

Prácticas y Aplicaciones con Microcontroladores/ 2: los PIC16F84 y PIC16C71

DR. EUGENIO MARTÍN CUENCA / ING. JOSÉ M^º MORENO BALBOA
Facultad de Ciencias. Universidad de Granada.

Continuando con la exposición teórica del primer artículo del Curso (Radorama, Noviembre de 1997), se describen a continuación los diferentes tipos de circuitos de reloj.

El PIC16F84 puede trabajar con uno de los cuatro siguientes tipos de osciladores (ver figura 1):

- RC, Oscilador Resistencia/Capacidad. Puede funcionar hasta 4 MHz, pero con una estabilidad de frecuencia menor.
- XT, Oscilador Estándar de cristal de cuarzo, desde una frecuencia de 100 kHz hasta un máximo de 4 MHz.
- HS, Alta velocidad, también con

crystal de cuarzo, hasta una frecuencia de 10 MHz o 20 MHz en el PIC16C71.

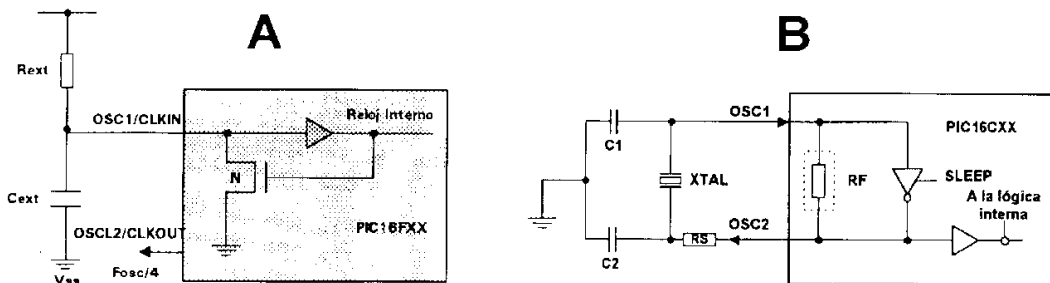
- LP, con cristal de cuarzo desde 32 a 200 kHz. Esto le proporciona la característica de muy bajo consumo.

En aplicaciones donde no es necesaria una temporización de gran exactitud, puede ser conveniente emplear el tipo RC al ser el más económico. La frecuencia en este caso depende de los valores de R, C y de la temperatura de funcionamiento. Los valores de R deben estar comprendidos entre 5 y 100 k, mientras que los valores de C no deben ser inferiores a 20 pF. Como se indica en la figura 1-A, el circuito

RC se conecta a la patita OSC1, mientras que en la patita OSC2 se dispone de FOSC/4.

Con los modos XT, HS y LP (figura 1-B) se conecta un cristal de cuarzo o un resonador cerámico a las patitas OSC1/CLKIN y OSC2/CLKOUT.

El tipo de oscilador empleado debe indicarse durante la grabación del microcontrolador en las opciones de configuración del programa PIC84.EXE. En ese momento se produce la programación de dos bits de configuración denominados FOSC1 y FOSC2. Todas las prácticas se han realizado con un reloj de 4 MHz, es decir tipo XT.



C	R	Frecuencia (5-25°C)	Tolerancia
20 pF	3.3 K	4.68 MHz	27 %
	5.1 K	3.94 MHz	25 %
	10 K	2.34 MHz	29 %
	100 K	250.16 KHz	33 %
100 pF	3.3 K	1.49 MHz	25 %
	5.1 K	1.12 MHz	25 %
	10 K	820.31 KHz	30 %
	100 K	90.25 KHz	25 %
300 pF	3.3 K	524.24 KHz	28 %
	5.1 K	415.52 KHz	30 %
	10 K	270.33 KHz	28 %
	100 K	25.37 KHz	25 %

Tipo de oscilador	Frecuencia del cristal	Rango de C1	Rango de C2
LP	33 kHz	33-68 pF	33-68 pF
	200 kHz	15-33 pF	15-33 pF
XT	100 kHz	68-100 pF	68-100 pF
	2 MHz	10-22 pF	10-22 pF
	4 MHz	10-22 pF	10-22 pF
HS	8 MHz	22-47 pF	22-47 pF
	20 MHz	22-47 pF	22-47 pF

Fig. 1 - Diferentes formas de conexionado del oscilador. (A) Tipo RC. (B) Tipos LP, XT y HS.

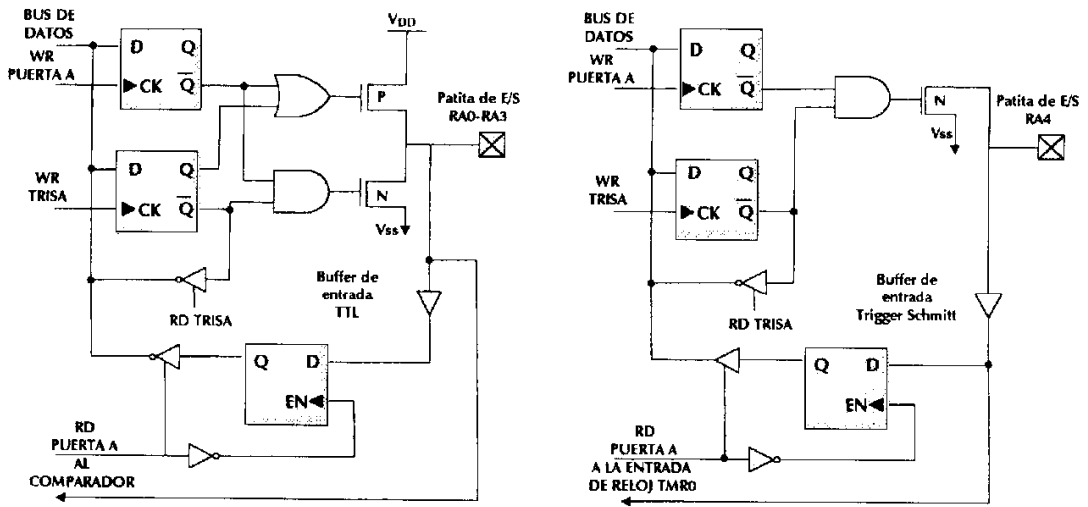


Fig. 2 - Diagrama de bloques de los pines del Puerto A. (A) a la izquierda pines RA0-RA3, (B) a la derecha pin RA4.

El PIC16F84 dispone de dos Puertos de E/S, el Puerto A de 5 bits y el Puerto B de 8 bits (ver las figuras 2 y 3 en la primera parte de esta serie de artículos). Cualquier línea de éstos puede ser programada de forma totalmente independiente tanto como entrada que como salida. Cada Puerto tiene dos registros: el de Datos denominado PORTA o PORTB, y el Registro de Control denominado TRISA o TRISB.

Los registros de Datos se pueden leer o escribir según que el Puerto correspondiente se utilice como entrada

o como salida. En los registros de Control se programa el sentido de funcionamiento de cada una de las líneas de entradas/salidas E/S.

Colocando un 0 en el correspondiente bit del registro TRISA o TRISB, la línea queda programada como salida, mientras que colocando un 1 la línea queda programada como entrada.

Cualquier línea puede funcionar como entrada o como salida. Sin embargo, si actúa como entrada la información que se introduce no se memoriza, pasando simplemente por un dis-

positivo triestado. Por ello, el valor de dicha información debe mantenerse hasta que sea leída, es decir: la lectura se realiza en tiempo real. Si la línea actúa como salida, el bit que procede del bus de datos se guarda en la báscula, con lo que la información que ofrece esta patita permanece invariable hasta que se reescriba este bit.

La máxima corriente que puede absorber y suministrar cada línea individual de los Puertos E/S es respectivamente de 25 y 20 mA. Teniendo en cuenta las limitaciones del chip, la máxi-

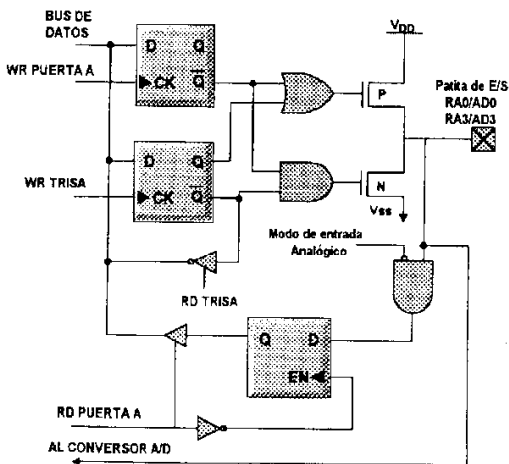
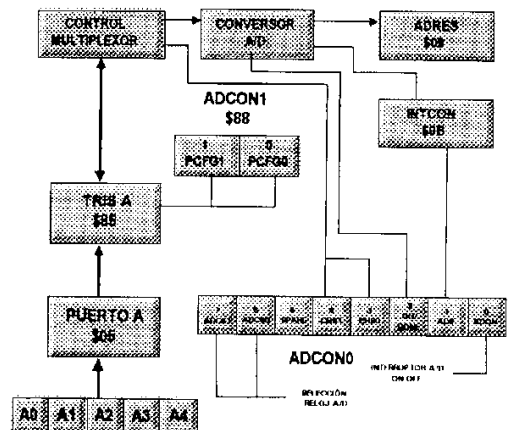


Fig. 3 - Diagrama esquemático del Puerto A en los PIC16C71.



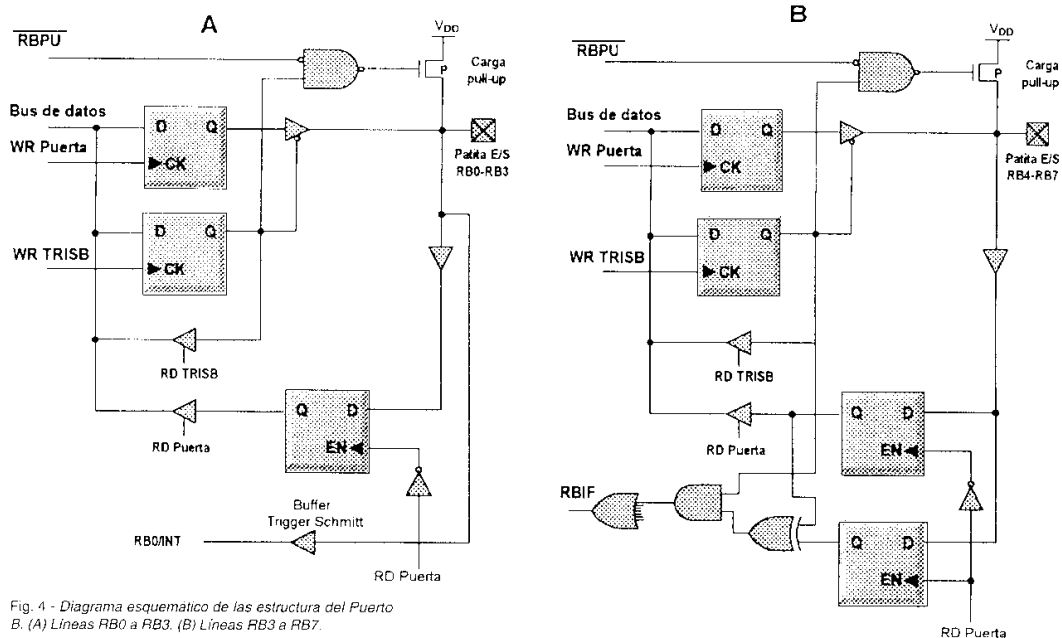


Fig. 4 - Diagrama esquemático de la estructura del Puerto B. (A) Líneas RB0 a RB3. (B) Líneas RB3 a RB7.

ma corriente que puede absorber el Puerto A es de 80 mA y suministrar 50 mA, mientras que para el Puerto B son respectivamente 150 mA y 100 mA.

Como se ha indicado, todas las patitas de E/S no empleadas deben llevarse a +5 voltios, regla de las entradas CMOS. Cuando se produce un reset, todas las líneas se programan automáticamente como entradas.

El Puerto A (fig. 2) dispone en todas las salidas de un buffer TTL de tres estados, aceptando como entradas niveles lógicos TTL. La quinta línea de este Puerto es compartida con el contador/temporizador TMR0 a través de la RA4 denominada TOCKI (Timer 0 Clock Input). Esta patita presenta una estructura de trigger Schmitt cuando se programa como entrada; cuando actúa como salida es una línea de colector abierto, por lo que debe colocarse una resistencia de polarización. Como salida, su lógica es invertida: un 0 escrito en la línea aparece como un 1, y no suministra corriente: sólo la acepta.

En el caso del PIC16C71, además las líneas RA0-RA3 son compartidas con el convertor A/D (fig. 3). Así, cuando se activa el convertor estas líneas funcionan como entradas analógicas. Este convertor es de 8 bits, por lo que

cualquier tensión presente a su entrada se convierte en una de 256 niveles. Por ejemplo, si se usa como referencia 5 V como fondo de escala, la mejor resolución es $5/256 = 19,5$ mV; una tensión de 4 V a su entrada será traducido en $(4 \times 256)/5 = 204,8$. Como se indicó en la primera parte, el convertor A/D tiene asociados los registros ADCON0, ADCON1 y ADRES, siendo en este último donde se depositan los resultados del proceso de conversión.

El Puerto B es bidireccional con un ancho de 8 bits. Las líneas RB0 a RB3 poseen la estructura mostrada en la figura 4-A, mientras que las de RB4 a RB7 aparecen en la figura 4-B. Esta diferencia se debe a que en el caso de las líneas RB4 a RB7 puede programarse una *interrupción por cambio de estado* de cualquiera de estas líneas. Cuando se programa una interrupción de este tipo, para lo cual las líneas deben estar además programadas como entradas, el hecho de que cualquier línea de RB4 a RB7 cambie de estado produce una interrupción (ver descripción del registro INTCON e Interrupciones). El valor de la patita en modo entrada es comparado con el almacenado en la báscula durante la última lectura del Puerto B. Los cambios

en las patitas se detectan realizando una operación OR para la generación de una interrupción por cambio de estado. Esta interrupción por cambio de estado puede «despertar - wake up» al μ C del modo de reposo SLEEP. El usuario debe borrar la interrupción en la Rutina de Servicio de Interrupción RSI de una de las siguientes maneras: a) borrando la bandera bit 3 de INTCON (RBIF); b) leyendo o escribiendo el PORTB y luego borrando el bit RBIF. Esto finaliza la condición mismacht y permite que se borre RBIF.

Una condición mismacht puede producir que el bit RBIF siga en 1. Leyendo el Puerto B puede finalizar la condición de mismacht y permitir que el bit RBIF sea borrado, es decir puesto a 0.

Todas las líneas del Puerto B disponen de resistencias internas de polarización programables. Cada una de las 8 resistencias puede ser activada o deshabilitada por medio del bit RBPU del registro especial OPTION, y se deshabilitan automáticamente si una línea es programada como salida, así como durante el proceso de Power on Reset.

La característica interrupción por cambio de estado, junto con las resistencias de polarización programables de estas cuatro patitas, permiten co-

```

.....
; Programa CONTAR.ASM      Fecha : 1 - Septiembre - 97
; Este programa cuenta en binario de 0 a 255 sacando el resultado a
; los LEDs de la Puerta B
; Revisión : 0.0          Programa para PIC16C84 y PIC16F84
; 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      ;El resultado se guarda en w
f EQU 1      ;El resultado se guarda en el registro
***** Igualdades de la CPU y del mapa de memoria *****
PORTA EQU 05h ; Puerta A
PORTB EQU 06h ; Puerta B
TRISA EQU 05h ; Registro Triestado Puerta A
TRISB EQU 06h ; Registro Triestado Puerta B
STATUS EQU 03h ; Registro Status
RP0 EQU 05h ; Bit RP0 del registro de STATUS
Contador1 EQU 0Ch ; Registro utilizado en el retardo
Contador2 EQU 0Dh ; Registro utilizado en el retardo
.....
ORG 00h ; Dirección del vector de Reset
GOTO Inicializa ; Comienza el programa después
; del vector de interrupción
Inicializa
ORG 05h ; Una posición detrás vector de Int.
BSF STATUS,RP0 ; Selecciona la página 1 de la memoria
CLRF TRISB ; Coloca la Puerta B como salida
; borrando TRIS B
BCF STATUS,RP0 ; Vuelve a la página 0
CLRF PORTB ; Apaga los LEDs borrando la Puerta B
Bucle
INCF PORTB,f ; Suma 1 registro de la Puerta B
CLRF Contador1 ; Borra contenido Contador1
MOVLW 0E0h ; Carga el acumulador con el valor 0E0H
MOVWF Contador2 ; Contenido W se guarda en Contador2
Bucle1
DECFSZ Contador1,f ; Resta 1 al contenido de Contador1
; Si Contador1 llega a cero: salta
; la instrucción GOTO Bucle1.
; Si Contador1 no llega a cero: ejecuta
; la instrucción GOTO Bucle1
GOTO Bucle1 ; Cierra el primer bucle de retardo
DECFSZ Contador2,f ; Lo mismo que el caso anterior, pero
; esta vez aplicado a Contador2
GOTO Bucle1 ; Cierra el segundo bucle de retardo
GOTO Bucle ; y otra vez, y otra vez ..
Terminado
SLEEP ; detiene la ejecución
END

```

nectar fácilmente un teclado matricial, permitiendo además que el microcontrolador «despierte» wake-up cuando se pulsa una tecla. Si se ha programado este tipo de interrupción, no es recomendable la realización de lecturas del Puerto B.

Por último, hay que indicar que la línea RB0 es la única compartida con la entrada de interrupción INT.

El Módulo Básico

Una vez que se dispone de las herramientas necesarias, como el programa compilador-ensamblador MPASM .EXE para ensamblar los programas, así como del programador PIC84/71, se puede comenzar con los diseños.

En primer lugar se propone el Módulo básico o comodín, a emplear en todas las experiencias. Contiene solo y exclusivamente el PIC16F84 o PIC16C71, así como la circuitería necesaria para su alimentación, partiendo de una entrada en alterna de 9-12 VAC o desde una pila de 9 V. Incluye además un conector de 17 pines donde están disponibles todas las líneas útiles del microcontrolador así como las señales de masa, +5 V y la tensión rectificificada y filtrada del transformador para futuros usos. También está dotado de una salida serie basada en un IC MAX-232 para la comunicación serie RS-232 con un ordenador personal.

El esquema eléctrico de este módulo se presenta en la figura 5 y la disposición de los componentes en la figura 6. Con sólo la sustitución del cristal de cuarzo puede hacerse funcionar el μ C

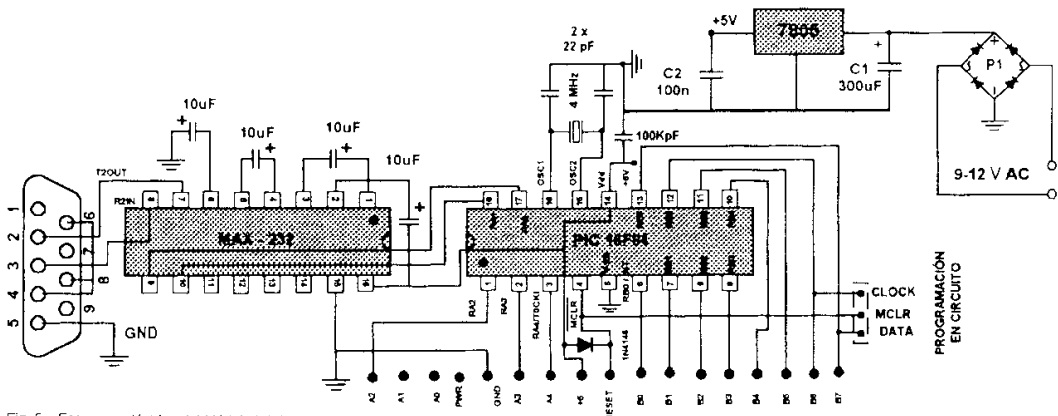


Fig. 5 - Esquema eléctrico del Módulo básico basado en un PIC-16F84 funcionando a 4 MHz y un IC MAX-232.

RAF

INSTRUCCIONES QUE MANEJAN REGISTROS

NEMÓNICO Y OPERANDOS	DESCRIPCIÓN	CICLOS	CÓDIGO OP 14 BITS				SEÑALIZADORES
			MSB		LSB		
ADDWF f,d	Suma W y f	1	00	0111	dfff	fff	C, DC, Z
ANDWF f,d	AND W con f	1	00	0101	dfff	fff	Z
CLRF f	Borra f	1	00	0001	1fff	fff	Z
CLRWF ---	Borra w	1	00	0001	0xxx	xxxx	Z
COMF f,d	Complementa f	1	00	1001	dfff	fff	Z
DECf f,d	Decrementa f	1	00	0011	dfff	fff	Z
DECFSZ f,d	Decrementa f, si es 0 salta	1 (2)	00	1011	dfff	fff	---
INCF f,d	Incrementa f	1	00	1010	dfff	fff	Z
INCFSZ f,d	Incrementa f, si es 0 salta	1	00	1111	dfff	fff	---
IORWF f,d	OR entre W y f	1	00	0100	dfff	fff	Z
MOVF f,d	Mueve f	1	00	1000	dfff	fff	Z
MOVWF f	Mueve W a f	1	00	0000	1fff	fff	---
NOP ---	No opera	1	00	0000	0xxx0	0000	---
RLF f,d	Rota f a la izqda. a través del acarreo	1	00	1101	dfff	fff	C
RRF f,d	Rota f a la dcha. a través del acarreo	1	00	1100	dfff	fff	C
SUBWF f,d	Resta a f w/ reg. W	1	00	0010	dfff	fff	C, DC, Z
SWAPF f,d	Intercambia f	1	00	1110	dfff	fff	---
XORWF f,d	XOR de W con f	1	00	0110	dfff	fff	Z

INSTRUCCIONES QUE MANIPULAN BITS

BCF f,b	Borra bit de f	1	01	00bb	bfff	fff	---
BSF f,b	Pone a 1 el bit de f	1	01	01bb	bfff	fff	---
BTFSC f,b	Testea un bit de f y salta si vale 0	1 (2)	01	10bb	bfff	fff	---
BTFSS f,b	Testea un bit de f y salta si vale 1	1 (2)	01	11bb	bfff	fff	---

INSTRUCCIONES DE CONTROL Y DE OPERANDOS INMEDIATOS

ADDLW k	Suma inmediata a W	1	11	111x	kkkk	kkkk	C, DC, Z
ANDLW k	AND inmediato con W	1	11	1001	kkkk	kkkk	Z
CALL k	Llamada a subrutina	2	10	0kkk	kkkk	kkkk	---
CLRWDI k	Borra el Perro guardián	1	00	0000	0110	0100	TO, PD
GOTO k	Salto incondicional	2	10	1kkk	kkkk	kkkk	---
IORLW k	OR inmediato con W	1	11	1000	kkkk	kkkk	Z
MOVLW k	Mueve a W un valor inmediato	1	11	00xx	kkkk	kkkk	---
RETFIE ---	Retorno desde interrupción	2	00	0000	0000	1001	---
RETLW k	Retorno y carga de W	2	11	01xx	kkkk	kkkk	---
RETURN ---	Retorno de subrutina	2	00	0000	0000	1000	---
SLEEP ---	Pasa a estado de reposo	1	00	0000	0110	0011	TO, PD
SUBLW k	Resta W de un inmediato	1	11	110x	kkkk	kkkk	C, DC, Z
XORLW k	OR Exclusiva a W	1	11	1010	kkkk	kkkk	Z

Tabla 1 - Repertorio de instrucciones de la gama media PIC.

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.

Completado el montaje del Módulo

básico, se pasa a la realización de la primera experiencia. Para ello, utilizando una placa protoboard, se conectarán las líneas RB0 - RB7 del módulo con una serie de resistencias de 330 ohmios además de un diodo LED rojo, uniendo estos últimos a masa (fig. 7).

La experiencia va a consistir en la escritura y posterior grabación del microcontrolador PIC16F84 con el programa CONTAR.ASM, cuyo listado se presenta en la página anterior. Su finalidad consiste en contar en binario a través del Puerto B (patitas RB0 - RB7),

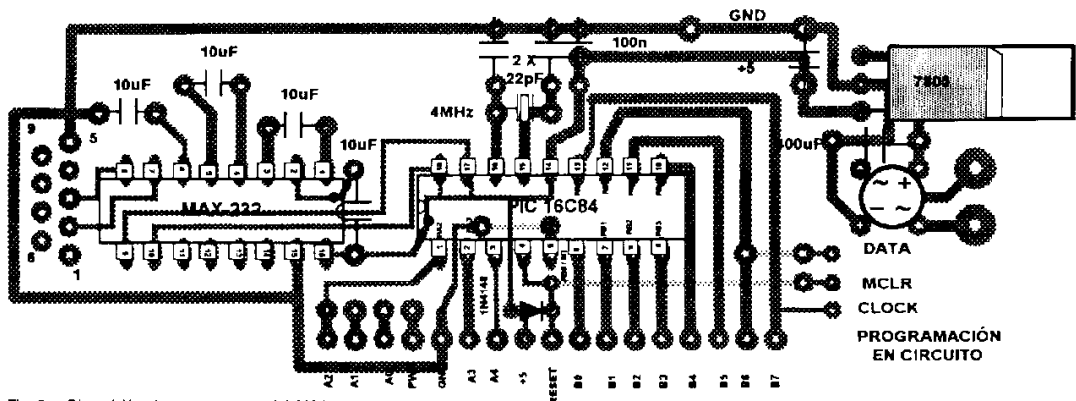


Fig. 6 - Disposición de componentes del Módulo básico.

iluminando los correspondientes diodos LEDs según el número de cuenta. Los pasos a seguir están esquematizados en la figura 8.

I.- Consiste en la escritura del programa CONTAR.ASM, por medio de un editor de textos ASCII como el EDIT del DOS o cualquier otro. Una vez escrito el programa, debe salvarse al disco en formato ASCII.

II.- El segundo paso consiste en ensamblar este fichero (CONTAR.ASM) empleando el compilador-ensamblador MPASM, como se indicó en el primer

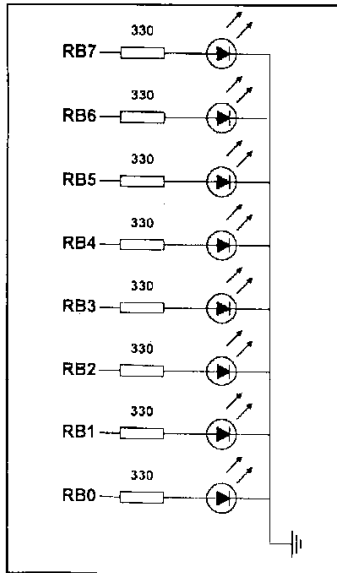


Fig. 7 - Montaje necesario para la realización de la primera experiencia.

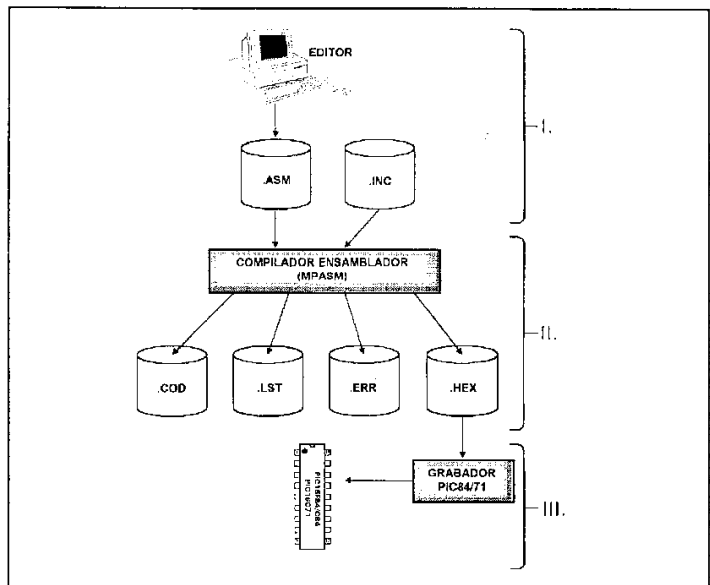


Fig. 8 - Esquema de la secuencia de pasos para la grabación del microcontrolador.

artículo de esta serie, con lo que se obtendrá el fichero CONTAR.HEX.

III.- Llegado este punto es el momento de la grabación del μ C. Conecte el programador a su ordenador, ejecute el programa PIC84/71, y a continuación grabe el microcontrolador con el programa CONTAR.HEX.

Retire el μ C del programador e insértelo correctamente en el Módulo básico, realizando las conexiones indicadas en la figura 6. Conecte la alimentación y su primer programa comienza

a funcionar. En la Tabla 1 se detallan todas la instrucciones del μ C y la descripción de su funcionamiento.

Para que el lector pueda comprender el funcionamiento del programa, se ha decidido recurrir a explicar detalladamente cada uno de los programas en su listado empleando gran profusión de comentarios en los mismos.

Bibliografía

J. M^o. Angulo, E. Martín Cuenca, I. Angulo (1997). «Microcontroladores PIC. La solución en un Chip». Paraninfo-ITP.

RA6