

## Usare MPLAB IDE V.8.83

Semplici note per essere subito operativi col programma MPLAB della Microchip  
A cura del prof. Giuseppe SPALIERNO – docente di Sistemi Elettronici Automatici presso  
I.T.T. “M. Panetti” – BARI – aprile 2012

È un ambiente integrato che consente di scrivere, assemblare e simulare un programma per microcontrollori della Microchip tra cui il famoso PIC 16F84A.

Si descrivono le fasi da compiere dopo aver installato correttamente il programma.

1) per incominciare a scrivere un programma sorgente: **File / New**

Per avere le linee numerate: **Edit/Proprietà/Editor/Line number.**

Per salvare: **File/Salva** ed inserire il nome del file e l'estensione **ASM**; esempio: **prova1.asm**

2) prima di assemblare: **Project/Project wizard**

step 1: Device: scegliere **PIC16F84A**

step 2: Selezionare come language tool suite: **MPASM Assembler**

step 3: Nome progetto: prova1, crea una nuova cartella dove vuoi col nome che vuoi: PICprg1

step 4: Aggiungi qualsiasi file esistente del tuo progetto (dovresti inserire almeno prova1.asm)

3) per assemblare: **Project/Make (F10)**

Se ci sono degli errori controllare in quali linee si sono verificati ed effettuare le correzioni, salvare il file ASM e ripetere l'assemblaggio.

4) Durante la simulazione può essere utile tenere sotto controllo i valori dei registri speciali, le istruzioni del programma, ecc. per cui conviene accedere al menù **View**:

**Disassembly** per vedere le istruzioni in linguaggio assembly che contengono indirizzi numerici per rappresentare i registri speciali ed i bit specifici

**Program memory** per vedere linee, indirizzo, codice hex, disassembly del programma sorgente

**File register** per visualizzare il contenuto della memoria RAM

**Special Function Register** per visualizzare il contenuto dei registri speciali, TRIS, PORT, PCL, TIMER, STATUS, OPTION, ecc.

5) Per la simulazione occorre far riferimento al menù **Debugger**:

Scegliere il simulatore: **Select Tool/Mplab SIM**

Per eseguire il programma: **Run**

Per l'esecuzione con animazione: **Animate**

Per eseguire una istruzione per volta: **Step into**

Per abilitare un breakpoint: **doppio click sul numero di istruzione** (ripetere doppio click per disabilitare il breakpoint). È possibile inserire più breakpoint.

Per inserire gli input durante la simulazione: **Debugger/Stimulus/New Workbook**

- Dalla scheda “Asynch”: click nella casella PIN e scegliere la linea di ingresso
- Nella casella “Action” spesso conviene scegliere “toggle” per cambiare stato al pin scelto ad ogni click di mouse durante la simulazione
- Compilare altre righe se ci sono altri PIN di ingresso (per personalizzazioni più spinte consultare il manuale)

6) Se la simulazione ha avuto buon esito si passa alla **programmazione reale del PIC**

- a. Inserire il PIC sullo zoccolo presente nel circuito programmatore
- b. Collegare il circuito programmatore alla porta seriale del PC
- c. Lanciare il programma che consente la programmazione del PIC
- d. Configurare e controllare se tutto è OK
- e. Eseguire il comando di programmazione del PIC

7) Rimuovere il PIC dallo zoccolo della scheda programmatrice e collocarlo su breadboard inserendo il quarzo e l'elettronica di I/O. Alimentare il PIC con Vdd=+5V e ..buona fortuna!!