

NAME

rgbfix — Game Boy header utility and checksum fixer

SYNOPSIS

```
rgbfix [-h]OsVv] [-C | -c] [-f fix_spec] [-i game_id] [-k licensee_str]
        [-L logo_file] [-l licensee_id] [-m mbc_type] [-n rom_version]
        [-o out_file] [-p pad_value] [-r ram_size] [-t title_str] [file ...]
```

DESCRIPTION

The **rgbfix** program changes headers of Game Boy ROM images, typically generated by *rgbblink*(1), though it will work with *any* Game Boy ROM. It also performs other correctness operations, such as padding. **rgbfix** only changes the fields for which it has values specified. Developers are advised to fill those fields with 0x00 bytes in their source code before running **rgbfix**, and to have already populated whichever fields they don't specify using **rgbfix**.

The input *asmfile* can be a path to a file, or - to read from standard input.

Note that options can be abbreviated as long as the abbreviation is unambiguous: `--color-o` is `--color-only`, but `--color` is invalid because it could also be `--color-compatible`. Options later in the command line override those set earlier. Accepted options are as follows:

- C, `--color-only`
Set the Game Boy Color-only flag (0x143) to 0xC0. This overrides -c if it was set prior.
- c, `--color-compatible`
Set the Game Boy Color-compatible flag: (0x143) to 0x80. This overrides -c if it was set prior.
- f *fix_spec*, `--fix-spec fix_spec`
Fix certain header values that the Game Boy checks for correctness. Alternatively, intentionally trash these values by writing their binary inverse instead. *fix_spec* is a string containing any combination of the following characters:
 - l Fix the Nintendo logo (0x104–0x133).
 - L Trash the Nintendo logo.
 - h Fix the header checksum (0x14D).
 - H Trash the header checksum.
 - g Fix the global checksum (0x14E–0x14F).
 - G Trash the global checksum.
- h, `--help`
Print help text for the program and exit.
- i *game_id*, `--game-id game_id`
Set the game ID string (0x13F–0x142) to a given string. If it's longer than 4 chars, it will be truncated, and a warning emitted.
- j, `--non-japanese`
Set the non-Japanese region flag (0x14A) to 0x01.
- k *licensee_str*, `--new-licensee licensee_str`
Set the new licensee string (0x144–0x145) to a given string. If it's longer than 2 chars, it will be truncated, and a warning emitted.
- L *logo_file*, `--logo logo_file`
Specify a logo file to use instead of the official Nintendo logo. The file must be 48 bytes of 1bpp tile data; the source image should be 48 pixels wide and 8 pixels tall.
- l *licensee_id*, `--old-licensee licensee_id`
Set the old licensee code (0x14B) to a given value from 0 to 0xFF. This value is deprecated and should be set to 0x33 in all new software.

- m *mbc_type*, --mbc-type *mbc_type*
Set the MBC type (*0x147*) to a given value from 0 to 0xFF.

This value may also be an MBC name. The list of accepted names can be obtained by passing `help` as the argument. Any amount of whitespace (space and tabs) is allowed around plus signs, and the order of "components" is free, as long as the MBC name is first. There are special considerations to take for the TPP1 mapper; see the “TPP1” section below.
- n *rom_version*, --rom-version *rom_version*
Set the ROM version (*0x14C*) to a given value from 0 to 0xFF.
- O, --overwrite
Allow overwriting different non-zero bytes in the header without a warning being emitted.
- o *out_file*, --output *out_file*
Write the modified ROM image to the given file, or '-' to write to standard output. If not specified, the input files are modified in-place, or written to standard output if read from standard input.
- p *pad_value*, --pad-value *pad_value*
Pad the ROM image to a valid size with a given pad value from 0 to 255 (0xFF). **rgbfix** will automatically pick a size from 32 KiB, 64 KiB, 128 KiB, ..., 8192 KiB. The cartridge size byte (*0x148*) will be changed to reflect this new size. The recommended padding value is 0xFF, to speed up writing the ROM to flash chips, and to avoid "nop slides" into VRAM.
- r *ram_size*, --ram-size *ram_size*
Set the RAM size (*0x149*) to a given value from 0 to 0xFF.
- s, --sgb-compatible
Set the SGB flag (*0x146*) to 0x03. This flag will be ignored by the SGB unless the old licensee code is 0x33! If this is given as well as -l, but is not set to 0x33, a warning will be printed.
- t *title*, --title *title*
Set the title string (*0x134–0x143*) to a given string. If the title is longer than the max length, it will be truncated, and a warning emitted. The max length is 11 characters if the game ID (-i) is specified, 15 characters if the CGB flag (-c or -C) is specified but the game ID is not, and 16 characters otherwise.
- V, --version
Print the version of the program and exit.
- v, --validate
Equivalent to -f lhg.

EXAMPLES

Most values in the ROM header do not matter to the actual console, and most are seldom useful anyway. The bare minimum requirements for a workable program are the header checksum, the Nintendo logo, and (if needed) the CGB/SGB flags. It is a good idea to pad the image to a valid size as well (“valid” meaning a power of 2, times 32 KiB).

The following will make a plain, non-color Game Boy game without checking for a valid size:

```
$ rgbfix -v foo.gb
```

The following will make a SGB-enabled, color-enabled game with a title of “foobar”, and pad it to a valid size. (The Game Boy itself does not use the title, but some emulators or ROM managers do.)

```
$ rgbfix -vcs -l 0x33 -p 255 -t foobar baz.gb
```

The following will duplicate the header of the game “Survival Kids”, sans global checksum:

```
$ rgbfix -cjsv -k A4 -l 0x33 -m 0x1B -p 0xFF -r 3 -t SURVIVALKIDAVKE SurvivalKids.gbc
```

TPP1

TPP1 is a homebrew mapper designed as a functional superset of the common traditional MBCs, allowing larger ROM and RAM sizes combined with other hardware features. Its specification, as well as more resources, can be found online at <https://github.com/aaaaaa123456789/tpp1>.

MBC name

The MBC name for TPP1 is more complex than standard mappers. It must be followed with the revision number, of the form `major.minor`, where both `major` and `minor` are decimal, 8-bit integers. There may be any amount of spaces or underscores between TPP1 and the revision number. **rgbfix** only supports 1.x revisions, and will reject everything else.

Like other mappers, the name may be followed with a list of optional, '+'-separated features; however, RAM should not be specified, as the TPP1 mapper implicitly requests RAM if a non-zero RAM size is specified. Therefore, **rgbfix** will ignore the RAM feature on a TPP1 mapper with a warning.

Special considerations

TPP1 overwrites the byte at `0x14A`, usually indicating the region destination (see `-j`), with one of its three identification bytes. Therefore, **rgbfix** will warn about and ignore `-j` if used in combination with TPP1.

BUGS

Please report bugs on *GitHub*: <https://github.com/gbdev/rgbds/issues>.

SEE ALSO

rgbasm(1), *rgbink*(1), *rgbgfx*(1), *gbz80*(7), *rgbds*(7)

HISTORY

rgbfix was originally written by Carsten Sørensen as a standalone program called GBFix, which was then packaged in ASMotor, and was later repackaged in RGBDS by Justin Lloyd. It is now maintained by a number of contributors at <https://github.com/gbdev/rgbds>.