

NAME

rgbasm — Game Boy assembler

SYNOPSIS

```
rgbasm [-Ehvw] [-b chars] [-D name[=value]] [-g chars] [-i path] [-o outfile]  
        [-p pad_value] file
```

DESCRIPTION

The **rgbasm** program creates an object file from an assembly source file. Its arguments are as follows:

-b *chars*

Change the two characters used for binary constants. The defaults are 01.

-D *name*[=*value*]

Add string symbol to the compiled source code. This is equivalent to *name* EQU *"value"* in code. If a value is not specified, a value of 1 is given.

-E Export all labels, including unreferenced and local labels.

-g *chars*

Change the four characters used for binary constants. The defaults are 0123.

-h By default, **rgbasm** inserts a ‘nop’ instruction immediately after any ‘halt’ instruction. The **-h** option disables this behavior.

-i *path*

Add an include path.

-o *outfile*

Write an object file to the given filename.

-p *pad_value*

When padding an image, pad with this value. The default is 0x00.

-v Be verbose.

-w Disable warning output.

EXAMPLES

Assembling a basic source file is simple:

```
$ rgbasm -o bar.o foo.asm
```

The resulting object file is not yet a usable ROM image — it must first be run through *rgblink*(1) and *rgbfix*(1).

SEE ALSO

rgbasm(5), *rgbfix*(1), *rgblink*(1), *rgbds*(5), *rgbds*(7), *gbz80*(7)

rgbasm assembly commands: <https://rednex.github.io/rgbds/asm.htm>

HISTORY

rgbasm was originally written by Carsten Sørensen as part of the ASMotor package, and was later packaged in RGBDS by Justin Lloyd. It is now maintained by a number of contributors at <https://github.com/rednex/rgbds>.