . . 24, 25
\foreach 1010, 1011, 1012
\ForEachX . . . 55, 61, 66
\Fp . . . . . *25*
\framebox . . . . . 44
\Frb 86, 116, 203, 208, 230, 591, 688, 743
\Frm . . . . . 85, 115, 187, 192, 195, 200, 223, 582, 689
\Frt 84, 114, 179, 184, 216, 573, 660, 690
\Fs . . . . . *27*
\Fw . . . . . *27*

**H**

\height . . . . . 1234, 1255, 1276, 1297, 1318, 1339, 1360, 1381, 1402, 1423, 1442, 1461, 1482, 1502, 1544, 1565, 1586, 1607, 1628, 1649, 1670, 1691, 1712, 1733, 1755, 1776, 1797, 1818, 1839, 1860, 1881, 1902, 2041, 2063, 2085, 2106, 2127, 2148, 2169, 2190
\hspace . . . . . 1177

**I**

\ifthenelse . . . . 56, 62, 67, 788, 797, 806, 815, 826, 845, 882, 901, 936, 953, 976, 993

**L**

\L . . . . . *25*
\l . . . . . *28*
\La . . . . . *27*
\Lc . . . . . *29*
\Llb 101, 126, 207, 346, 393, 517, 730, 745
\Llm 100, 125, 191, 199, 339, 386, 507, 731
\Llt 99, 124, 183, 332, 379, 497, 652, 732
\Lm . . . . . *26*
\Lmb . . 101, 126, 207, 348, 395, 520, 746
\Lmm . . 100, 125, 191, 199, 341, 388, 510
\Lmt . . . 99, 124, 183, 334, 381, 500, 651
\Lp . . . . . *25*
\Lrb . . . . . 101, 126, 207, 350, 397, 523, 669, 747, 863
\Lrm . . . . . 100, 125, 191, 199, 343, 390, 513, 670, 862
\Lrt . . . . . 99, 124, 183, 336, 383, 503, 650, 671, 861
\Ls . . . . . *27*
\Lw . . . . . *27*
\lx . . 494, 498, 499, 501, 502, 504, 505, 508, 509, 511, 512, 514, 515, 518, 519, 521, 522, 524, 525
\ly . . 495, 498, 499, 501, 502, 504, 505, 508, 509, 511, 512, 514, 515, 518, 519, 521, 522, 524, 525

**M**

\M . . . . . *26*
\MB . . . . . *26*
\mbox . . . . 74, 75, 76, 77
\MD . . . . . *26*
\MF . . . . . *26*
\ML . . . . . *26*
\Mp . . . . . *26*
\MR . . . . . *26*
\MU . . . . . *26*

**N**

\ncubes . . . . . 1007, 1010, 1011, 1012
\NeedsTeXFormat . . . . 4
\newcommand 15, 16, 24, 25, 26, 28, 39, 52, 53, 54, 59, 65, 70, 71, 72, 73, 74, 75, 76, 77, 78, 83, 88, 93, 98, 103, 108, 113, 118, 123, 128, 133, 144, 152, 153, 154, 156, 164, 165,

173, 174, 175,
176, 177, 178,
182, 186, 190,
194, 198, 202,
206, 210, 233,
280, 281, 328,
375, 423, 458,
493, 528, 563,
595, 629, 637,
643, 662, 681,
700, 719, 738,
757, 758, 759,
760, 761, 762,
763, 764, 765,
766, 767, 768,
769, 770, 771,
772, 773, 774,
775, 776, 777,
781, 782, 783,
785, 786, 795,
804, 813, 822,
833, 838, 839,
840, 841, 852,
857, 858, 859,
860, 865, 866,
867, 868, 873,
874, 875, 876,
877, 878, 889,
894, 895, 896,
897, 908, 913,
914, 919, 920,
921, 922, 927,
928, 929, 930,
931, 932, 943,
948, 949, 960,
965, 966, 967,
968, 969, 970,
971, 972, 983,
988, 989, 1000,
1005, 1006,
1015, 1037,
1038, 1041,
1052, 1063,
1074, 1085,
1094, 1103,
1112, 1121,
1134, 1135,
1136, 1137,
1138, 1139,
1140, 1141,
1142, 1143,
1144, 1145,
1146, 1147,
1148, 1149,
1150, 1151,
1152, 1153,
1154, 1155,
1156, 1157,
1158, 1159,
1160, 1162,
1164, 1166,
1168, 1170,
1172, 1177,
1178, 1180,
1181, 1182,
1183, 1184,
1185, 1186,
1187, 1188,
1189, 1190,
1191, 1192,
1193, 1194,
1195, 1196,
1197, 1198,
1199, 1200,
1201, 1202,
1203, 1204,
1205, 1206,
1207, 1208,
1209, 1210,
1211, 1212,
1213, 1214,
1215, 1216,
1217, 1218,
1219, 1220,
1221, 1222,
1223, 1224,
1226, 1234,
1236, 1244,
1245, 1247,
1255, 1257,
1265, 1266,
1268, 1276,
1278, 1286,
1287, 1289,
1297, 1299,
1307, 1308,
1310, 1318,
1320, 1328,
1329, 1331,
1339, 1341,
1349, 1350,
1352, 1360,
1362, 1370,
1371, 1373,
1381, 1383,
1391, 1392,
1394, 1402,
1404, 1412,
1413, 1415,
1423, 1425,
1433, 1434,
1436, 1442,
1444, 1452,
1453, 1455,
1461, 1463,
1471, 1472,
1474, 1482,
1484, 1492,
1493, 1495,
1502, 1504,
1512, 1514,
1515, 1516,
1517, 1518,
1519, 1520,
1521, 1522,
1523, 1524,
1525, 1526,
1527, 1528,
1529, 1530,
1531, 1532,
1533, 1534,
1536, 1544,
1546, 1554,
1555, 1557,
1565, 1567,
1575, 1576,
1578, 1586,
1588, 1596,
1597, 1599,
1607, 1609,
1617, 1618,
1620, 1628,
1630, 1638,
1639, 1641,
1649, 1651,
1659, 1660,
1662, 1670,
1672, 1680,
1681, 1683,
1691, 1693,

1701, 1702, 2018, 2019, 2248, 2249,
1704, 1712, 2020, 2021, 2250, 2251,
1714, 1722, 2022, 2023, 2252, 2253,
1723, 1725, 2024, 2025, 2254, 2255,
1733, 1735, 2026, 2027, 2257, 2259,
1743, 1744, 2028, 2029, 2261, 2263,
1746, 1755, 2030, 2031, 2264, 2265,
1757, 1765, 2033, 2041, 2266, 2267,
1766, 1768, 2043, 2051, 2268, 2269,
1776, 1778, 2052, 2054, 2270, 2271,
1786, 1787, 2063, 2065, 2272, 2273,
1789, 1797, 2074, 2075, 2274, 2275,
1799, 1807, 2077, 2085, 2276, 2277,
1808, 1810, 2087, 2095, 2278, 2279,
1818, 1820, 2096, 2098, 2280, 2281,
1828, 1829, 2106, 2108, 2282, 2283,
1831, 1839, 2116, 2117, 2284, 2285,
1841, 1849, 2119, 2127, 2286, 2287,
1850, 1852, 2129, 2137, 2288, 2289,
1860, 1862, 2138, 2140, 2290, 2291,
1870, 1871, 2148, 2150, 2292, 2293,
1873, 1881, 2158, 2159, 2294, 2295,
1883, 1891, 2161, 2169, 2296, 2297,
1892, 1894, 2171, 2179, 2298, 2299,
1902, 1904, 2180, 2182, 2300, 2301,
1912, 1913, 2190, 2192, 2309, 2310,
1914, 1915, 2200, 2201, 2311, 2312,
1916, 1917, 2202, 2203, 2320, 2321,
1918, 1919, 2204, 2205, 2322, 2323,
1920, 1921, 2206, 2207, 2331, 2332,
1922, 1923, 2208, 2209, 2333, 2334,
1924, 1925, 2210, 2211, 2342, 2343,
1933, 1934, 2212, 2213, 2344, 2345,
1935, 1936, 2214, 2215, 2353, 2354,
1944, 1945, 2216, 2217, 2355, 2356,
1946, 1947, 2218, 2219, 2364, 2365,
1955, 1956, 2220, 2221, 2366, 2367,
1957, 1958, 2222, 2223, 2375, 2376,
1966, 1967, 2224, 2225, 2377, 2378,
1968, 1969, 2226, 2227, 2386, 2387,
1977, 1978, 2228, 2229, 2388, 2389,
1979, 1980, 2230, 2231, 2390, 2391,
1988, 1989, 2232, 2233, 2392, 2393,
1990, 1991, 2234, 2235, 2394, 2395,
1999, 2000, 2236, 2237, 2396, 2397,
2001, 2002, 2238, 2239, 2398, 2399,
2010, 2011, 2240, 2241, 2400, 2401,
2012, 2013, 2242, 2243, 2402, 2403,
2014, 2015, 2244, 2245, 2404, 2405,
2016, 2017, 2246, 2247, 2406, 2407,

2408, 2409, 2411, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2459, 2460, 2461, 2462, 2470, 2471, 2472, 2473, 2481, 2482, 2483, 2484, 2492, 2493, 2494, 2495, 2503, 2504, 2505, 2506, 2514, 2515, 2516, 2517, 2525, 2526, 2527, 2528, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2561, 2570, 2571, 2572, 2574, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2642, 2643, 2644, 2645, 2653, 2654, 2655, 2656, 2664, 2665, 2666, 2667, 2675, 2676, 2677, 2678, 2686, 2687, 2688, 2689, 2697, 2698, 2699, 2700, 2708, 2709, 2710, 2711, 2719, 2720, 2721, 2722, 2723, 2724, 2725, 2726, 2727, 2728, 2729, 2730, 2731, 2732, 2733, 2734, 2735, 2736, 2737, 2738, 2739, 2740, 2741, 2742, 2743, 2744, 2745, 2746, 2747, 2748, 2749, 2750, 2751, 2752, 2753, 2754, 2755, 2756, 2757, 2758, 2759, 2760, 2761, 2762, 2763, 2764, 2765, 2766, 2767, 2768, 2769, 2770, 2771, 2772, 2773, 2774, 2775, 2776, 2777, 2778, 2779, 2780, 2781

`\newlength` . . . 27, 1175

`\no@sidebar` . . . . . 784, 788, 797, 806, 815, 826, 845, 882, 901, 936, 953, 976, 993

`\node` . . . . . 1479

`\NoSidebar` . . . . *42*, 785

**O**

`\overbrace` . . . . . 74, 76

**P**

`\pgfmathsetmacro` . . . . . 424, 425, 459, 460, 494, 495, 529, 530, 596, 597, 775, 776, 777, 781, 782, 783, 789, 790, 791, 798, 799, 800, 807, 808, 809, 816, 817, 818, 827, 828, 829, 846, 847, 848, 883, 884, 885, 902, 903, 904, 937, 938, 939, 954,

955, 956, 977, 978, 979, 994, 995, 996, 1007, 1016, 1019, 1022, 1037, 1038, 1124, 1125, 1126, 1127, 1128, 1129, 1130, 1131, 1132, 1133
\ProvidesPackage ... 5

## R

\R ... *25*
\r ... *28*
\Ra ... *27*
\raisebox ... 1159, 1161, 1163, 1165, 1167, 1169, 1171, 1174, 1186, 1187, 1191, 1192, 1196, 1197, 1201, 1202, 1206, 1207, 1211, 1212, 1216, 1217, 1221, 1222, 1234, 1255, 1276, 1297, 1318, 1339, 1360, 1381, 1402, 1423, 1442, 1461, 1482, 1502, 1516, 1517, 1521, 1522, 1526, 1527, 1531, 1532, 1544, 1565, 1586, 1607, 1628, 1649, 1670, 1691, 1712, 1733, 1755, 1776, 1797, 1818, 1839, 1860, 1881, 1902, 1915, 1916, 1920, 1921, 2013, 2014, 2018, 2019, 2023, 2024, 2028, 2029, 2041, 2063, 2085, 2106, 2127, 2148, 2169, 2190, 2255, 2257, 2259, 2260, 2261, 2262, 2389, 2390, 2394, 2395, 2399, 2400, 2404, 2405, 2407, 2408, 2410, 2412, 2539, 2540, 2544, 2545, 2549, 2550, 2554, 2555, 2624, 2626, 2629, 2631, 2722, 2723, 2727, 2728, 2732, 2733, 2737, 2738
\Rc ... *29*
\RCfiledate ... 3, 5
\RCfileversion ... 2, 5
\relax ... 31, 43
\RequirePackage 8, 11, 14
\Rlb ... 91, 121, 204, 273, 299, 552, 673, 751, 917
\Rlm ... 90, 120, 188, 196, 266, 292, 542, 674, 916
\Rlt ... 89, 119, 180, 259, 285, 532, 654, 675, 915
\Rm ... *26*
\Rmb ... 91, 121, 204, 275, 301, 555, 750
\Rmm ... 90, 120, 188, 196, 268, 294, 545
\Rmt ... 89, 119, 180, 261, 287, 535, 655
\Rp ... *25*
\rr ... *25*, 1180
\Rrb 91, 121, 204, 277, 303, 558, 726, 749
\rrB ... 1184
\rrb ... 2249
\rrBa ... 1214
\rrBap ... 1219
\rrBc ... 2293
\rrBcp ... 2296
\rrBm ... 2397
\rrBmp ... 2402
\rrBp ... 1189
\rrbp ... 2250
\rrBs ... 1204
\rrBsp ... 1209
\rrBw ... 1194
\rrBwp ... 1199
\rrCB ... 2443
\rrCBp ... 2446
\rrCD ... 2431
\rrCDp ... 2434
\rrCF ... 2437
\rrCFp ... 2440
\rrCL ... 2419
\rrCLp ... 2422
\rrCR ... 2413
\rrCRp ... 2416
\rrCU ... 2425
\rrCUp ... 2428
\rrD ... 1224
\rrd ... 2225
\rrDa ... 1350
\rrDap ... 1371
\rrDc ... 2281
\rrDcp ... 2284
\rrDm ... 2365
\rrDmp ... 2376
\rrDp ... 1245
\rrdp ... 2226
\rrDs ... 1308
\rrDsp ... 1329
\rrDw ... 1266
\rrDwp ... 1287
\rrE ... 1392
\rrEp ... 1413
\rrF ... 1434
\rrf ... 2243
\rrFa ... 1524
\rrFap ... 1529
\rrFc ... 2287
\rrFcp ... 2290

\rrFm . . . . . 2387
\rrFmp . . . . . 2392
\rrFp . . . . . 1453
\rrfp . . . . . 2244
\rrFs . . . . . 1514
\rrFsp . . . . . 1519
\rrFw . . . . . 1472
\rrFwp . . . . . 1493
\rrh . *25*, 1180, 2558, 2559, 2571, 2572
\rrhB . . . . . 1184
\rrhb . . . . . 2249, 2756
\rrhBa . . . . . 1214
\rrhBap . . . . . 1219
\rrhBc . . . . . 2293, 2768
\rrhBcp . . . . 2296, 2769
\rrhBm . . . . . 2397
\rrhBmp . . . . . 2402
\rrhBp . . . . . 1189
\rrhbp . . . . . 2252, 2757
\rrhBs . . . . . 1204
\rrhBsp . . . . . 1209
\rrhBw . . . . . 1194
\rrhBwp . . . . . 1199
\rrhCB . . . . . 2443, 2780
\rrhCBp . . . . 2446, 2781
\rrhCD . . . . . 2431, 2776
\rrhCDp . . . . 2434, 2777
\rrhCF . . . . . 2437, 2778
\rrhCFp . . . . 2440, 2779
\rrhCL . . . . . 2419, 2770
\rrhCLp . . . . 2422, 2771
\rrhCR . . . . . 2413, 2772
\rrhCRp . . . . 2416, 2773
\rrhCU . . . . . 2425, 2774
\rrhCUp . . . . 2428, 2775
\rrhD . . . . . 1224
\rrhd . . . . . 2225, 2752
\rrhDa . . . . . 1350
\rrhDap . . . . . 1371
\rrhDc . . . . . 2281, 2764
\rrhDcp . . . . 2284, 2765
\rrhDm . . . . . 2365
\rrhDmp . . . . . 2376
\rrhDp . . . . . 1245
\rrhdp . . . . . 2228, 2753
\rrhDs . 1308, 2699, 2708
\rrhDsp 1329, 2710, 2719
\rrhDw . . . . . 1266
\rrhDwp . . . . . 1287
\rrhE . . . 1392, 1935, 1944, 1946, 1955, 2355, 2364, 2366, 2375, 2505, 2514, 2516, 2525
\rrhEp . . 1413, 1924, 1933, 1957, 1966, 2344, 2353, 2377, 2386, 2494, 2503, 2527, 2536
\rrhF . . . . . 1434
\rrhf . . . . . 2243, 2754
\rrhFa . . . . . 1524
\rrhFap . . . . . 1529
\rrhFc . . . . . 2287, 2766
\rrhFcp . . . . 2290, 2767
\rrhFm . . . . . 2387
\rrhFmp . . . . . 2392
\rrhFp . . . . . 1453
\rrhfp . . . . . 2246, 2755
\rrhFs . . . . . 1514
\rrhFsp . . . . . 1519
\rrhFw . . . . . 1472
\rrhFwp . . . . . 1493
\rrhL . . . . . 1534
\rrhl . . . . . 2231, 2746
\rrhLa . . . . . 1660
\rrhLap . . . . . 1681
\rrhLc . . . . . 2269, 2758
\rrhLcp . . . . 2272, 2759
\rrhLm . . . . . 2321
\rrhLmp . . . . . 2332
\rrhLp . . . . . 1555
\rrhlp . . . . . 2234, 2747
\rrhLs . 1618, 2655, 2664
\rrhLsp 1639, 2666, 2675
\rrhLw . . . . . 1576
\rrhLwp . . . . . 1597
\rrhM . . . 1702, 1968, 1977, 2001, 2010, 2311, 2320, 2322, 2331, 2461, 2470, 2472, 2481
\rrhMB . . . . . 2547
\rrhMBp . . . . . 2552
\rrhMD . . . . . 2515
\rrhMDp . . . . . 2526
\rrhMF . . . . . 2537
\rrhMFp . . . . . 2542
\rrhML . . . . . 2471
\rrhMLp . . . . . 2482
\rrhMp . . 1723, 1979, 1988, 1990, 1999, 2300, 2309, 2333, 2342, 2450, 2459, 2483, 2492
\rrhMR . . . . . 2449
\rrhMRp . . . . . 2460
\rrhMU . . . . . 2493
\rrhMUp . . . . . 2504
\rrhR . . . . . 1744
\rrhr . . . . . 2237, 2748
\rrhRa . . . . . 1871
\rrhRap . . . . . 1892
\rrhRc . . . . . 2263, 2760
\rrhRcp . . . . 2266, 2761
\rrhRm . . . . . 2299
\rrhRmp . . . . . 2310
\rrhRp . . . . . 1766
\rrhrp . . . . . 2240, 2749
\rrhRs . 1829, 2633, 2642
\rrhRsp 1850, 2644, 2653
\rrhRw . . . . . 1787
\rrhRwp . . . . . 1808
\rrhS . . . . . 1913
\rrhSB . . . . . 2730
\rrhSb . . . . . 2021
\rrhSBp . . . . . 2735
\rrhSbp . . . . . 2026
\rrhSD . . . . . 2698
\rrhSd . . . . . 1945
\rrhSDp . . . . . 2709
\rrhSdp . . . . . 1956
\rrhSF . . . . . 2720
\rrhSf . . . . . 2011
\rrhSFp . . . . . 2725
\rrhSfp . . . . . 2016
\rrhSL . . . . . 2654
\rrhSl . . . . . 1967
\rrhSLp . . . . . 2665
\rrhSlp . . . . . 1978
\rrhSp . . . . . 1918
\rrhSR . . . . . 2632
\rrhSr . . . . . 1989
\rrhSRp . . . . . 2643
\rrhSrp . . . . . 2000

\rrhSU . . . . . 2676
\rrhSu . . . . . 1923
\rrhSUp . . . . . 2687
\rrhSup . . . . . 1934
\rrhTB . . . . . 2622
\rrhTBp . . . . . 2627
\rrhTD . . . . . 2606
\rrhTDp . . . . . 2610
\rrhTF . . . . . 2614
\rrhTFp . . . . . 2618
\rrhTL . . . . . 2582
\rrhTLp . . . . . 2586
\rrhTR . . . . . 2590
\rrhTRp . . . . . 2594
\rrhTU . . . . . 2598
\rrhTUp . . . . . 2602
\rrhU . . . . . 2031
\rrhu . . . . . 2219, 2750
\rrhUa . . . . . 2159
\rrhUap . . . . . 2180
\rrhUc . . . . . 2275, 2762
\rrhUcp . . . . 2278, 2763
\rrhUm . . . . . 2343
\rrhUmp . . . . . 2354
\rrhUp . . . . . 2075
\rrhup . . . . . 2222, 2751
\rrhUs . . . . . 2117, 2138, 2677, 2686
\rrhUsp . . . . . 2148, 2158, 2688, 2697
\rrhUw . . . . . 2052
\rrhUwp . . . . . 2096
\rrhx . . . . . 2201, 2740
\rrhxp . . . . . 2204, 2741
\rrhy . . . . . 2207, 2742
\rrhyp . . . . . 2210, 2743
\rrhz . . . . . 2213, 2744
\rrhzp . . . . . 2216, 2745
\rrL . . . . . 1534
\rrl . . . . . 2231
\rrLa . . . . . 1660
\rrLap . . . . . 1681
\rrLc . . . . . 2269
\rrLcp . . . . . 2272
\rrLm . . . . . 2321
\rrLmp . . . . . 2332
\rrLp . . . . . 1555
\rrlp . . . . . 2232
\rrLs . . . . . 1618
\rrLsp . . . . . 1639
\rrLw . . . . . 1576
\rrLwp . . . . . 1597
\Rrm 90, 120, 188, 196, 270, 296, 548, 727
\rrM . . . . . 1702
\rrMB . . . . . 2547
\rrMBp . . . . . 2552
\rrMD . . . . . 2515
\rrMDp . . . . . 2526
\rrMF . . . . . 2537
\rrMFp . . . . . 2542
\rrML . . . . . 2471
\rrMLp . . . . . 2482
\rrMp . . . . . 1723
\rrMR . . . . . 2449
\rrMRp . . . . . 2460
\rrMU . . . . . 2493
\rrMUp . . . . . 2504
\rrR . . . . . 1744
\rrr . . . . . 2237
\rrRa . . . . . 1871
\rrRap . . . . . 1892
\rrRc . . . . . 2263
\rrRcp . . . . . 2266
\rrRm . . . . . 2299
\rrRmp . . . . . 2310
\rrRp . . . . . 1766
\rrrp . . . . . 2238
\rrRs . . . . . 1829
\rrRsp . . . . . 1850
\rrRw . . . . . 1787
\rrRwp . . . . . 1808
\rrS . . . . . 1913
\rrSB . . . . . 2730
\rrSb . . . . . 2021
\rrSBp . . . . . 2735
\rrSbp . . . . . 2026
\rrSD . . . . . 2698
\rrSd . . . . . 1945
\rrSDp . . . . . 2709
\rrSdp . . . . . 1956
\rrSF . . . . . 2720
\rrSf . . . . . 2011
\rrSFp . . . . . 2725
\rrSfp . . . . . 2016
\rrSL . . . . . 2654
\rrSl . . . . . 1967
\rrSLp . . . . . 2665
\rrSlp . . . . . 1978
\rrSp . . . . . 1918
\rrSR . . . . . 2632
\rrSr . . . . . 1989
\rrSRp . . . . . 2643
\rrSrp . . . . . 2000
\rrSU . . . . . 2676
\rrSu . . . . . 1923
\rrSUp . . . . . 2687
\rrSup . . . . . 1934
\Rrt 89, 119, 180, 263, 289, 538, 656, 728
\rrTB . . . . . 2622
\rrTBp . . . . . 2627
\rrTD . . . . . 2606
\rrTDp . . . . . 2610
\rrTF . . . . . 2614
\rrTFp . . . . . 2618
\rrTL . . . . . 2582
\rrTLp . . . . . 2586
\rrTR . . . . . 2590
\rrTRp . . . . . 2594
\rrTU . . . . . 2598
\rrTUp . . . . . 2602
\rrU . . . . . 2031
\rru . . . . . 2219
\rrUa . . . . . 2159
\rrUap . . . . . 2180
\rrUc . . . . . 2275
\rrUcp . . . . . 2278
\rrUm . . . . . 2343
\rrUmp . . . . . 2354
\rrUp . . . . . 2075
\rrup . . . . . 2220
\rrUs . . . . . 2117
\rrUsp . . . . . 2138
\rrUw . . . . . 2052
\rrUwp . . . . . 2096
\rrx . . . . . 2201
\rrxp . . . . . 2204
\rry . . . . . 2207
\rryp . . . . . 2210
\rrz . . . . . 2213
\rrzp . . . . . 2216
\Rs . . . . . *27*
\Rubik . . . . . *25*, 1180
\RubikB . . . . . 1184
\Rubikb . . . . . 2249
\RubikBa . . . . . 1214
\RubikBap . . . . . 1219
\RubikBc . . . . . 2293
\RubikBcp . . . . . 2296

\RubikBm . . . . . 2397
\RubikBmp . . . . . 2402
\RubikBp . . . . . 1189
\Rubikbp . . . . . 2254
\RubikBs . . . . . 1204
\RubikBsp . . . . . 1209
\RubikBw . . . . . 1194
\RubikBwp . . . . . 1199
\RubikCB . . . . . 2443
\RubikCBp . . . . . 2446
\RubikCD . . . . . 2431
\RubikCDp . . . . . 2434
\RubikCF . . . . . 2437
\RubikCFp . . . . . 2440
\RubikCL . . . . . 2419
\RubikCLp . . . . . 2422
\RubikCR . . . . . 2413
\RubikCRp . . . . . 2416
\RubikCU . . . . . 2425
\Rubikcube . . . . . 16
\rubikcube . . . . . *11*, 15
\RubikCubeGray . . . . 152
\RubikCubeGrayAll . 174
\RubikCubeGrayWB . . 164
\RubikCubeGrayWY . . 154
\RubikCubeGrey . . . . 144
\RubikCubeGrey.. . . *19*
\RubikCubeGreyAll . . . . . *19*, *23*, 173
\RubikCubeGreyWB *23*, 144
\RubikCubeGreyWY *23*, 144
\RubikCubeSolved *21*, 175
\RubikCubeSolved.. . *19*
\RubikCubeSolvedWB . . . . . *22*, 175
\RubikCubeSolvedWY . . . . . *21*, 175
\RubikCUp . . . . . 2428
\RubikD . . . . . 1224
\Rubikd . . . . . 2225
\RubikDa . . . . . 1350
\RubikDap . . . . . 1371
\RubikDc . . . . . 2281
\RubikDcp . . . . . 2284
\RubikDm . . . . . 2365
\RubikDmp . . . . . 2376
\RubikDp . . . . . 1245
\Rubikdp . . . . . 2230
\RubikDs . . . . . 1308
\RubikDsp . . . . . 1329
\RubikDw . . . . . 1266
\RubikDwp . . . . . 1287
\RubikE . . . . . 1392
\RubikEp . . . . . 1413
\RubikF . . . . . 1434
\Rubikf . . . . . 2243
\RubikFa . . . . . 1524
\RubikFace.. . . . . *19*
\RubikFaceBack . *20*, 78, 103, 150, 162
\RubikFaceBackAll *20*, 108, 133, 143, 171
\RubikFaceDown . . . . *20*, 78, 93, 148, 160
\RubikFaceDownAll *20*, 108, 128, 139, 169
\RubikFaceFront . . . *20*, 78, 83, 149, 161
\RubikFaceFrontAll . . . . . *20*, 108, 113, 142, 170
\RubikFaceLeft . . . . *20*, 78, 98, 146, 158
\RubikFaceLeftAll *20*, 108, 123, 140, 167
\RubikFaceRight . . . *20*, 78, 88, 145, 157
\RubikFaceRightAll . . . . . *20*, 108, 118, 141, 166
\RubikFaceUp . . . . . *20*, 78, 78, 147, 159
\RubikFaceUpAll *20*, 108, 108, 138, 168
\RubikFap . . . . . 1529
\RubikFc . . . . . 2287
\RubikFcp . . . . . 2290
\RubikFm . . . . . 2387
\RubikFmp . . . . . 2392
\RubikFp . . . . . 1453
\Rubikfp . . . . . 2248
\RubikFs . . . . . 1514
\RubikFsp . . . . . 1519
\RubikFw . . . . . 1472
\RubikFwp . . . . . 1493
\RubikL . . . . . 1534
\Rubikl . . . . . 2231
\RubikLa . . . . . 1660
\RubikLap . . . . . 1681
\RubikLc . . . . . 2269
\RubikLcp . . . . . 2272
\RubikLm . . . . . 2321
\RubikLmp . . . . . 2332
\RubikLp . . . . . 1555
\Rubiklp . . . . . 2236
\RubikLs . . . . . 1618
\RubikLsp . . . . . 1639
\RubikLw . . . . . 1576
\RubikLwp . . . . . 1597
\RubikM . . . . . 1702
\RubikMB . . . . . 2547
\RubikMBp . . . . . 2552
\RubikMD . . . . . 2515
\RubikMDp . . . . . 2526
\RubikMF . . . . . 2537
\RubikMFp . . . . . 2542
\RubikML . . . . . 2471
\RubikMLp . . . . . 2482
\RubikMp . . . . . 1723
\RubikMR . . . . . 2449
\RubikMRp . . . . . 2460
\RubikMU . . . . . 2493
\RubikMUp . . . . . 2504
\RubikR . . . . . 1744
\Rubikr . . . . . 2237
\RubikRa . . . . . 1871
\RubikRap . . . . . 1892
\RubikRc . . . . . 2263
\RubikRcp . . . . . 2266
\RubikRm . . . . . 2299
\RubikRmp . . . . . 2310
\RubikRp . . . . . 1766
\Rubikrp . . . . . 2242
\RubikRs . . . . . 1829
\RubikRsp . . . . . 1850
\RubikRw . . . . . 1787
\RubikRwp . . . . . 1808
\RubikS . . . . . 1913
\RubikSB . . . . . 2730
\RubikSb . . . . . 2021
\RubikSBp . . . . . 2735
\RubikSbp . . . . . 2026
\RubikSD . . . . . 2698
\RubikSd . . . . . 1945
\RubikSDp . . . . . 2709
\RubikSdp . . . . . 1956
\RubikSF . . . . . 2720
\RubikSf . . . . . 2011
\RubikSFp . . . . . 2725
\RubikSfp . . . . . 2016

\RubikSideBarLength 782
\RubikSidebarLength . *42*, 775, 776, 779
\RubikSideBarSep .. 783
\RubikSidebarSep .. . *42*, 775, 777, 780
\RubikSideBarWidth . 781
\RubikSidebarWidth . . *42*, 775, 775, 778
\RubikSL ..... 2654
\RubikSl ..... 1967
\RubikSlice.. ..... *19*
\RubikSliceBottomL . 178
\RubikSliceBottomR . 178
\RubikSliceBottomX . *23*
\RubikSliceEquatorL 198
\RubikSliceEquatorR 194
\RubikSliceMiddleL . 178
\RubikSliceMiddleR . 178
\RubikSliceMiddleX . *23*
\RubikSliceTopL ... 178
\RubikSliceTopR ... 178
\RubikSliceTopX ... *23*
\RubikSLp ..... 2665
\RubikSlp ..... 1978
\RubikSolvedConfig . ..... *21*, 165, 173, 175, 177
\RubikSp ..... 1918
\RubikSR ..... 2632
\RubikSr ..... 1989
\RubikSRp ..... 2643
\RubikSrp ..... 2000
\RubikSU ..... 2676
\RubikSu ..... 1923
\RubikSUp ..... 2687
\RubikSup ..... 1934
\RubikTB ..... 2622
\RubikTBp ..... 2627
\RubikTD ..... 2606
\RubikTDp ..... 2610
\RubikTF ..... 2614
\RubikTFp ..... 2618
\RubikTL ..... 2582
\RubikTLp ..... 2586
\RubikTR ..... 2590
\RubikTRp ..... 2594
\RubikTU ..... 2598
\RubikTUp ..... 2602
\RubikU ..... 2031
\Rubiku ..... 2219
\RubikUa ..... 2159
\RubikUap ..... 2180
\RubikUc ..... 2275
\RubikUcp ..... 2278
\RubikUm ..... 2343
\RubikUmp ..... 2354
\RubikUp ..... 2075
\Rubikup ..... 2224
\RubikUs ... 2117, 2138
\RubikUsp ..... 2150
\RubikUw ..... 2052
\RubikUwp ..... 2096
\Rubikx ..... 2201
\Rubikxp ..... 2204
\Rubiky ..... 2207
\Rubikyp ..... 2210
\Rubikz ..... 2213
\Rubikzp ..... 2216
\rule ..... 1174
\Rw ..... *27*
\rx .. 529, 533, 534, 536, 537, 539, 540, 543, 544, 546, 547, 549, 550, 553, 554, 556, 557, 559, 560
\ry .. 530, 533, 534, 536, 537, 539, 540, 543, 544, 546, 547, 549, 550, 553, 554, 556, 557, 559, 560

## S

\S ..... *26*
\SB ..... *27*
\Sb ..... *27*
\SD ..... *27*
\Sd ..... *27*
\SequenceBraceA . *47*, 74
\SequenceBraceAF *47*, 76
\SequenceBraceB . *47*, 74
\SequenceBraceBF *47*, 77
\SequenceInfo ..... 70
\SequenceLong ..... 70
\SequenceName ..... 70
\SequenceShort .... 70
\setlength ..... . 29, 30, 40, 41, 42, 1172, 1173, 1176
\SF ..... *27*
\Sf ..... *27*
\ShowCube ..... *16*, 27
\ShowCubeF ..... *16*, 27
\ShowSequence ... *45*, 52
\ShowSequenceF .. *45*, 52
\ShowSequencef .. *45*, 52
\side@barB ... 658, 659, 660, 677, 678, 679, 696, 697, 698, 715, 716, 717, 734, 735, 736, 753, 754, 755, 784, 813, 869, 870, 871
\side@barL ... 650, 651, 652, 669, 670, 671, 688, 689, 690, 707, 708, 709, 726, 727, 728, 745, 746, 747, 784, 786, 861, 862, 863
\side@barR ... 654, 655, 656, 673, 674, 675, 692, 693, 694, 711, 712, 713, 730, 731, 732, 749, 750, 751, 784, 795, 915, 916, 917
\side@barRubikBbackLD ..... 989, 1001, 1002, 1003
\side@barRubikBbackRD . 949, 961, 962, 963
\side@barRubikLbackLD . 972, 984, 985, 986
\side@barRubikLbackLU . 878, 890, 891, 892
\side@barRubikRbackRD . 932, 944, 945, 946
\side@barRubikRbackRU . 822, 834, 835, 836
\side@barRubikTbackLU . 897, 909, 910, 911

\side@barRubikTbackRU . 841, 853, 854, 855
\side@barT . . . 646, 647, 648, 665, 666, 667, 684, 685, 686, 703, 704, 705, 722, 723, 724, 741, 742, 743, 784, 804, 923, 924, 925
\SL . . . . . *27*
\Sl . . . . . *27*
\Sp . . . . . *26*
\space . . . . . 5
\Square . . . . 2565, 2578
\SquareB . . . . . 1184
\SquareBa . . . . . 1214
\SquareBap . . . . . 1219
\SquareBm . . . . . . 2398, 2399, 2400
\SquareBmp . . . . . . 2403, 2404, 2405
\SquareBp . . . . . 1189
\SquareBs . . . . . 1204
\SquareBsp . . . . . 1209
\SquareBw . . . . . 1194
\SquareBwp . . . . . 1199
\SquareD . . . . . 1224
\SquareDa . . . . . 1350
\SquareDap . . . . . 1371
\SquareDp . . . . . 1245
\SquareDs . . 1308, 2704
\SquareDsp . 1329, 2715
\SquareDw . . . . . 1266
\SquareDwp . . . . . 1287
\SquareE 1392, 1940, 1951, 2360, 2371, 2510, 2521
\SquareEp 1413, 1929, 1962, 2349, 2382, 2499, 2532
\SquareF . . . . . 1434
\SquareFa . . . . . 1525, 1526, 1527
\SquareFap . . . . . 1530, 1531, 1532
\SquareFm . . . . . 2388, 2389, 2390
\SquareFmp . . . . . 2393, 2394, 2395
\SquareFp . . . . . 1453
\SquareFs . . . . . 1515, 1516, 1517
\SquareFsp . . . . . 1520, 1521, 1522
\SquareFw . . . . . 1472
\SquareFwp . . . . . 1493
\SquareL . . . . . 1534
\SquareLa . . . . . 1660
\SquareLap . . . . . 1681
\SquareLp . . . . . 1555
\SquareLs . . 1618, 2660
\SquareLsp . 1639, 2671
\SquareLw . . . . . 1576
\SquareLwp . . . . . 1597
\SquareM 1702, 1973, 2006, 2316, 2327, 2466, 2477
\SquareMB . . . . . 2548, 2549, 2550
\SquareMBp . . . . . 2553, 2554, 2555
\SquareMF . . . . . 2538, 2539, 2540
\SquareMFp . . . . . 2543, 2544, 2545
\SquareMp 1723, 1984, 1995, 2305, 2338, 2455, 2488
\SquareR . . . . . 1744
\SquareRa . . . . . 1871
\SquareRap . . . . . 1892
\SquareRp . . . . . 1766
\SquareRs . . 1829, 2638
\SquareRsp . 1850, 2649
\SquareRw . . . . . 1787
\SquareRwp . . . . . 1808
\SquareS . . . . . 1913
\SquareSB . . . . . 2731, 2732, 2733
\SquareSb . . . . . 2022, 2023, 2024
\SquareSBp . . . . . 2736, 2737, 2738
\SquareSbp . . . . . 2027, 2028, 2029
\SquareSF . . . . . 2721, 2722, 2723
\SquareSf . . . . . 2012, 2013, 2014
\SquareSFp . . . . . 2726, 2727, 2728
\SquareSfp . . . . . 2017, 2018, 2019
\SquareSp . . . . . 1918
\SquareTB . . . . . 2623, 2624, 2626
\SquareTBp . . . . . 2628, 2629, 2631
\SquareU . . . . . 2031
\SquareUa . . . . . 2159
\SquareUap . . . . . 2180
\SquareUp . . . . . 2075
\SquareUs . . 2117, 2682
\SquareUsp . 2138, 2693
\SquareUw . . . . . 2052
\SquareUwp . . . . . 2096
\SR . . . . . *27*
\Sr . . . . . *27*
\string . . . . . 1179
\SU . . . . . *27*
\Su . . . . . *27*

## T

\TB . . . . . *28*
\TD . . . . . *28*
\textCubieLD . . . . . *39*, 1085, 1103
\textCubieLU . . . . . *39*, 1085, 1112
\textCubieRD . . . . . *39*, 1085, 1094
\textCubieRU . . . . . *39*, 1085, 1085
\textRubik . . . *25*, 1180
\textRubikB . . . . . 1184
\textRubikb . . . . . 2756
\textRubikBa . . . . . 1214
\textRubikBap . . . . 1219
\textRubikBc . . . . . 2768
\textRubikBcp . . . . 2769
\textRubikBm . . . . . 2397
\textRubikBmp . . . . 2402
\textRubikBp . . . . . 1189
\textRubikbp . . . . . 2757
\textRubikBs . . . . . 1204
\textRubikBsp . . . . 1209
\textRubikBw . . . . . 1194
\textRubikBwp . . . . 1199
\textRubikCB . . . . . 2780

\textRubikCBp .... 2781
\textRubikCD ..... 2776
\textRubikCDp .... 2777
\textRubikCF ..... 2778
\textRubikCFp .... 2779
\textRubikCL ..... 2770
\textRubikCLp .... 2771
\textRubikCR ..... 2772
\textRubikCRp .... 2773
\textRubikCU ..... 2774
\textRubikCUp .... 2775
\textRubikD ..... 1224
\textRubikd ..... 2752
\textRubikDa ..... 1350
\textRubikDap .... 1371
\textRubikDc ..... 2764
\textRubikDcp .... 2765
\textRubikDm ..... 2365
\textRubikDmp .... 2376
\textRubikDp ..... 1245
\textRubikdp ..... 2753
\textRubikDs ..... 1308
\textRubikDsp .... 1329
\textRubikDw ..... 1266
\textRubikDwp .... 1287
\textRubikE ..... 1392
\textRubikEp ..... 1413
\textRubikF ..... 1434
\textRubikf ..... 2754
\textRubikFa ..... 1524
\textRubikFap .... 1529
\textRubikFc ..... 2766
\textRubikFcp .... 2767
\textRubikFm ..... 2387
\textRubikFmp .... 2392
\textRubikFp ..... 1453
\textRubikfp ..... 2755
\textRubikFs ..... 1514
\textRubikFsp .... 1519
\textRubikFw ..... 1472
\textRubikFwp .... 1493
\textRubikL ..... 1534
\textRubikl ..... 2746
\textRubikLa ..... 1660
\textRubikLap .... 1681
\textRubikLc ..... 2758
\textRubikLcp .... 2759
\textRubikLm ..... 2321
\textRubikLmp .... 2332
\textRubikLp ..... 1555
\textRubiklp ..... 2747
\textRubikLs ..... 1618
\textRubikLsp .... 1639
\textRubikLw ..... 1576
\textRubikLwp .... 1597
\textRubikM ..... 1702
\textRubikMB ..... 2547
\textRubikMBp .... 2552
\textRubikMD ..... 2515
\textRubikMDp .... 2526
\textRubikMF ..... 2537
\textRubikMFp .... 2542
\textRubikML ..... 2471
\textRubikMLp .... 2482
\textRubikMp ..... 1723
\textRubikMR ..... 2449
\textRubikMRp .... 2460
\textRubikMU ..... 2493
\textRubikMUp .... 2504
\textRubikR ..... 1744
\textRubikr ..... 2748
\textRubikRa ..... 1871
\textRubikRap .... 1892
\textRubikRc ..... 2760
\textRubikRcp .... 2761
\textRubikRm ..... 2299
\textRubikRmp .... 2310
\textRubikRp ..... 1766
\textRubikrp ..... 2749
\textRubikRs ..... 1829
\textRubikRsp .... 1850
\textRubikRw ..... 1787
\textRubikRwp .... 1808
\textRubikS ..... 1913
\textRubikSB ..... 2730
\textRubikSb ..... 2021
\textRubikSBp .... 2735
\textRubikSbp .... 2026
\textRubikSD ..... 2698
\textRubikSd ..... 1945
\textRubikSDp .... 2709
\textRubikSdp .... 1956
\textRubikSF ..... 2720
\textRubikSf ..... 2011
\textRubikSFp .... 2725
\textRubikSfp .... 2016
\textRubikSL ..... 2654
\textRubikSl ..... 1967
\textRubikSLp .... 2665
\textRubikSlp .... 1978
\textRubikSp ..... 1918
\textRubikSR ..... 2632
\textRubikSr ..... 1989
\textRubikSRp .... 2643
\textRubikSrp .... 2000
\textRubikSU ..... 2676
\textRubikSu ..... 1923
\textRubikSUp .... 2687
\textRubikSup .... 1934
\textRubikTB ..... 2622
\textRubikTBp .... 2627
\textRubikTD ..... 2606
\textRubikTDp .... 2610
\textRubikTF ..... 2614
\textRubikTFp .... 2618
\textRubikTL ..... 2582
\textRubikTLp .... 2586
\textRubikTR ..... 2590
\textRubikTRp .... 2594
\textRubikTU ..... 2598
\textRubikTUp .... 2602
\textRubikU ..... 2031
\textRubiku ..... 2750
\textRubikUa ..... 2159
\textRubikUap .... 2180
\textRubikUc ..... 2762
\textRubikUcp .... 2763
\textRubikUm ..... 2343
\textRubikUmp .... 2354
\textRubikUp ..... 2075
\textRubikup ..... 2751
\textRubikUs ..... 2117
\textRubikUsp .... 2138
\textRubikUw ..... 2052
\textRubikUwp .... 2096
\textRubikx ..... 2740
\textRubikxp ..... 2741
\textRubiky ..... 2742
\textRubikyp ..... 2743
\textRubikz ..... 2744
\textRubikzp ..... 2745
\textsc ..... 15, 16
\TF ..... *28*
\the ..... 32, 45
\thislevelcount ... 53
\thislevelitem .... 52
\TL ..... *28*
\TR ..... *28*
\TU ..... *28*
\typeout ..... 7, 10, 13

**U**

\U . . . . . *25*
\u . . . . . *28*
\Ua . . . . . *27*
\Uc . . . . . *29*
\Ulb 81, 111, 251, 415, 447, 665, 705, 923
\Ulm . . . . . 80, 110, 244, 408, 437, 704
\Ult . . . 79, 109, 237, 401, 427, 703, 724
\Um . . . . . *26*
\Umb . . . 81, 111, 253, 417, 450, 666, 924
\Umm 80, 110, 246, 410, 440
\Umt . . . . . 79, 109, 239, 403, 430, 723
\underbrace . . . . . 75, 77
\Up . . . . . *25*
\Urb 81, 111, 255, 419, 453, 667, 684, 925
\Urm . . . . . 80, 110, 248, 412, 443, 685
\Urt . . . 79, 109, 241, 405, 433, 686, 722
\Us . . . . . *27*
\usefont . . . . . 24, 25
\Uw . . . . . *27*
\ux . . 424, 428, 429, 431, 432, 434, 435, 438, 439, 441, 442, 444, 445, 448, 449, 451, 452, 454, 455
\uy . . 425, 428, 429, 431, 432, 434, 435, 438, 439, 441, 442, 444, 445, 448, 449, 451, 452, 454, 455

**X**

\x . . . . . *28*, 52, 56, 62, 67, 1010, 1013
\xcount . . 53, 56, 62, 67

**Y**

\y . . . . . *28*, 1011, 1013

**Z**

\z . . . . . *28*, 1012, 1013