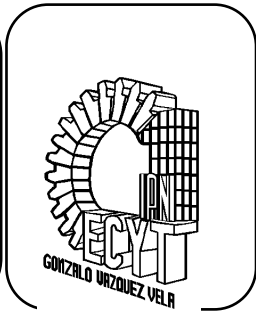


INSTITUTO POLITÉCNICO NACIONAL
 Centro de Estudios Científicos y Tecnológicos N° 1
 “Gonzalo Vázquez Vela”
 Academia de Sistemas Digitales
 Prácticas de Arquitectura de Microprocesadores y Microcontroladores



NOMBRE DEL ALUMNO: _____
 Apellido Paterno

_____ Apellido Paterno Nombre

N° DE BOLETA: _____ GRUPO: _____

ASIGNATURA: **Arquitectura de Microp. y Microcontr.**

HOJA	DE	FECHA			EVALUACION
		DIA	MES	AÑO	

PROFESOR: _____

Práctica 3

Introducción al Ambiente de desarrollo MPLAB

Competencia De La Unidad:

Emplea las instrucciones del microprocesador y/o microcontrolador para la elaboración de programas básicos diferenciando los modos de direccionamiento.

Resultado De Aprendizaje Propuesto (RAP):

- Indica el principio de funcionamiento de las instrucciones que posee el microprocesador y/o microcontrolador con el fin de identificar los modos de direccionamiento
- Interpreta las instrucciones de transferencia de información internas y externas del microprocesador y/o microcontrolador en sus aplicaciones prácticas para el desarrollo diagramas de flujo y algoritmos en casos prácticos

Objetivos De La Práctica:

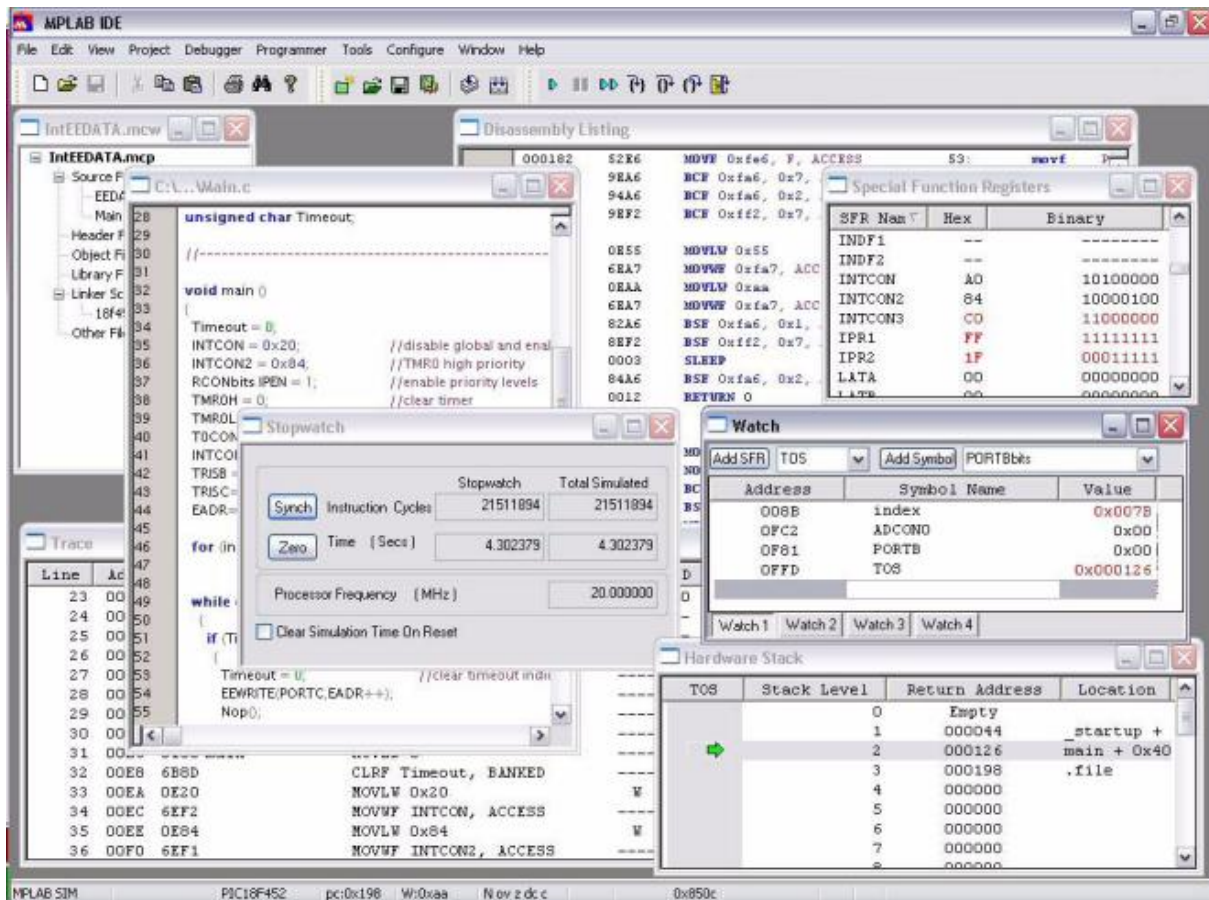
1. Conocer la manera de instalar el ambiente desarrollo MPLAB
2. Conocer los elementos del ambiente de desarrollo MPLAB.
3. Realizar un programa en ensamblador para comprobar su funcionamiento.

<p>Equipo Necesario</p> <p>Computadora (con el Software MPLAB IDE)</p>	<p>Material Necesario</p> <p>Hojas de especificaciones del PIC16F887</p>
---	---

Introducción Teórica

El ambiente integrado MPLAB (MPLAB IDE) es un conjunto de herramienta para desarrollar aplicaciones embebidas empleando microcontroladores PIC® y dsPIC® de la compañía Microchip. MPLAB IDE trabaja como una aplicación de 32 bits en el sistema operativo Windows®. Trabajar con las herramientas de ambiente es muy sencillo, así como su actualización. Una característica que se tiene en el entorno es que las distintas versiones del software tiene la misma interfaz.

En entorno de desarrollo MPLAB se observa de la manera siguiente:



El MPLAB incluye las utilidades necesarias para desarrollo de proyectos con microcontroladores PIC, permite editar el archivo fuente (asm, C) del proyecto, así como ensamblarlo y simularlo en pantalla para comprobar el funcionamiento de la aplicación antes de grabarlo en el dispositivo.

El ambiente MPLAB cuenta con las siguientes características:

- Editor de textos para efectuar cambios de forma muy flexible.
- Un ensamblador denominado MPASM
- Un simulador llamado MPLAB SIM
- Un Depurador (Debug) sencillo y poderoso
- Sus componentes son gratuitos (Editor, Compilar para lenguaje C, Depurador, etc)

- Esta desarrollado para soportar componentes adicionales de software y hardware de la compañía Microchip.

ACTIVIDADES PRÁCTICAS

1. Instalación/Desinstalación del MPLAB IDE

- Para instalar MPLAB en su computadora

Descargue MPLAB IDE desde el sitio web de Microchip (www.microchip.com). Identifique MPLAB IDE en el sitio, localice el archivo (.zip), seleccione el archivo y guárdelo en el disco duro su PC. Descomprima el archivo y ejecute el archivo de instalación setup.exe.

- Para desinstalar MPLAB

Seleccione Inicio>Configuración>Panel de Control abra el panel de control.

Doble click en agregar y quitar programas. Localice MPLAB IDE en la lista y de click sobre este.

Posteriormente, de un click en Cambiar/Quitar para quitar el programa de su sistema

2. Iniciar a utilizar MPLAB IDE

- A. Antes de iniciar el programa es necesario crear un carpeta en el directorio raíz llamada PIC (o algún nombre parecido), que será el lugar donde se guardaran los programas y aplicaciones que se van realizando, La ruta absoluta o *path* no puede ser de mayor a 62 caracteres ya que en programa marcara un error al momento de ensamblarlo, es importante considerar lo anterior ya que en Windows las rutas son bastantes largas. Por lo tanto, el subdirectorío que utilizaremos para guarda cada practica de aquí en adelante será **C:\PIC**
- B. Para iniciar MPLAB, de doble clic sobre el icono ubicado el escritorio o seleccione inicio>programas>Microchip>MPLAB vx.xx>MPLAB IDE. Y entrara a una pantalla similar a la siguiente (figura 1.1). Maximice la pantalla

¿Cuál es la versión de MPLAB que utilizas en la práctica?: _____

Todas las aplicaciones con microcontroladores PIC por lo general se realizan en el MPLAB, por lo tanto es necesario realizar una serie de etapas para obtener un desempeño adecuado en nuestro proyecto. A continuación se mencionan las etapas a realizar:

- *Crear un proyecto.*
- *Habilitar el simulador MPLAB SIM.*
- *Crear y agregar un archivo fuente en el proyecto.*
- *Configuración del ambiente de trabajo.*
- *Escribir el programa fuente (código) de la aplicación.*
- *Construir en proyecto (Ensamblar el código fuente).*
- *Simulación de la aplicación (programa)*

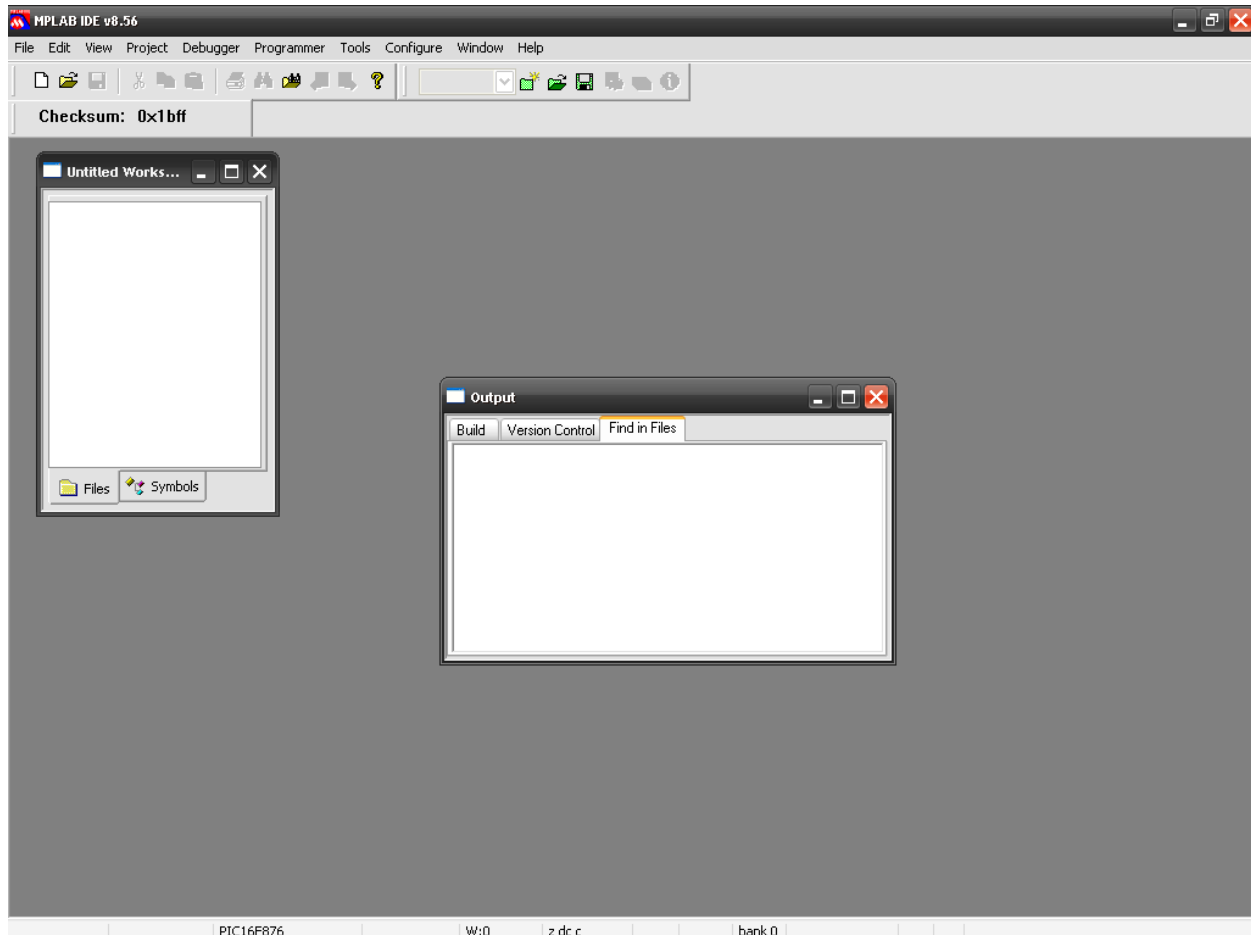


Fig. 1.1 Pantalla de inicio de MPLAB

3. Crear un proyecto

- A. Para crear un proyecto se utilizar el **Project Wizard**, (figura 1.2) y dar siguiente y seleccionar
 - Seleccione el tipo de microcontrolador a utilizar. Para esto elija el dispositivo **PIC16F887**, seleccionar siguiente hasta que aparezca la opción
 - Crear nuevo archivo de proyecto, que en este caso será **prac3** y guárdalo en la carpeta **C:\PIC\ practica3** con la opción *examinar (Browse)* dar siguiente, apareciendo la imagen de la figura1.3 y finalizar.

- B. Otra manera de crear un proyecto es de forma manual utilizando
 - Project > New (figura 1.3a)
 - Darle al proyecto el nombre de **prac3** y guárdalo en la carpeta **c:\PIC\ practica3**. (figura1.3b)

- Y posteriormente, es necesario seleccionar el tipo de microcontrolador a utilizar. Para esto se accede al menú *Configure*>*Select device* y escoger el dispositivo *PIC16F887*, como muestra en la figura 1.3c.

- C. Enseguida realice la configuración del la palabra de configuración del dispositivo (Configuration Word) que se escribe durante el proceso grabación. Seleccione *Configure* >*Configuration bits*. Modificar únicamente el tipo de oscilador a XT y apagar el watchdog y cerrar. Debe de aparecer una pantalla como la figura 1.4a y figura 1.4b

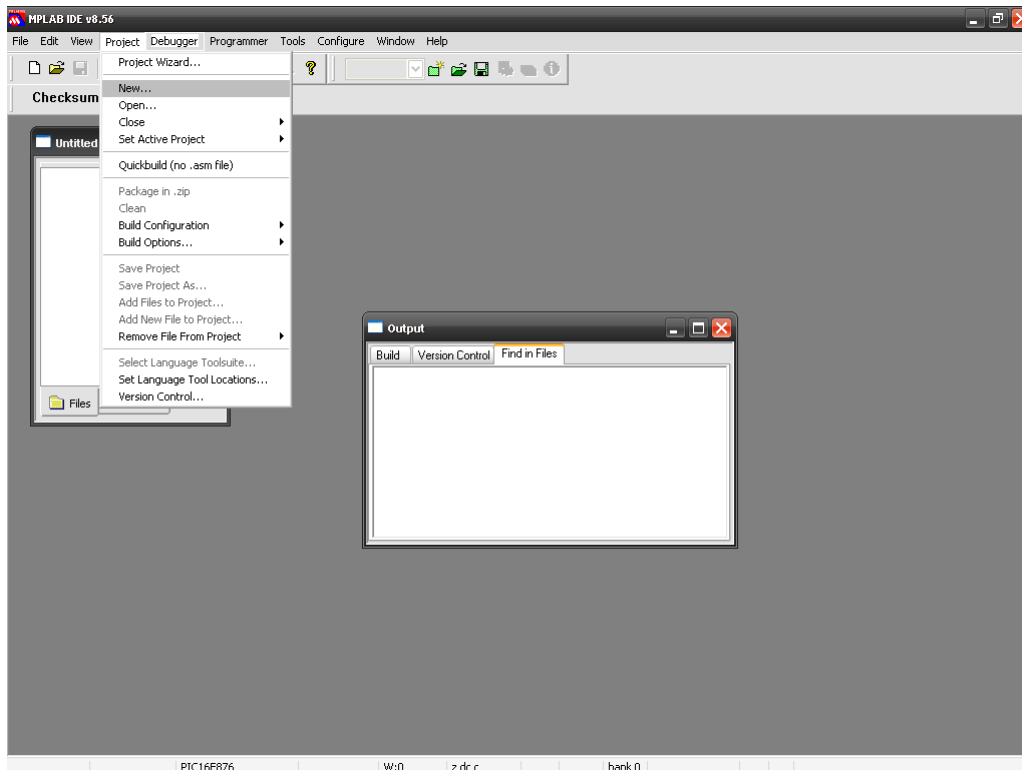


Fig. 1.2 Crear Nuevo proyecto (Seleccionar Project Wizard)

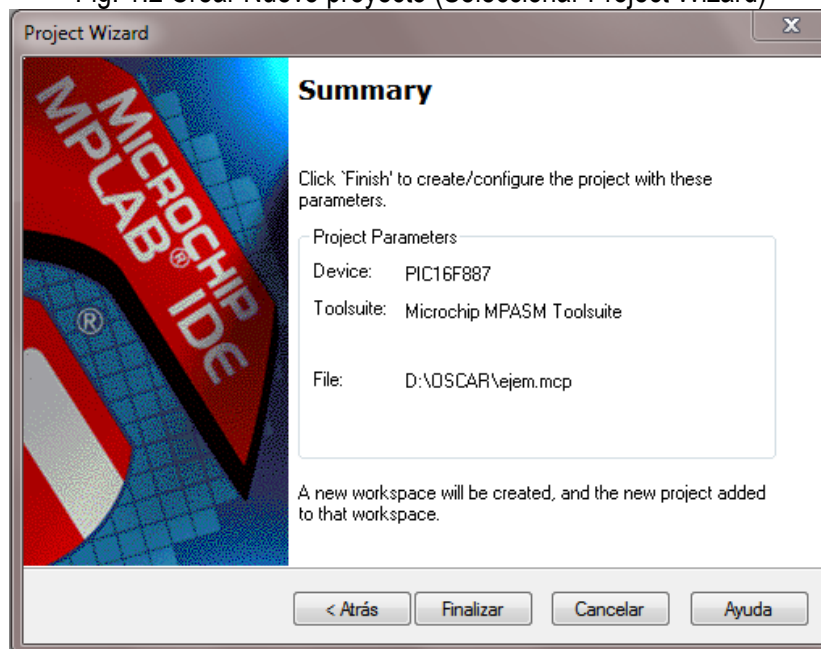


Fig. 1.3 Resumen de creación de Nuevo proyecto

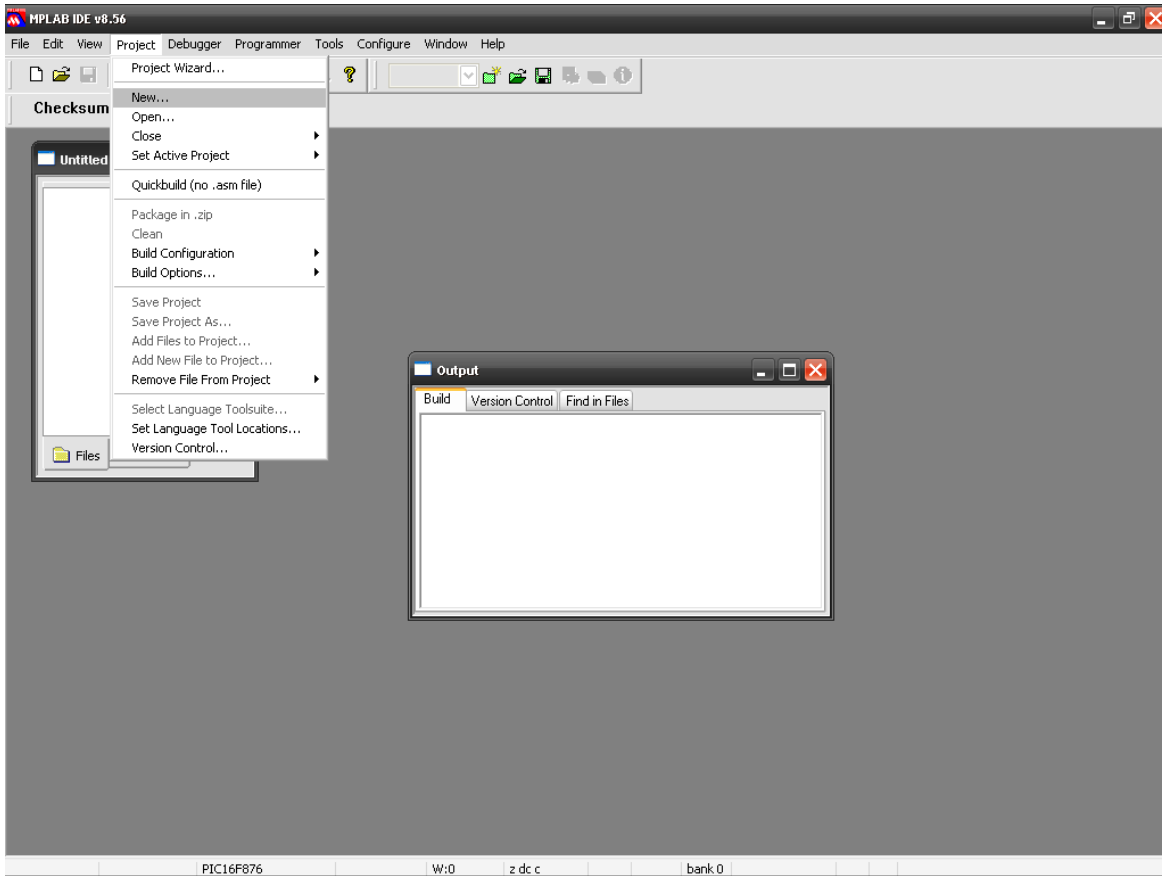


Fig. 1.3a Crear Nuevo proyecto

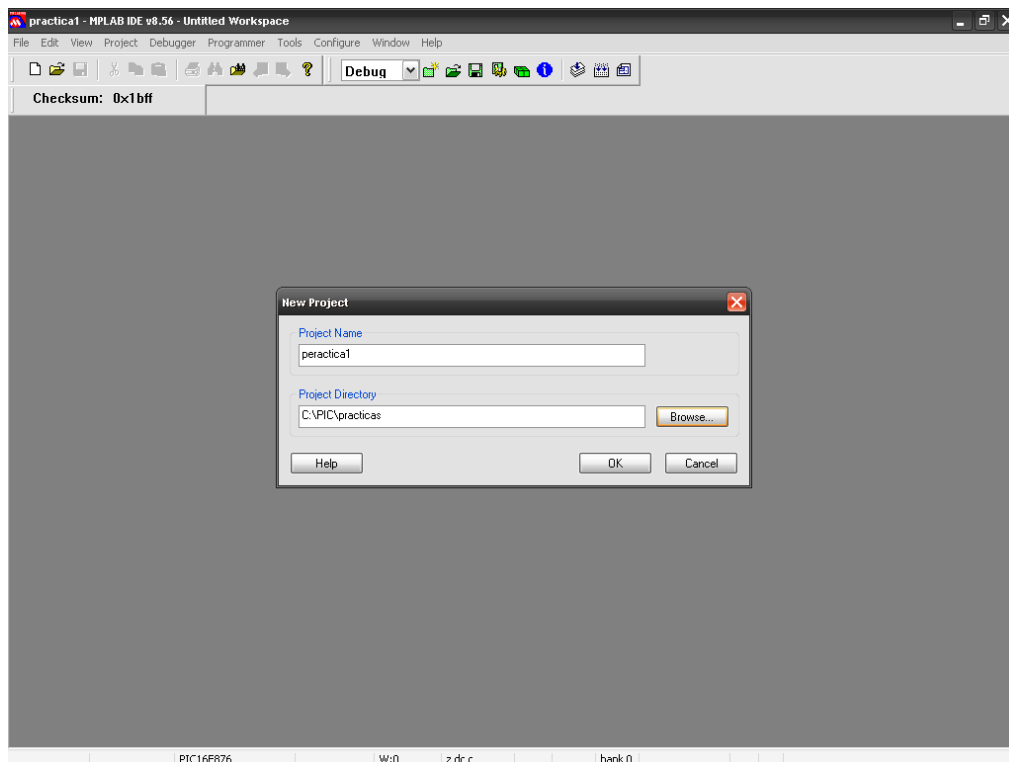


Fig. 1.3b Dar nombre al proyecto

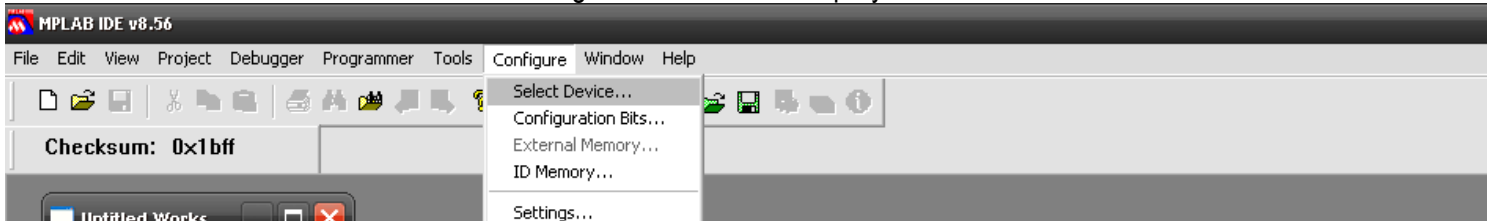


Fig. 1.3c Selección del microcontrolador

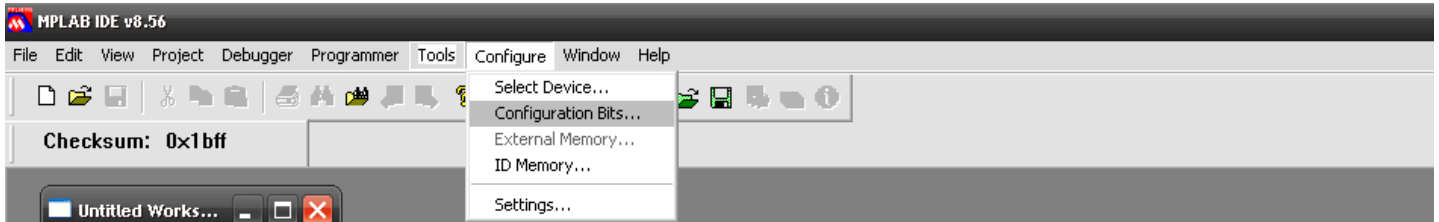


Fig. 1.4a Configuración de bits

Address	Value	Field	Category	Setting		
2007	3FF1	OSC	Oscillator	XT oscillator: Crystal/resonat		
		WDT	Watchdog Timer	WDT disabled		
		PUT	Power Up Timer	PWRT disabled		
		MCLRE	Master Clear Enable	RE3/MCLR pin function is MCLR		
		CP	Code Protect	Off		
		CPD	Data EE Read Protect	Data memory code protection is		
		BOREN	Brown Out Reset Selec	BOR enabled		
		IESO	Internal External Swi	Internal/External Switchover n		
		FCMEN	Monitor Clock Fail-sa	Fail-Safe Clock Monitor is ene		
		LVP	Low Voltage Program	RB3/PGM pin has PGM function,		
		2008	3FFF	BOR4V	Brown Out Reset Selec	Brown out Reset set to 4.0V
				WRT	Flash Program Memory	Write protection off

Fig. 1.4b Selección de oscilador y deshabilitar watchdog

D. Habilitar el simulador MPLAB SIM.

- A. Para habilitar el simulador, se tiene que elegir el menú *Debugger > Select Tool > MPLAB SIM* (Figura 1.5)
- B. Es conveniente modificar la frecuencia de trabajo del simulador, para que coincida con el circuito que se desea simular. Para modificarlo acceda a *Debugger > Setting > Osc/Trace* y modifique el valor del reloj. Para nuestro caso colocar 4 MHz o la frecuencia que utilizara en su circuito. Y aceptar. (figura 1.6)

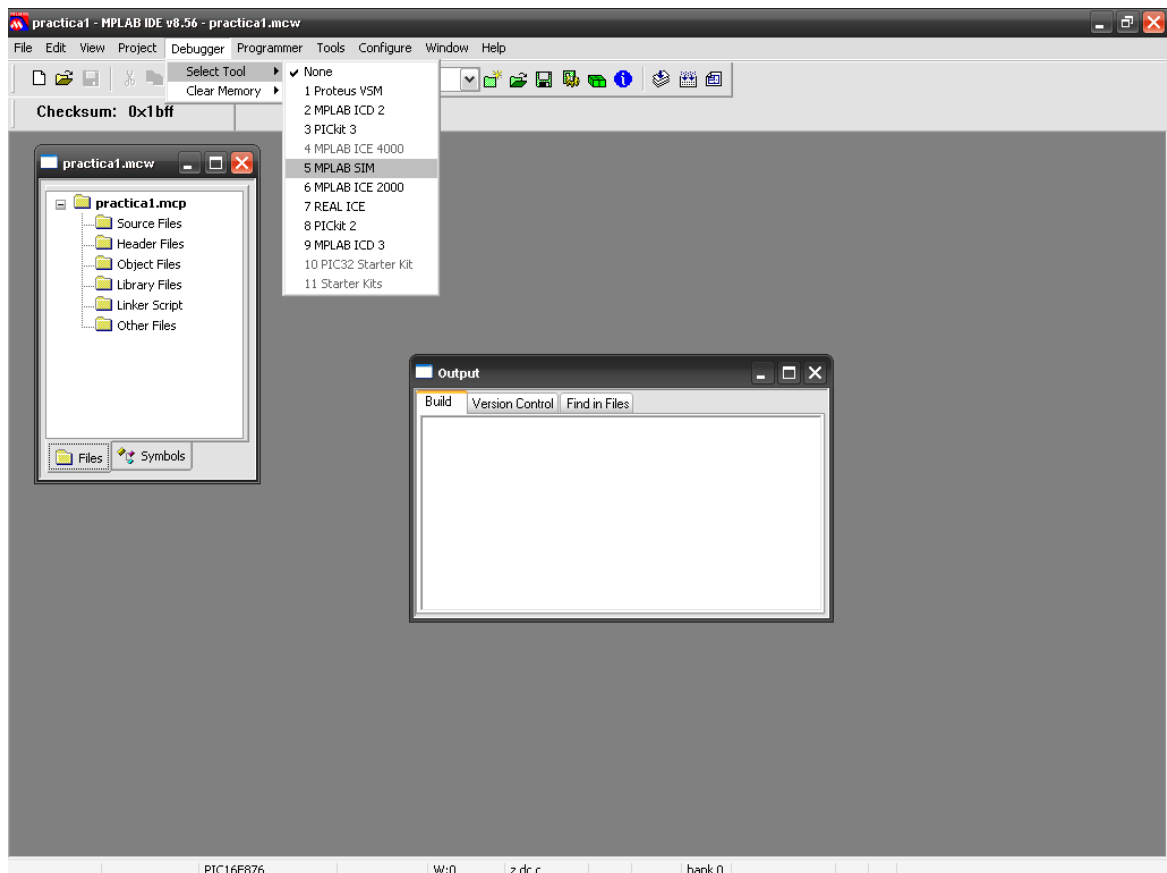


Fig. 1.5 Habilitar Simulador

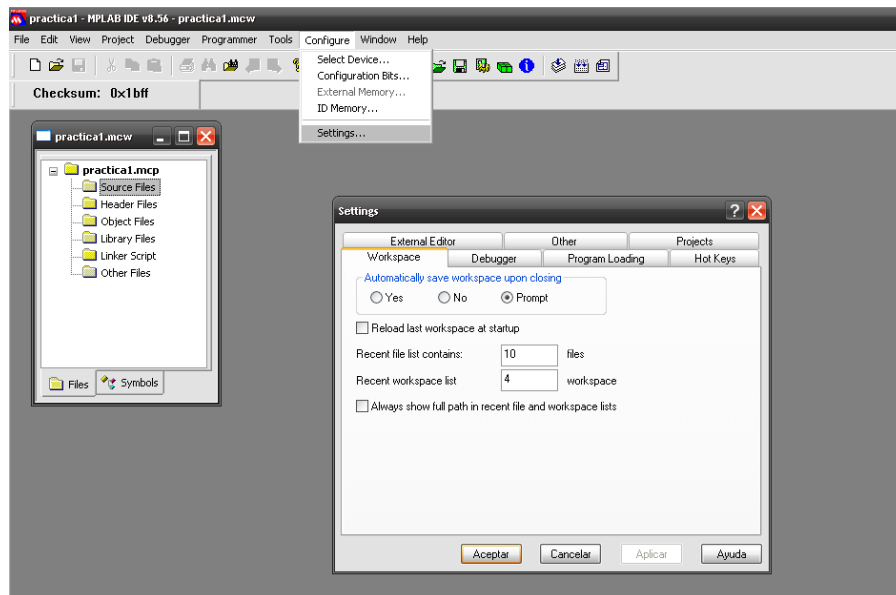


Fig. 1.6 Opciones del Depurador (Debugger) y Modificar la velocidad del reloj

E. Crear y agregar en el proyecto un archivo fuente.

- A. Para crear un archivo fuente nuevo, es necesario ir al menú *File > New* y aparecerá una pantalla en blanco donde se escribirá el programa.(figura 1.7)

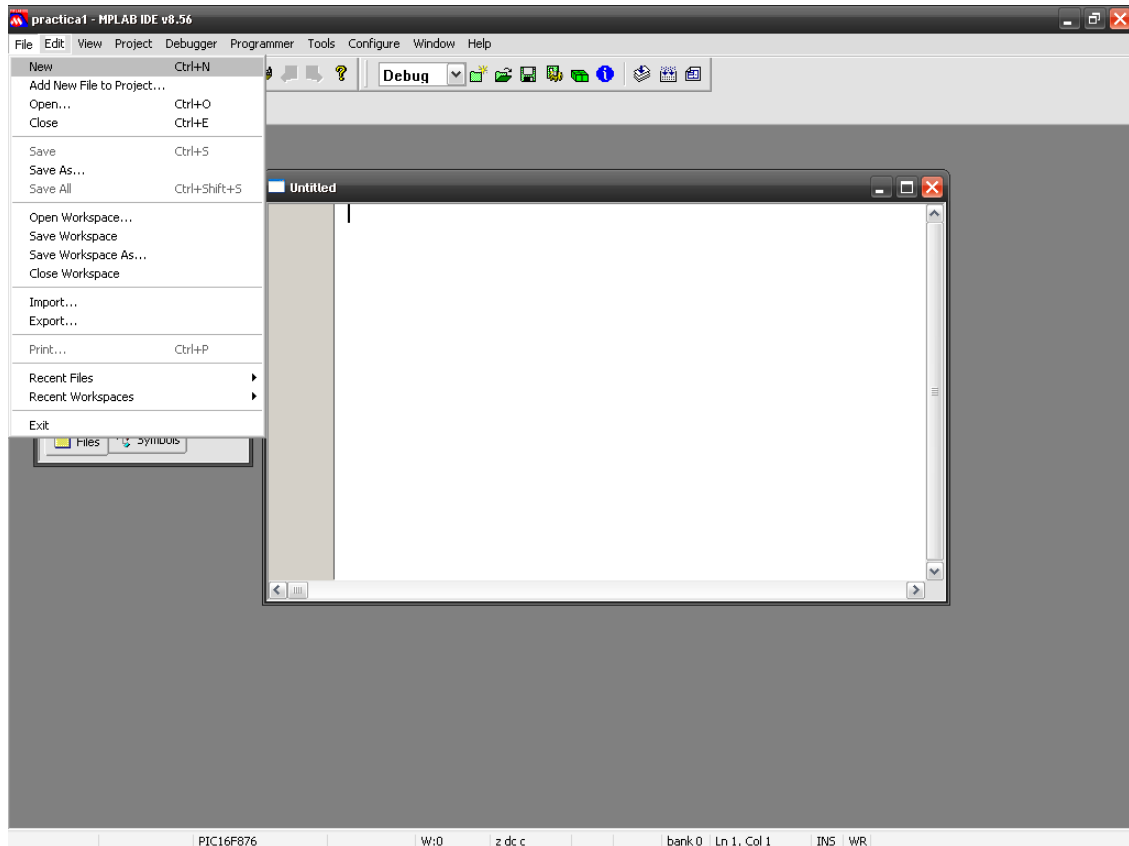


Fig. 1.7 archivo fuente nuevo

- B. A continuación se da el nombre y se almacena accediendo a *File > Save as* y aparecerá el cuadro de dialogo que solicita el nombre. Denomine al archivo *practica3.asm*. y guárdelo en la carpeta **C:\PIC\practica3**. La extensión ASM es necesaria para que el ambiente reconozca que es un programa realizado en ensamblador. ¹ (figura 1.8)
- C. En seguida, para agregar el archivo fuente creado o algún otro previamente desarrollado en el proyecto. Se puede realizar de dos formas:
- Ir a *File > add file new to project* y seleccionar el archivo fuente (asm) correspondiente, para nuestro caso **practica3.asm**, y guardar. Observara la siguiente pantalla. (figura 1.9)
 - Colocarse en la ventana xxx.mcw (en este caso **prac3.mcw**) y en **Source file** con el botón derecho del mouse seleccionar **add file** (agregar archivo) y escoger en el cuadro de dialogo el archivo fuente (asm) correspondiente, para nuestro caso **practica3.asm**, y aceptar. Y observara la pantalla de la figura 1.10.

¹ En algunas ocasiones el programa fuente se guarda con extensión .C, por que se pueden implementar el programa en lenguaje C, pero se requiere más experiencia para realizarlo.

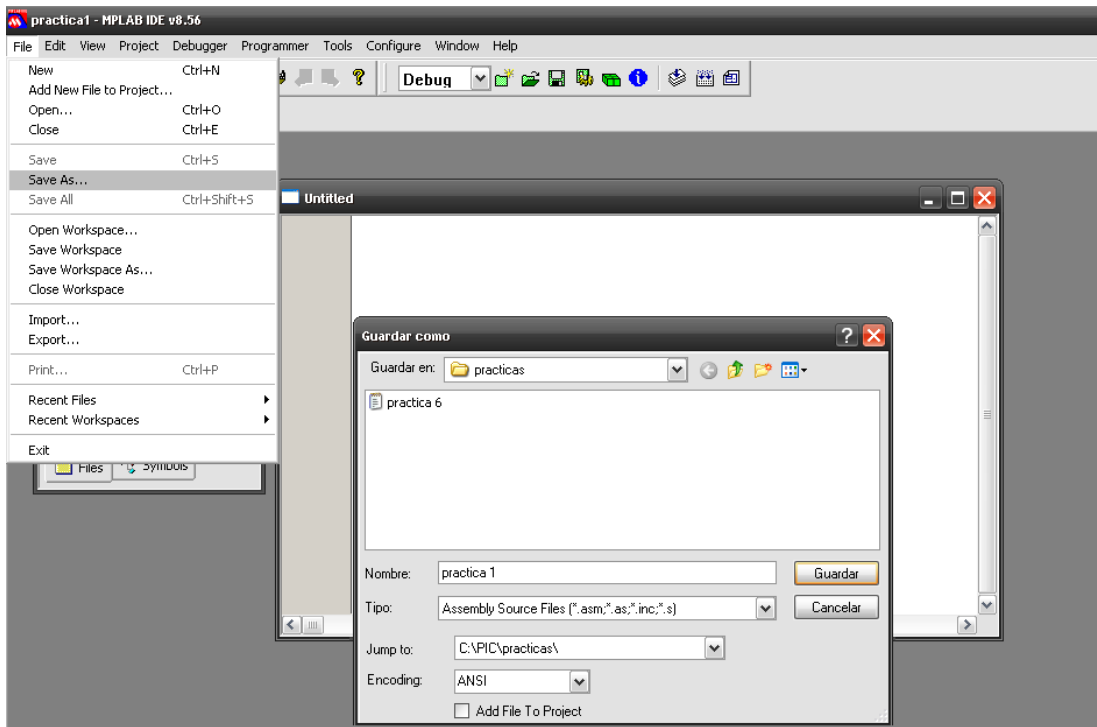


Fig. 1.8 Guardar el archivo fuente

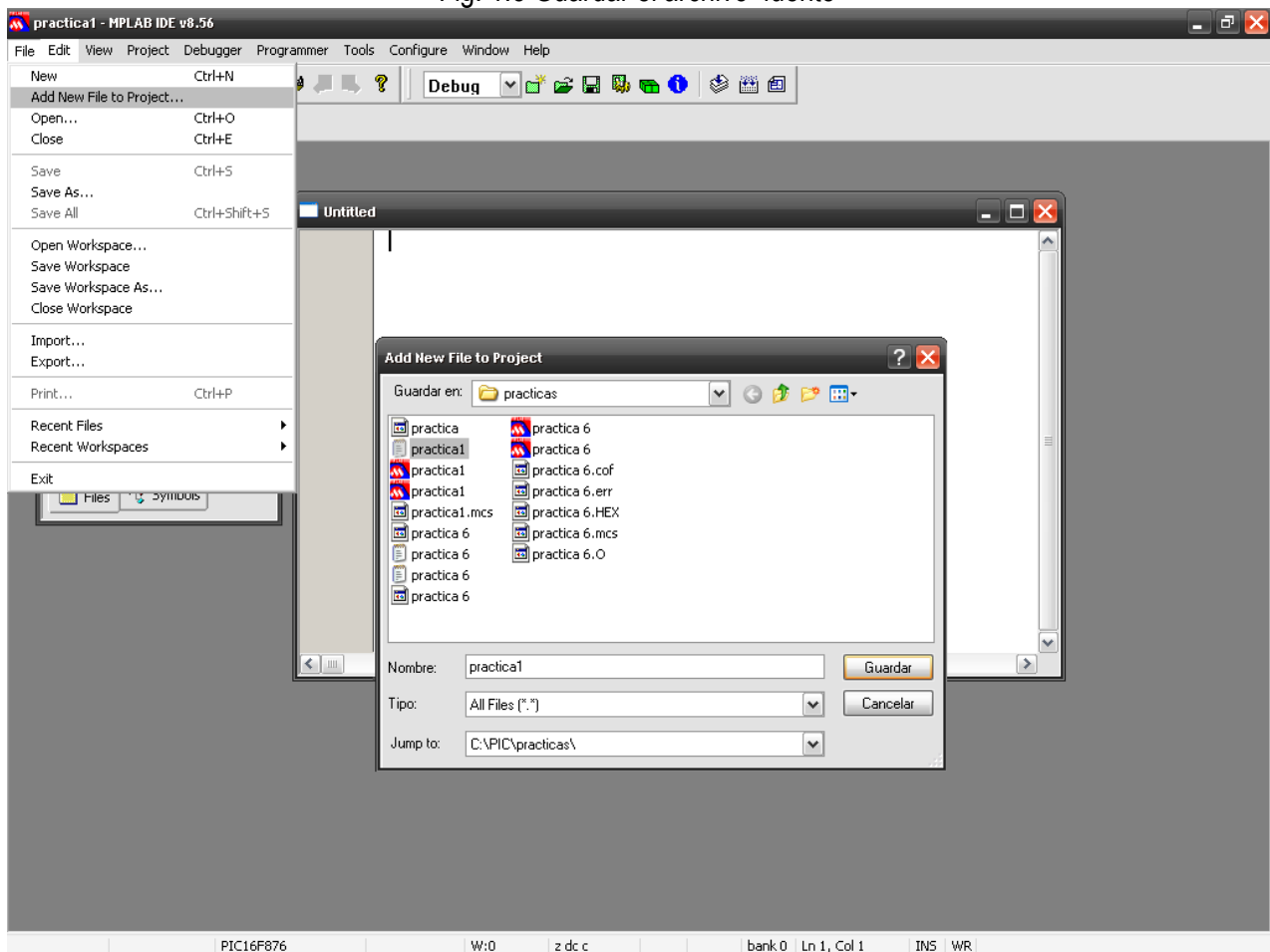


Fig. 1.9 Agregar un archivo fuente (asm)

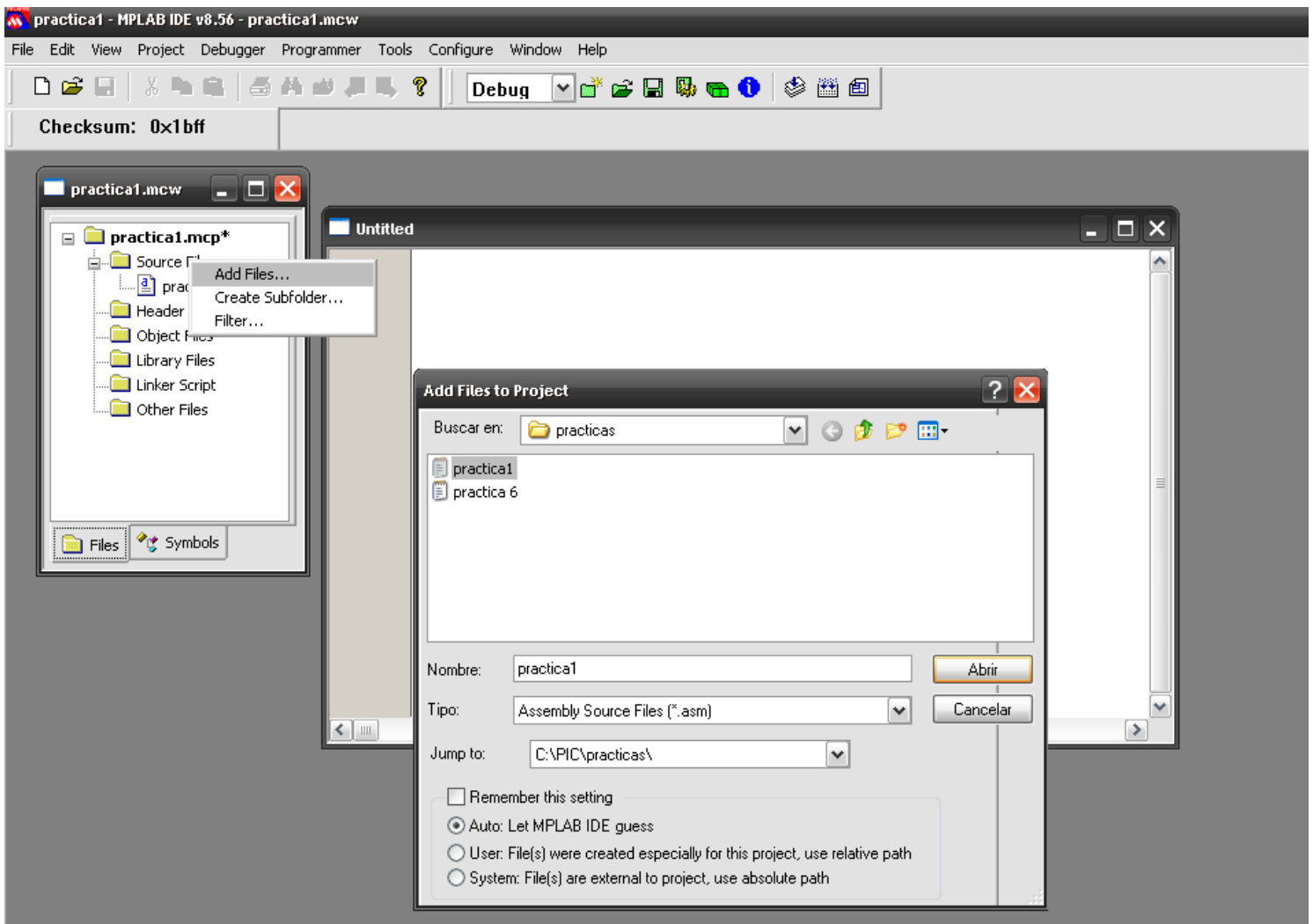


Fig. 1.10 Agregar un archivo fuente (asm) al proyecto-opción b

F. Configuración del ambiente de trabajo.

Enseguida realizaremos algunas recomendaciones para poder trabajar en el ambiente con mayor comodidad.

- A. Para trabajar con mayor comodidad es conveniente, es conveniente visualizar el número de línea. Para esto elija el menú *Edit > Properties*. Dentro de la ventana *Editor Options* y la pestaña *'ASM' File Types* active la casilla *line numbers* como se indica en la figura 1.11
- B. Para elegir el tipo de letra. Seleccione el menú *Edit > Properties*. Dentro de la ventana *Editor Options* y la pestaña *Text*, elija *select font* y escoja el tipo de letra (figura 1.12) o aquel con el que se trabaje de manera más cómoda.

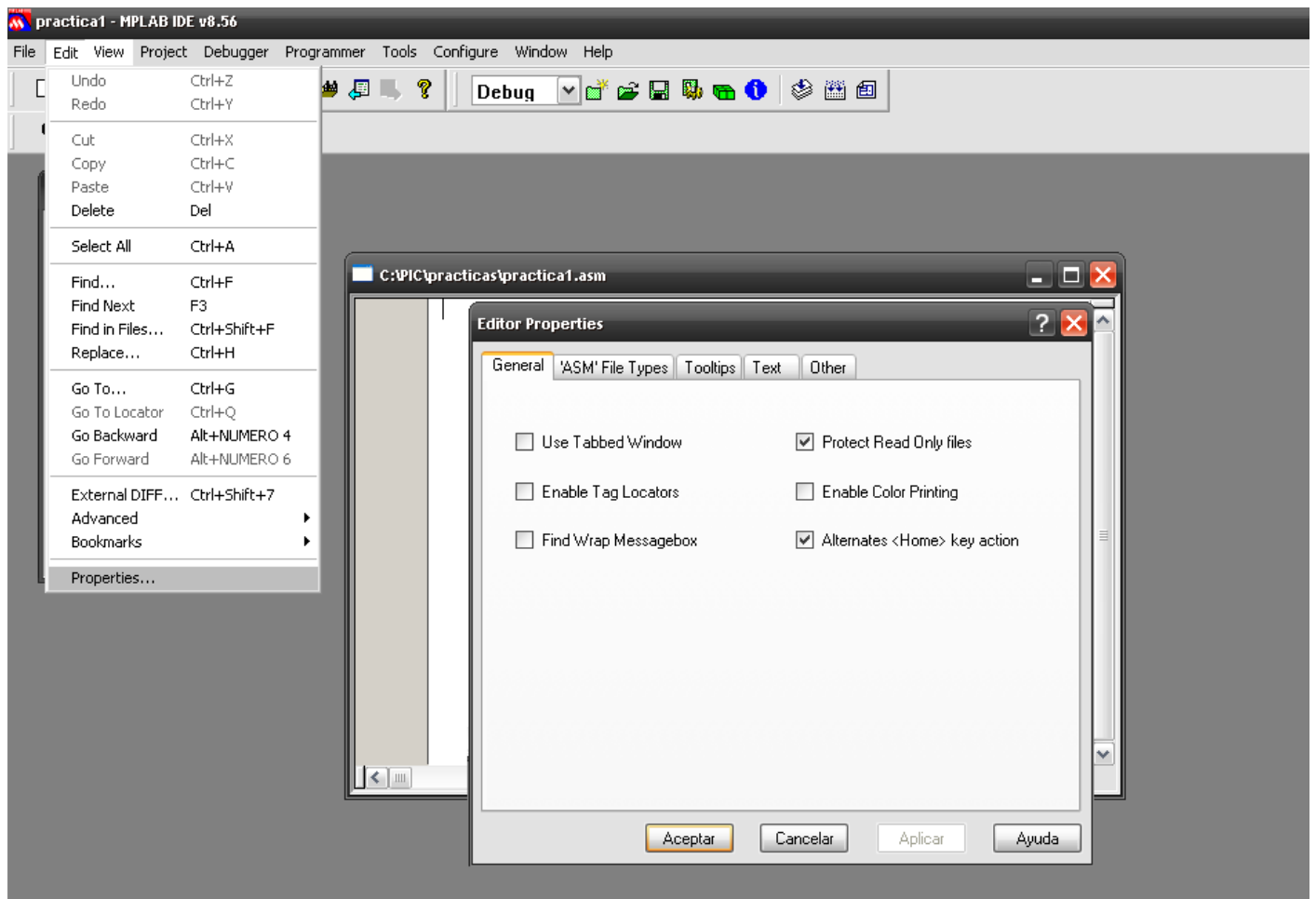


Fig. 1.11 Agregar número de línea

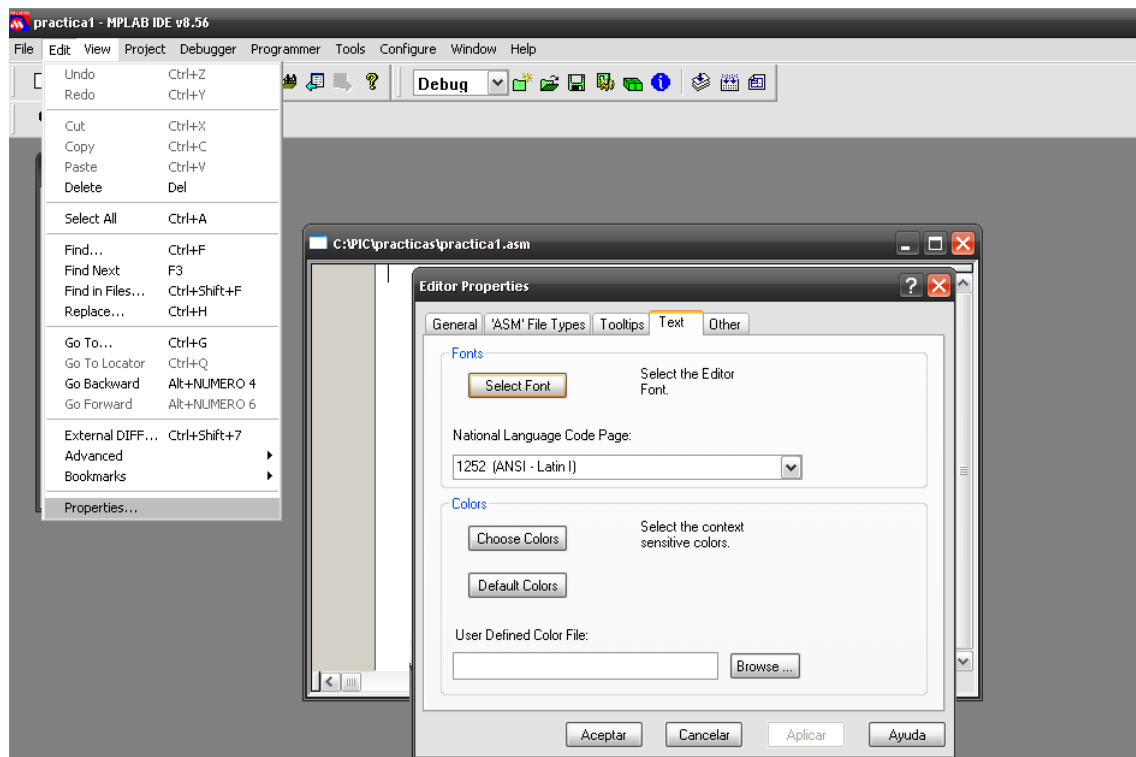


Fig. 1.12 Modificar tamaño y tipo de letra

G. Estructura general de un programa (código fuente).

- A. En la pantalla en blanco con la leyenda `c:\PIC\ practica3/ practica3.asm` comenzar a escribir el código del programa, aparece un asterisco en la leyenda, este indica que el archivo se está modificando. Tome en cuenta los espacios y tabuladores.

```
LIST P=16F887      ;microcontrolador a utilizar
#include "p16F887.inc"
cblock 025

dato
dato1
TEMP

endc

inicio    movlw 07H
          movwf dato
          incf dato,1
          movlw 0F1H
          movwf dato1
          movf  dato1,0
          movwf TEMP
          decf  TEMP,1
          goto inicio

          end
```

- B. Guarde el programa y el archivo asterisco desaparecerá

- La instrucción `LIST P=16F887` se utiliza para definir el microcontrolador que se utilizara para desarrollar la aplicación.
- `#include "p16F887.inc"`, indica el fichero donde se localiza las etiquetas que nombran los diferentes registros propios del microcontrolador y el valor que le corresponde a cada uno. Este archivo se localiza en el directorio principal del programa ensamblador.
- `cblock 025` y `endc` indican las localidades de memoria que se utilizaran de los registros de propósito general localizados en la memoria RAM, en este caso inician en la dirección 25H.
- `End`, indica el final de programa fuente, es decir le dice al ensamblador que termine de ensamblar el programa fuente.

H. Construir en proyecto (Ensamblar el código fuente.)

- A. Para construir el proyecto seleccionar `Project > Build all (ctrl. +F10)` o el icono (figura 1.13)

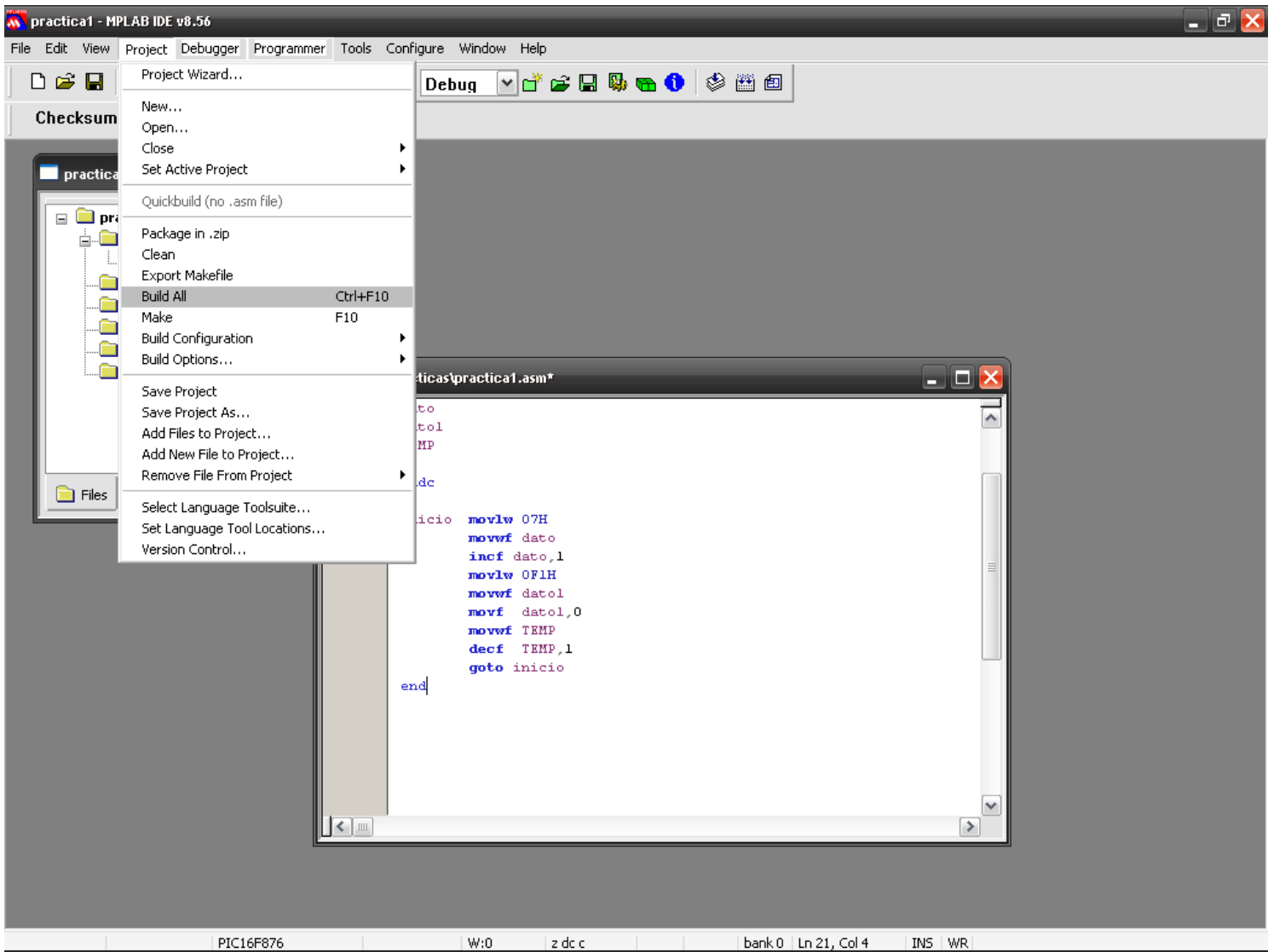


Fig. 1.13 Construir un proyecto

- B. Al realizar el paso anterior si el programa no genera ningún error debe aparecer “Build succeeded” en la ventana de *output* o un barra en verde en versiones anteriores (figura 1.14). los errores se pueden observar en la ventana *output*.

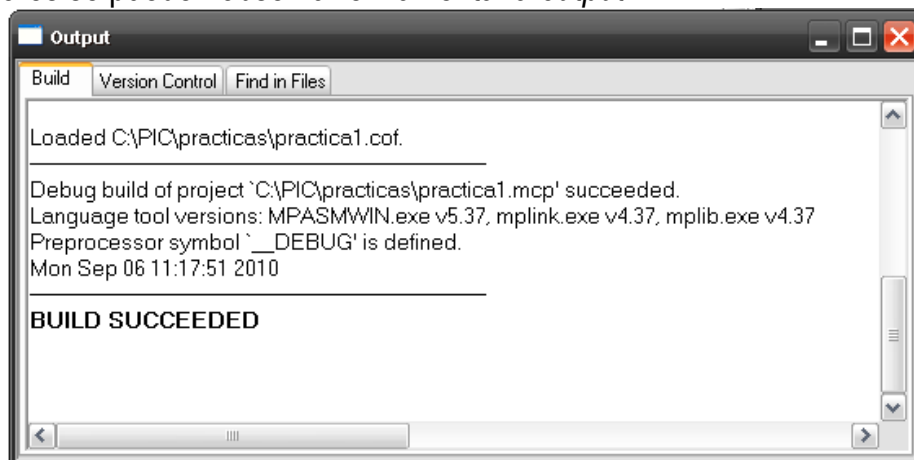


Fig. 1.14 Ensamblado correcto

I. Simulación de una aplicación (programa).

- A. Al realizar el ensamblado, debe aparecer un indicador (flecha verde) en la primera instrucción del programa. (Figura 1.15) para poder iniciar la simulación.



```
dato
dato1
TEMP
endc

inicio movlw 07H
        movwf dato
        incf dato,1
        movlw 0F1H
        movwf dato1
        movf dato1,0
        movwf TEMP
        decf TEMP,1
        goto inicio

end
```

Fig. 1.15 Indicación de inicio de simulación

- B. Para iniciar simulación se puede seleccionar Debugger > run (F10) o en icono correspondiente. Esta opción se utilizara en posteriores prácticas, ya que se combina con la opción Breakpoint (punto de ruptura). Y la opción *run* ejecuta el programa como si el circuito estuviera funcionando y no permite observar el funcionamiento del programa instrucción por instrucción, por lo cual no cumple los fines educativos.

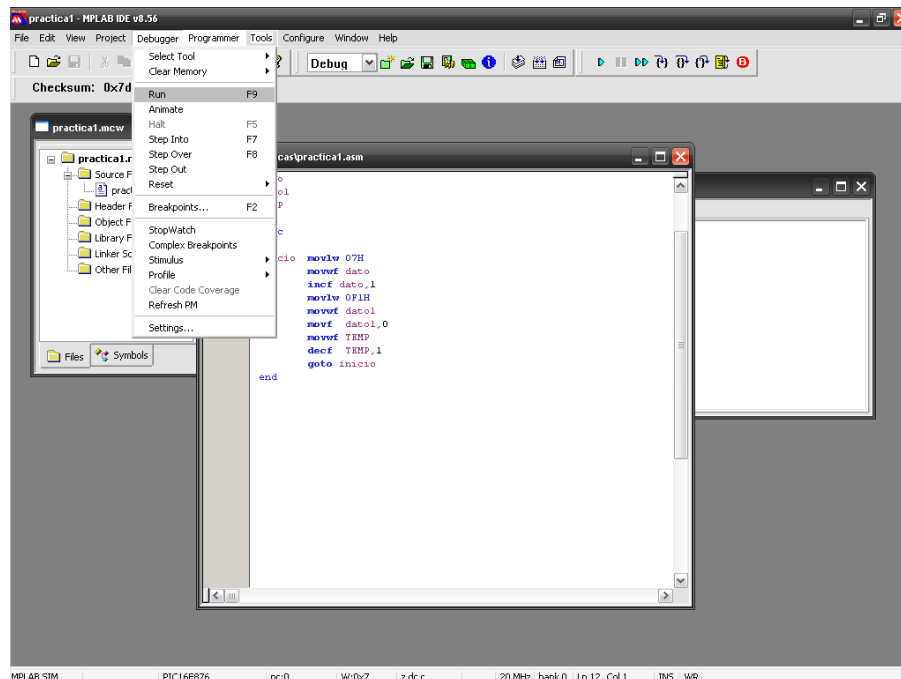



Fig. 1.16 Comando run

- C. Un opción del depurador que nos permite observar el funcionamiento del programa instrucción por instrucción es la modalidad paso a paso, la cual se puede ejecutar seleccionado Debugger > step into (F7) o en icono  (figura 1.17). Y cada vez que se ejecuta una instrucción el indicador cambio de posición.

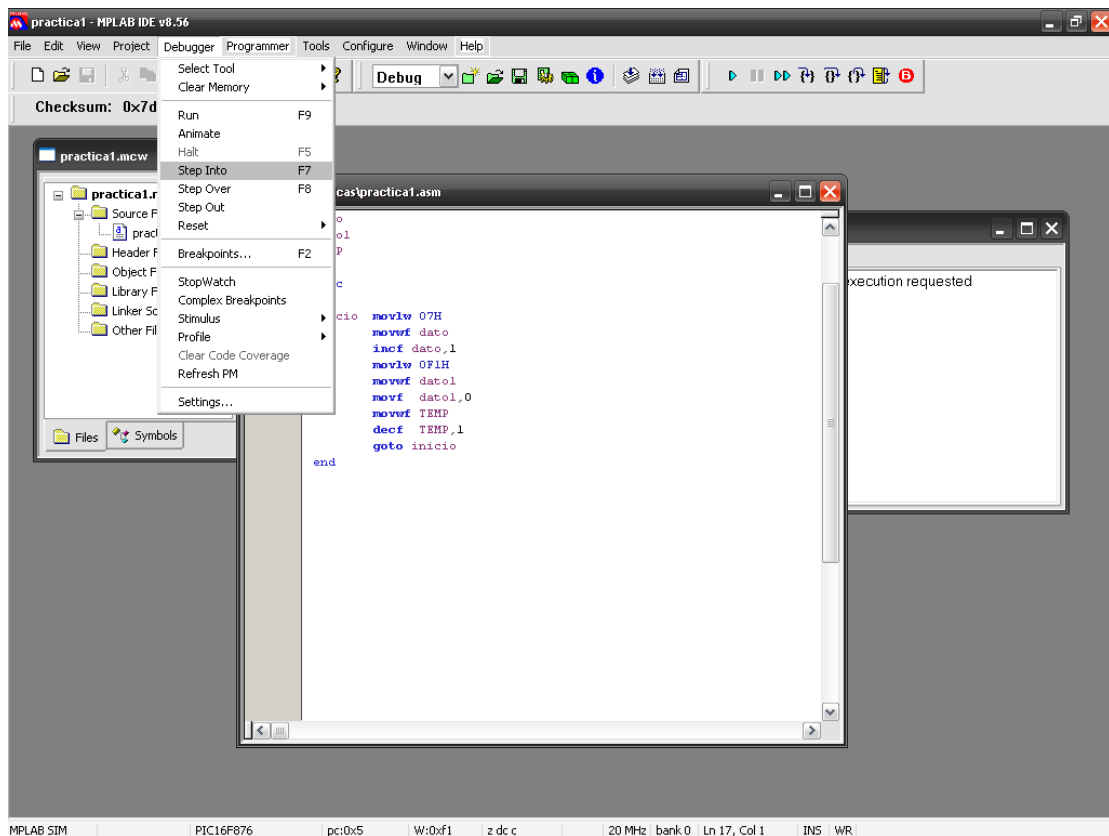


Fig. 1.17 Comando de modalidad paso a paso (step into)

- D. Una herramienta que ayuda bastante al realizar simulación es la ventana de watch, que nos permite visualizar el valor de los registros de funciones especiales (SFR) y de los registros de propósito general (GPR). Para visualizarlo escoja View > watch. (Figura 1.19). Y debe observarse la pantalla de la figura siguiente.

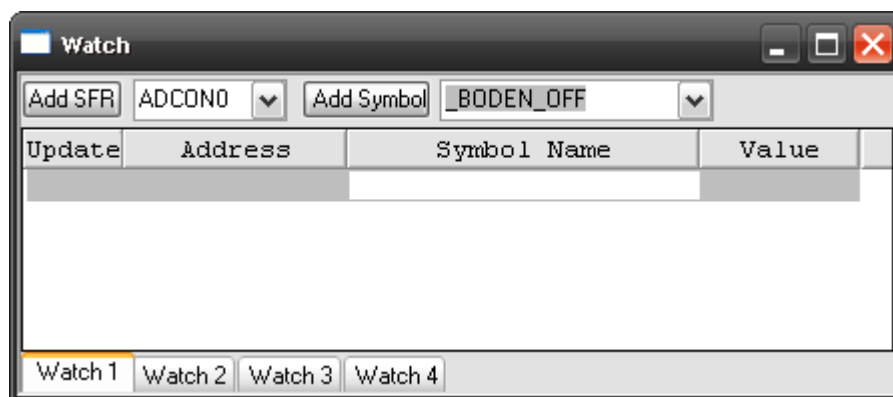


Fig. 1.18 Ventana Watch

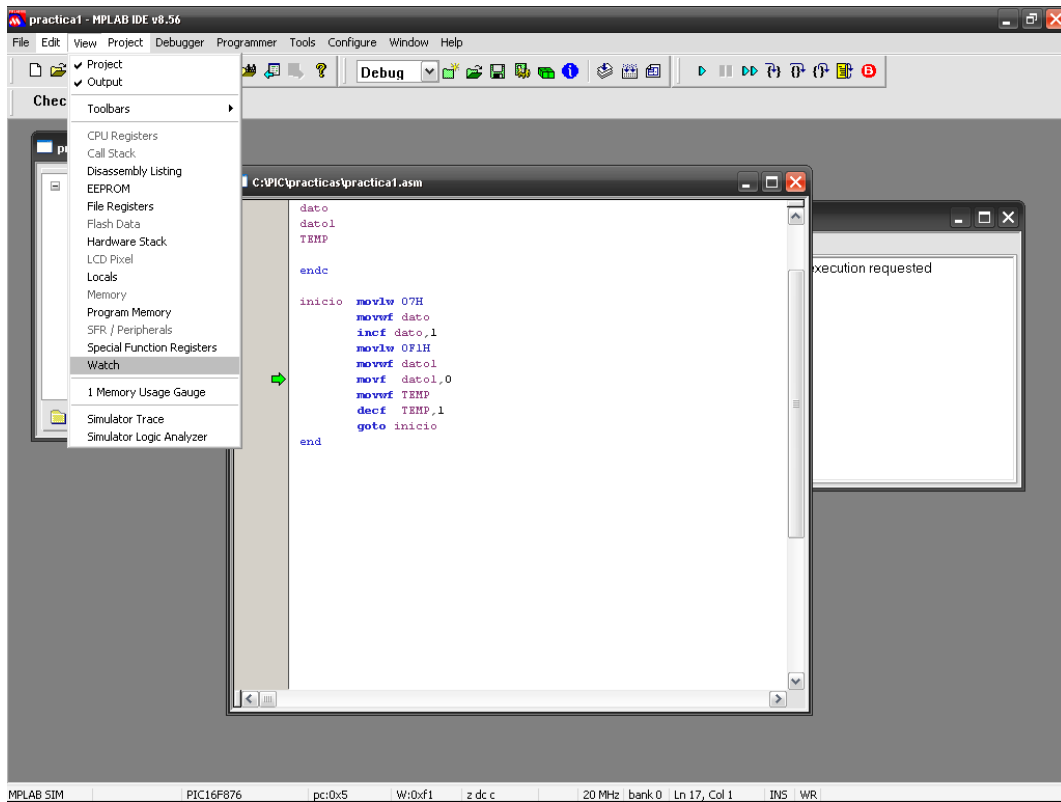


Fig. 1.19 Para seleccionar la opción watch

E. Para visualizar los registros SFR en la ventana, escoger el registro en la primer pestaña y dar click en Add SFR.

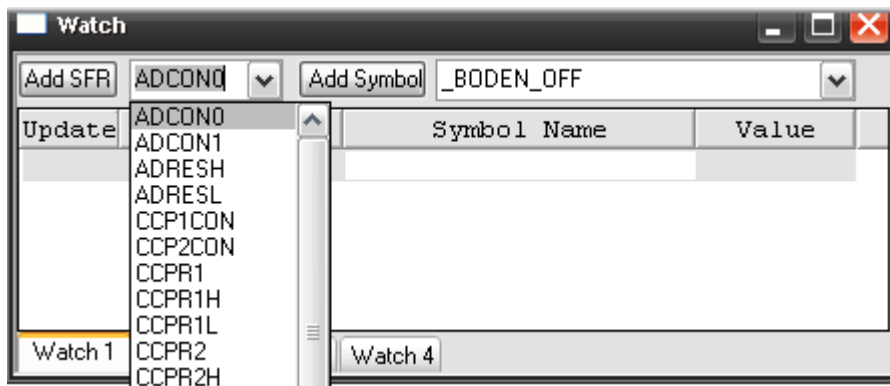


Fig. 1.20 Para seleccionar de los registro SFR

F. Para visualizar los registros GFR, seleccionar la variable o registro en la segunda pestaña y dar click en Add symbol.

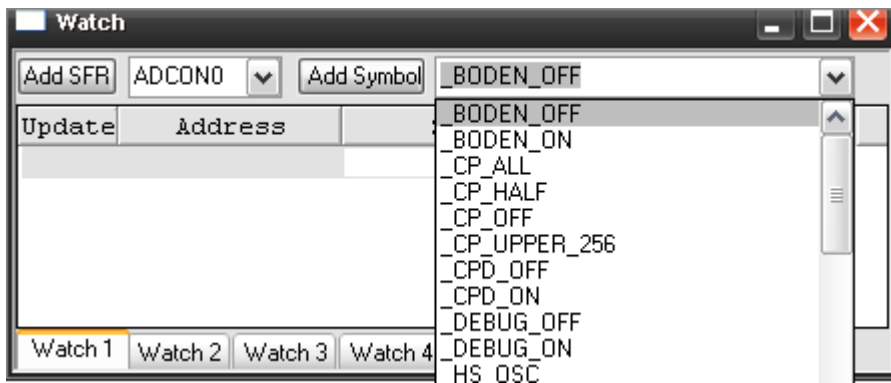
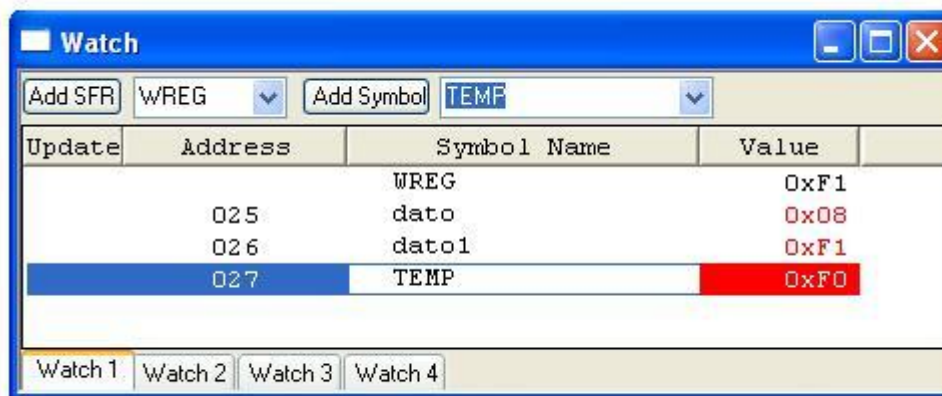


Fig. 1.21 Para seleccionar de los registro GPR

- G. Escoger de los registros **SRF**, el registro **WREG** y de los **GPR** (add symbol) los registros que se nombraron como **dato**, **dato1** y **TEMP**. Debe observar lo siguiente.



- H. Simular el programa con la metodología anterior y obtenga el valor de los registros **dato**, **dato1** y **TEMP** al finalizar el programa, es decir ejecutarlo una sola vez (Sugerencia utilice el modo paso a paso).

Registro (GPR)	Valor
dato	
dato1	
TEMP	

- J. Simular y observar el valor de cada registro del siguiente programa al ser realizarse una vez, utilizando los pasos comentados anteriormente (Sugerencia utilice modo paso a paso).

```

LIST P=16F887      ;microcontrolador a utilizar
#include "p16F887.inc"
cblock 025

var1
var2
var3

endc

```

```

inicio
    movlw 2FH
    movwf var1
    incf var1,1
    movlw 05H
    movwf var2
    addwf var1,0
    movwf var3
    goto inicio

end

```

Registro (GPR)	Valor
var1	
var2	
var3	
W	

K. Conclusiones

A. Realizar conclusiones de manera individual.

L. Cuestionario

- ¿Cuál es la versión de MPLAB que utilizas en la práctica?
- Menciona los paso para crear un proyecto en MPLAB IDE
- Cuales la longitud máxima de la ruta de un proyecto o programa para que no genere ningún error.
- Con que extensión debe de guardar el archivo fuente y que significa
- ¿Qué ventaja tiene realizar una simulación utilizando la opción paso a paso?
- Que indica directiva END
- ¿Cual es la función del la directiva INCLUDE?
- ¿Por qué en Cblock se le coloca el valor de 25?
- ¿Por qué en la ventana de Watch el registro W no tiene dirección?.
- Menciona, ¿Por qué en la configuración escoger en tipo de oscilador XT
- ¿Por es necesario apagar o deshabilitar el watchdog?

Comentarios Finales

- El alumno entrega un reporte de la práctica, como el profesor lo indique.
- Además, en el reporte deben anexarse las conclusiones y cuestionario contestado.