

**gputils 0.10.0**

James Bowman

January 20 2002

# Contents

<b>1</b>	<b>gpasm</b>	<b>3</b>
1.1	Running gpasm . . . . .	3
1.1.1	Using gpasm with “make” . . . . .	3
1.1.2	Dealing with errors . . . . .	4
1.2	Syntax . . . . .	4
1.2.1	File structure . . . . .	4
1.2.2	Expressions . . . . .	4
1.2.3	Numbers . . . . .	5
1.2.4	Preprocessor . . . . .	6
1.3	Directives . . . . .	
1.2.1s23.490(.)Tj 7.generat.TdPrepr1.181rs	. . . . .	4 Tf -399.207 -31.88 Td (1.

# Introduction

gputils is a collection of tools for Microchip (TM) PIC



```
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

---

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

### **1.3.3 Conditional assembly**

The IF, IFNDEF, IFDEF, ELSE and ENDIF directives enable you to assemble certain sections of code only if a condition is met. In themselves, they do not cause gpasm to emit any PIC



```
IF (this) ; {  
    ...  
ELSE      ; }{  
    ...  
ENDIF     ; }
```

After you’ve done this, you can use your text editor’s show-matching-brace to check matching parts of the IF structure. In vi this command is “%”, in emacs it’s M-C-f and M-C-b.

### 1.3.6 Directive summary

#### **\_\_BADRAM**

`__BADRAM <expression>`

**DA**

```
<label> DA <expression> [, <expression>]*
```

Stores Strings in program memory. The data is stored as one 14 bit word (ord)Tj 16.4196 0 Td (representing)Tj 5

**ENDC**

ENDC

Marks the end of a CBLOCK.

See also: CBLOCK

**ENDIF**

ENDIF

Ends a conditional assembly block.

See also: IFDEF, IFNDEF, ELSE, ENDIF

**ENDM**

ENDM

Ends a macro definition.

See also: MACRO

**ENDW**

ENDW

Ends a while loop.

See also: WHILE

**EQU**

<label> EQU <expression>

## **EXITM**

EXITM

Immediately return from macro expansion during assembly.

See also: ENDM

## **EXPAND**

EXPAND

Expand the macro in the listing file.

See also: ENDM

## **FILL**

<label> FILL <expression>,<count>

GenILL

option	description
b=nnn	Sets the tab spaces
f=<format>	





## 1.4.2 Instruction set summary

### 12 bit Devices (PIC12C5XX)

Syntax	Description
ADDWF <f>,<dst>	Add W to <f>, result in <dst>
ANDLW <f>,<dst>	And W and literal, result in W
ANDWF <f>,<dst>	And W and <f>, result in <dst>
BCF <f>,<bit>	Clear <bit> of <f>

BS1.34980.1539 0 Td (<f>,<bit>)Tj ET Q q 4.8 0 0 -120 1876.68 6202.68 cm BI /IM true /W 1 /H 1 /BPC 1 ID EI Q q 10 0 0



**14 Bit Devices (PIC16CXX)**

Syntax	Description
ADDLW <imm8>	Add immediate to W
ADDWF <f>,<dst>	Add W to <f>, result in <dst>
ANDLW <f>,<dst>	And W and <f>, result in <dst>
BCF <f>,<bit>	Clear <bit> of <f>
BSF <f>,<bit>	Set <bit> of <f>
BTFSC <f>,<bit>	Skip <del>not</del> instruction if <bit> of <f> <del>is</del> clear
BTFSS <f>,<bit>	Skip <del>not</del> instruction if <bit> of <f> is set
CALL <addr>	

### **Special macros**

There are also a number of standard additional macros. These macros are:



## **1.5.2 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 ID locations value specified is too large.

**305** Using default destination of 1 (file).

No destination was specified so the default location was used.

**308** Warning level superseded by command line

The warning level was specified on the command line and in the source file. The command line has precedence.

**309** Macro expansion superseded by command line

Macro expansion was specified on the command line and in the source file. The command line has precedence.



## Chapter 3

# gplink

gplink is a new and incomplete tool. When complete, it will provide the ability to link relocatable objects to produce an executable object.







```
gpvo test.obj > test.dump
```

This ~~script~~ ~~script~~ the file “test.dump”.

# Index

ASCII, 6

B