

CHƯƠNG 3

HỘ VI ĐIỀU KHIỂN 8051

3.4 Timer (Mạch định thời)

HOÀNG Trang
hoangtrang@hcmut.edu.vn

Reference and thank to: thầy Hồ Trung Mỹ

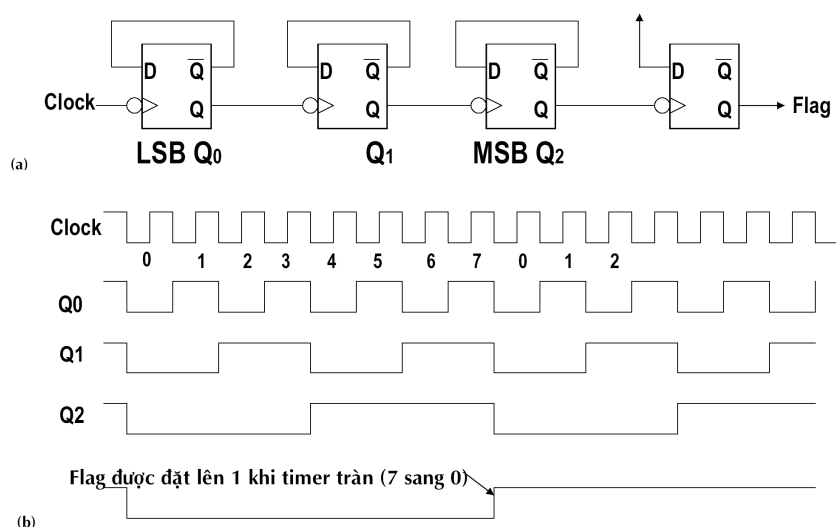
1

cuu duong than cong . com

Giới thiệu về timer

- Timer: một chuỗi các flip-flop chia đôi tần số mắc nối tiếp với nhau.
- => chúng nhận tín hiệu vào làm clk. CLK qua flip-flop thứ nhất để chia đôi tần số xung nhịp. Ngõ ra của flip-flop thứ nhất làm xung nhịp cho flip-flop thứ hai (cũng làm việc chia đôi tần số), và v.v...
- Timer có n tầng: cho xung ra có tần số là tần số xung nhịp chia cho $2^{\exp(n)}$
- Ngõ ra của tầng cuối làm xung nhịp cho flip-flop báo tràn của timer (còn gọi là cờ timer TF [Timer Flag]).
- Giá trị nhị phân trong các flip-flop của timer có thể xem như số đếm số xung nhịp (hoặc các sự kiện) từ khi timer bắt đầu chạy. Thí dụ timer 16 bit sẽ đếm lên từ 0000H đến FFFFH. Cờ báo tràn sẽ lên 1 khi số đếm tràn từ FFFFH đến 0000H.₂

TD: Timer 3 bit



Timer trong 8051

- 8051/8031 có hai timer 16 bit (T0 và T1), mỗi timer có bốn chế độ hoạt động.
- Mỗi timer 16 bit có 16 tầng hay tầng cuối cùng chia tần số xung nhịp cho $2^{16}=65536$. Nguồn xung nhịp cho các timer là xung vuông có tần số bằng 1/12 tần số xung nhịp cung cấp cho 8051.
- **Ứng dụng của timer:**
 - a) định khoảng thời gian,
 - b) đếm sự kiện
 - c) tạo tốc độ baud cho cổng nối tiếp có sẵn trong 8051/8031.

Timer trong 8051

- **a/ Định khoảng thời gian:** lập trình cho timer tràn ở một khoảng đều đặn và đặt cờ tràn timer lên 1. Cờ được dùng để đồng bộ hóa hoặc đo thời gian trôi qua giữa hai sự kiện (ví dụ: đo độ rộng xung).
- **b/ Đếm sự kiện:** để xác định số lần xảy ra của một sự kiện hơn là đo khoảng thời gian trôi qua giữa các sự kiện. Một “sự kiện” là bất cứ tác động bên ngoài nào có thể cung cấp một chuyển tiếp 1 sang 0 từ chân T0, T1(ở P3) của 8051/8031.
- **c/ Các timer cũng có thể cung cấp xung nhịp tốc độ baud** cho cổng nối tiếp có sẵn trong 8051/8031.

5

cuu duong than cong . com

Các thanh ghi timer

- Timer (T0/T1) của 8051 được tạo bởi 2 thanh ghi 8 bit.
- Timer 0 (T0) được tạo bởi
 - TL0 và TH0 (L=Low và H=High)
 - Truy cập chúng như các thanh ghi khác, TD:


```
MOV TL0, #55H
MOV R1, TH0
```
- Timer 1 (T1) được tạo bởi
 - TL1 and TH1

6

Các SFR của timer

Timer SFR	Mục đích	Địa chỉ	Địa chỉ bit
TCON	Điều khiển	88H	Có
TMOD	Chế độ (hoạt động)	89H	Không
TL0	Byte thấp của Timer 0	8AH	Không
TL1	Byte thấp của Timer 1	8BH	Không
TH0	Byte cao của Timer 0	8CH	Không
TH1	Byte cao của Timer 1	8DH	Không
T2CON*	Điều khiển Timer 2	C8H	Có
RCAP2L*	Bắt byte thấp của Timer 2	CAH	Không
RCAP2H*	Bắt byte cao của Timer 2	CBH	Không
TL2*	Byte thấp của Timer 2	CCH	Không
TH2*	Byte cao của Timer 2	CDH	Không

* Với 8032/8052.

7

Thanh ghi chế độ timer (TMOD)

cho Timer 1

Bảng 3.17 Tóm tắt thanh ghi TMOD.

Bit	Tên	Timer	Mô tả
7	GATE	1	Bit mở cổng. Khi bit này 1, timer chỉ chạy trong khi $\overline{INT1}$ ở mức cao. GATE=0: Timer hoạt động bình thường
6	C/ $\overline{T}$	1	Bit chọn counter (bộ đếm) hay timer 1 = bộ đếm sự kiện Timer nhận xung clk từ bên ngoài 0 = timer khoảng thời gian Timer nhận clk từ động nội
5	M1	1	Bit 1 của chọn chế độ (Xem slide kế)
4	M0	1	Bit 0 của chọn chế độ (Xem slide kế)
3	GATE	0	Bit mở cổng cho timer 0.
2	C/ $\overline{T}$	0	Bit chọn counter (bộ đếm) hay timer của timer 0
1	M1	0	Bit 1 của chọn chế độ của timer 0
0	M0	0	Bit 0 của chọn chế độ của timer 0

cho Timer 0

8

Các chế độ hoạt động của timer

Bảng 3.18 Các chế độ hoạt động của timer.

M1	M0	Chế độ	Mô tả
0	0	0	Chế độ timer 13 bit (chế độ 8048)
0	1	1	Chế độ timer 16 bit
1	0	2	Chế độ timer 8 bit tự nạp lại giá trị đầu
1	1	3	Chế độ tách timer

ít sử dụng
hiện nay

5 bit thấp của TLx + 8 bit THx

Timer 0: TL0 là timer 8 bit được điều khiển bằng các bit chế độ của timer 0; TH0 là timer 8 bit được điều khiển bằng các bit chế độ của timer 1.

Timer 1: bị dừng lại

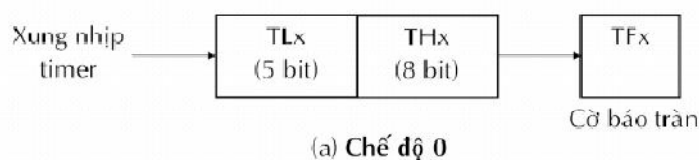
9

Thanh ghi điều khiển timer (TCON)

Bảng 3.19 Tóm tắt thanh ghi TCON.

Bit	Ký hiệu	Địa chỉ bit	Mô tả
TCON.7	TF1	8FH	Cờ báo tràn timer 1. Đặt lên 1 bởi phần cứng khi tràn; được xóa về 0 bởi phần mềm hoặc phần cứng khi bộ xử lý chỉ đến chương trình phục vụ ngắt.
TCON.6	TR1	8EH	Bit điều khiển Timer 1 chạy. Đặt/xóa bằng phần mềm để cho timer chạy/ngừng.
TCON.5	TF0	8DH	Cờ báo tràn Timer 0.
TCON.4	TR0	8CH	Bit điều khiển Timer 0 chạy.
TCON.3	IE1	8BH	Cờ cạnh ngắt 1 bên ngoài. Đặt bởi phần cứng khi phát hiện có cạnh xuống ở INT1; xóa bằng phần mềm hoặc bằng phần cứng khi CPU chỉ đến chương trình phục vụ ngắt.
TCON.2	IT1	8AH	Cờ kiểu ngắt 1 bên ngoài. Đặt/xóa bằng phần mềm để ngắt ngoài tích cực cạnh xuống/mức thấp.
TCON.1	IE0	89H	Cờ cạnh ngắt 0 bên ngoài.
TCON.0	IT0	88H	Cờ kiểu ngắt 0 bên ngoài.

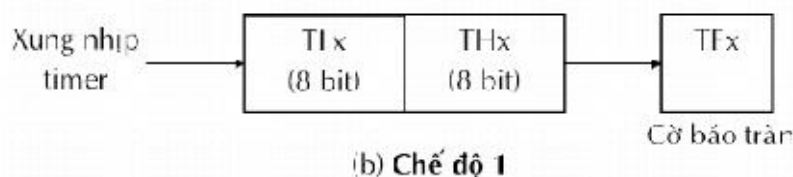
Chế độ timer 13 bit (Chế độ 0)



- Chế độ 0 là chế độ timer 13 bit để tương thích với bộ xử lý trước 8051 là 8048 (hiện nay, ít sử dụng)
- Với các thiết kế mới người ta ít dùng chế độ hoạt động này
- Byte cao của timer (THx) nối tiếp với 5 bit thấp của byte thấp của timer (TLx) để tạo thành timer 13 bit
- Ba bit cao của TLx không được sử dụng

11

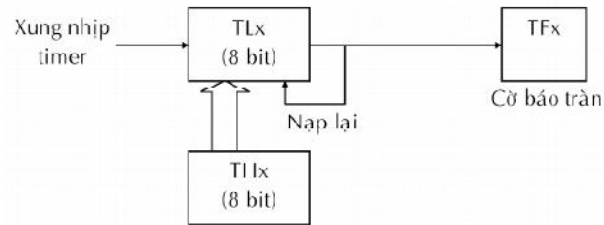
Chế độ timer 16 bit (chế độ 1)



- Chế độ 1 là chế độ timer 16 bit, sử dụng THx và TLx.
- Tràn xảy ra khi có chuyển tiếp từ FFFFH sang 0000H trong số đếm và nó đặt cờ báo tràn timer lên 1. Timer tiếp tục đếm tiếp. Sau khi qua tràn, TFx vẫn là 1.
- Cờ báo tràn là bit TFx trong TCON mà người ta có thể đọc ra hoặc ghi vào bằng phần mềm

12

Chế độ timer 8 bit tự nạp lại trị đầu (chế độ 2)

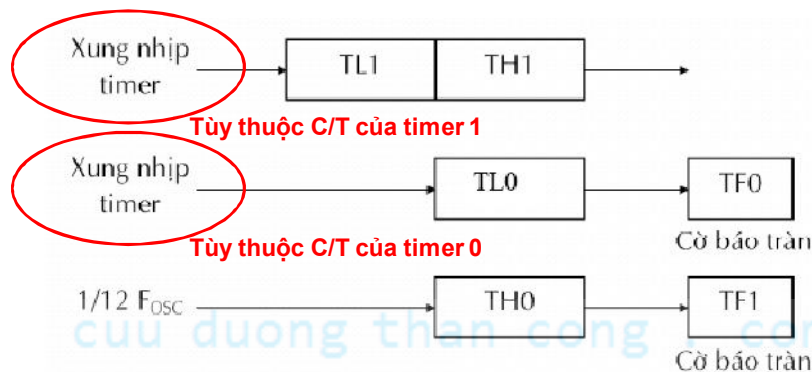


(c) Chế độ 2

- Chế độ 2 là chế độ tự động nạp giá trị đầu.
 TLx làm việc như timer 8 bit
 THx giữ giá trị cần nạp lại.
- Khi bộ đếm tràn từ FFH sang 00H thì
 + cờ timer được đặt lên 1 VÀ
 + giá trị trong THx còn được nạp vào TLx,
 việc đếm tiếp tục từ giá trị này đến chuyển tiếp từ FFH sang 00H kế, và cứ tiếp tục như vậy.
- Chế độ này tiện lợi vì tràn timer xảy ra theo những khoảng thời gian có chu kỳ một khi TMOD và THx đã được khởi tạo trị

13

Chế độ tách timer (chế độ 3)



- Chế độ 3 là chế độ tách timer thì khác nhau với mỗi timer. Timer 0 ở chế độ 3 được tách thành 2 timer 8 bit.
- TL0 và TH0 làm việc như 2 timer độc lập với các báo tràn đặt các cờ TF0 và TF1 tương ứng

14

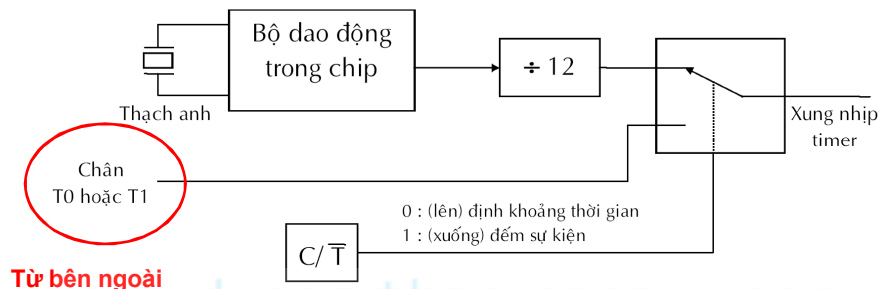
Đặc điểm của chế độ 3

- Timer 1 bị dừng ở chế độ 3, nhưng có thể cho nó chạy bằng cách chuyển sang 1 trong các chế độ khác.
- => Giới hạn duy nhất: cờ báo tràn thông thường của Timer 1 (TF1) không bị ảnh hưởng bởi sự báo tràn trong Timer1 vì TF1 đã được nối vào TH0. **(không có cờ báo tràn TF1)**
- Chế độ 3 chủ yếu cho thêm 1 timer 8 bit (8051 giống như có thêm timer thứ ba). Khi Timer 0 ở chế độ 3, ta có thể bật hay tắt Timer 1 bằng cách chuyển nó ra khỏi hay đi vào chính chế độ 3 của nó. Nó vẫn có thể được công nối tiếp sử dụng làm bộ tạo tốc độ baud hoặc có thể được sử dụng theo bất cứ cách nào mà không cần ngắt **(vì nó không còn TF1)**

15

cuu duong than cong . com

Các nguồn tạo xung nhịp



Từ bên ngoài

- Có 2 nguồn xung nhịp, mà ta có thể chọn bằng cách ghi vào bit chọn bộ đếm/timer trong TMOD khi tạo các trị khởi động timer.
- Một nguồn xung nhịp được dùng để định thì khoảng thời gian, còn nguồn kia để đếm sự kiện.

16

Định thì khoảng thời gian

- Nếu $C/\overline{T} = 0$, hoạt động timer liên tục được chọn và timer được cấp xung nhịp từ mạch dao động trên chip. Một tầng chia 12 được thêm vào để giảm tần số xung nhịp xuống giá trị thích hợp cho phần lớn các ứng dụng trong điều khiển.
- Khi chọn hoạt động timer liên tục thì timer được sử dụng để **định thì khoảng thời gian** (interval timing). Các thanh ghi timer (TLx/THx) tăng nội dung thêm 1 cứ theo tần số là 1/12 tần số dao động trên chip; như vậy với thạch anh 12 MHz thì xung nhịp của timer là 1 MHz.
- Tràn timer xảy ra cứ sau một số các xung nhịp cố định mà tùy theo giá trị đầu được nạp vào các thanh ghi TLx/THx

17

cuu duong than cong . com

Đếm sự kiện

- Nếu $C/\overline{T} = 1$, timer được cấp xung nhịp từ nguồn bên ngoài. Trong phần lớn các ứng dụng, nguồn bên ngoài này cung cấp timer một xung khi xảy ra sự kiện—timer được sử dụng để **đếm sự kiện** (event counting). Số sự kiện được xác định trong phần mềm bằng cách đọc các thanh ghi TLx/THx, từ đó giá trị 16 bit trong các thanh ghi này tăng thêm 1 cho mỗi sự kiện.
- Nguồn xung nhịp bên ngoài được cung cấp bằng cách đưa vào các có chức năng thay thế ở Port 3: ngõ vào xung nhịp cho Timer 0 là T0 (chân P3.4). hay cho timer là T1 (chân P3.5)
- *Trong các ứng dụng bộ đếm, các thanh ghi timer được tăng thêm 1 khi có chuyển tiếp 1 sang 0 ở ngõ vào bên ngoài Tx. Ngõ vào bên ngoài này được lấy mẫu trong S5P2 của mọi chu kỳ máy; như vậy khi ngõ vào ở mức 1 trong 1 chu kỳ và mức 0 trong chu kỳ kế thì số đếm được tăng thêm 1. Giá trị mới xuất hiện trong các thanh ghi timer trong S3P1 của chu kỳ theo sau chu kỳ mà chuyển tiếp được phát hiện. Từ đó nó mất 2 chu kỳ máy (2 μ s) để ghi nhận chuyển tiếp 1 sang 0, do đó tần số bên ngoài tối đa là 500 KHz (giả sử hoạt động 12 MHz)*

18

Khởi tạo và truy cập các thanh ghi timer

Các tác vụ:

- Đặt chế độ làm việc
- Cho timer chạy
- Dừng timer
- Kiểm tra cờ tràn
- Xóa cờ báo tràn
- Đọc và cập nhật các thanh ghi timer

19

cuu duong than cong . com

Đặt chế độ làm việc – Cho timer chạy – Dừng timer

TD:

- **Đặt chế độ làm việc**

```
MOV  TMOD, #00010000B
```

timer 1: Gate=0, C/T=0, M1M0=01 (mode 1)

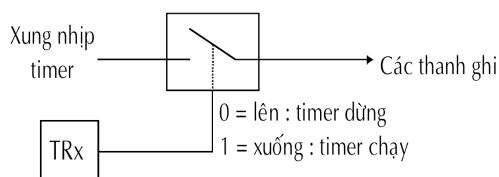
timer 0: Gate=0, C/T=0, M1M0=00 (mode 0)

- **Cho timer chạy**

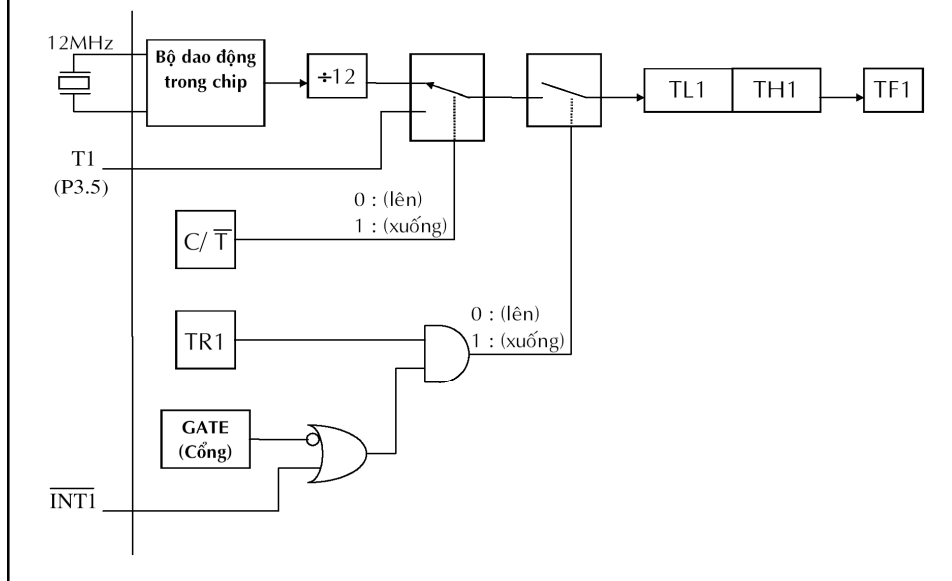
```
SETB TR1
```

- **Dừng timer**

```
CLR  TR1
```



Chú ý khi có sử dụng ngắt ngoài (ví dụ: chế độ 1)



Kiểm tra cờ và truy cập các thanh ghi timer

TD:

- **Kiểm tra cờ và xóa cờ**

WAIT: JNB TF1, WAIT

CLR TR1 ; dừng timer T1

CLR TF1 ; xóa cờ báo tràn

- **Cập nhật các thanh ghi timer**

MOV TL1, #9CH

MOV TH1, #0FFH

Đọc giá trị đếm của timer đang chạy (on the fly)

- $R7 \leftarrow TH1, R6 \leftarrow TL1$
- Trước hết đọc byte cao
- Vấn đề có thể xảy ra:
khi tràn byte thấp $\rightarrow$ đọc lại lần nữa

```
AGAIN: MOV  A, TH1
        MOV  R6, TL1
        CJNE A, TH1, AGAIN
        MOV  R7, A
```

23

Các bước để lập trình Timer mode 0 hay 1

1. Nạp trị cho TMOD
2. Nạp trị cho các thanh ghi TL và TH
3. Cho timer chạy (SETB TR0 hay SETB TR1)
4. Kiểm tra/theo dõi TF $\rightarrow$ TF = 1 (tràn)
5. Dừng timer (CLR TR0 hay CLR TR1)
6. Xóa TF
7. Quay về bước 2

24

Các bước để lập trình Timer mode 2

1. Nạp trị cho TMOD
2. Nạp trị cho các thanh ghi TH
3. Cho timer chạy (SETB TR0 hay SETB TR1)
4. Kiểm tra/theo dõi TF $\rightarrow$ TF = 1 (tràn)
5. Dừng timer (CLR TR0 hay CLR TR1)
6. Xóa TF
7. Quay về bước 4

25

cuu duong than cong . com

Định thì khoảng thời gian ngắn và khoảng thời gian dài

Bảng 3.20 Các kỹ thuật lập trình khoảng thời gian (hoạt động 12 MHz)

Khoảng thời gian tối đa (theo μ s)	Kỹ thuật
≈ 10	Điều chỉnh phần mềm
256	Timer 8 bit với tự động nạp lại trị đầu
65536	Timer 16 bit
Không giới hạn	Timer 16 bit cộng với các vòng lặp phần mềm

26

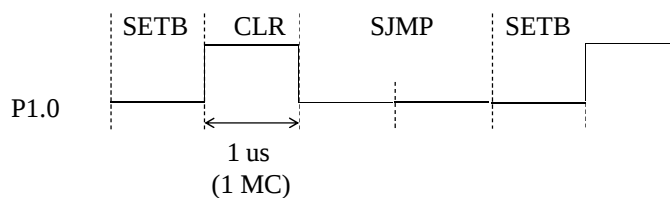
Thí dụ: Tạo dạng xung trên chân P1.0

Viết một chương trình tạo ra dạng sóng có chu kỳ trên P1.0 với tần số cao nhất có thể được. Tần số và chu kỳ nhiệm vụ của dạng sóng này là bao nhiêu?

(giả sử ta dùng XTAL = 12MHz $\rightarrow$ 1 MC = 1us)

$\rightarrow$ Với dạng sóng rất ngắn: không cần timer

```
ORG 8000H
LOOP: SETB P1.0 ; 1 MC
      CLR  P1.0 ; 1 MC
      SJMP LOOP ; 2 MC
```

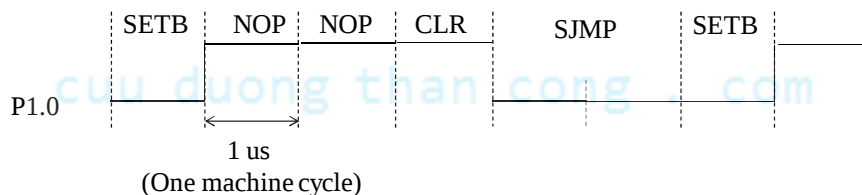


- Tần số = 250 KHz (1/ 4 us)
- $T_{ON} = 1 \text{ us}$, $T_{OFF} = 3 \text{ us}$ $\rightarrow$ duty cycle = $T_{ON}/(T_{ON}+T_{OFF}) = 25\%$ (1/4)

27

Thí dụ: Tạo sóng vuông trên chân P1.0

```
ORG 8000H
LOOP: SETB P1.0 ; 1 MC
      NOP      ; 1 MC
      NOP      ; 1 MC
      CLR  P1.0 ; 1 MC
      SJMP LOOP ; 2 MC
```



Tần số = 166.7 KHz (1/ 6 us)

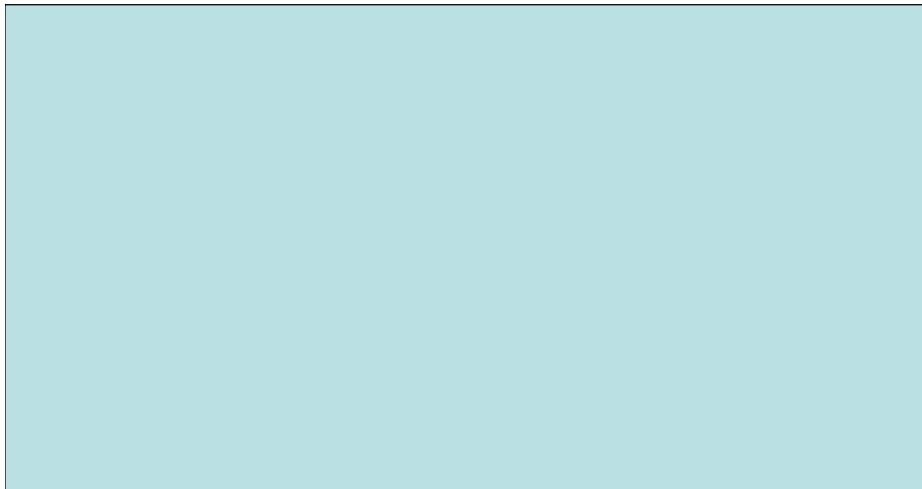
$T_{ON} = 3 \text{ us}$, $T_{OFF} = 3 \text{ us}$ $\rightarrow$ duty cycle = $T_{ON}/(T_{ON}+T_{OFF}) = 50\%$ (3/6)

28

Tạo sóng vuông 10 KHz ở chân P1.0

Tạo sóng vuông 10 KHz

Tần số = 10KHz $\rightarrow$ chu kỳ $T = 1/10000 = 100 \text{ us}$



cuu duong than cong . com

Tạo sóng vuông 1 KHz trên chân P1.0

Tạo sóng vuông 1 KHz

Tần số = 1KHz $\rightarrow$ chu kỳ $T = 1/1000 = 1000 \text{ us}$

TON = 500 us, TOFF= 500us

Dùng mode 1 (16 bit mode), vì khoảng thời gian > 256 us.

```

                                ORG    8000H
                                MOV     TMOD, #01H    ; chế độ 1 (16 bit)
LOOP:                          MOV     TH0, #0FEH
                                MOV     TL0, #0CH      ; -500 (-01F4H)
                                SETB    TR0            ; cho timer T0 chạy
WAIT:                          JNB     TF0, WAIT      ; đợi timer tràn
                                CLR      TR0           ; dừng timer 0
                                CLR      TF0          ; xóa cờ báo tràn
                                CPL      P1.0         ; đảo bit cổng
                                SJMP     LOOP          ; lặp lại
                                END

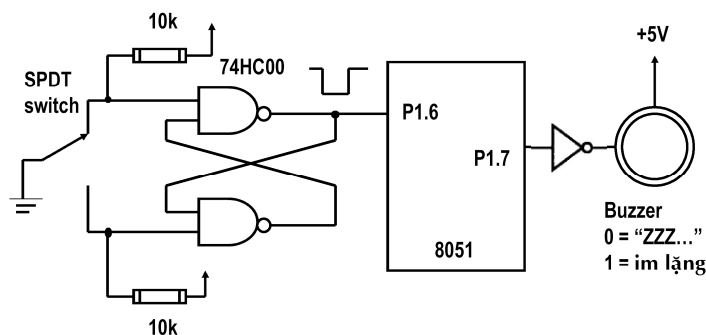
```

Chú ý: `MOV TH0, #0FEH` $\equiv$ `MOV TH0, #HIGH(-500)`
`MOV TL0, #0CH` $\equiv$ `MOV TL0, #LOW(-500)`

30

TD: Giao tiếp buzzer (1/2)

Một buzzer được nối vào chân P1.7 và một công tắc không nảy (debounce switch) được nối vào chân P1.6 (xem hình 3.28). Viết chương trình đọc mức logic do công tắc cấp và tạo ra âm thanh ở buzzer trong 1 giây sau mỗi lần phát hiện chuyển trạng thái từ 1 xuống 0.



31

TD: Giao tiếp buzzer (2/2)

Địa chỉ	Mã máy	Dòng lệnh	Nhân	Lệnh	Chú thích
		6	HUNDRED	EQU 100	; 100 x
		7	COUNT	EQU 10000	; 10000μs = 1s
8100		8		ORG 8100H	
8100	758901	7		MOV TMOD, #01H	; chế độ 1
8103	3096FD	9	LOOP:	JNB P1.6, LOOP	; đợi mức 1
8106	758A0C	10	WAIT:	JB P1.6, WAIT	; đợi mức 0
8109	D297	11		SETB P1.7	; bật buzzer ON
810B	128112	12		CALL DELAY	; đợi 1 giây
810E	C297	14		CLR P1.7	; tắt buzzer
8110	80F1	15		SJMP LOOP	
		16			
8112	7F64	17	DELAY:	MOV R7, #HUNDRED	
8114	758C27	18	AGAIN:	MOV TH0, #HIGH(-COUNT)	
8117	758A10	19		MOV TL0, #LOW(-COUNT)	
811A	D28C	20		SETB TR0	
811C	308DFD	21	WAIT2:	JNB TF0, WAIT2	
811F	C28D	22		CLR TF0	
8121	C28C	23		CLR TR0	
8123	DFEF	24		DJNZ R7, AGAIN	
8125	22	25		RET	
		26		END	

Lập trình counter

- Ở chế độ counter: TMOD, TH, TL giống như với timer
- Các chế độ timer cũng giống
- Tuy nhiên thay vì dùng tần số thạch anh, counter đếm xung từ bên ngoài đưa vào 8051.
- Bit C/T trong TMOD quyết định nguồn xung nhịp.
- Khi C/T = 1, counter đếm lên khi có xung đưa vào các chân T1 và T0 (ở P3)

Pin	Port Pin	Function	Description
14	P3.4	T0	Timer/Counter 0 external input
15	P3.5	T1	Timer/Counter 1 external input
(MSB) (LSB)			
GATE	C/T	M1	M0
Timer 1		Timer 0	

33

TD: lập trình counter ở chế độ 2

- Giả sử có xung nhịp đưa vào chân T1, viết chương trình dùng counter 1 ở chế độ 2 và xuất số xung đếm được ra P2.
- **Bài giải**

```

MOV    TMOD, #01100000B ; counter 1 chế độ 2
MOV    TH1, #0
SETB   P3.5      ; làm cho T1 là chân nhập
AGAIN: SETB   TR1    ; cho counter 1 chạy
BACK:  MOV    A, TL1  ; lấy số xung đếm được
        MOV    P2, A   ; xuất ra P2
        JNB    TF1, BACK
        CLR    TR1
        CLR    TF1
        SJMP   AGAIN

```

34