

METODOLOGÍA DESARROLLADA PARA PRÁCTICAS DE PROGRAMACIÓN DE MICROCONTROLADORES PIC

M.A. VICENTE, C. FERNÁNDEZ., y M.N. ROBLES

*Dpto. de Ingeniería de Sistemas Industriales. Universidad Miguel Hernández. Elche, España.
suni@umh.es, c.fernandez@umh.es*

En esta ponencia se presenta la metodología utilizada en la asignatura Sistemas Electrónicos y Automáticos, de la Universidad Miguel Hernández, para la realización de prácticas de programación de microcontroladores PIC. Se propone, básicamente, la realización de prácticas mediante el uso de un entrenador o placa de evaluación de microcontroladores y también mediante la simulación software de dicho entrenador.

1. Introducción

En este artículo se presenta la metodología utilizada en la asignatura Sistemas Electrónicos y Automáticos [1], de la Universidad Miguel Hernández, para la realización de prácticas de programación de microcontroladores PIC [2].

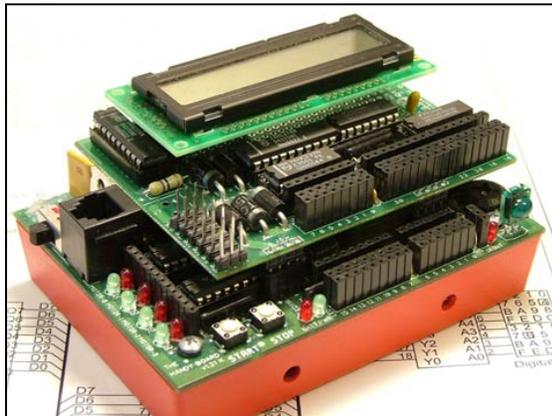
La asignatura Sistemas Electrónicos y Automáticos se imparte actualmente en cuarto curso de la titulación de Ingeniería Industrial. Se trata de una asignatura troncal de 10.5 créditos, siendo 6 de ellos teóricos y 4.5 prácticos. En el programa de la asignatura se pueden distinguir dos bloques temáticos diferenciados correspondientes a cada uno de los cuatrimestres del curso: en el primer cuatrimestre se imparte el bloque temático correspondiente a Sistemas de Control en el Espacio de Estado, y en el segundo cuatrimestre se imparte el temario relativo a Instrumentación de control, cuyo temario se basa fundamentalmente en la programación de microcontroladores PIC (80% del total de este cuatrimestre).

2. Experiencia en la docencia de programación de microcontroladores

La docencia universitaria en programación de microcontroladores es compleja y rápidamente cambiante, a diferencia de asignaturas clásicas como Teoría de Circuitos, por poner un ejemplo, donde los conceptos teóricos (leyes de Kirchoff, teoremas de Thevenin, Superposición, etc) y las prácticas no tienen porqué cambiar todos los años, salvo los enfoques particulares de cada profesor. En las asignaturas relativas al estudio de microcontroladores es necesario actualizar tanto el temario como las prácticas, puesto que los diferentes microcontroladores que aparecen en el mercado cambian rápidamente con el tiempo, dejando obsoleto el temario y los equipos de prácticas.

En el caso concreto de la asignatura objeto de este artículo, las prácticas comenzaron en el curso 2000-2001, con el microcontrolador tipo Von-Neumann Motorola 68HC11 [3] y el entorno de desarrollo HandyBoard [4]. En poco tiempo, los microcontroladores tipo Harvard coparon el mercado, y en concreto la familia de microcontroladores PIC de Microchip [2] se ha expandido hasta convertirse en líder del mercado. Ante esta situación, en el curso 2003-2004 se adquirieron nuevos equipos de prácticas basados en el entrenador MicroPIC Trainer [5] (gama media de la familia PIC). Durante dos cursos académicos se

realizaron prácticas con los equipos antiguos y los equipos nuevos simultáneamente, lo que permitía que los alumnos conociesen las particularidades de cada tipo de microcontrolador; sin embargo los resultados no fueron buenos dado que conocer dos juegos de instrucciones y dos arquitecturas completamente diferentes resultaba excesivamente complicado para un solo cuatrimestre de la asignatura. La decisión final ha sido utilizar exclusivamente los microcontroladores de la familia PIC para las prácticas.



(a)



(b)

Figura 1. Detalles de los entrenadores a) Handy Board y b) MicroPIC Trainer.

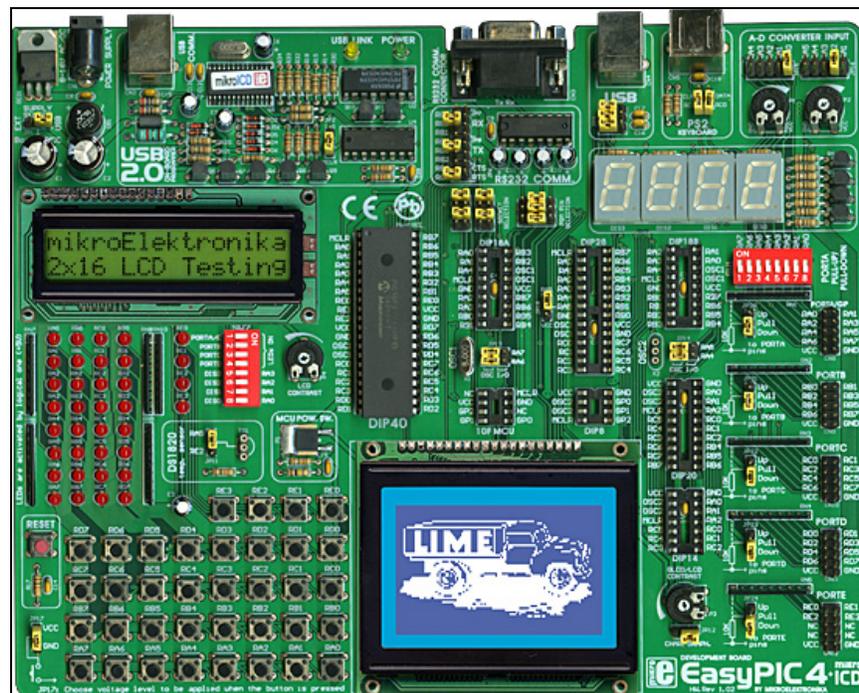


Figura 2. Detalle de la placa de evaluación EasyPIC4

Incluso dentro de la familia PIC, ha sido necesaria una renovación. A partir del pasado curso 2006-2007 se han adquirido unas nuevas placas entrenadoras (EasyPIC4 [6]), que ofrecen diversas ventajas respecto de las anteriores placas MicroPIC Trainer, entre las que destacan:

- En los entrenadores EasyPIC4, el software de grabación funciona bajo sistema operativo Windows (en el caso de las placas MicroPIC Trainer era necesario trabajar bajo sistema operativo DOS).
- Disponen de más dispositivos gráficos, como un GLCD y hasta cuatro displays 7 segmentos, además de otros tipos de periféricos.
- Es posible conectar en sus zócalos cualquier microcontrolador PIC de gama media (en el caso de las placas MicroPIC Trainer las opciones eran más escasas). En concreto, el sistema EasyPIC4 admite microcontroladores de 8, 14, 18, 28 y 40 pines.

Actualmente, en la asignatura Sistemas Electrónicos y Automáticos, se utilizan como modelos de microcontroladores PIC a estudiar el PIC16F84A y el PIC16F877A, pertenecientes ambos a la gama media de Microchip.

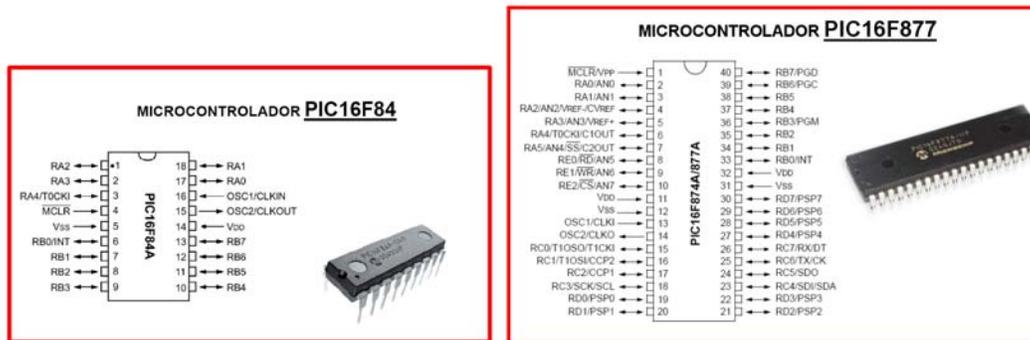


Figura 3. Microcontroladores PIC utilizados en la asignatura

A continuación se muestran las características generales del PIC16F877A:

- Procesador de arquitectura RISC avanzada:
 - Juego de sólo 35 instrucciones con 14 bits de longitud. Todas ellas se ejecutan en un ciclo de instrucción, menos las de salto que tardan dos.
 - Hasta 8K palabras de 14 bits de memoria de programa tipo FLASH.
 - Hasta 368 Bytes de memoria de datos RAM.
 - Hasta 256 Bytes de memoria de datos EEPROM.
 - Pines de salida compatibles con los microcontroladores PIC16CXXX y PIC16FXXX.
- Recursos analógicos:
 - Conversor Analógico/Digital de 10 bits.
 - Reset de Brown-Out (BOR).
 - Módulo de comparador analógico.
- Recursos especiales:
 - Código de protección programable.
 - Modo SLEEP de bajo consumo.

- Perro Guardián (WDT).
- Programación serie en circuito con dos pines. Sólo necesita 5V para programarlo en este modo.
- Recursos periféricos.:
 - Timer0: Temporizador-contador de 8 bits con prescaler de 8 bits.
 - Timer1: Temporizador-contador de 16 bits con prescaler, puede incrementarse en modo sleep de forma externa por un cristal/clock.
 - Timer2: Temporizador-contador de 8 bits con registro de periodo, prescaler y postescaler.
 - Dos módulos de Captura, Comparación, PWM.
 - Puerto Serie Síncrono (SSP) con SPI (Modo maestro) e I2C (Master/Slave).
 - USART/SCI (Universal Synchronous Asynchronous Receiver Transmitter) con 9 bits.
 - Puerta Paralela Esclava (PSP) con control externo RD, WR y CS (sólo en encapsulados con 40 pines).
- Tecnología CMOS.:
 - Voltaje de alimentación comprendido entre 2,0V y 5,5V.
 - Bajo consumo.

Para más información sobre el microcontrolador se recomienda ver la hoja de características del componente disponible [2].

3. Entorno de desarrollo EasyPIC4

El sistema de desarrollo EasyPIC4 consiste en un entrenador o placa didáctica de evaluación para aplicaciones basadas en los microcontroladores Microchip PIC. Se ha diseñado para permitir a estudiantes e ingenieros explorar y trabajar con las capacidades de los microcontroladores PIC. Permite además, concentrarse principalmente en el desarrollo del software puesto que las conexiones entre microcontroladores PIC y circuitos externos son muy sencillas de realizar.

Dispone de una serie de periféricos básicos de E/S con los que se puede verificar el funcionamiento de una aplicación, así como los circuitos necesarios para la grabación de diversos modelos de microcontroladores PIC. En la figura 2 se aprecia el aspecto del equipo EasyPIC4 y en las figuras 4 y 5 algunos detalles de los periféricos.

El sistema de desarrollo EasyPIC4 se presenta totalmente montado, a excepción del LCD, GLCD y el sensor de temperatura, con un manual donde se incluye un tutorial con diversos ejemplos de demostración.

También se incluye un CD-ROM con las diferentes herramientas de diseño así como los programas fuentes de los ejemplos propuestos en el manual. En la tabla 1 se enumeran las principales características del sistema EasyPIC4.



Figura 4. Detalle de los leds y el LCD de la EasyPIC

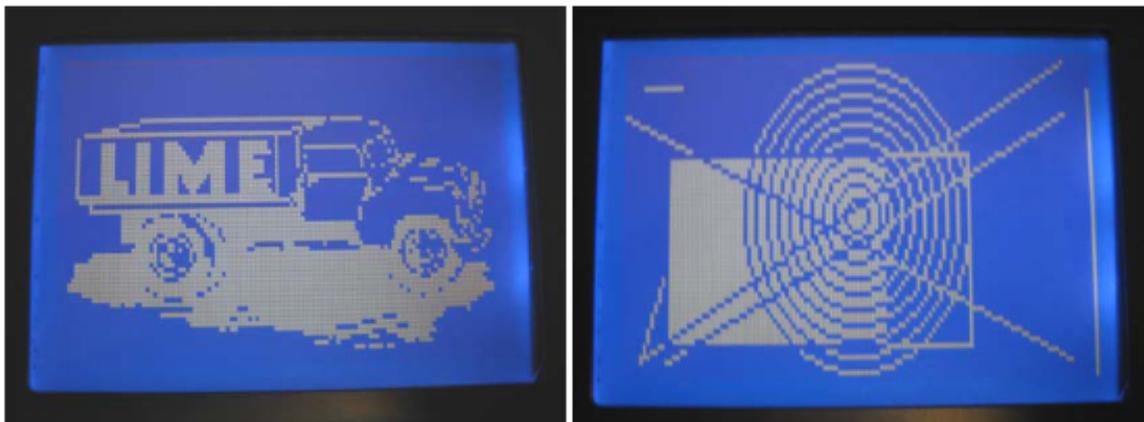


Figura 5. Imágenes mostradas en el GLCD de la EasyPIC

4. Simulación software del entrenador EasyPIC4

Además del problema de la actualización continua de los equipos, existe otro problema en esta asignatura: la dificultad de programar los equipos de prácticas es normalmente demasiado elevada como para que las prácticas den resultados satisfactorios para la mayor parte de los alumnos. El tiempo disponible durante las sesiones de laboratorio es normalmente escaso para corregir todos los errores de programación en que puede incurrir un programador inexperto. Por esta razón, se ha decidido facilitar a los alumnos un entorno de simulación en el que pueden realizar las prácticas sin necesidad de acceder al equipo hardware. Este entorno de simulación está basado en el software Proteus [7], que permite la simulación de circuitos electrónicos con microcontroladores.

Se ha desarrollado un esquema Proteus que simula exactamente la tarjeta de entrenamiento EasyPIC4. Con este software los alumnos pueden trabajar en los programas de sus prácticas tanto durante las sesiones de laboratorio como desde sus hogares, si lo desean. De este modo, disponen de tiempo suficiente para depurar sus programas. Adicionalmente, también se ha desarrollado un esquema que simula parcialmente el funcionamiento del entrenador (se han eliminado algunos de los periféricos). En la figura 6 podemos ver un detalle de este entrenador básico, y en la figura 7 el esquema completo para el entrenador EasyPIC4.

1. Alimentación mediante fuente de alimentación externa AC/DC de 8 a 16 V.
2. Selector de fuente de alimentación externa o vía USB.
3. Programador USB integrado.
4. Sensor de temperatura DS1820.
5. Comunicación RS232 con selección TX y RX para microcontroladores pequeños.
6. Potenciómetros P1 y P2. Algunos pines están conectados a dichos potenciómetros pudiendo utilizarse para medir tensiones.
7. Mediante los interruptores SW1 se conecta el PORTA a una red de resistencias. Si un interruptor se encuentra en la posición OFF, el pin asociado no estará conectado con la resistencia pull-up o pull-down. Esto es importante ya que permite que el PORTA pueda ser utilizado en modo analógico como un conversor A/D o como un puerto digital de E/S.
8. Jumpers. Al situar el jumper en la posición de arriba (pull-up) aparece un uno digital en el puerto correspondiente. Si el jumper se encuentra en la posición de abajo (pull-down), los pines reciben un cero lógico (pull-down).
9. Conexión de un LCD en modo 4 bits.
10. Conexión de un LCD gráfico o un LCD en modo de 8 bits.
11. Zócalos para situar los microcontroladores en DIP8, DIP14, DIP18, DIP20, DIP28 y DIP40.
12. 36 botones para controlar todos los pines del microcontrolador.
13. Jumper para seleccionar la forma en la que afectará el apretar el botón al pin.
14. LEDs, cada pin del microcontrolador tiene asociado un LED.
15. Displays de 7 segmentos en modo multiplexor.
16. Interruptores que encienden o apagan los LEDs de los puertos PORTA, PORTB, PORTC, PORTD y PORTE.
17. Selección de contraste del LCD.
18. Control de fuente de alimentación.
19. Comunicación USB para MCU con soporte USB
20. Conector para el teclado.
21. Circuito de reset.

Tabla 1. Principales características de la placa de evaluación EasyPIC4

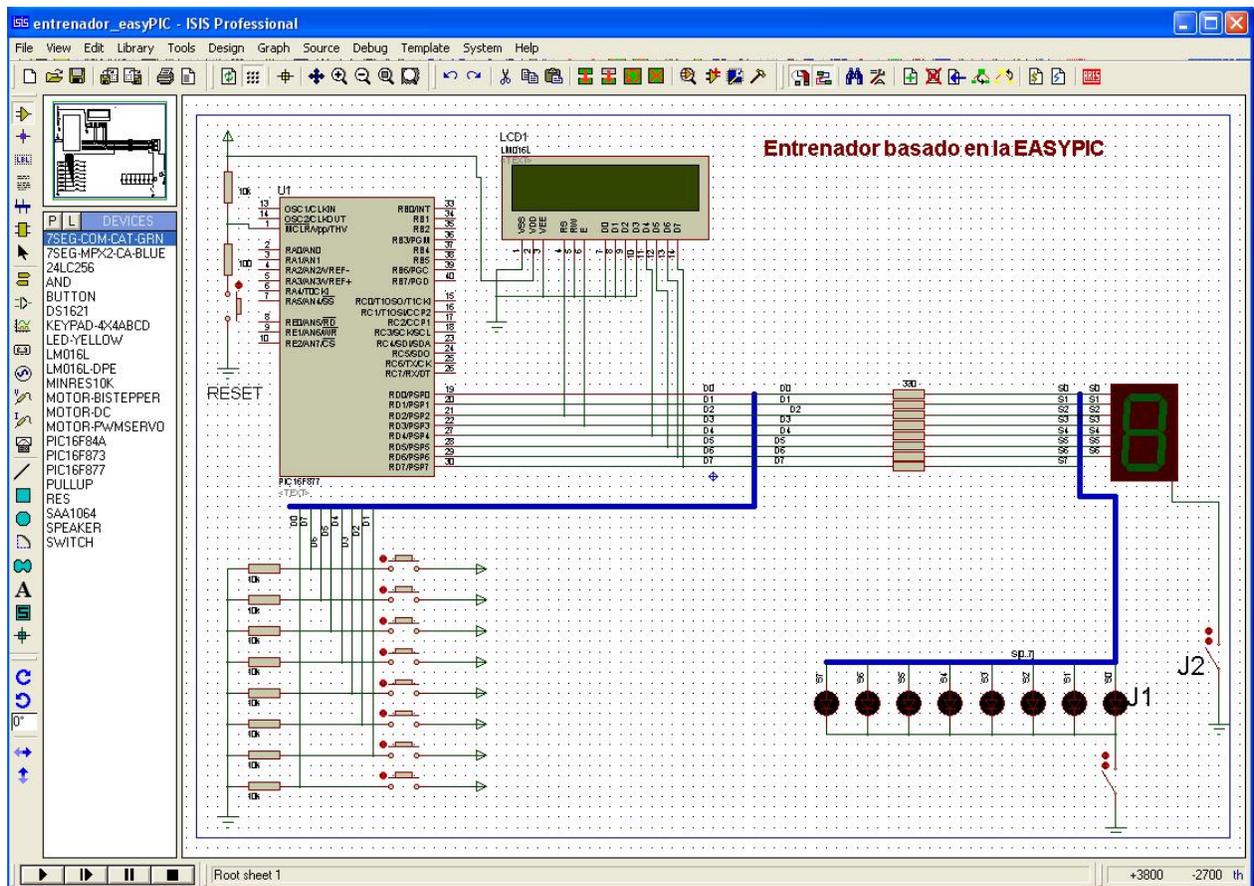


Figura 6. Esquema Proteus del entrenador básico inspirado en EasyPIC4

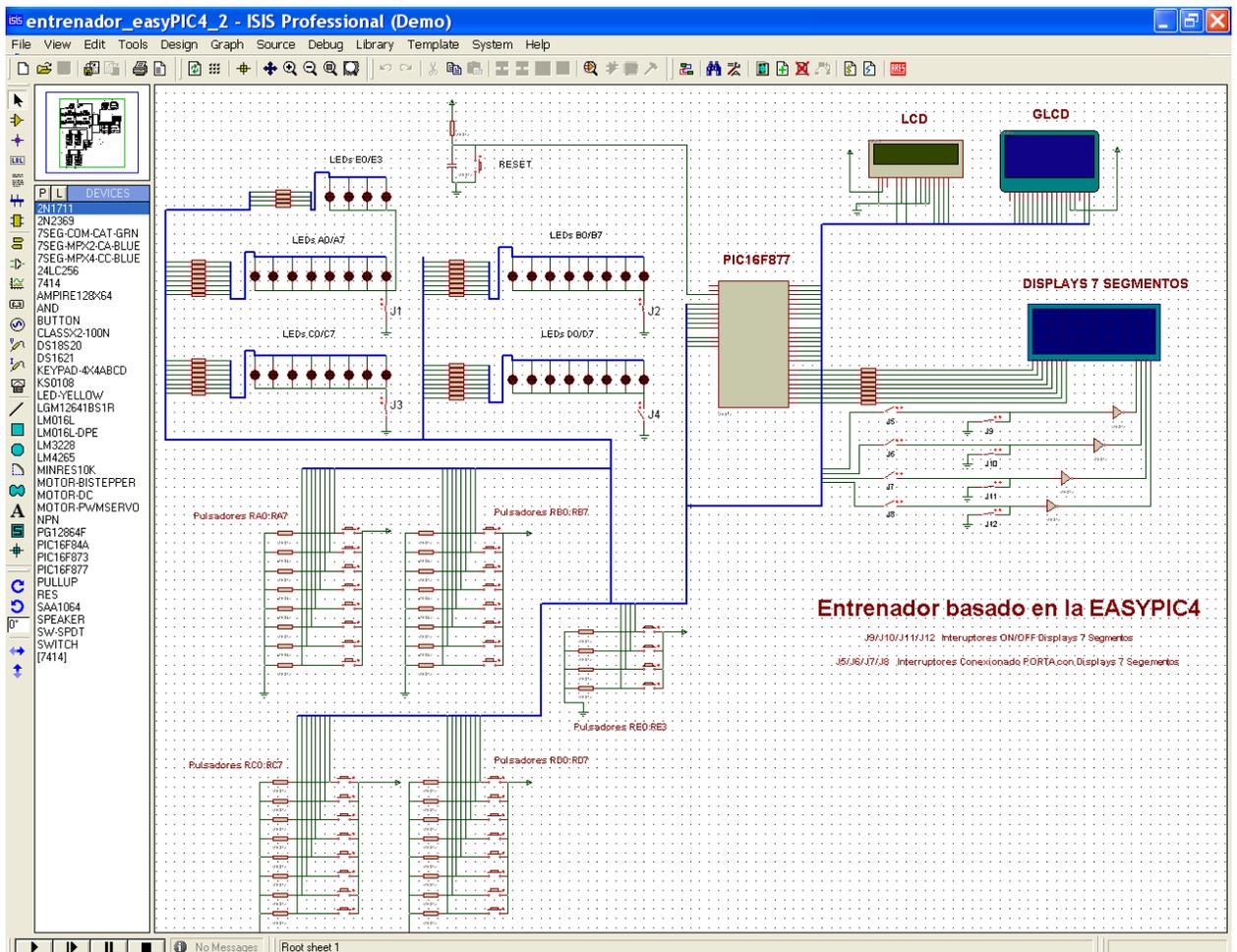


Figura 7. Esquema Proteus del entrenador EasyPIC4

5. Programa práctico propuesto para la asignatura

Actualmente, el programa práctico de la asignatura consta de diez sesiones de dos horas de duración cada una, en las que se profundiza paulatinamente en la programación de microcontroladores. A continuación se enumeran los títulos de estas prácticas (el contenido completo de las mismas es accesible desde la página web de la asignatura [1]):

1. Entorno de desarrollo MPLAB-IDE. Primeros ejemplos de programación.
2. Programas básicos. Tipos de direccionamiento.
3. Bucles.
4. Entorno de desarrollo EasyPIC4.
5. Medida del tiempo en un PIC.
6. Gobierno del display 7 segmentos.
7. El módulo LCD.
8. El módulo LCD (II).
9. Interrupciones.
10. Interrupciones (II).

Puede apreciarse que las tres primeras sesiones prácticas sirven únicamente como introducción a la programación en ensamblador, y no se desarrollan sobre los entrenadores EasyPIC4. Para estas tres prácticas se utiliza el entorno de desarrollo gratuito MPLAB-IDE [2] (ver detalle en Figura 8), que los alumnos pueden instalar en sus ordenadores personales para realizar las prácticas desde sus hogares. El resto de las sesiones se pueden llevar a cabo tanto sobre los equipos hardware disponibles en el laboratorio (EasyPIC4) como en simulación utilizando los esquemas Proteus que se les suministran (básico y completo). De este modo, los alumnos tienen la posibilidad de trabajar desde sus casas en todas las prácticas de la asignatura.

La mayor parte de los programas propuestos en las prácticas están basados en los magníficos ejemplos propuestos en [8]. Los programas originales están desarrollados para el PIC 16F84A, pero en nuestro caso hemos decidido utilizar un PIC superior para poder hacer uso de más periféricos disponibles en el entrenador EasyPIC4. Por esa razón se han adaptado los programas para el PIC 16F877A. También ha sido necesario un trabajo de adaptación para que los programas propuestos en [8], cada uno de los cuales hace referencia a un hardware específico, funcionen sobre un mismo equipo (el entrenador EasyPIC4).

6. Conclusiones

En este artículo se ha descrito la metodología que se utiliza actualmente en la asignatura Sistemas Electrónicos y Automáticos, de la Universidad Miguel Hernández, para la docencia teórica y práctica de programación de microcontroladores. Como hemos visto, se utilizan los microcontroladores PIC de gama media como modelos a estudiar y, en las prácticas de laboratorio se hace uso de un entrenador o placa de evaluación de microcontroladores para ayudar al desarrollo y comprensión de los programas en ensamblador.

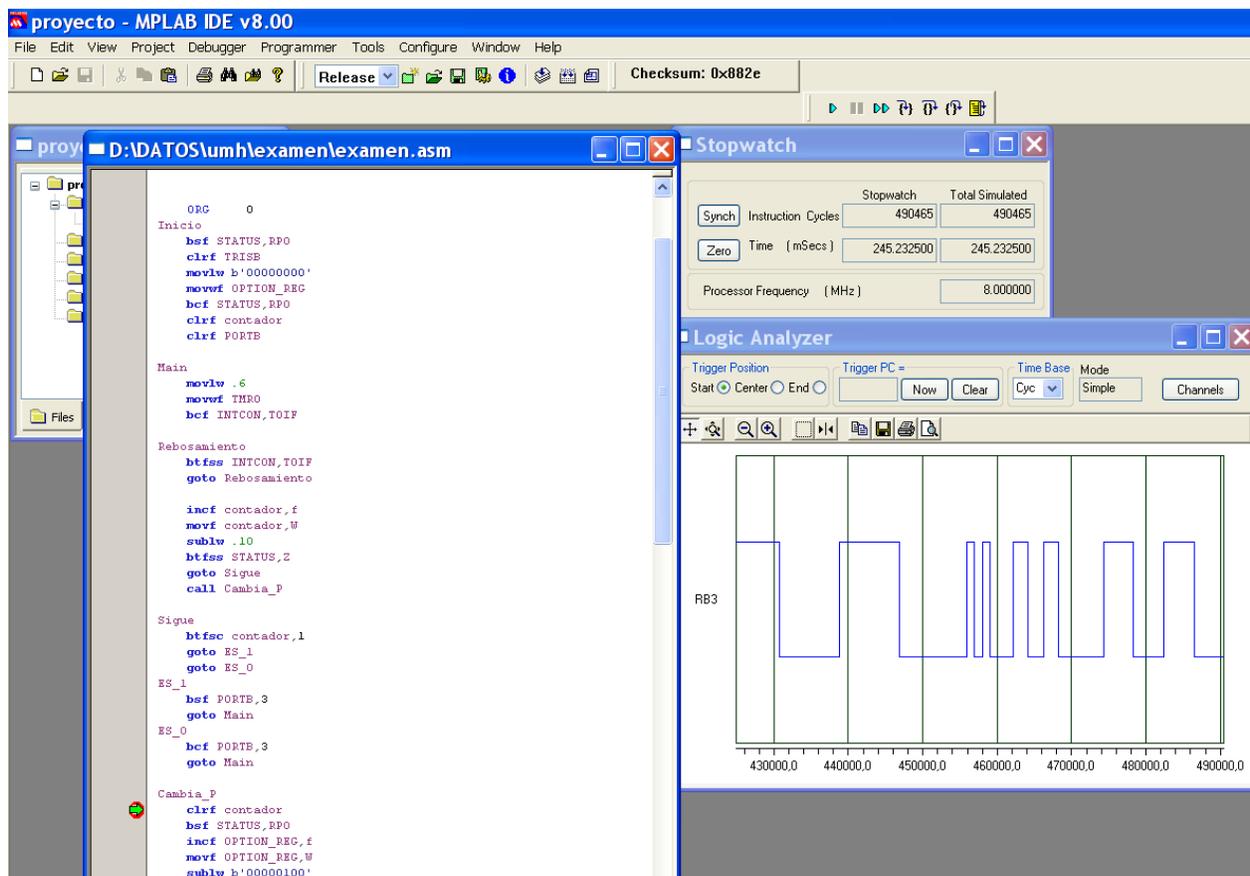


Figura 8. Detalle del software de edición, depuración, ensamblado y simulación MPLAB-IDE

Referencias

- [1] Asignatura Sistemas Electrónicos y Automáticos en la Universidad Miguel Hernández, http://coolab.umh.es/teaching/sea_oficial/default.htm
- [2] Microchip, <http://www.microchip.com/>
- [3] Motorola 68HC11, <http://isa.umh.es/isa/es/temas/micros/doc/doc.html#HC11>
- [4] Handy Board, <http://isa.umh.es/isa/es/temas/micros/doc/doc.html#HB>
- [5] MicroPIC Trainer http://www.msebilbao.com/tienda/product_info.php?products_id=111
- [6] Entorno de desarrollo EasyPIC4, <http://www.mikroe.com/en/tools/easypic4/>
- [7] Proteus, <http://www.ieeproteus.com/>
- [8] E. Palacios, F. Remiro y L.J. López, *Microcontrolador PIC16F84, Desarrollo de proyectos*, Editorial Ra-Ma (2004)