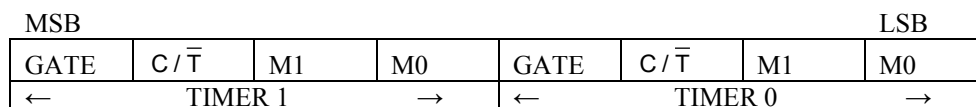


Timere / Numărătoare

Circuitele 80C51 au două Timere / Numărătoare: Timerul 0 și Timerul 1. Ambele pot fi configurate pentru a lucra fie ca timere, fie ca numărătoare de evenimente.



GATE Control poartă când este setat. Timerul / numărător "x" este activat numai când pinul \overline{INTx} este HIGH și bitul de control "TRx" este setat. Când este șters, timerul "x" este activat dacă se setează bitul de control "TRx".

C / \bar{T} Selector timer sau numărător șters pentru funcționare ca timer (intrare de la sistemul de clock intern). Setat pentru funcționare ca numărător (intrare pe pinul "Tx").

M1	M0	FUNȚIONARE
0	0	Timerul 8048 "TLx" servește pentru prescalare pe 5 biți
0	1	Timerul / numărător de 16 biți "THx" și "TLx" sunt în cascadă; nu există prescalare
1	0	Timerul / numărător "THx" de 8 biți cu autoreîncărcare păstrează o valoare ce trebuie reîncărcată în "TLx" ori de câte ori el depășește
1	1	(Timer 0) TL0 este un timer / numărător de 8 biți controlat de către biții de control ai timerului 0 standard. TH0 este un timer de 8 biți controlat numai de biții de control ai timerului 1.
1	1	(Timer 1) timerul / numărător 1 este oprit

Registrul de control mod Timer / Numărător (TMOD)

În funcția "Timer", registrul este incrementat la fiecare ciclu mașină. Astfel, s-ar putea crede că funcționează ca numărător de cicluri mașină. Întrucât un ciclu mașină constă din 12 perioade de oscilator, rata de numărare este 1 / 12 din frecvența oscilatorului.

În funcția "numărător", registrul este incrementat ca răspuns al unei tranziții 1 în 0 la pinul său de intrare corespunzător T0 sau T1. În această funcționare, intrările externe sunt eșantionate pe durata S5P2 a fiecărui ciclu mașină.

Când eșantionarea găsește un nivel "HIGH" într-un ciclu și "LOW" în ciclul următor, numărătorul este incrementat. Noua valoare a numărului apare în registru pe durata S3P1 a ciclului care urmează celui în care a fost detectată tranziția. Întrucât ia 2 cicluri mașină (24 perioade de oscilator) pentru a recunoaște o tranziție 1 la 0, viteza maximă de numărare este 1 / 24 din frecvența oscilatorului. Nu sunt restricții în privința factorului de umplere al semnalului de intrare extern, decât cel de a asigura că un nivel dat este eșantionat cel puțin o dată înainte ca el să se schimbe. El trebuie să se mențină cel puțin pentru un ciclu complet. Suplimentar selecției "Timer" sau "Numărător", Timerul 0 și Timerul 1 au patru moduri de funcționare ce pot fi selectate.

Timerul 0 și Timerul 1

Funcția "Timer" sau "Numărător" poate fi selectată de către biții de control C / \bar{T} în registrul de funcție specială TMOD. Aceste două Timere / Numărătoare au patru moduri de funcționare, ce sunt selectate de perechea de biți (M1, M0) din TMOD. Modurile 0, 1 și 2 sunt aceleași pentru ambele circuite Timer / Numărător. Modul 3 este diferit. Cele patru moduri de funcționare sunt descrise în cele ce urmează.

Modul 0

Programarea modului 0 a oricărui Timer va determina ca acesta să semene cu un Timer 8048 care este un numărător de 8 biți cu un circuit de *prescalare* divizor prin 32. În Fig. 1 se prezintă modul de funcționare atunci când acesta se aplică numărătorului 1.

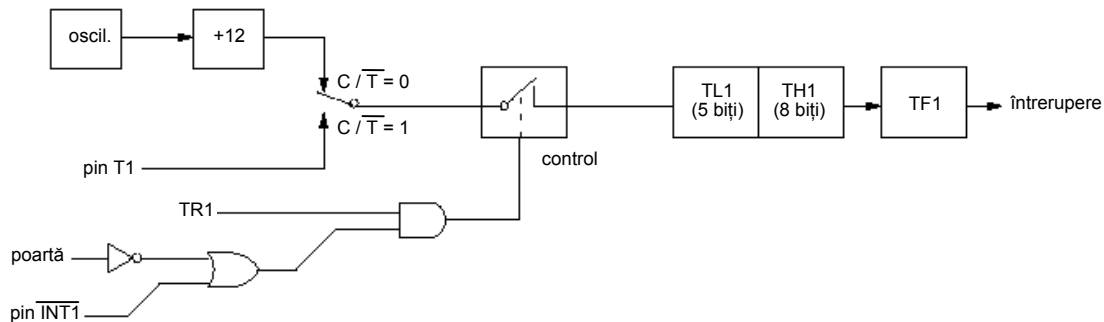


Fig. 1 Timer / Numărător mod 0: numărător de 13 biți

MSB				LSB			
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT	SIMBOL	FUNCȚIE
TCON.7	TF1	Flag depășire Timer 1. Setat hardware pe depășire Timer / Numărător. Șters hardware când procesorul vectorizează spre rutina de întrerupere sau prin ștergerea bitului prin software
TCON.6	TR1	Bit de control funcționare timer 1. Setat / Șters prin software pentru a comuta timerul / numărător on / off.
TCON.5	TF0	Flag depășire timer 0. Setat hardware pe depășire timer / numărător. Șters hardware când procesorul vectorizează spre rutina de întrerupere sau prin ștergerea bitului prin software
TCON.4	TR0	Bit de control funcționare timer 0. Setat / Șters prin software pentru a comuta timerul / numărător on / off.
TCON.3	IE1	Flag front întrerupere tip 1. Setat hardware când se detectează frontul unei întreruperi externe. Șters când s-a procesat întreruperea.
TCON.2	IT1	Bit control întrerupere tip 1. Setat / șters software pentru a preciza front căzător / nivel LOW pentru delanșarea întreruperilor externe.
TCON.1	IE0	Flag front întrerupere tip 0. Setat hardware când se detectează frontul unei întreruperi externe. Șters când s-a procesat întreruperea.
TCON.0	IT0	Bit control întrerupere tip 0. Setat / șters software pentru a preciza front căzător / nivel LOW pentru delanșarea întreruperilor externe.

Registrul de control timer / numărător (TCON)

În acest mod, registrul Timerului este configurat ca un registru de 13 biți. Când numărătorul comută din starea toți biții "1" în starea toți biții "0", se setează flagul de întrerupere timer TF1. Pornirea Timerului este activată când TR1 = 1 și fie GATE = 0, fie INT1 = 1 (setând GATE = 1 se permite timerului să fie controlat de către intrarea externă INT1 pentru a facilita măsurarea lățimii impulsurilor. TR1 este un bit de control din registrul de funcții speciale TCON . GATE se găsește în registrul de funcție specială TMOD.

Registrul de 13 biți constă din toți cei 8 biți din TH1 și cei 5 biți inferiori din TL1. Cei trei biți superiori din TL1 sunt nedeterminați și trebuie ignorați. Setarea flagului de funcționare (TR1) nu șterge registrul.

Modul 0 de funcționare este același pentru Timerul 0 și pentru Timerul 1. Se substituie TR0, TF0 și $\overline{INT0}$ pentru semnalele corespunzând Timerului 1. Există doi biți diferiți GATE, unul pentru Timerul 1 (TMOD.7) și unul pentru Timerul 0 (TMOD.3).

Modul 1

Modul 1 este asemănător cu modul 0, cu excepția faptului că registrul Timerului va lucra cu toți de 16 biți.

Modul 2

Modul 2 configurează registrul timer ca un numărător de 8 biți (TL1) cu **reîncărcare automată**, așa cum se prezintă în Fig. 2. Depășirea din TL1 nu numai că setează TF1, dar de asemenea **reîncarcă** TL1 cu conținutul TH1, care este presetat din software. Reîncărcarea lasă TH1 neschimbat.

Modul 2 este identic și pentru Timerul / Numărător 0.

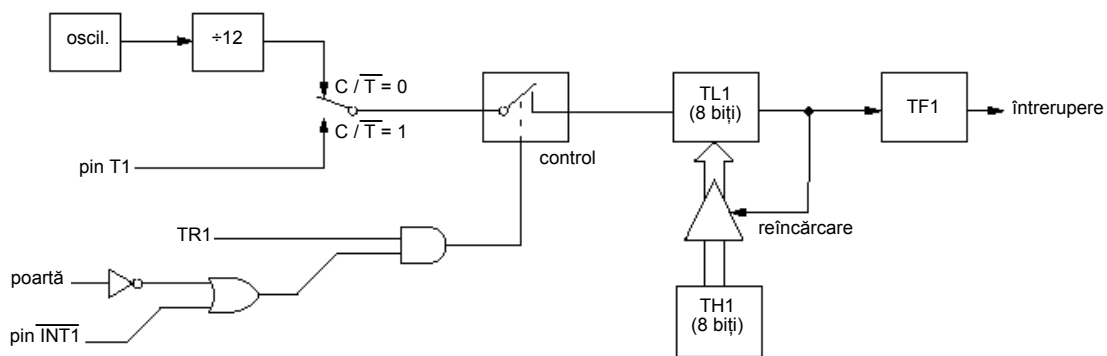


Fig. 2 Timer / Numărător mod 2: 8 biți cu auto-încărcare

Modul 3

Timerul 1 în *mod 3* numai își păstrează conținutul. Efectul este același ca la fixarea valorii TR1=0.

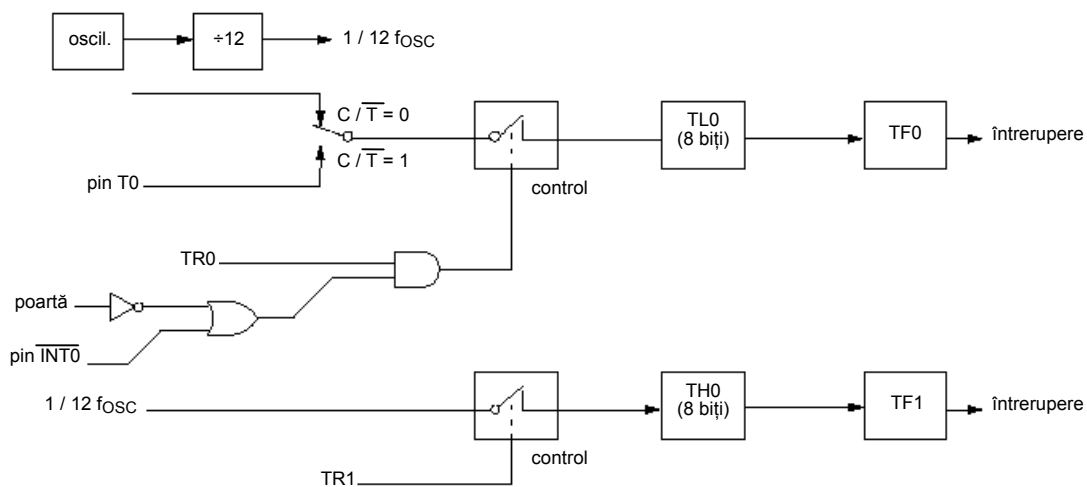


Fig. 3 Timerul / Numărător 0 în mod 3: două numărătoare de 8 biți

Timerul 0 în mod 3 stabilește TL0 și TH0 ca **două numărătoare separate**. Logica pentru modul 3 a Timerului 0 este prezentată în Fig. 3. TL0 folosește biții de control ai timerului 0: C/T, GATE, R0, INT0 și TF0. TH0 este blocat într-o funcție de timer (ce numără cicluri mașină) și preia pentru folosire TR1 și TF1 de la timerul 1. Astfel, TH0 controlează întreruperea "timer 1".

Modul 3 este prevăzut pentru aplicații ce necesită un timer suplimentar de 8 biți pe numărător. Cu Timerul 0 în mod 3, un circuit 80C51 poate fi privit ca și cum ar avea 3 timere / numărătoare. Când Timerul 0 este în mod 3, Timerul 1 poate fi cuplat și decuplat, deconectându-l și conectându-l în propriul său mod 3, sau mai poate fi folosit de către portul serial ca generator de control al vitezei de transfer, sau în orice aplicație ce nu solicită o întrerupere.

Aplicație

În cele ce urmează este prezentat un program ce simulează o sirenă de tipul celor montate pe mașinile „salvare”. Datorită faptului că frecvența semnalului audio se modifică în timp (de exp. frecvența semnalului sa crească timp de 3 secunde și apoi să scadă tot în 3 secunde), am împărțit cele 3 secunde în 100 de trepte. „Treptele” le realizez cu ajutorul Timerului T0. Dacă Timerul T0 ar fi zero (TL0=0 și TH0=0), incrementat fiind la fiecare 1us el s-ar umple (TL0=0FFH, TH0=0FFH) în 65.535 microsecunde. O astfel de treaptă este prea mare (treapta calculată este de 30ms) ($3.000.000\text{us}/100=30\text{ms}$; $65.535\text{us}\cdot 30.000=35.535\text{us}=A8\text{CF}$). Ca urmare sunt nevoit să fac un artificiu, voi încărca TH0 cu A8 și TL0 cu CF, astfel încât la incrementări de 1us, acesta s-ar „umple” mai repede (FFFF). La depășirea (umplerea numărătorului) se setează flagul TF0. Programul pornește de la o frecvență joasă ce constituie prima treaptă. În registru R2 am încărcat numărul de trepte.

```

name SIRENA_lab_6_ETC
ORG 0000H           ;baza memoriei Program
JMP START          ;salt la eticheta START

ORG 000BH           ;adresa vectorului de intrerupere Timer T0
JMP INTER          ;salt la eticheta INTER

START:              ;eticheta START
    BUZZ EQU P3.0
    MOV IE,#82H     ;activez intreruperea globala si ce-a data de Timerul T0
    MOV TMOD,#01H   ;setez Timerul T0, in Mod1
    MOV TCON,#10H   ;TR0=1, pornesc Timerul T0
    MOV R2,#100     ;incarc cele 100 de "trepte"
    MOV R3,#0       ;folositi la panta descrescatoare a sunetului
    MOV R4,#0       ;incarc constante pentru operatiuni aritmetice
    MOV R5,#4H      ;incarc constante pentru operatiuni aritmetice
    MOV R6,#3       ;incarc o valoare de start pentru generarea sunetului
    MOV R7,#238     ;incarc o valoare de start pentru generarea sunetului

MAIN:
    SETB BUZZ       ;rutina MAIN este folosita pentru bascularea pinului P3.0 = cu scop
    CALL DELAY      ;de generarea a unui sunet, de o anumita frecventa.
    CLR BUZZ
    CALL DELAY
    JMP MAIN

DELAY:
    MOV 00H,R6      ;Atentie! 00H este Adresa Registrului R0 (se poate scrie si asa)
    MOV 01H,R7      ;01H este Adresa Registrului R1
DELAY0:
    DJNZ 01H,DELAY0 ;valorile lu' R6 si R7 se modifica in rutinele ADUN si SCAD !
    DJNZ 00H,DELAY0
    RET
    
```

```

INTER:                                ;cand am depasire de Timer T0, intru in aceasta rutina

    DJNZ R2,SCAD                       ;decrementeaza pe R2(unde sunt "treptele"), si sari daca NU este zero
    INC R3                              ;AJUNG AICI NUMAI DACA R2 ESTE ZERO!!! ;incrementeaza R3
    INC R2                              ;                               ;incrementeaza R2
    CJNE R3,#100,adun                  ;sunt pe panta crescatoare a frecv. sunetului: am ajuns la treapta 100?
    MOV R2,#100                        ;acici, R3=100, atunci incarca si R2=100
    MOV R3,#0                          ;incarca treapta "0" pentru urmatoarea crestere de frecventa
    RETI                               ;revenire din intrerupere
    
```

```

ADUN:
    clr c                               ;sterge flagul CARRY - se poate seta la adunare cu depasire
    MOV A,R7                            ;rutina de adunare intre doi octetzi
    ADD A,R5
    MOV R7,A
    MOV A,R6
    ADDC A,R4                          ;aduna cu carry
    MOV R6,A
    CLR C
    mov th0,0A8H                       ;preincarc TH0=A8CF
    mov tl0,0CFH
    RETI
    
```

```

SCAD:
    CLR C
    MOV A,R7
    SUBB A,R5
    MOV R7,A
    MOV A,R6
    SUBB A,R4
    MOV R6,A
    CLR C
    mov th0,0A8H                       ;A8CF
    mov tl0,0CFH
    RETI
    
```

END