gputils 0.11.7

James Bowman

October 19, 2003

Contents

Chapter 1

Chapter 2

gpasm

2.1 Running gpasm

The general syntax for running gpasm is

```
gpasm [options] asm-file
```

Where options can be

2.1.1 Using gpasm with “make”

On most operating systems, you can build a project using the make utility. To use gpasm with make, you might have a “makefile” like this:

```
tree.hex: tree.asm treedef.inc
        gpasm tree.asm
```

This will rebuild “tree.hex” whenever either of the “tree.asm” or “treedef.inc” files change. A more comprehensive example of using gpasm with makefiles is included as example1 in the gpasm source distribution.

2.1.2 Dealing with errors

gpasm doesn’t specifically create an error file. This can be a problem if you want to keep a record of errors, or if your assembly produces so many errors e file ”e962j /R198399ef.inc

gpasm

base	general syntax	21 decimal written as
binary	B’[01]*’	B’10101’
octal	O’[0-7]*’	O’25’
decimal		

2.2.5 Processor header files

gputils distrib

__CONFIG

```
__CONFIG <expression>
```

Sets the PIC processor's configuration fuses.

__IDLOCS

```
__IDLOCS <expression> or __IDLOCS <expression1>,<expression2>
```

Sets the PIC processor's identification locations. For 12 and 14 bit processors, the four id locations are set to the hexadecimal value of expression. For 18cxx de

MACRO

```
<label> MACRO [ <symbol> [ , <symbol> ]* ]
```

Declares a macro with name <label>. gpasm replaces any occurrences of <symbol> in the macro definition with the parameters given at macro invocation.

See also: LOCAL, ENDM

MESSG

```
MESSG <string>
```

Writes <string> to the list file, and to the standard error output.

See also: ERROR

NOEXPAND

UDATA_ACS

```
<label> UDATA_ACS <expression>
```

Only for relocatable mode. Creates a new uninitialized accessbank data section in the output object file. <label> specifies the name of the section. If <label> is not specified the default name ".udata_acs" will be used. <expression> is optional and specifies the absolute address of the section.

See also: CODE, IDATA, UDATA

UDATA_OVR

```
<label> UDATA_OVR <expression>
```

Only for relocatable mode. Creates a new uninitialized overlaid data section in the output object file. <label> specifies the name of the section. If <label> is not specified the default name ".udata_ovr" will be used. <expression> is optional and specifies the absolute address of the section.

See also: CODE, IDATA, UDATA

UDATA_SHR

def `<label> UDATA_SHR <expression>`

Only

2.4.2 Instruction set summary

12 bit Devices (PIC12C5XX)

Syntax	Description
ADDWF <f>,<dst>	

Spe

The s are:

SETZ	Set zero
MOVFW <f>	Move file to W
NEGF <f>	

115 Duplicate Label

Duplicate label or redefining a symbol that can not be redefined.

124 Illegal Argument

gpasm encountered an illegal argument in an expression.

125 Illegal Condition

An illegal condition like a missing ENDIF or ENDW has been encountered.

126 Argument out of

2.5.2 Warnings

201 Symbol not previously defined.

The symbol being #undefined was not previously defined.

202 Argument out of range

The argument does not fit in the allocated space.

211 Extraneous arguments

Extra arguments were found on the line.

215 Processor superseded by command line

The processor was specified on the command line and in the source file. The command line has prece-

The ID locations value specified is too large.

305 Using default destination

Chapter 3

gplink

gplink relocates and links gpasm COFF objects and generates an absolute executable COFF.

3.1 Running gplink

The general syntax for running gplink is

```
gplink [options] [objects] [libraries]
```

If the user does not specify a linker script, gplink will

Chapter 4

gplib

gplib creates, modifies and extracts COFF archi

4.4 Archive format

The file format is a standard COFF archive. A header is added to each member and the unmodified object is copied into the archive.

Being a standard archive they do include a symbol index. It provides a simple why to determine which member should be extract to resolve external references. This index is not included in mplib archives. So using gplib archives with Microchip Tools will probably cause problems unles the “-n” option is added when the archive is created.

Chapter 5

Utilities

5.1 gpdasm

gpdasm is a disassembler for gputils. It converts hex files generated by gpasm and gplink into disassembled instructions.

5.1.1 Running gpdasm

The general syntax for running gpdasm is

```
gpdasm [options] hex-file
```

Where

5.2 gpvc

gpvc is cod file viewer for gputils. It provides an easy way to view the contents of the cod files generated by gpasm and gplink.

5.2.1 Running gpvc

The general syntax for running gpvc is

```
gpvc [options] cod-file
```

Where options can be one of:

Option	Meaning
a	Display all information
d	

d