

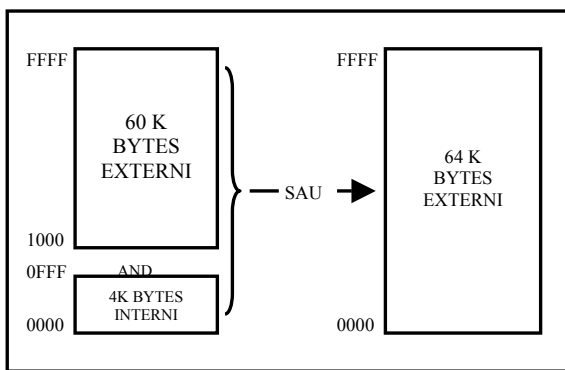
Microcontroler – arhitectură 8051 – generalități

După cum a-ți putut constata din Laboratorul 1, un sistem de calcul are nevoie de Memorie de Date (RAM) și Memorie Program (ROM). Ne vom opri la cele două tipuri de memorii. In Memoria Program este stocat programul ce urmează a fi executat de Unitatea Aritmetico Logică (ALU) , conține instrucțiuni (ex. adunare, scădere ,salt ,...) și operanzi (memorați pe bit sau octet: 1,2,3, ...). Memoria de Date este o zonă de memorie volatilă, de dimensiune

- adresată direct/indirect = 128 octeți „inferiori”
- adresată indirect = 128 octeți „superiori”
- adresată direct = 128 octeți „superiori” – această zonă se mai numește și Regiștri cu Funcții Speciale (SFR).

In Fig. 1 este prezentată o variantă de Memorie Program cu 4K „on-chip” și restul de memorie de 60K să fie externă, SAU toată Memoria Program să fie externă.

In Fig. 2 este prezentată Memoria de Date internă (observați maparea celor 128 octeți superiori), SAU toată Memoria de Date să fie externă.



1 Memoria Program

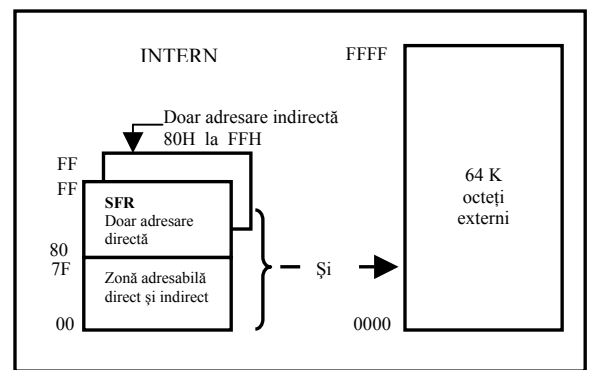


Fig. 2 Memoria de Date

Cei 128 octeți „inferiori” din Memoria de Date pot fi adresați direct și/sau indirect. – Fig. 3.

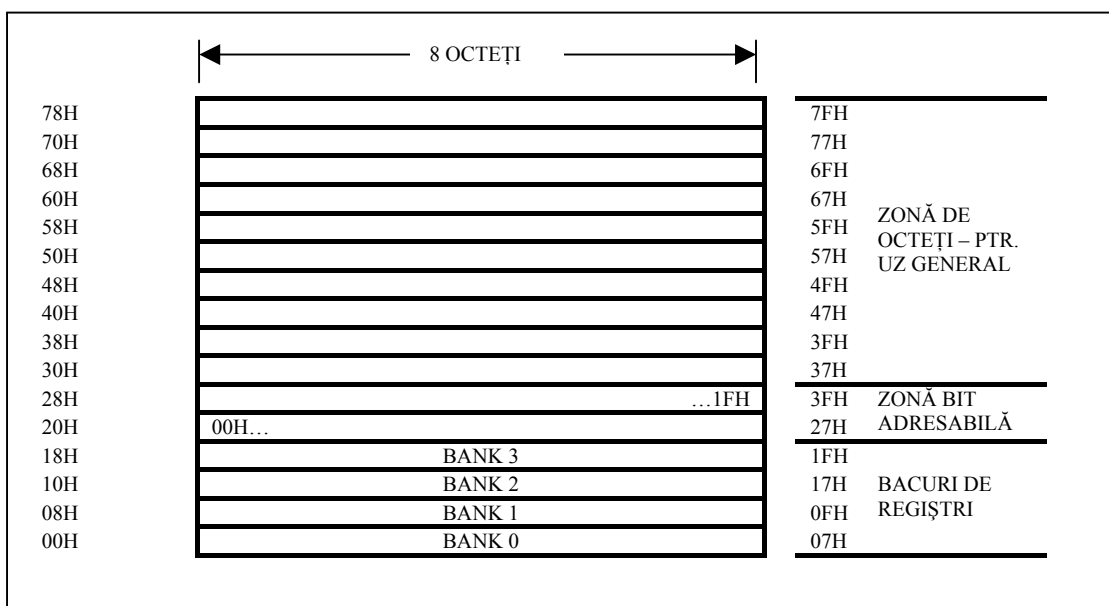


Fig. 3
 128 octeți (00H-7FH) „inferiori”, zonă adresabilă direct și indirect

Informații suplimentare găsiți la pagina:

http://81.180.222.26/site/Microcontrolere/Referate_laborator/pagAduC832.pdf

Regiștrii cu Funcții Speciale se află în spațiul memoriei de date, de la adresa 80H la adresa FFH și pot fi accesați doar prin adresare directă.

Harta Memoriei Regiștrilor cu Funcții Speciale (SFR)

8 Bytes

F8								FF
F0	B							F7
E8								EF
E0	ACC							E7
D8								DF
D0	PSW							D7
C8								CF
C0								C7
B8	IP							BF
B0	P3							B7
A8	IE							AF
A0	P2							A7
98	SCON	SBUF						9F
90	P1							97
88	TCON	TMOD	TL0	TL1	TH0	TH1		8F
80	P0	SP	DPL	DPH			PCON	87

Fig. 4

Regiștrii din prima coloana pot fi adresați la nivel de bit.

Acumulatorul. ACC este registrul acumulator. Mnemonicele pentru instrucțiuni specifice cu acumulatorul se referă la acumulator în mod simplificat ca fiind A.

Registrul B. Registrul B este utilizat pe durata operațiilor de înmulțire și de împărțire. Pentru alte operații poate fi tratat ca alt registru de memorie.

Cuvântul de stare a programului PSW. Registrul PSW conține informații asupra stării programului, așa cum se detaliază în cele ce urmează.

Indicatorul de stivă SP (Stack Pointer). Indicatorul de stivă este un registru de 8 biți. El este incrementat înainte ca data să fie stocată în timpul executării instrucțiunilor PUSH sau CALL. Cum stiva poate fi plasată oriunde în interiorul memorie RAM on-chip, indicatorul de stivă este inițializat la valoarea 07H după reset. Aceasta determină ca stiva să înceapă de la locația 08H.

Indicatorul de date DPTR (Data Pointer). Indicatorul de date (DPTR) este format dintr-un octet superior (DPH) și un octet inferior (DPL). Funcția lor a fost în intenție de a păstra adrese de 16 biți. El poate fi manipulat ca un registru de 16 biți sau ca două registre de 8 biți.

Porturile 0 la 3. P0, P1, P2 și P3 sunt latchurile porturilor 0, 1, 2 și 3. Scrierea unui "1" într-un bit al unui port din SFR (P0, P1, P2 sau P3) determină ca pinul de ieșire corespunzător din port să comute la nivel HIGH. Scrierea unui zero va determina trecerea la nivel LOW a pinului corespunzător. Atunci când este utilizat ca intrare, starea externă a unui pin dintr-un port va fi înscrisă în portul SFR (de exemplu, dacă starea externă a pinului este LOW, bitul corespunzător din portul SFR va conține un "0"; dacă nivelul este HIGH, bitul va conține un "1").

Bufferul de date seriale. Bufferul de date seriale este format în realitate din două registre separate, un buffer transmițător și un buffer receptor. Când data este transferată în SBUF, ea merge în bufferul transmițător și este păstrată pentru transmisia serială. Când data este transferată din SBUF, ea provine de la bufferul receptor.

Registrele timerelor de bază ale 80C51. Registrele pereche (TH0, TL0) și (TH1, TL1) sunt registre numărătoare de 16 biți pentru circuitele timer / numărător 0, respectiv 1.

Registreele de control pentru 80C51. Registreele de funcții speciale IP, IE, TMOD, TCON, SCON și PCON conțin biții de control și de stare pentru sistemul de întreruperi, timere / numărătoare, pentru portul serial și pentru controlul alimentării.

Registrul cuvânt de stare program (PSW)

MSB						LSB	
CY	AC	F0	RS1	RS0	OV	-	P

BIT	SIMBOL	FUNCTIE
PSW.7	CY	Flag transport
PSW.6	AC	Flag transport auxiliar
PSW.5	F0	Flag 0 (disponibil utilizatorului pentru aplicații generale)
PSW.4	RS1	Bit 1 de control pentru selecția bancului (vezi nota)
PSW.3	RS0	Bit 0 de control pentru selecția bancului Setat/șters software pentru a preciza bancul de registre (vezi nota)
PSW.2	OV	Flag depășire
PSW.1	-	Flag ce poate fi definit de utilizator
PSW.0	P	Flag de paritate Setat/șters prin hardware la fiecare ciclu de instrucțiune pentru a indica un număr par / impar de biți "1" din acumulator, și anume par

Valorile din RS0 și RS1 selectează bancul de regiștri corespunzător tabelului de mai jos:

RS1	RS0	Register Bank	Adresa
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

Moduri de adresare

Modurile de adresare din setul de instrucțiuni 80C51 sunt următoarele:

Adresarea directă. În adresarea directă, operandul este specificat de un câmp de adresă de 8 biți în instrucțiune. Adresarea directă se face numai pentru memoria internă RAM și pentru registrele de funcții speciale SFR.

Exp. MOV 32H, 31H

Adresare imediată. Valoarea unei constante poate urma unui cod de operare în memoria program. De exemplu:

Exp. MOV A, #25

încarcă acumulatorul cu valoarea zecimală 25. Același număr poate fi specificat în hex prin 19H.

Adresarea indirectă. În adresarea indirectă, instrucțiunea specifică un registru care conține adresa operandului. Atât memoria internă cât și cea externă RAM pot fi adresate indirect.

Registrul de adresă pentru adresele de 8 biți pot fi R0 sau R1 din bancul selectat sau indicatorul de stivă (*Stack Pointer*). Registrul de adresă pentru adrese de 16 biți poate fi numai registrul de 16 biți DPTR (*Data Pointer*) - indicator de date.

Exp. MOV R0, #32H

MOV @R0, 31H

Instrucțiuni registre. Bancurile pentru registre, conținând registrele R0 la R7, pot fi accesate prin anumite instrucțiuni ce conțin 3 biți de precizare a registrului în interiorul codului de operare al instrucțiunii. Instrucțiunile ce accesează registrele în acest mod sunt coduri eficiente, întrucât acest mod elimină un octet de adresă. Când instrucțiunea este executată, este accesat unul din cele opt registre din bancul selectat. La momentul executării instrucțiunii este selectat unul din cele patru bancuri de registre de către cei doi biți de selecție a bancului din PSW.

Exp. ADD A,Rn

Instrucțiuni registre speciale. Câteva instrucțiuni sunt specifice unor anumite registre. De exemplu, câteva instrucțiuni operează întotdeauna cu acumulatorul, cu indicatorul de date (Data pointer) etc., și nu mai este necesar un octet de adresă pentru precizare. Codul de operare însuși realizează acest fapt. Instrucțiunile ce se referă la acumulator ca A, asamblează coduri de operare specifice pentru acumulator.

Exp. INC A
DEC A
MUL AB

Adresarea indexată. Cu adresare indexată se poate accesa numai memoria program și numai în modul citire.

Acest mod de adresare este conceput pentru citirea tabelor de informații din memoria program. Un registru bază de 16 biți (fie DPTR sau Contorul Program) fixează baza tabelului, iar acumulatorul este încărcat cu numărul de intrare în tabel.

Adresa de intrare în tabelul din memoria program se formează prin adunarea datei din acumulator la indicatorul bazei.

Exp. MOV DPTR,#TAB1
MOV A,#01H
MOVC A,@A+DPTR
MOV P0,A

Sistemul de întreruperi

Microcontrolerul 80C51 asigură 5 surse de întrerupere. Acestea sunt prezentate în Fig. 1. Întreruperile externe $\overline{INT0}$ și $\overline{INT1}$ pot fi fiecare activate fie pe nivel, fie pe front (tranziție), funcție de biții IT0 și IT1 din registrul TCON. Flagurile care generează în realitate aceste întreruperi sunt biții IE0 și IE1 din TCON. Când se generează o cerere de întrerupere externă, flagul care a generat-o este șters de către hardware atunci când rutina de servire este executată, numai dacă întreruperea a fost activată de o tranziție. Dacă întreruperea a fost activată pe nivel, atunci sursa externă a cererii este cea care controlează flagul de cerere, și nu partea hardware on-chip. (în acest caz, în rutina de deservire a întreruperii, programatorul, va trebui să șteargă soft acest flag – în caz contrar, la revenirea din întrerupere, flagul este găsit iar setat, și iar se duce în întrerupere – fără ca să se fi cerut o întrerupere).

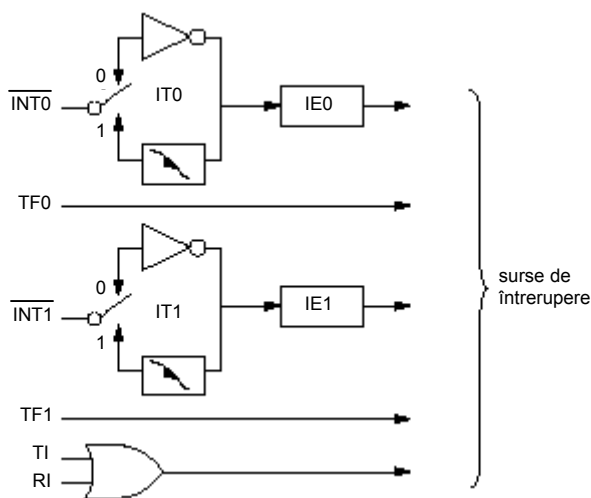


Fig. 5 Surse de întrerupere 80C51

Înteruperile timer 0 și timer 1 sunt generate de TF0 și TF1, ce sunt setați de o trecere la depășire în respectivele registre ale circuitelor timer / numărător. Când se generează o întrerupere de timer, flagul care o generează este șters de hardwareul on-chip atunci când rutina de servire este vectorizată (deservită).

Cererea de întrerupere de port serial este generată de operația logică SAU între RI și TI. Niciunul dintre aceste flaguri nu este șters prin hardware atunci când rutina de deservire este vectorizată. De fapt, rutina de deservire va trebui în mod normal să stabilească dacă întreruperea a fost determinată de RI sau TI și bitul care va trebui să fie șters prin software.

Toți acești biți care generează întreruperi pot fi setați sau șterși prin software, cu același efect ca și când ar fi fost setați sau șterși prin mijloace hardware. Astfel, întreruperile pot fi generate sau întreruperile nerezolvate pot fi anulate prin software.

Fiecare din aceste surse de întreruperi pot fi activate sau dezactivate în mod individual prin setarea sau ștergerea unui bit în registrul funcției speciale IE (Fig. 6). IE conține de asemenea un bit de dezactivare globală, EA, care dezactivează simultan toate întreruperile.

MSB				LSB			
EA	X	X	ES	ET1	EX1	ET0	EX0

BIT	SIMBOL	FUNCȚIE
IE.7	EA	Dezactivează toate întreruperile. Dacă EA = 0 nu va fi confirmată nici o întrerupere. Dacă EA = 1, fiecare sursă de întrerupere este activată în mod individual prin setarea sau ștergerea bitului său de activare
IE.6	–	Rezervat
IE.5	–	Rezervat
IE.4	ES	Activează sau dezactivează întreruperea pentru portul serial. Dacă ES = 0, întreruperea pentru portul serial este dezactivată.
IE.3	ET1	Activează sau dezactivează întreruperea la depășirea timerului 1. Dacă ET1 = 0 întreruperea pentru timerul 1 este dezactivată
IE.2	EX1	Activează sau dezactivează întreruperea externă 1. Dacă EX1 = 0, întreruperea externă 1 este dezactivată.
IE.1	ET0	Activează sau dezactivează întreruperea la depășirea timerului 0. Dacă ET0 = 0 întreruperea pentru timerul 0 este dezactivată
IE.0	EX0	Activează sau dezactivează întreruperea externă 0. Dacă EX0 = 0, întreruperea externă 0 este dezactivată.

Fig. 6 Registrul de activare a întreruperilor (IE)

Structura nivelului de prioritate

Fiecare sursă de întrerupere poate fi de asemenea programată în mod individual la unul din două nivele de prioritate prin setarea sau ștergerea unui bit în registrul funcției speciale IP (Fig. 7). O întrerupere de nivel inferior poate fi ea însăși întreruptă de o întrerupere de nivel superior, dar nu și de o altă întrerupere de nivel inferior. O întrerupere de nivel superior nu poate fi întreruptă de nici o altă sursă de întrerupere.

MSB				LSB			
X	X	X	PS	PT1	PX1	PT0	PX0

BIT	SIMBOL	FUNCȚIE
IP.7	–	Rezervat
IP.6	–	Rezervat
IP.5	–	Rezervat
IP.4	PS	Definește nivelul de prioritate pentru întreruperea de port serial. PS = 1 îl programează la cel mai înalt nivel de prioritate.
IP.3	PT1	Definește nivelul de prioritate pentru întreruperea de timer 1. PT1 = 1 îl programează la cel mai înalt nivel de prioritate.
IP.2	PX1	Definește nivelul de prioritate pentru întreruperea externă 1. PX1 = 1 îl programează la

IP.1	PT0	cel mai înalt nivel de prioritate. Activează sau dezactivează nivelul de prioritate al întreruperii pentru timerul 0. PT0 = 1 îl programează la cel mai înalt nivel de prioritate.
IP.0	PX0	Definește nivelul de prioritate pentru întreruperea externă 0. PX0 = 1 îl programează la cel mai înalt nivel de prioritate.

Fig. 7 Registrul priorităților întreruperilor (IP)

Dacă două cereri pe diferite nivele de prioritate sunt recepționate simultan, cererea de prioritate mai înaltă este servită. Dacă se recepționează simultan cereri de întrerupere de același nivel, o secvență internă de polling (testare) determină care cerere va fi servită. Astfel, în interiorul fiecărui nivel de prioritate există o a doua structură de prioritate determinată prin secvența polling, după cum urmează:

Sursă	Prioritate în interiorul nivelului
1. IE0	cea mai înaltă
2. TF0	
3. IE1	
4. TF1	
5. RI + TI	cea mai scăzută

De notat că "prioritatea în interiorul nivelului" este o structură utilizată numai pentru rezolvarea cererilor simultane pe același nivel de prioritate.

Registrul IP conține un număr de biți neimplementați. Astfel IP.7, IP.6 și IP.5 sunt biți rezervați în 80C51. Software-ul utilizator **nu** trebuie să scrie "1" în aceste poziții, întrucât biții respectivi pot fi folosiți în alte produse ale familiei 8051.

În tabelul de mai jos este dată adresa vectorilor de întrerupere, adresă ce se încarcă în PC (Program Counter) în momentul unei cereri de întrerupere. Din acest motiv, programatorul trebuie să fie atent de la ce adresă plasează programul în memoria program (ex. ORG 0000H – dacă nu lucrez în întreruperi, dacă DA, sar peste adresa vectorului de întrerupere corespunzător întreruperii solicitate).

Sursă	Adresă vector
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
RI + TI	0023H

Întreruperi externe

Sursele externe de întrerupere pot fi programate pentru a fi activate pe nivel sau pe front, prin setarea sau ștergerea biților IT1 sau IT0 în registrul TCON. Dacă ITx = 0, întreruperea externă x este declanșată prin detectarea unui nivel LOW la pinul INTx. Dacă ITx = 1, întreruperea x este declanșată pe front. În acest mod, dacă eșantioanele succesive ale pinului ITx arată un nivel HIGH într-un ciclu și LOW în ciclul următor, este setat flagul de cerere a întreruperii IEx din TCON. După aceasta bitul IEx solicită o întrerupere.

Întrucât pinul întreruperii externe este eșantionat o dată la fiecare ciclu mașină, o intrare HIGH sau LOW trebuie să se mențină pentru cel puțin 12 perioade de oscilator pentru a asigura eșantionarea. Dacă întreruperea externă este activată pe tranziție, sursa externă trebuie să păstreze la pinul de cerere nivelul HIGH pentru cel puțin un ciclu, iar apoi să fie menținut LOW pentru cel puțin un ciclu. Acest lucru se face pentru a asigura faptul că tranziția este "văzută", astfel încât va fi setat flagul de cerere de întrerupere IE. IE va fi șters în mod automat de către CPU când se face apelul rutinei de servire.

Dacă întreruperile externe sunt activate pe nivel, sursa externă trebuie să mențină cererea activată până când întreruperea solicitată este generată în realitate. Apoi trebuie să se dezactiveze cererea înainte ca rutina de servire a întreruperii să fie completată, altfel va fi generată o nouă întrerupere.

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.

Fig. 8 Registrul de control timer / numărător (TCON)

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.

MSB				LSB			
GATE	C/ \bar{T}	M1	M0	GATE	C/ \bar{T}	M1	M0
← TIMER 1 →				← TIMER 0 →			

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	FUNCȚIONARE
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

Fig. 9 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. 10 se prezintă modul de funcționare atunci când acesta se aplică numărătorului 1.

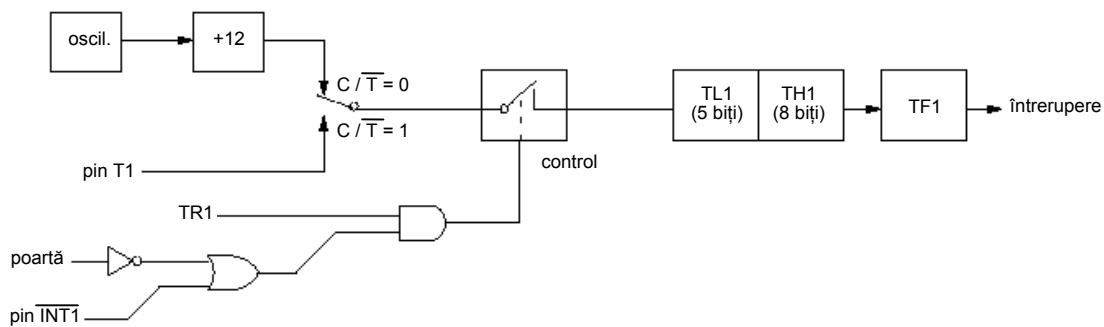


Fig. 10 Timer / Numărător mod 0: numărător de 13 biți

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.

Fig. 11 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 $\overline{INT1} = 1$ (setând GATE = 1 se permite timerului să fie controlat de către intrarea externă $\overline{INT1}$ 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. 11. 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.

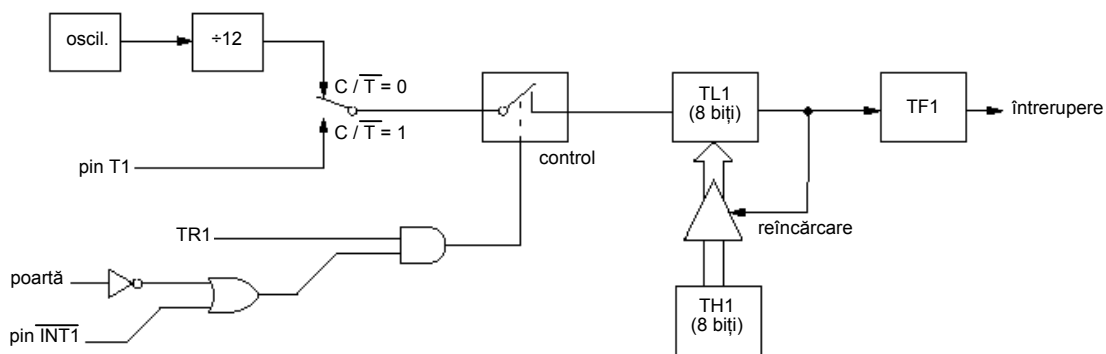


Fig. 12 Timer / Numărător mod 2: 8 biți cu auto-încărcare

Modul 3

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

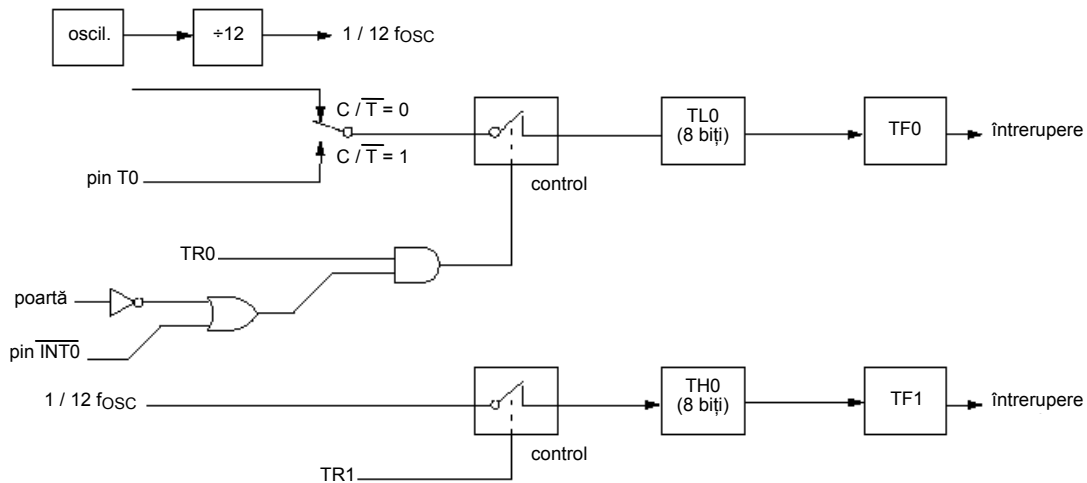


Fig. 13 Timerul / Numărător 0 - în mod 3: două numărătoare de 8 biți

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.

Utilizarea programului uVision2 - Keil Software - Crearea unui Proiect

Programul folosit în laborator este o variantă gratis oferită de compania Keil Software, având limitare de asamblare (compilare) a programelor ce depășesc 2K octeți. Acest program este un simulator de microcontrolere, ajutând programatorul să-și testeze programul, în timp real sau pas cu pas, înainte de a-l înscrie în microcontroler.

Utilizarea programului:

Se pornește de la crearea unui „proiect” : „project” → „new project” . Se dă nume proiectului și se salvează. În acest moment apare „Select device for target” , și se alege din librărie microcontrolerul pentru care se editează programul. Apoi se creează o pagină nouă „file” → „new”. Se salvează această pagină cu extensia: „.c”, apoi din „target 1” → „source group 1” , se dă click dreapta la mouse și se alege „Add files to Group ‘source group 1’ ”, se deschide o nouă fereastră de unde se alege fișierul mai devreme salvat. Atenție la extensia fișierului! Odată ales fișierul, se apasă butonul „close” pentru a închide fereastra de dialog.

Începând din acest moment, se poate trece la editarea programului. Programul oferă o interfață ușoară de lucru cu utilizatorul, de exp. mnemonicele din program se vor colora în albastru.

După ce s-a scris programul, se va trece la asamblarea acestuia, apăsând „F7” (build target). În acest moment se assemblează programul și dacă nu sunt erori se poate trece la simularea programului.