

# Primeros pasos

Programación y diseño de dispositivos mediante microcontroladores PIC

Dr. Eugenio Martín Cuenca  
Ing. José María Moreno Balboa

Comenzamos con los microcontroladores PIC. ¿Pero con cuál o cuáles de entre ellos? Hemos elegido el microcontrolador PIC16F84 por las características que se explicarán más adelante. Ahora que ya disponemos de nuestro microcontrolador PIC16F84 (ver «Figura 1»), que es un chip de 18 pines, se nos plantean una serie de preguntas: ¿qué hacemos con él?, ¿cómo lo hacemos funcionar?, ¿cómo introducimos el programa dentro del mismo?

Las respuestas a estas preguntas son al menos dos. Por un lado necesitamos un dispositivo que permita grabar el programa elaborado dentro del PIC16F84, es decir, un grabador o programador. Además, es necesario un compilador que produzca el código ejecutable de dicho programa. Una vez introducido el programa en la memoria del microcontrolador, éste no funcionará así porque sí, necesita un mínimo de componentes para poder hacerlo (ver «Figura 2»).

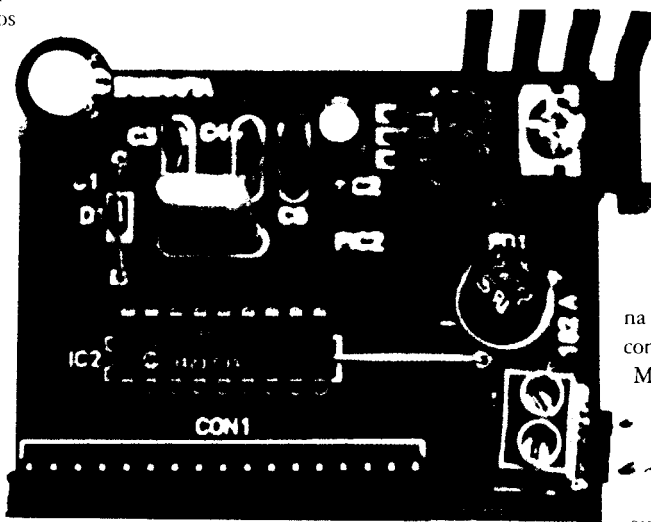
Los autores de este artículo hemos pensado en los lectores y desde estas páginas proporcionaremos las herramientas imprescindibles para poder realizar no sólo los dispositivos que se presentan, sino que también serán capaces de finalizar con éxito sus propios proyectos.

Para dicho fin se ha diseñado y realizado un sistema de aprendizaje modular que consta como mínimo del grabador MultiPIC (ver «Foto 2»), el Módulo-01 (ver «Foto 1») que contiene el microcontrolador y el Módulo de aprendizaje (ver «Foto 3»). Además, el software se ha desarrollado para su uso tanto bajo MS-DOS, lo que permite que el lector pueda emplear cualquier ordenador anticuado, como bajo Windows 95/98. Todo este material se irá presentando a lo largo de la serie de artículos.

Las herramientas imprescindibles para ponernos a trabajar son:

—Si desarrollamos bajo MS-DOS es necesario un editor de texto ASCII como el Edit. Si trabajamos con Windows 95/98 el programa del grabador MultiPIC ya incorpora un editor ASCII.

*En el número anterior se introdujo el concepto de microcontrolador, haciendo una breve referencia a los asuntos que se tratarán en esta sección. Se prestará especial interés a la programación de los microcontroladores en lenguajes de alto nivel como son el Basic y el C, aunque no por ello se dejará completamente de lado la programación en ensamblador.*



«Foto 1». Imagen del kit del Módulo-01 montando un PIC16C84. Se puede apreciar el conector CN1.

editor ASCII, así como el compilador ensamblador MPLAB que proporciona Microchip.

—Un microcontrolador PIC16F84 junto con la electrónica necesaria para hacerlo funcionar como por ejemplo el Módulo-01.

—Opcionalmente el Módulo de aprendizaje.

Nuestra intención es colocar en la página web «www.ugr.es/~emartin/Microcontrol-01.htm» los programas MPASM y MPLAB. Aunque pretendemos atender la mayor cantidad de consultas o dudas, dada la imposibilidad de una atención individualizada y personal, en la página web se responderá a las preguntas más frecuentes. De todas formas, no dudéis en comunicarnos vía e-mail (emartin@goliat.ugr.es) con nosotros siempre que lo creáis necesario.

—Ensamblador MPASM para MS-DOS o el MPLAB para Windows que proporciona gratuitamente la empresa Microchip.

—Un compilador PicBasic o C.

—Grabador o programador MultiPIC.

—Software de grabación MultiPIC, ya sea en su versión para DOS o Windows 95/98. La versión de Windows incluye un

## PIC16F84. Módulo-01

Comencemos por el principio. El microcontrolador PIC 16F84 pertenece a la gama media del fabricante y posee unas características óptimas para su empleo en fases de aprendizaje. Este microcontrolador dispone de memoria Flash y está fabricado con tec-

DESCRIPCIÓN PINES PIC16F84 16C84					
Nombre	Nº	Tipo I/O/P	Tipo Buffer	Descripción	
OSC1 CLKIN	18	I	ST CMOS	Entrada oscilador a cristal / fuente externa de reloj.	
OSC1 CLKOUT	15	O	—	Salida oscilador a cristal. Conectar al cristal de cuarzo.	
MCLR	4	I/P	ST	Reset a nivel 0. Entrada voltaje de programación.	
RA0	17	I/O	TTL	El Puerto A es bidireccional.	
RA1	18	I/O	TTL	El pin RA4 del Puerto A, si se programa como salida es de colector abierto. Como entrada puede seleccionarse en funcionamiento normal o como entrada del contador / temporizador TMRC es decir TOCK1.	
RA2	1	I/O	TTL		
RA3	2	I/O	TTL		
RA4 TOCK1	3	I/O	ST		
RB0 INT	8	I/O	TTL / ST	El pin 8 del Puerto B, RB0 puede programarse como entrada de interrupciones externas INT.	
RB1	7	I/O	TTL	El Puerto B es bidireccional: mediante software, cuando se programa como entrada pueden activarse resistencias de polarización internas.	
RB2	8	I/O	TTL	Los pines RB4 a RB7 pueden programarse para que respondan a interrupción por cambio de estado.	
RB3	9	I/O	TTL		
RB4	10	I/O	TTL		
RB5	11	I/O	TTL		
RB6	12	I/O	TTL / ST	Modo programación entrada RELO.	
RB7	13	I/O	TTL / ST	Modo programación DATOS.	
V <sub>ss</sub>	5	P	—	Pin de masa GND.	
V <sub>dd</sub>	14	P	—	Pin de alimentación positiva.	

«Figura 1». Breve descripción y disposición de patillas del PIC 16F84 y 16C84.

## PC Práctico

### Microcontroladores

nología CMOS. Su consumo es muy reducido, siendo un circuito estático que permite que su reloj pueda detenerse completamente sin pérdida de los contenidos de los registros. El consumo típico es de 2 mA a 4 MHz y unos 40  $\mu$ A funcionando en modo Sleep.

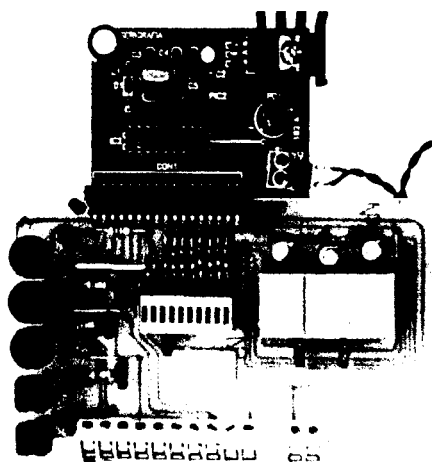
El encapsulado de 18 pines así como la descripción del patillaje puede verse en la «Figura 1». Existe un modelo de 4 MHz, el PIC16F84-04/P, y otro de 10 MHz, el PIC16F84-10/P. Para el desarrollo de las prácticas se ha diseñado el Módulo-01, que se muestra en la «Foto 1», mientras en la «Figura 2» se presenta el esquema eléctrico del mismo para aquellos aficionados a la electrónica y al soldador.

Se propone en primer lugar el Módulo-01 o comodín, ya que se emplea en casi todas las experiencias a realizar. Contiene sólo y exclusivamente el microcontrolador PIC16F84 así como la circuitería necesaria para su alimentación partiendo de una entrada en alterna de 9-12 V A.C. o desde una pila de 9V. Incluye además un conector de 17 pines (CN1) donde se encuentran todas las líneas útiles del microcontrolador así como las señales de masa, +5V y el voltaje rectificado y filtrado del transformador PWR.

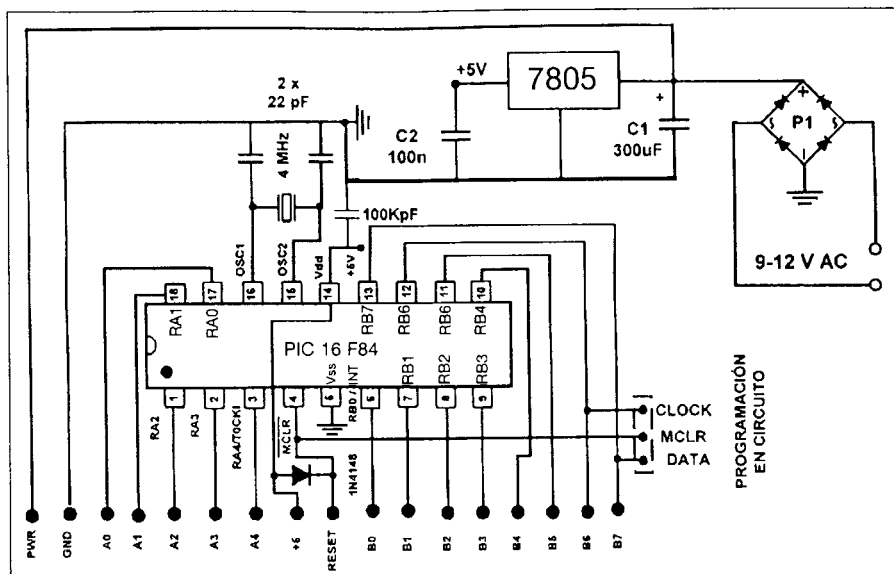
Con la sola sustitución del cristal de cuarzo y dos condensadores puede hacerse funcionar el microcontrolador a 4 o 10 MHz. Los módulos empleados en los ejemplos funcionan a 4 MHz; por lo tanto, si el lector los usa a una velocidad de 10 MHz debe tener en cuenta esta mayor velocidad a la hora de escribir sus programas.

#### Arquitectura interna

Las altas prestaciones de los microcontroladores PIC derivan de las características



«Foto 3». Módulo-01 y Módulo de aprendizaje interconectados a través del conector CN1.

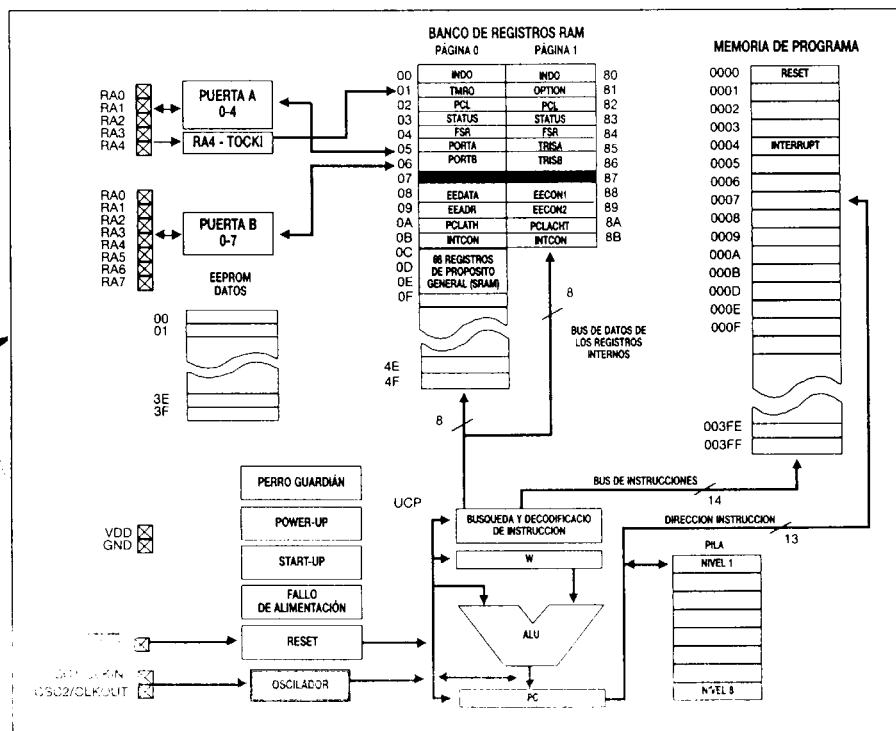


«Figura 2». Esquema eléctrico del Módulo-01.

de su arquitectura (ver «Figura 3»). Están basados en una arquitectura tipo Harvard, es decir, estos microcontroladores disponen de dos memorias independientes: una que contiene sólo instrucciones y otra sólo con datos. Ambas poseen sus respectivos sistemas de buses y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ellas. Esto los hace más rápidos que los microcontroladores basados en la arquitectura tradicional de

Von Neumann, permitiendo además que las instrucciones tengan un tamaño de 14 bits. Otra característica es su juego de instrucciones reducido (35 instrucciones) RISC, que en su mayoría se ejecutan en un solo ciclo de reloj excepto las instrucciones de salto que necesitan dos ciclos.

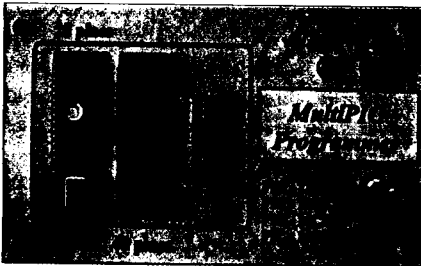
Los microcontroladores PIC y por tanto el 16F84 poseen dos bloques de memoria separados, la memoria de programa y los bancos de registros. (ver «Figura 3»). La



«Figura 3». Diagrama esquemático de la arquitectura interna del PIC16F84.

memoria de programa está organizada con palabras de 14 bits con un total de 1K. Es del tipo EEPROM y en funcionamiento es de sólo lectura. Únicamente se ejecutará el código contenido en esta memoria, pudiéndose además almacenar en ella una cantidad limitada de datos como parte de la instrucción RETLW. En una sola palabra se agrupan el código de la instrucción y el operando o su dirección.

Los bancos de registros RAM son de 8 bits (byte) a excepción del contador del programa que es de 13 bits. Como puede verse en la «Figura 3» las direcciones bajas contienen los registros especiales y los registros de recursos, encontrándose a continuación los registros de propósito general. El PIC16F84 posee 68 registros que van de la dirección \$00 a la \$4F, aunque no todas estas direcciones se usan. El PIC16C84 sólo posee 36 registros RAM (esta es una de las pocas diferencias entre el PIC16C84 y el PIC16F84).



«Foto 2». El programador MultiPIC está diseñado para la grabación de múltiples modelos de microcontroladores PIC.

El vector de *reset* se encuentra en la posición 0000h y el de interrupciones en la 0004h, por lo que la memoria de usuario propiamente dicha se extiende desde la dirección 0005h a la 03ffh.

La pila es de ocho niveles. No existe ninguna bandera que indique que esté llena, por lo que será el programador el responsable de controlar que no se produzca su desbordamiento.

Posee además una memoria EEPROM de datos de 8 bits. Ésta no forma parte del espacio normal direccionable y sólo es accesible en lectura y escritura a través de dos registros: para los datos el EEDATA que se encuentra en la posición 0008h del banco de registros RAM y para las direcciones el EEADR en la 0009h. Para definir el modo de funcionamiento de esta memoria se utilizan dos registros especiales, el EECON1 dirección 0088h y el EECON2 en 0089h. Esta memoria EEPROM no emplea ningún recurso externo de alimentación. Su programa-

```

PROGRAMA CONTAR.ASM
;*****
; Programa CONTAR.ASM                               Fecha :
*
; Este programa cuenta de 0 a 255 sacando el dato binario a los LED *
; conectados a la Puerta B introduciendo un retardo durante la cuenta *
; Revisión : 0.0                                     Programa para PIC16C84/16F84 *
; Velocidad del reloj : 4 MHz                       Reloj Instrucción: 1 MHz = 1 uS *
; Perro guardián : Deshabilitado                   Tipo de Reloj : XT *
; Protección del código : OFF
*
;*****
;***** IGUALDADES *****
;***** Igualdades que designa los destinos *****
w      EQU      0
f      EQU      1
;***** Igualdades de la UCP y del Mapa de memoria *****
PORTA      EQU      05h ; Puerta A
PORTB      EQU      06h ; Puerta B
TRISA      EQU      05h ; Registro triestado Puerta A. Página 1
TRISB      EQU      06h ; Registro triestado Puerta B. Página 1
STATUS     EQU      03h ; Registro STATUS
RP0        EQU      05h ; Bit 5 registro STATUS
Contador1  EQU      0Ch ; Registro para primer contador
Contador2  EQU      0Dh ; Registro para segundo contador
;***** Sección Código de reset *****
ORG      00h ; Dirección del Vector de reset
GOTO     Inicializa ; Comienza el programa destrás
; del Vector interrupción
ORG      05h ; Una posición a continuación
; del vector de interrupción
;***** Sección Inicializa *****
Inicializa BSF      STATUS,RP0 ; Selecciona página 1
           CLRF     TRISB      ; Programa Puerta B todo salida
           BCF      STATUS,RP0 ; Vuelve a página 0
           CLRF     PORTB     ; Apaga los LEDs borrando la Puerta B
           CLRF     Contador1 ; Inicializa (borra) Contador1 (0Ch)
           CLRF     Contador2 ; y Contador2 (0Dh)
;***** Sección Principal *****
Principal INCF     PORTB,f ; Incrementa en 1 la Puerta B
;***** Sección Bucle *****
Bucle     DECFSZ   Contador1,f ; Decrementa Contador1 y chequea si = 0
           GOTO     Bucle      ; Si no 0, decrementa Contador1
           DECFSZ   Contador2,f ; Si es 0, decrementa Contador2
           GOTO     Bucle      ; Si Contador2 no es 0, decrementa
           GOTO     Principal ; Contador1 otras 256 veces
           GOTO     Principal ; Actualiza los LED
           GOTO     Principal ; después de 65536 iteraciones
END
    
```

ción dura unos 10 ms y se controla mediante un temporizador interno.

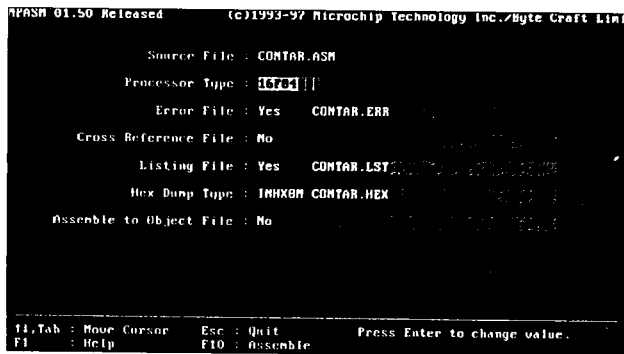
Mencionar por último que el microcontrolador dispone de dos puertos de entrada y salida (E/S). El puerto A con 5 líneas de la RA0 a la RA4 en la que hay que distinguir la A4 o T0CKI (*Timer 0 Clock Input*). Como puede verse en la breve descripción de la «Figura 1», esta línea es compartida con la entrada para el timer 0 (TMRO) a través de un *trigger* de Schmitt y que cuando se configura como salida es de colector abierto. El puerto B dispone de 8 líneas de entrada/salida que van de la RB0 a la RB7. En éste cabe

mencionar que la línea RB0 o INT es con partida con la entrada de interrupción externa (ver «Figura 1»).

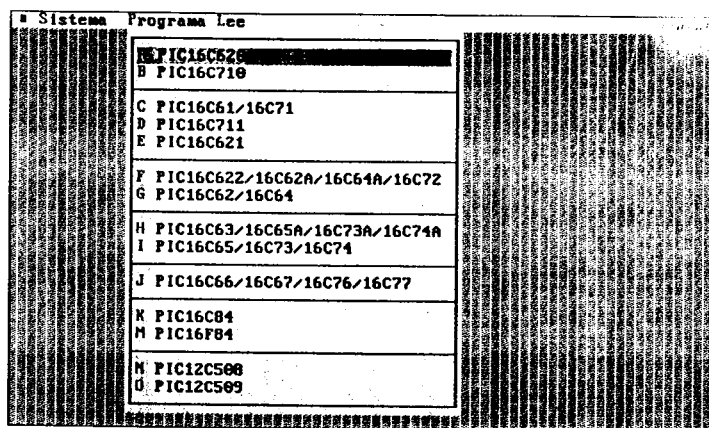
### El programador MultiPIC

En la «Foto 2» puede verse el nuevo programador MultiPIC que permite grabar mayoría de los microcontroladores PIC, abarcando desde los de gama baja de 8 pines con el 12C508 que se encuentra en el sistema ant copia de la PlayStation, hasta los de 40 pines como el 16C74 de la gama media.

El grabador se conecta al puerto paralelo de cualquier PC: no importa el tipo, pue-

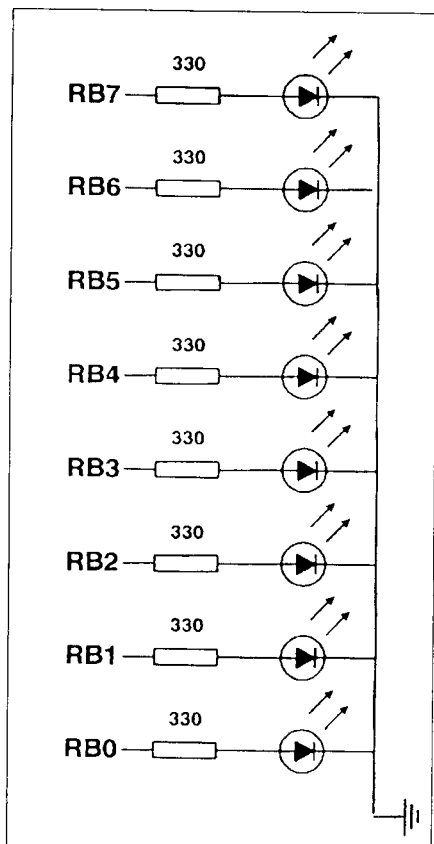


Aspecto que presenta la pantalla del ordenador cuando se emplea el programa ensamblador MPASM. Como se observa en el ejemplo, en la línea Source File se indica el nombre del fichero a ensamblar CONTAR.ASM, en la Processor Type 16F84 y en la línea Hex Dump Type el fichero que se creará (CONTAR.HEX) en formato INHX8M.



Aspecto de la pantalla del ordenador durante el uso del programa de grabación MULTIPIC.EXE.

ser un antiguo XT o un moderno Pentium III. En el frontal, además del interruptor de encendido/apagado y de los zócalos para la programación se dispone de dos diodos LED para ver las señales de potencia (Vdd) y actividad (Vpp). El LED amarillo indica que el programador se encuentra bajo tensión de alimentación Vdd mientras que el LED verde de actividad señala que está siendo aplicada al PIC la tensión de programación Vpp (12.6 V). Estas tensiones se encuentran bajo control del programa.



«Figura 4». Montaje necesario para la realización de la experiencia.

### Primera experiencia

Aunque pueda parecer una incoherencia, el primer programa que haremos estará escrito en ensamblador, dado que debe dar la menor cantidad de pasos posible y esto le ayudará posteriormente con los lenguajes de alto nivel. Necesitará el Módulo-01 o un montaje similar. Lo primero que hay que hacer es realizar el montaje que vemos en la «Figura 4» conectando las líneas RB0 - RB7 (conector CN1) del Módulo-01 con una serie de resistencias de 330 ohmios además de un diodo LED rojo.

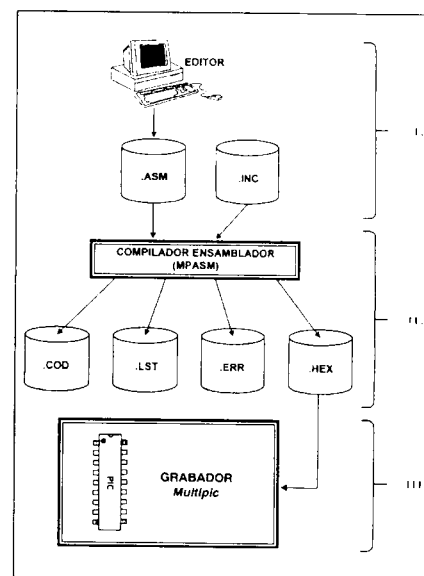
La experiencia va a consistir en la escritura y posterior grabación del microcontrolador PIC16F84 con el programa CONTAR.ASM, cuyo listado publicamos. La finalidad del mismo consiste en contar en binario a través del puerto B (patitas RB0 - RB7), iluminando los correspondientes diodos LED. Los pasos a seguir están esquematizados en la «Figura 7».

El primer paso consiste en la escritura del programa CONTAR haciendo uso de un editor de textos ASCII o del incorporado en la versión Windows del grabador del programa CONTAR haciendo uso de un editor de textos ASCII o del incorporado en la versión Windows del grabador del programa CONTAR.

Una vez escrito el programa debe salvarse al disco. El segundo paso es el ensamblado de este fichero (CONTAR.ASM) empleando el compilador-ensamblador MPASM.EXE que distribuye libremente el fabricante. El ensamblador produce varios ficheros; aunque de entre ellos el que

**Más información**  
Sólo hay que insertar o retirar el PIC del grabador cuando el LED de actividad está apagado. No se debe insertar nunca el PIC con el LED de actividad encendido. También es conveniente, si tenemos que realizar repetidas grabaciones durante el aprendizaje, colocar el PIC en un zócalo de patillas torneadas que soportará el esfuerzo de las repetidas inserciones y retiradas del zócalo de programación, evitando daños en las patillas del propio microcontrolador.

Martin Cuenca, E; Angulo, JM; Angulo, I. (1998).  
-Microcontroladores PIC. La solución en un chip-. 2ª Edición. Paraninfo-ITP.  
Martin Cuenca, E y Moreno Balboa, JM. -Diseño y realización de aplicaciones industriales con microcontroladores PIC- (en preparación).  
Martin Cuenca, E y Moreno Balboa, JM. (1998). -Fundamentos de electrónica moderna. Teoría y diseño de circuitos-.



«Figura 7». Esquema de la secuencia de pasos para la grabación del microcontrolador.

necesitamos es el fichero CONTAR.HEX.

Llegados a este punto es el momento de la grabación del microcontrolador. Conecta el programador a tu ordenador, ejecuta el programa MultiPIC y a continuación graba en el microcontrolador el programa CONTAR.HEX.

Ahora retira el microcontrolador del grabador e insértalo correctamente en el Módulo-01. A continuación conecta el Módulo-01 con el Módulo de aprendizaje tal como se muestra en las «Foto 3». En caso de no disponer del mismo, realiza ahora las conexiones que se te han indicado en la «Figura 4». Conecta la alimentación y verás cómo tu primer programa comienza a funcionar.

**Dr. Eugenio Martín Cuenca**  
(emartin@goliat.ugr.es)  
**Ing. José María Moreno Balboa**