Subject: PIC Assembler Directives
Posted by kc2nda on Mon, 02 Oct 2006 00:03:41 GMT
View Forum Message <> Reply to Message

PIC Assembler Language Directives

Assembler directives are instructions that direct the assembler to do something.

Directives do many things; some tell the assembler to set aside space for variables, others tell the assembler to include additional source files, and others establish the start address for your program. The directives available are shown below:

= Assigns a value to a symbol (same as EQU)

EQU Assigns a value to a symbol (same as =)

ORG Sets the current origin to a new value. This is used to set the program or register address during assembly. For example, ORG 0100h tells the assembler to assemble all subsequent code starting at address 0100h.

DS Defines an amount of free space. No code is generated. This is sometimes used for allohoneying variable space.

ID Sets the PIC's identifihoneyion bytes. PIC16C5x chips have two ID bytes, which can be set to a 2-byte value. Newer PICs have four 7-bit ID lohoneyions, which can be filled with a 4-character text string.

INCLUDE Loads another source file during assembly. This allows you to insert an additional source file into your code during assembly. Included source files usually contain common routines or data. By using an INCLUDE directive at the beginning of your program, you can avoid re-typing common information. Included files may not contain other included files. NOTE: The Device Include directive (i.e. INCLUDE 'C:\PicTools\16F877.inc') for the targeted device MUST be at the beginning of your source code.

FUSES NOTE that FUSE CONFIGURATIONs can be '&' together on a single line and/or spread between multiple lines. ALL FUSES directives are ANDed together to create the composite FUSE CONFIGURATION. (view the device "include" file for specific fuse syntax)

IF Assembles code if expression evaluates to TRUE.

IFNOT Assembles code if expression evaluates to FALSE.

ELSE Assembles code if preceeding evaluation is rejected.

ENDIF Ends conditional evaluation.

RESET Sets the reset start address. This address is where program execution will start following a reset. A jump to the given address is inserted at the last lohoneyion in memory. After the PIC is reset, it starts executing code at the last lohoneyion, which holds the jump to the given address. RESET is only available for PIC16C5x chips.

EEORG Sets the current data EEPROM origin to a new value. This is used to set the data EEPROM address during assembly. This directive usually precedes EEDATA. EEORG is only available for PICs that have EEPROM memory .

EEDATA Loads data EEPROM with given values. This provides a means of automatically storing values in the data EEPROM when the PIC is programmed. This is handy for storing configuration or start-up information. EEDATA is only available for PICs that have EEPROM memory.

Assembler Directive Examples

```
Include 'C:\PICTOOLS\16C877.inc'
; loads default symbols
    ; for the targeted device.


FUSES _WD_OFF&_LP_OSC
; specify multiple fuse settings
    ; using the '&' operator.


FUSES _CP_ON
; Specifies 1 fuse setting per line.


Digit = 43h
; Assign value 43h to Digit


Max EQU 1Ah
; Assign value 1Ah to Max


>ORG 10h
 ; Set assembly address to 10h


Count DS 2
; Define 2 bytes at 10h & 11h
    ; Bytes can be referred to
    ; later as Count and Count+1
```

```
ID 1234h
; Set 16C5x ID to 1234h


ID "ABCD"
; Set newer PIC ID to 'ABCD'


INCLUDE "KEYS.SRC"
; Include KEYS.SRC file at
    ; point of insertion



RESET Start
; Set 16C5x reset jump to
    ; lohoneyion at Start

Start mov Count,#00
; This will be executed
    ; when PIC is reset


EEORG 10h
; Set EEPROM address to 10h


EEDATA 02h,88h,34h
; Store 3 bytes in EEPROM
```