

NAME

rgbgfx — Game Boy graphics converter

SYNOPSIS

```
rgbgfx [-DfFhPTv] [-o outfile] [-d depth] [-p palfile] [-t mapfile]
      [-x tiles] file
```

DESCRIPTION

The **rgbgfx** program converts PNG images into the Nintendo Game Boy's planar tile format. The arguments are as follows:

- D** Debug features are enabled.
- f** Fix the input PNG file to be a correctly indexed image.
- F** Same as **-f**, but additionally, the input PNG file is fixed to have its parameters match the command line's parameters.
- d** *depth*
The bitdepth of the output image (either 1 or 2). By default, the bitdepth is 2 (two bits per pixel).
- h** Lay out tiles horizontally rather than vertically.
- o** *outfile*
The name of the output file.
- p** *palfile*
Raw bytes (8 bytes for two bits per pixel, 4 bytes for one bit per pixel) containing the RGB15 values in the little-endian byte order and then ordered from lightest to darkest.
- P** Same as **-p**, but the pallete file output name is made by taking the input filename, removing the file extension, and appending `.pal`.
- t** *mapfile*
If any tiles are the same, don't place the repeat tiles in the output file, and make a tilemap file.
- T** Same as **-t**, but the tilemap file output name is made by taking the input filename, removing the file extension, and appending `.tilemap`.
- u** Truncate repeated tiles. Useful with tilemaps.
- v** Verbose. Print errors when the command line parameters and the parameters in the PNG file don't match.
- x** *tiles*
Trim the end of the output file by this many tiles.

EXAMPLES

The following will take a PNG file with a bitdepth of 1, 2, or 8, and output planar 2bpp data:

```
$ rgbgfx -o out.2bpp in.png
```

The following creates a planar 2bpp file with only unique tiles, and its tilemap `out.tilemap`:

```
$ rgbgfx -T -u -o out.2bpp in.png
```

The following will do nothing:

```
$ rgbgfx in.png
```

SEE ALSO

`rgbds(7)`, `rgbasm(1)`, `rgblink(1)`, `rgbfix(1)`, `gbz80(7)`

HISTORY

rgbgfx was created by stag019 to be included in RGBDS. It is now maintained by a number of contributors at <https://github.com/rednex/rgbds>.