

Temporización

Ejemplo:

Calcular el tiempo de la siguiente rutina. Suponer $f_{xtal} = 11.0592 \text{ MHz}$

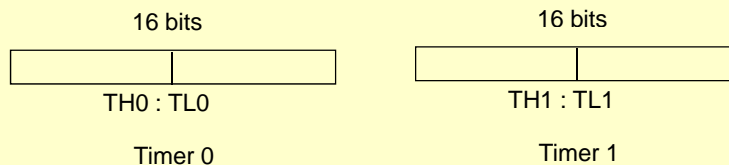
DELAY:	MOV R3,#250	;	1 ciclo m
ACA:	NOP	;	1 ciclo m
	NOP	;	1 ciclo m
	NOP	;	1 ciclo m
	NOP	;	1 ciclo m
	DJNZ R3,ACA	;	2 ciclo m
	RET	;	1 ciclo m

Solución:

$$250 \times (1+1+1+1+2) + 2 \times 1.085 \text{ us} = 1627.5 \text{ us}$$

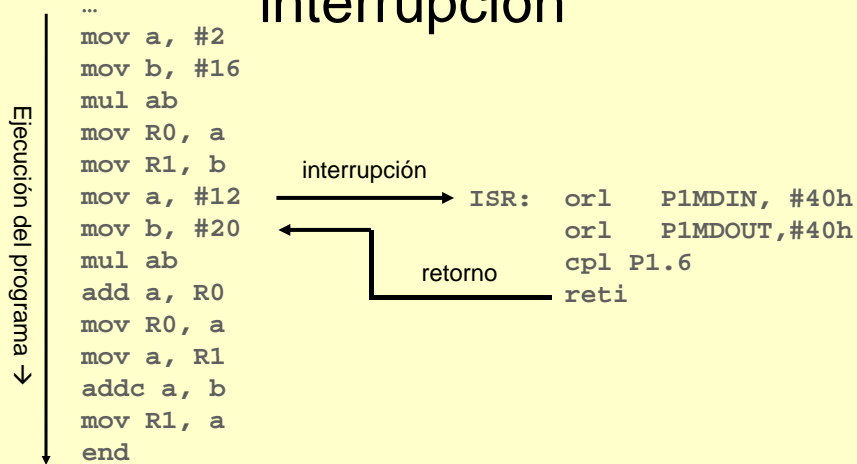
Internal Timers

El 8051 tiene 2 temporizadores de 16 bits



- Los temporizadores se incrementan cada ciclo de reloj del sistema (NO habitualmente igual al de cristal)
- Los registros de los temporizadores (TH0, TL0, TH1, TL1) pueden ser leídos o escritos
- Los desbordes (hacia arriba) de los temporizadores producen cambios en bits de los SFRs asociados que pueden ser empleados para generar interrupciones

Anticipo del concepto de interrupción



Microcontroladores - Parte 2

3

Anticipo del concepto de interrupción

El 8051 tiene cinco fuentes de interrupción

- Timer 1 overflow
- Timer 2 overflow
- Interrupción externa 0
- Interrupción externa 1
- Eventos de la comunicación serie (buffer full, buffer vacío, etc)

Las interrupciones se habilitan y deshabilitan por medio de los correspondientes SFRs

Microcontroladores - Parte 2

4

Interrupciones

SOURCE	POLLING PRIORITY	REQUEST BITS	HARDWARE CLEAR?	VECTOR ADDRESS
X0	1	IE0	N (L) ¹ Y (T) ²	03H
T0	2	TP0	Y	0BH
X1	3	IE1	N (L) Y (T)	13H
T1	4	TF1	Y	1BH
PCA	5	CF, CCFn n = 0-4	N	33H
SP	6	RI, TI	N	23H
T2	7	TF2, EXF2	N	2BH

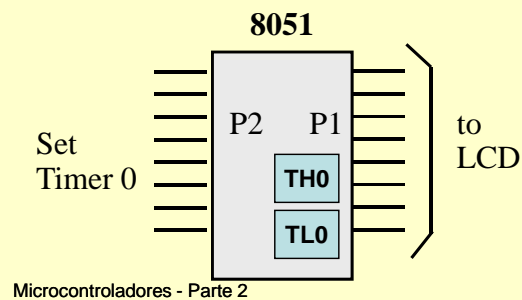
NOTES:
1. L = Level activated
2. T = Transition activated

Programación de Timers /Counters

- el 8051 tiene 2 timers/counters: timer/counter 0 y timer/counter 1. Ellos pueden utilizarse:
 1. el **timer** puede utilizarse como un generador de demoras.
 - La fuente de reloj es la frecuencia **interna** de reloj, derivada del cristal del 8051.
 2. Como **contador** de eventos
 - **Entrada externa** de una pata de entrada y el contada por registros.
 - Pueden ser el número de personas ingresando a un estadio, la rotación de un eje, o cualquier evento convertido en pulsos.

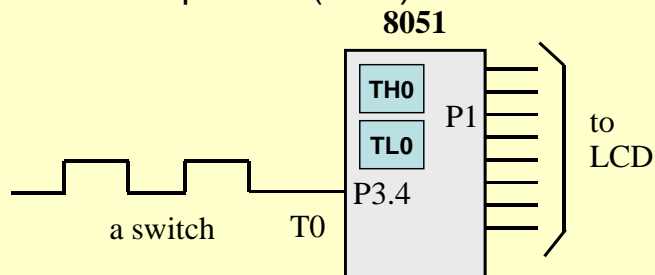
Timer

- Inicializar los registros del timer
- Arrancar el timer y el 8051 cuenta.
- Utiliza con señal de conteo el reloj interno (ciclos de máquina)
- Cuando los registros valen 0 el 8051 setea un bit para marcar que transcurrió el tiempo



Contador

- Cuenta el número de eventos
 - Muestra el número de eventos en los registros
 - Entrada externa para T0 (P3.4)
 - Entrada externa para T1(P3.5)



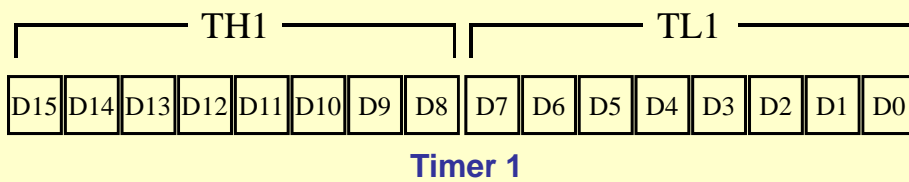
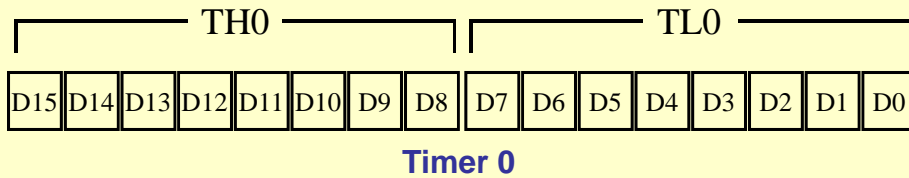
Registros Utilizados

- TH0, TL0, TH1, TL1
- TMOD (Timer mode register)
- TCON (Timer control register)
- Para el 8052 que tiene 3 timers/counters, aparecerán otros registros.
 - T2CON (Timer 2 control register)
 - TH2 y TL2 sólo utilizados en 8052.

Registros básicos de los timers

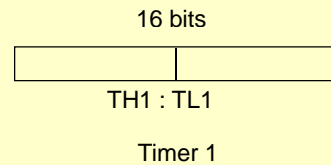
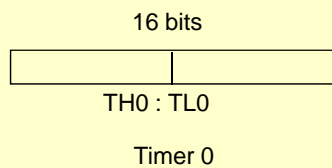
- Timer 0 y timer 1 son registros de 16 bits.
 - Los registros almacenan:
 - Demora como timer
 - Número de eventos como contador
 - Timer 0: **TH0 & TL0**
 - Timer 0 high byte, timer 0 low byte
 - Timer 1: **TH1 & TL1**
 - Timer 1 high byte, timer 1 low byte
 - Se acceden por separado como registros de 8 bits: low byte y high byte.

Registros



Internal Timers

Original 8051 has 2 timers



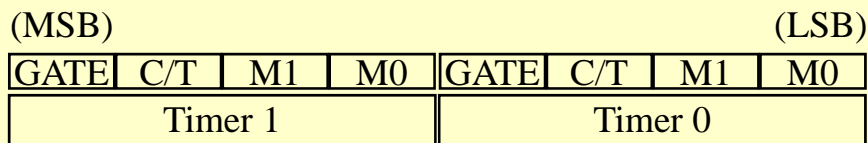
Timers increment on each system clock
Timer registers (TH0, TL0, TH1, TL1) can be read or written to
Timer overflow can cause "interrupts" or set SFR bits high

Registro TMOD

- Timer mode register: **TMOD**

`MOV TMOD,#21H`

- Registro de 8 bits
- Define el uso de los dos timers
 - Utiliza los 4 bits menores para el Timer 0 (Poner a 0000 si no se usa)
 - Utiliza los 4 bits mayores para el Timer 1 (Poner a 0000 si no se usa)
- No bit-addressable



Microcontroladores - Parte 2

13

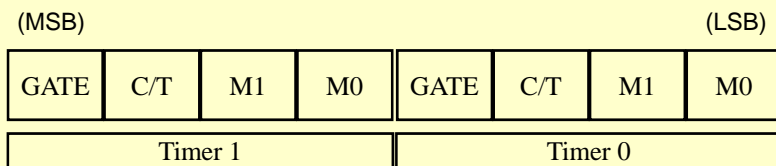
Registro TMOD

GATE Control de disparo. Se habilita Timer/counte solo cuando la pata INTx = 1 bit de control TRx = 1. Si vale 0 el timer esta habilitado sin importar .

C/T Timer o counter. = 0 para timer (entrada del reloj interno). =1 para contador (entrada de la pata Tx).

M1 Bit de modo 1

M0 Bit de modo 0



Microcontroladores - Parte 2

14

M1, M0

- **M0 y M1 seleccionan el modo de trabajo de los temporizadores 0 y 1.**

M1	M0	Modo	Operación
0	0	0	Modo Timer de 13-bits 8-bit THx + 5-bit TLx (x= 0 or 1)
0	1	1	Modo Timer de 16-bits 8-bit THx + 8-bit TLx
1	0	2	8-bit autorrecarga (Timer/counter); THx contiene el valor que se recarga en TLx cada vez que desborda.
1	1	3	Modo de Timer partido

Microcontroladores - Parte 2

15

TCON (1/2)

- **Timer control register: TCON**
 - Nibbles superior para timer/counter, nibble bajo para interrupciones
- **TR (run control bit)**
 - TR0 para Timer/counter 0; TR1 para Timer/counter 1.
 - TR = 1 por programa para habilitar/deshabilitar timer/counter.

(MSB)	– TR=0: off (stop)	(LSB)					
	– TR=1: on (start)						
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
Timer 1		Timer0		for Interrupt			

Microcontroladores - Parte 2

16

TCON (2/2)

- **TF (timer flag, control flag)**

- TF0 para timer/counter 0; TF1 para timer/counter 1.
- TF es como un carry. Originalmente, TF=0. Cuando TH-TL pasa de FFFFH a 0000, el TF = 1.
 - TF=0 : no llegó
 - TF=1: llegó
 - Si habilitamos interrupciones, TF=1 disparará ISR.

(MSB)							(LSB)
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
Timer 1		Timer0		Para Interrupciones			

Microcontroladores - Parte 2

17

Gate

- Cada timer tiene una forma de start y stop.
 - GATE=0
 - Control **Interno**
 - el arranque y parada del timer se controlan por **software**.
 - Set/clear el flag TR para arrancar/parar el timer.
 - GATE=1
 - Control **Externo**.
 - El uso de hardware para arrancar y parar el timer por **software y por una fuente externa**.
 - El Timer/counter es habilitado solo cuando la pata INT esta alta y el flag TR = 1.

Microcontroladores - Parte 2

18

Timer - Modo 1

- Utilizaremos como ejemplo el timer 0.
- Timer **16-bits** (TH0 y TL0)
- TH0-TL0 se incrementan continuamente cuando TR0 = 1. El 8051 deja de incrementar TH0-TL0 cuando TR0 = 0.
- El timer trabaja con el reloj interno. O sea cuenta en cada ciclo de máquina.
- Cuando el timer (TH0-TL0) alcanza su máximo de FFFFH, pasa a 0000, y TF0 = 1.
- El programa debería verificar TF0 y detener el timer 0.

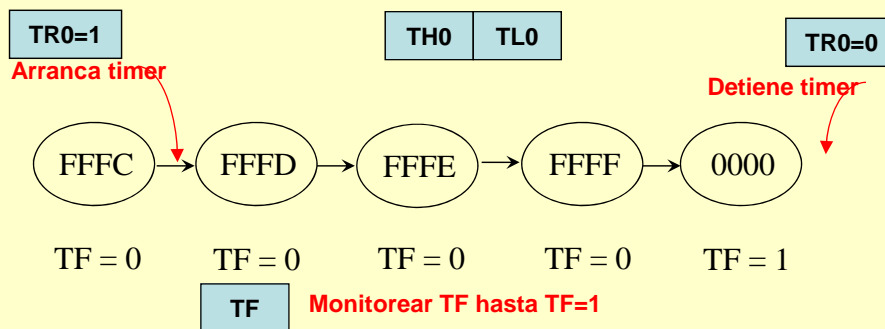
Pasos para Modo 1 (1/3)

1. Elegir modo 1 timer 0
 - `MOV TMOD,#01H`
2. Inicializar TH0 y TL0.
 - `MOV TH0,#FFH`
 - `MOV TL0,#FCH`
3. Por las dudas limpiar TF0: TF0=0.
 - `CLR TF0`
4. Arrancar el timer.
 - `SETB TR0`

Pasos para Modo 1 (2/3)

5. El 8051 inicia el conteo incrementando el par TH0-TL0.

• TH0-TL0= FFFCH, FFFDH, FFFEh, FFFFH, 0000H



Microcontroladores - Parte 2

23

Pasos para Modo 1 (3/3)

6. Cuando TH0-TL0 **pasa de FFFFH a 0000**, el 8051 coloca TF0=1.

- TH0-TL0= FFFEh, FFFFH, 0000H (Con TF0=1)

7. Monitorear el timer flag (TF) para esperar que valga 1.

- Pepe: JNB TF0, Pepe

8. TR0 = 0 para detener el proceso.

- CLR TR0

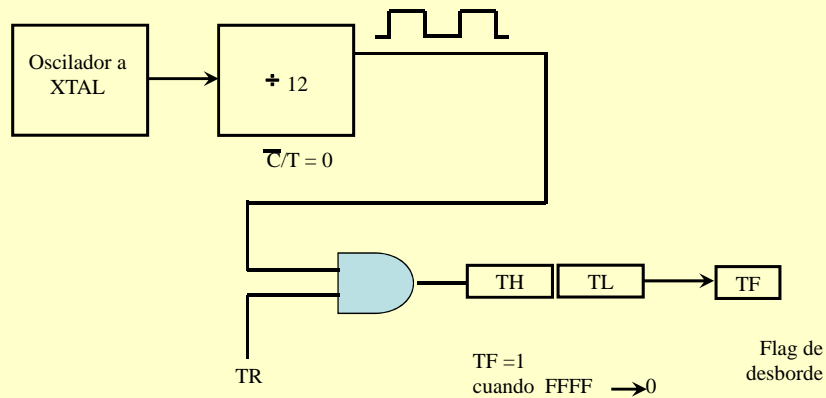
9. TF = 0 para la próxima operación.

- CLR TF0

Microcontroladores - Parte 2

24

Programación del Modo 1



Microcontroladores - Parte 2

25

Cálculo de la demora con XTAL = 11.0592 MHz

(a) en hex

$(FFFF - YYXX + 1) \times 1.085 \mu s$ donde YYXX son los valores iniciales de TH, TL respectivamente.

Nótese que los valores de YYXX son hex.

(b) en decimal

Convertir los valores YYXX de TH, TL a decimal para obtener un número decimal NNNNN

$(65536 - NNNNN) \times 1.085 \mu s$

Microcontroladores - Parte 2

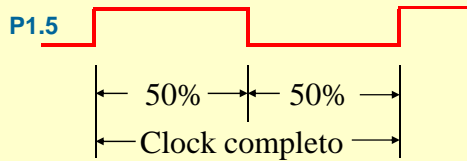
26

Ejemplo 2 (1/2)

Crear una señal rectangular con un ciclo de actividad del 50% en el bit P1.5. Utilizar el Timer 0 para generar los tiempos.

;Cada pasada es un semiciclo

```
MOV TMOD,#01 ;Timer 0,modo 1(16-bit)
ACA: MOV TL0,#0F2H ;Timer = FFF2H
      MOV TH0,#0FFH
      CPL P1.5
      ACALL DEMORA
      SJMP ACA
```



Microcontroladores - Parte 2

27

Ejemplo 2 (2/2)

;Generar la demora con el timer 0

DEMORA:

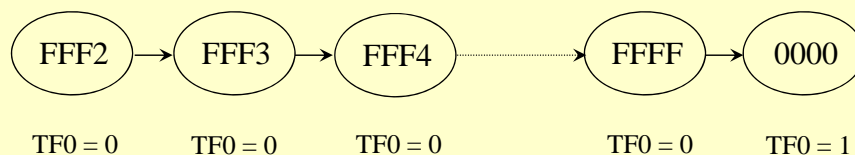
```
      SETB TR0 ;iniciar el timer 0
```

```
pepe: JNB TF0,pepe
```

```
      CLR TR0 ;detener timer 0
```

```
      CLR TF0 ;limpiar flag timer 0
```

```
      RET
```



Microcontroladores - Parte 2

28

Ejemplo 3 (1/2)

El siguiente programa genera una señal rectangular continua en P1.5 empleando el timer 1.

Calcular la frecuencia si $f_{XTAL} = 11.0592 \text{ MHz}$.

```
MOV  TMOD,#10H ;timer 1, modo 1
Recar:MOV  TL1,#34H ;timer =3476H
      MOV  TH1,#76H
      SETB TR1      ;iniciar
pepe: JNB  TF1,pepe
      CLR  TR1      ;detener
      CPL  P1.5     ;siguiente semiperíod
      CLR  TF1     ;limpiar TF1
      SJMP Recar   ;recargar timer1
```

Microcontroladores - Parte 2

29

Ejemplo 3 (2/2)

Solución:

En Modo 1 se debe recargar TH1, TL1 en cada desborde si deseamos señal continua.

$FFFFH - 7634H + 1 = 89CCH = 35276$ cuentas

Medio período = $35276 \times 1.085 \mu s = 38.274 \text{ ms}$

Todo el período = $2 \times 38.274 \text{ ms} = 76.548 \text{ ms}$

Frecuencia = $1 / 76.548 \text{ ms} = 13.064 \text{ Hz}$.

Notar que ambos semiperíodos son iguales.

Microcontroladores - Parte 2

30

Calcular los valores del timer

- Supongamos conocer la demora y $f_{XTAL} = 11.0592$ MHz .
- ¿Cómo se calculan los valores de TH, TL?
 1. Dividimos la demora deseada por **1.085 μ s**.
 2. Calcular **65536 -n**, donde **n** es el valor decimal calculado en el paso 1.
 3. Convertir el resultado del resultado del paso 2 a hexadecimal, donde **yyxx** es el valor inicial a ser cargado en los registros.
 4. Elegir el modo del temporizador.
 5. Si fue modo 1, cargar **TH = yy** y **TL = xx**.

Ejemplo 4 (1/2)

Con f_{Xtal} 11.0592 MHz, escriba un programa para generar una señal rectangular en P2.3.

Solución:

- (a) El período de la señal será $= 1 / 50 \text{ Hz} = 20 \text{ ms}$.
- (b) El semiperíodo será 10 ms.
- (c) $10 \text{ ms} / 1.085 \mu\text{s} = 9216$
 $65536 - 9216 = 56320$ en decimal = DC00H en hex.
- (d) TL1 = 00H y TH1 = DCH.

Ejemplo 4 (2/2)

```
MOV    TMOD,#10H    ;timer 1, modo 1
lazo:  MOV    TL1,#00    ;Timer = DC00H
        MOV    TH1,#0DCH
        SETB  TR1        ;iniciar timer
pepe:  JNB    TF1,pepe
        CLR    TR1        ;detener
        CPL   P2.3
        CLR    TF1        ;clear timer flag 1
        SJMP  lazo        ;recargar timer ya
                          ;que modo 1 NO es
                          ;autorrecarga
```

Microcontroladores - Parte 2

33

Ejemplo 5

Para grandes demoras.

```
MOV    TMOD,#10H
MOV    R3,#200
LAZO  : MOV    TL1,#08
        MOV    TH1,#01
        SETB  TR1
PEPE:  JNB    TF1,PEPE
        CLR    TR1
        CLR    TF1
        DJNZ  R3,LAZO
```

Solución:

TH - TL = 0108H = 264 en decimal 65536 - 264 = 65272.

UNA demora = 65272 × 1.085 μs = 70.820 ms

Demora Total = 200 × 70.820 ms = 14.164024 segundos

Microcontroladores - Parte 2

34

Lapsos Prolongados

- Temporizadores encadenados
- Contadores en memoria

Modo 2 del Timer

- Timer de 8-bits
 - Permite cargar valores 00 a FFH en TH0.
- Auto-recarga
- TL0 se incrementa continuamente si TR0=1.

Ejemplo 6 - Modo 2 (1/2)

1. Elegimos Modo 2 timer 0
 - `MOV TMOD,#02H`
2. Cargamos el valor inicial en TH0.
 - `MOV TH0,#38H`
3. Por las dudas, TF0=0.
 - `CLR TF0`
4. Después de cargar TH0, el 8051 lo copia a TL0.
 - `TL0=TH0=38H`
5. Iniciar el timer.
 - `SETB TR0`

Microcontroladores - Parte 2

37

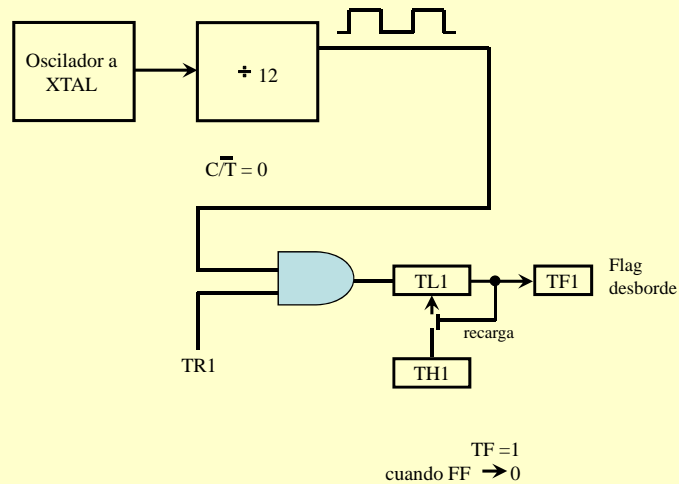
Ejemplo 6 - Modo 2 (2/2)

6. el 8051 inicia la cuenta incrementando TL0.
 - `TL0= 38H, 39H, 3AH,.....`
 7. Cuando TL0 pasa de FFH a 00, TF0=1. Además TL0 es recargado automáticamente con el valor de TH0.
 - `TL0= FEH, FFH, 00H (Con TF0=1)`
 - el 8051 auto recarga `TL0=TH0=38H`.
 - Vuelve al paso 6 (TL0 se incrementa continuamente).
- Nótese que si fuera necesario debemos limpiar TF0 cuando TL0 desborda, para poder monitorear TF0 en la próxima etapa.
 - `TR0 = 0` detiene el proceso.

Microcontroladores - Parte 2

38

Timer 1 Modo 2 con entrada externa



Microcontroladores - Parte 2

39

Ejemplo 7

Calcular la frecuencia de la señal rectangular generada en la pata P1.0.

```

MOV  TMOD,#2H  ;Timer 0,modo 2
MOV  TH0,#0
LAZO :MOV  R5,#250 ;cuanta 250 veces
      ACALL DEMORA
      CPL  P1.0
      SJMP LAZO
DEMORA:SETB TR0 ;inicio
pepe: JNB  TF0,pepe
      CLR  TR0 ;detener
      CLR  TF0 ;limpio TF
      DJNZ R5,DEMORA ;timer 2: auto-recarga
      RET

```

$T = 2 (250 \times 256 \times 1.085 \mu\text{s}) = 138.88 \text{ ms}$, y frecuencia = 72 Hz.

Microcontroladores - Parte 2

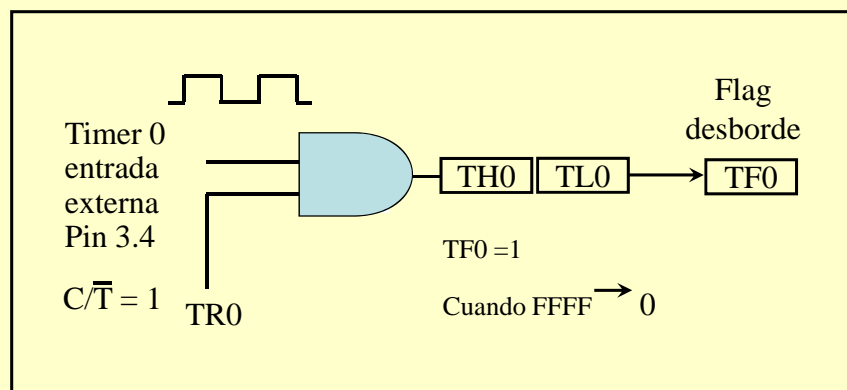
40

Patas de P3 utilizadas por los Timers 0 y 1

Pata	Pin Puerto	Función	Descripción
14	P3.4	T0	Timer/Counter 0 entrada externa
15	P3.5	T1	Timer/Counter 1 entrada externa

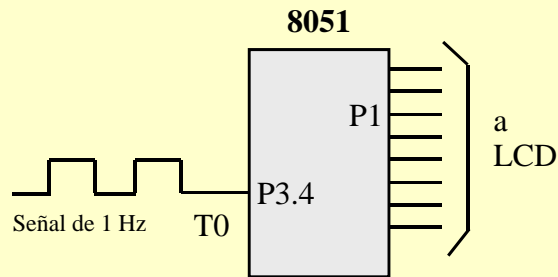
(MSB)				(LSB)			
GATE	C/T=1	M1	M0	GATE	C/T=1	M1	M0
Timer 1				Timer 0			

Timer 0 Con entrada externa (Modo 1)



Ejemplo 8 (1/3)

Supongamos conectar **un pulso de 1 Hz a P3.4**.
Escriba el programa que coloque el valor del contador 0 en un LCD. TH0 inicial = -60.



Microcontroladores - Parte 2

43

Ejemplo 8 (2/3)

```
ACALL LCD_SET_UP ;inicializar LCD
MOV  TMOD,#00000110B ;Counter 0,modo 2
MOV  TH0,#-60
SETB P3.4 ;T0 entrada
LAZO :SETB TR0 ;inicia el contador
BACK: MOV  A,TL0 ;cada 60 eventos
ACALL CONV ;conversión en R2,R3,R4
JNB  TF0,BACK ;loopea si TF0=0
CLR  TR0 ;detener
CLR  TF0
SJMP LAZO
```

Microcontroladores - Parte 2

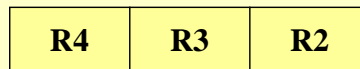
44

Ejemplo 8 (3/3)

;convertir binario 8-bit a ASCII

```

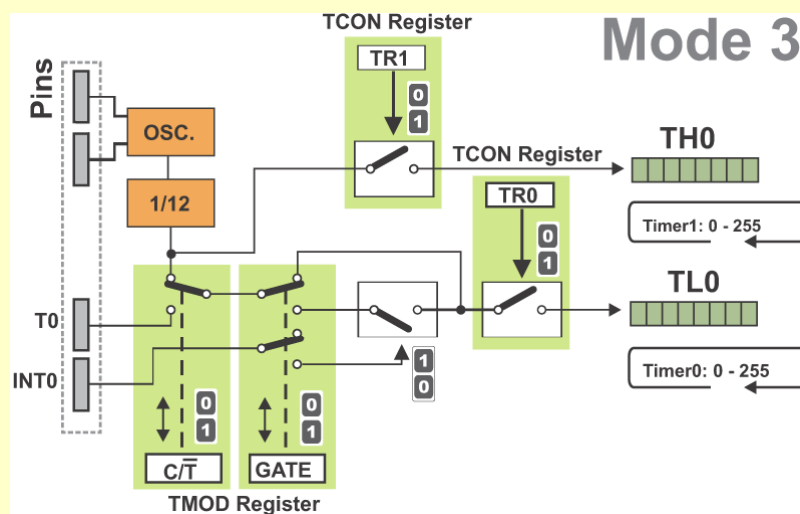
CONV: MOV  B,#10      ;divide por 10
      DIV  AB
      MOV  R2,B       ;guarda dígito bajo
      MOV  B,#10      ;divide por 10 otra vez
      DIV  AB
      ORL  A,#30H     ;convierte a ASCII
      MOV  R4,A
      MOV  A,B
      ORL  A,#30H
      MOV  R3,A
      MOV  A,R2
      ORL  A,#30H
      MOV  R2,A      ;ACALL LCD_DISPLAY
      RET
    
```



Microcontroladores - Parte 2

45

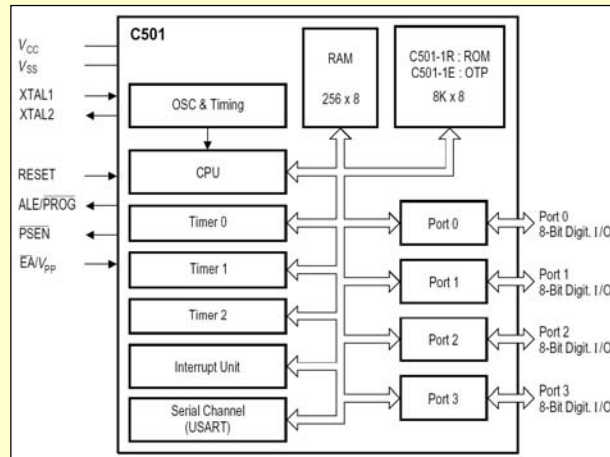
Modo 3



Microcontroladores - Parte 2

46

8x52



Microcontroladores - Parte 2

47

8052

- 256 bytes de memoria R/W
 - MOV R0,#90h
 - MOV A,@R0
- Un tercer temporizador con funciones especiales
- SFRs para este temporizador

Microcontroladores - Parte 2

48

SFRs para el 8052

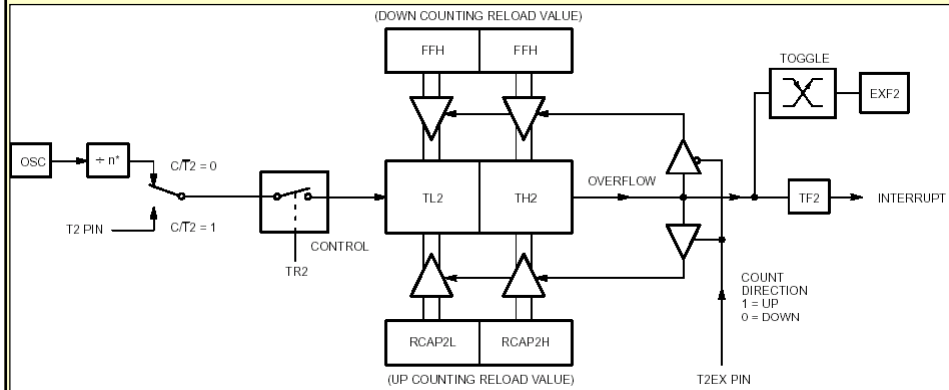
00	P0	SP	DPL	DPH				PCON	07
08	TCON	IMOD	TL0	TL1	TH0	TH1			0F
10	P1								17
18	SCON	SBUF							1F
20	P2								27
28	IE								2F
30	P3								37
38	IP								3F
40									47
48	T2CON		RCAP2L	RCAP2H	TL2	TH2			4F
50	PSW								57
58									5F
60	ACC								6F
68									77
70	P								7F
78									87

Blue background are I/O port SFRs
 Yellow background are control SFRs
 Green background are other SFRs

T2CON

Bit	Nombre	DirBIT	DESCRIPCIÓN
7	TF2	CFh	Timer 2 Overflow. Valdrá 1 en el desborde de T2. Cuyo la interrupción de T2 esta habilitada producirá una interrupción. Será activada por TCLK o RCLK.
6	EXF2	CEh	Timer 2 External Flag. Valdrá 1 por una recarga o captura causada por una transición 1-0 en T2EX (P1.1), pero solo cuyo EXEN2 vale 1. Cuyo esta habilitada la interrupción de T2, producirá la generación de interrupción..
5	RCLK	CDh	Timer 2 Receive Clock. Cuyo vale 1, el temporizador 2 será utilizado como generador de la velocidad de la comunicación serie de entrada. Si vale 0 se utilizará el temporizador 1.
4	TCLK	CCh	Timer 2 Transmit Clock. Cuyo vale 1, el temporizador 2 será utilizado como generador de la velocidad de la comunicación serie de salida. Si vale 0 se utilizará el temporizador 1.
3	EXEN2	CBh	Timer 2 External Enable. Cuyo vale 1, una transición 1-0 en T2EX (P1.1) causará una captura o recarga.
2	TR2	CAh	Timer 2 Run. Cuyo vale 1, se activará el temporizador 2. Es caso contrario estará desactivado.
1	C/T2	C9h	Timer 2 Counter/Interval Timer. Si vale 0, el temporizador 2 es un contador interno, si vale 1, se incrementará en cada transición 1-0 en T2 (P1.0).
0	CP/RL2	C8h	Timer 2 Capture/Reload. Si vale 0, se producirá una autorrecarga en el desborde del temporizador 2 o en una transición 1-0 en T2EX. Si EXEN2 es 1 una captura ocurrirá en cada transición 1-0 de T2EX.

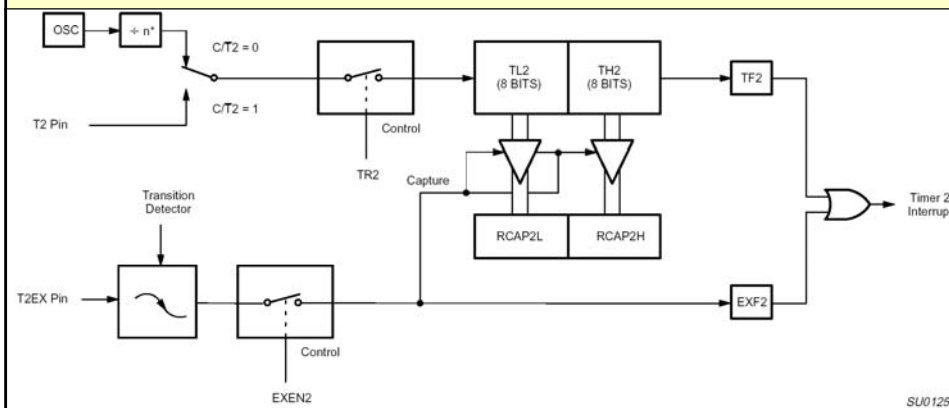
Temporizador 2 – DCEN = 1



Microcontroladores - Parte 2

51

Temporizador 2-Captura



SU01252

Microcontroladores - Parte 2

52