

# Curs 3

# Sisteme Încorporate

## 3 Periferia internă

### 3.1 Contoarele/ temporizatoarele 0,1

- Microcontrolerul 80C51 dispune de 2 contoare/ temporizatoare implementate cu două numărătoare independente, pe 16 biți, notate cu **TIMER 0** și **TIMER 1**;
- **TIMER 0** este alcătuit din numărătoarele **TH0** și **TL0** iar **TIMER 1** este alcătuit din numărătoarele **TH1** și **TL1**;
- Există 2 surse de tact pentru fiecare din cele două contoare/ temporizatoare;
- Dacă funcția îndeplinită este cea de **temporizator (timer mode)** atunci sursa este **tactul** microcontrolerului,  **$f_{osc}$  divizat la 12** deci numărătorul va fi incrementat în fiecare ciclu mașină;
- Dacă funcția îndeplinită este cea de **contor (counter mode)** atunci sursa este un **semnal extern, T0, respectiv T1**;
- **Ti** este testat în fiecare S5P2 a unui ciclu mașină și un front căzător al acestui semnal va incrementa numărătorul.

# Sisteme Încorporate

- Frontul căzător este detectat dacă într-un ciclu mașină se citește valoarea 1 la terminalul Ti și în ciclul următor se citește valoarea 0 la același terminal Ti; întrucât sunt necesare 2 cicluri mașină (2 internal machine cycles) pentru o detectare, rezultă că **frecvența maximă de numărare este de 1/ 24 din frecvența tactului**;
- Funcționarea contoarelor TIMER 0 și TIMER 1 este controlată de 2 registre:

- **TMOD** ("Timer Mode Register") cu structura:

GATE	C//T	M1	M0	GATE	C//T	M1	M0
------	------	----	----	------	------	----	----

- TMOD.0-1: M0, M1: fixează modul de lucru pentru TIMER 0.
- TMOD.2: C/ /T: stabilește sursa tactului pentru numărare:
  - 0: sursa tactului este  $f_{OSC}$  divizat la 12, 1: sursa tactului este T0.
- TMOD.3: GATE: controlează pornirea și oprirea numărării:
  - 0: pornirea/ oprirea este controlată de rangul TR0, 1: numărarea este pornită dacă TR0 = /INT0 = 1 și este oprită dacă TR0 = 0 sau /INT0 = 0.
- TMOD.4-5: idem cu TMOD.0-1 dar pentru TIMER 1.
- TMOD.6: idem cu TMOD.2 dar pentru TIMER 1.
- TMOD.7: idem cu TMOD.3 dar pentru TIMER 1.

# Sisteme Încorporate

- **TCON** ("Timer Control Register"), cu structura:

7							0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

- TCON.0: IT0: stabilește dacă întreruperea externă care sosește pe intrarea /INT0 va fi luată în considerare pe frontul căzător (1) sau pe nivel (0).
- TCON.1: IE0: indicator pentru întreruperea externă 0; este setat dacă s-a detectat întreruperea externă /INT0; este anulat dacă întreruperea este deservită, atunci când aceasta este luată în considerare pe front, adică IT0 = 1.
- TCON.2-3: idem cu TCON.0-1 dar pentru TIMER 1.
- TCON.4: TR0: rang de control pentru pornirea/ oprirea numărării la TIMER 0: 1 pornește numărarea iar 0 o oprește.
- TCON.5: TF0: indicator de depășire a numărătorului; este poziționat la depășirea capacității numărătorului provocată de o incrementare din starea FFFFH în starea 0000 și este o sursă pentru cererea de întrerupere de la TIMER 0; este anulat automat atunci când întreruperea este deservită.
- TCON.6-7: idem cu TCON.4-5 dar pentru TIMER 1.

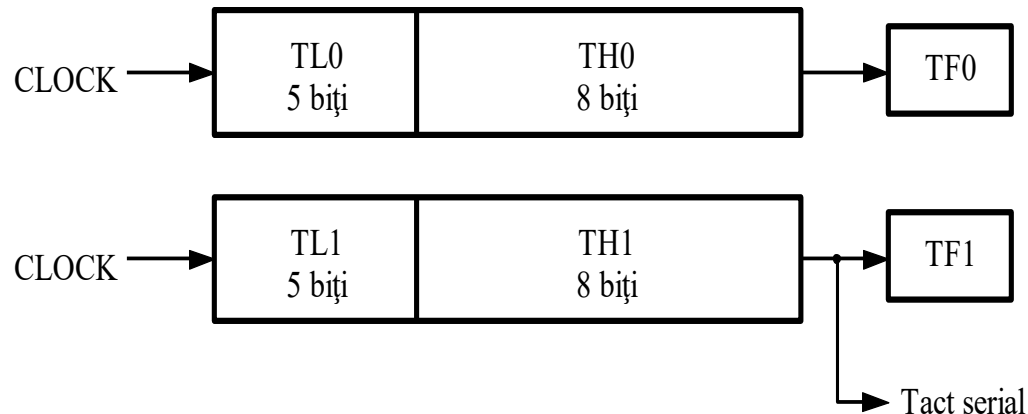
# Sisteme Încorporate

- **Controlul numărării** este realizat de rangurile TMOD.3 și TMOD.7, GATE, pentru TIMER 0, respectiv TIMER 1;
- Dacă GATE = 0 atunci rangurile TCON.4, TRO, respectiv TCON.6, TR1, vor controla numărarea; dacă TRi = 1, numărarea pornește iar dacă TRi = 0, numărarea se oprește;
- Dacă GATE = 1 atunci numărarea pornește dacă TRi & /INTi = 1 și se oprește dacă TRi = 0 sau /INTi = 0; în acest caz controlul numărării poate fi realizat extern, prin /INTi; o aplicație este aceea că se pot astfel măsura durate de semnale externe.
- Selectarea tactului este realizată de TMOD.2, respectiv TMOD.6, C/ /T;
- Dacă C/ /T = 0 atunci sursa tactului este  $f_{osc}/12$  iar dacă C/ /T = 1 atunci sursa tactului este un semnal extern aplicat la terminalele T0, respectiv T1.
- Există 4 moduri de lucru pentru TIMER 0 și 1, selectabile prin rangurile corespunzătoare M1 și M0.

# Sisteme Încorporate

## Modul 0:

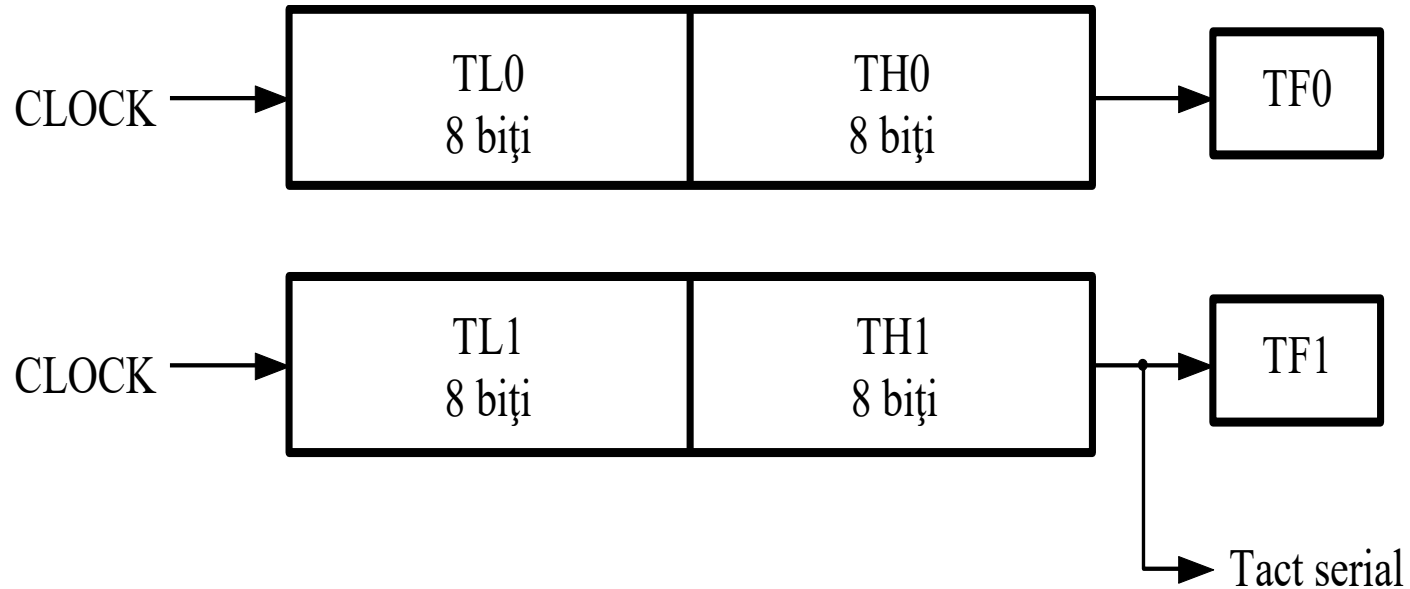
- TIMER 0 și 1 sunt configurate ca numărătoare pe 13 biți; TL0 și TL1 sunt numărătoare alcătuite din 5 ranguri, cele mai puțin semnificative iar TH0 și TH1 sunt numărătoare pe 8 biți; indicatorii de depășire TF0 și TF1 sunt poziționați atunci când numărătorul corespunzător conține valoarea maximă și este incrementat, anulându-se astfel conținutul său;
- Semnalul care apare la ieșirea de depășire a lui TIMER 1 este sursă pentru tactul serial.



# Sisteme Încorporate

## □ Modul 1:

- Este identic cu modul 0, cu excepția faptului că numărătoarele sunt pe 16 biți.



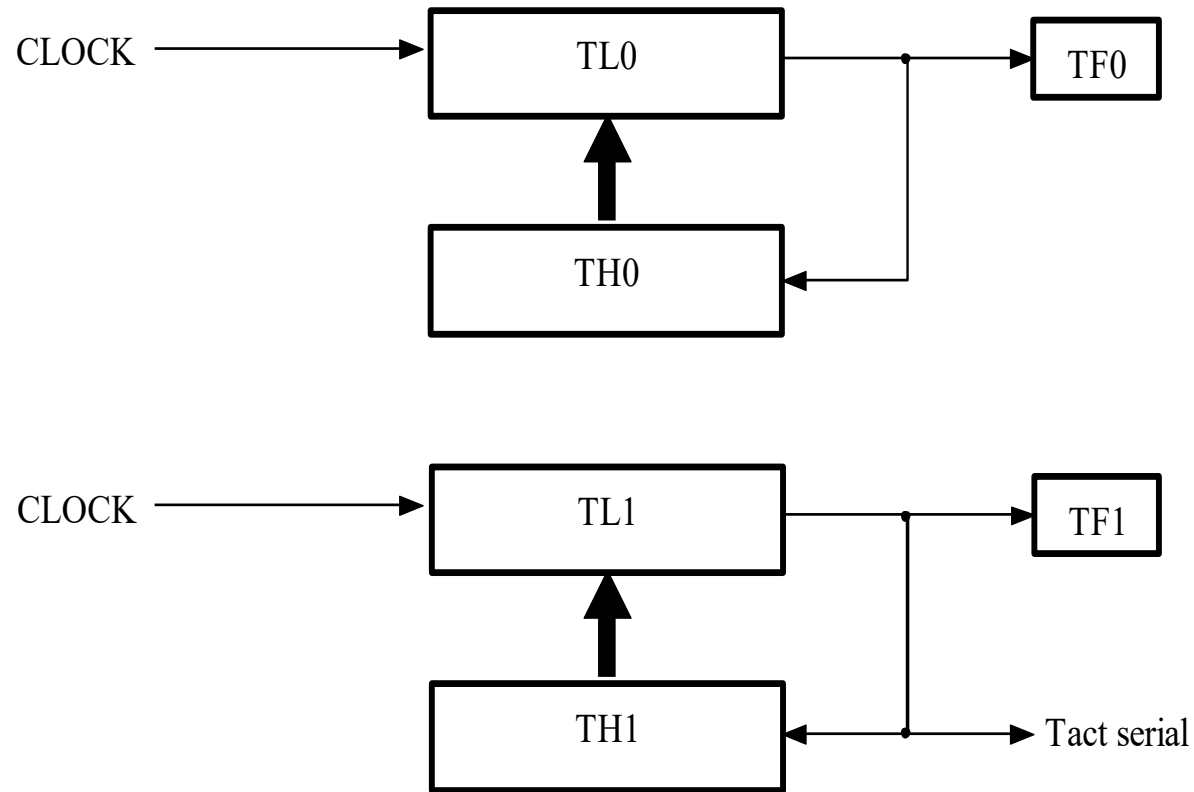
# Sisteme Încorporate

## □ Modul 2:

- TIMER 0 și 1 sunt configurate ca numărătoare pe 8 biți cu auto-încărcare;
- TL0 și TL1 sunt folosite ca numărătoare iar TH0 și TH1 sunt folosite ca registre pentru păstrarea constantei care se va încărca;
- Semnalul de depășire generat de TL0 sau TL1 va provoca preîncărcarea acestora și numărarea va continua de la această valoare;
- TH0 și TH1 sunt încărcate prin program; indicatorii TF0 și TF1 sunt poziționați de semnalele de depășire de la numărătoarele TL0, respectiv TL1;
- Semnalul de depășire generat de numărătorul TL1 este sursă pentru tactul serial;
- Figura următoare prezintă pe TIMER 0 și 1 în modul 2:



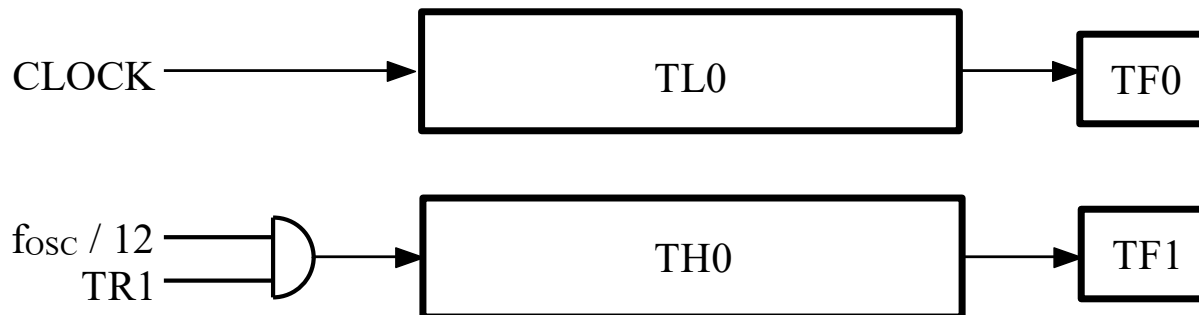
# Sisteme Încorporate



# Sisteme Încorporate

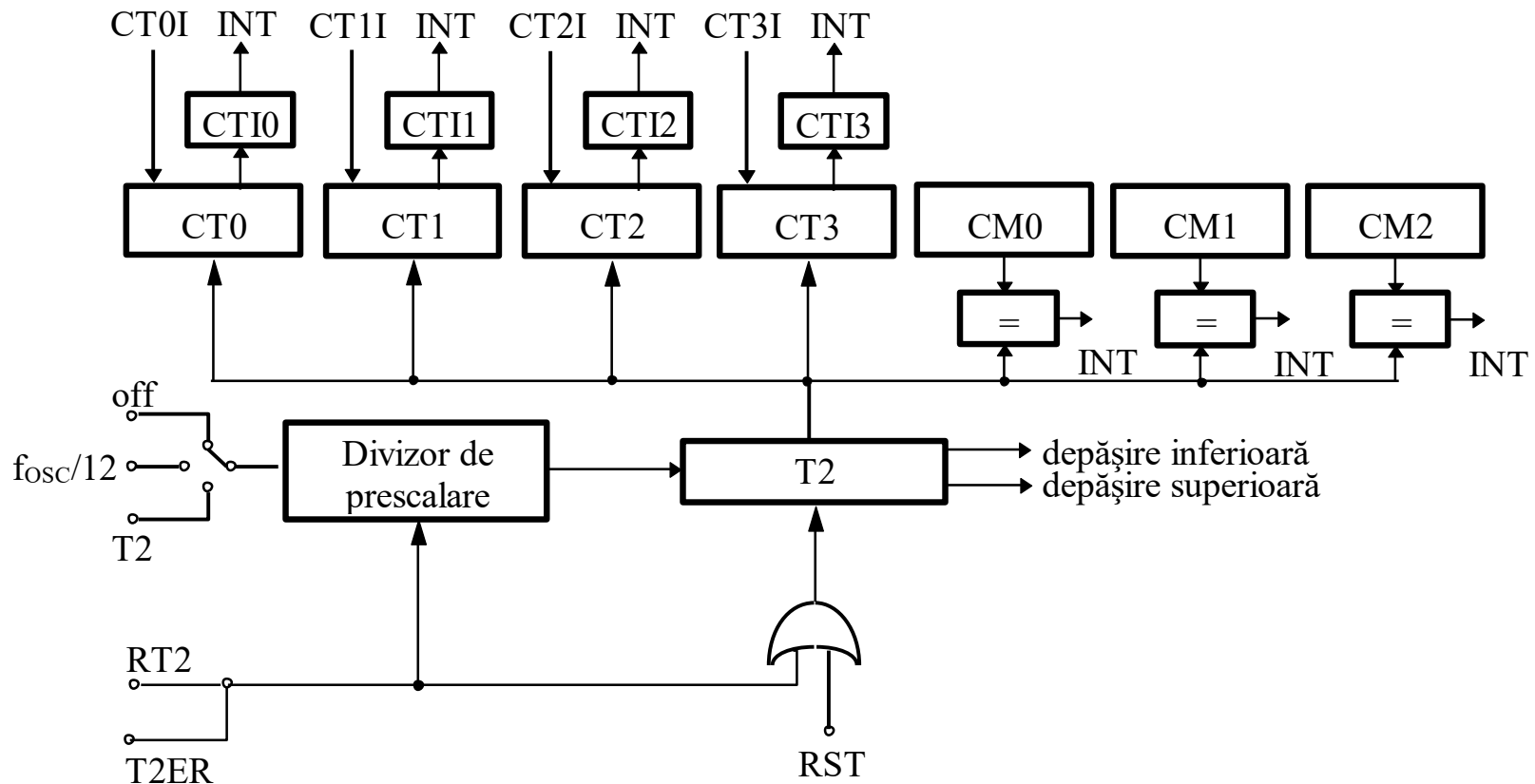
## □ Modul 3:

- Modul poate fi cerut doar pentru TIMER 0; dacă i se cere lui TIMER 1, acesta nu va număra (se obține același efect ca și când  $TR1 = 0$ );
- TIMER 0 în modul 3 lucrează ca două numărătoare independente pe 8 biți; TL0 folosește rangurile de control ale lui TIMER 0, adică C/ /T, GATE, TR0 și /INT0 și comandă pe TF0 iar TH0 folosește pe TR1 și comandă pe TF1;
- Modul este util în aplicații în care se cere un numărător suplimentar; TIMER 0 furnizează 2 contoare pe 8 biți iar TIMER 1 poate fi folosit în modurile 0, 1 sau 2 fără a putea, însă, să poziționeze pe TF1 deci nu va putea să genereze cerere de întrerupere.



# Sisteme Încorporate

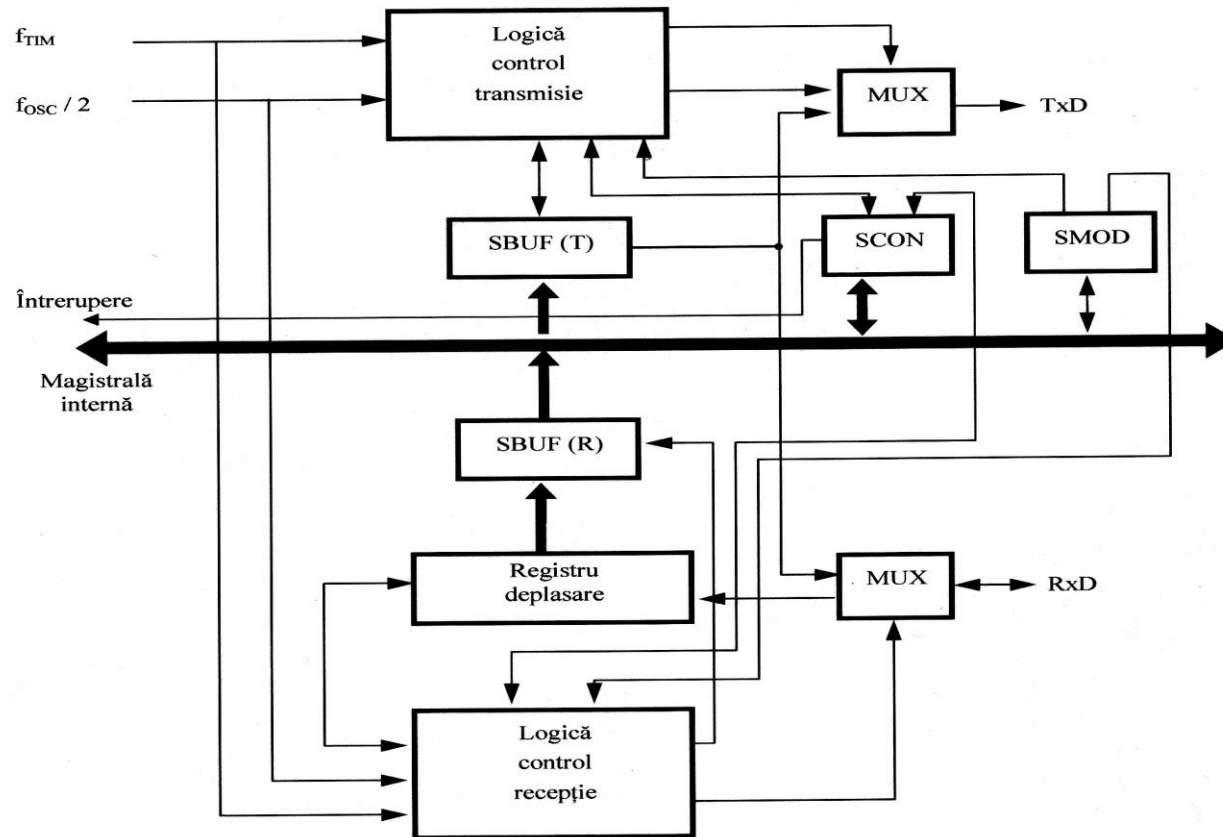
- Un exemplu de contor/ temporizator cu facilități de captare și comparare:



# Sisteme Încorporate

## 3.2 Interfața serială

- De tip **UART**; schema bloc este:

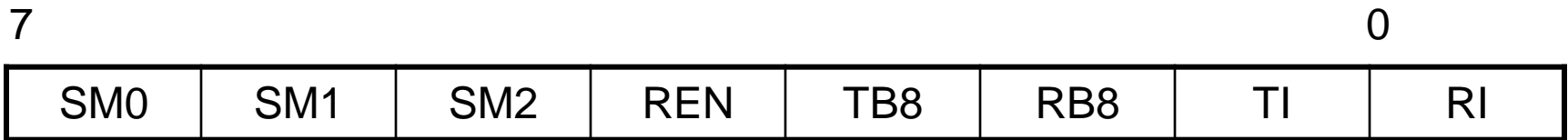


# Sisteme Încorporate

- Cuprinde:
  - logica de control pentru transmisie și recepție,
  - registrul de control **SCON** ("Serial Control Register"),
  - registrele tampon **SBUF** ("Serial Buffer Register") și
  - rangul **SMOD**;
- Registrul **SCON** conține biți de control pentru a specifica modurile de operare, activarea/ dezactivarea recepției de date, memorarea celui de-al 9-lea bit de date și starea interfeței seriale; fiecare rang poate fi adresat individual.
- Registrul **SBUF** este tamponul pentru date; deși îi corespunde o singură adresă, este alcătuit din 2 registre: unul pentru transmisie și unul pentru recepție; corespunzător, SBUF poate fi specificat în instrucțiuni ca registru sursă sau ca registru destinație; transmisia este anclanșată prin înscrierea datei în SBUF iar la recepție, data este citită din SBUF.
- Rangul **SMOD** este rangul 7 din registrul **PCON** ("Power Control Register") și are rolul să divizeze cu 2 ( $SMOD = 1$ ) sau nu ( $SMOD = 0$ ), rata de transfer.

# Sisteme Încorporate

## □ Structura registrului **SCON**



- SCON.0: RI: este un indicator activat de interfața serială atunci când a terminat de asamblat un caracter și îl poate oferi unității centrale; împreună cu rangul TI sunt intrări pentru o poartă SAU, a cărei ieșire poate fi folosită ca cerere de întrerupere; ștergerea se face doar prin program;
- SCON.1: TI: indicator activat de interfața serială atunci când a terminat de transmis un caracter și poate primi altul de la unitatea centrală; împreună cu rangul RI constituie intrări pentru o poartă SAU a cărei ieșire poate fi folosită ca cerere de întrerupere; ștergerea se face doar prin program;
- SCON.2: RB8: rang pentru memorarea celui de-al 9-lea bit recepționat dintr-un caracter pe 11 biți, folosit în modurile 2 și 3; în modul 1 va reține bitul STOP dacă rangul SM2 = 0 iar în modul 0 nu este folosit;
- SCON.3: TB8: rang pentru memorarea celui de-al 9-lea bit de date ce se va transmite într-un caracter pe 11 biți, folosit în modurile 2 și 3; este poziționat prin program.
- SCON.4: REN: trebuie să fie setat pentru ca recepția să fie activată;

# Sisteme Încorporate

- SCON.5: SM2: validează facilitatea de comunicare în sistem multiprocesor, în modurile 2 sau 3; dacă  $SM2 = 1$ , în modurile 2 sau 3 (cu 11 biți/ caracter) și al 9-lea bit de date este 1 atunci data recepționată este acceptată și este încărcată în SBUF și RB8 și indicatorul RI este activat; dacă al 9-lea bit de date este 0, data este ignorată și SBUF, RB8 și RI rămân nemodificate; dacă  $SM2 = 1$  în modul 1 (cu 10 biți/ caracter), se va încărca bitul de STOP în RB8, data este preluată și indicatorul RI este activat; dacă  $SM2 = 0$ , data este acceptată indiferent de valoarea celui de-al 9-lea bit de date sau a bitului de STOP;
- SCON. 6-7: SM1, SM0: setează modul de lucru al interfeței seriale, conform tab.:

SM0	SM1	Mod	Descriere	Rata de transfer
0	0	0	I/E extins	$f_{osc}/12$
0	1	1	UART cu 10 biți	variabilă
1	0	2	UART cu 11 biți	$f_{osc}/12$ sau $f_{osc}/64$
1	1	3	UART cu 11 biți	variabilă

# Sisteme Încorporate

- Interfața serie poate lucra în modurile:
  - I/ E (intrare/ ieșire) extins sau
  - UART;
  - În modul I/ E extins, transferul se face sincron cu tactul microcontrolerului;
  - În modul UART se pot transfera caractere pe 10 sau 11 biți; caracterul pe 11 biți este alcătuit dintr-un bit de START, 8 biți de date, un al 9-lea bit programabil și bitul de STOP;
  
- **Modul 0:**
  - Este modul numit și I/ E extins în care se transferă date pe 8 biți, sincronizate cu tactul microcontrolerului;
  - Terminalul TxD este folosit doar pentru a genera tactul iar terminalul RxD este folosit pentru a transfera date în ambele sensuri;
  - Rata de transfer este:  $R = f_{OSC} / 12$ .



## □ Modul 1:

- Este un mod UART în care se transferă caractere pe 10 biți: 1 bit de START, 8 biți de date, primul fiind cel mai puțin semnificativ și 1 bit de STOP;
- La recepție, bitul de STOP se încarcă în RB8;
- Rata de transfer este determinată de frecvența semnalului de la ieșirea de depășire a lui TIMER 1,  $f_{TIM}$ ;
- Dacă TIMER 1 lucrează în modul 0 sau 1 atunci rata de transfer este:

$$R = 2^{SMOD} \times f_{TIM} / 32;$$

- Dacă TIMER 1 lucrează în modul 2 atunci rata de transfer este:

$$R = 2^{SMOD} \times f_{osc} / 32 \times 12 \times (256 - (TH1)),$$

(TH1) este constanta de preîncărcare;

# Sisteme Încorporate

- TIMER 1 poate lucra ca și contor sau ca temporizator și este necesară dezactivarea întreruperii generată de TF1;
- Transmisia se realizează pe terminalul TxD și este anclanșată de o scriere în tamponul SBUF iar recepția se realizează pe terminalul RxD și este anclanșată la apariția bitului de START;
- Data se preia din tamponul SBUF;
- Încărcarea datei în SBUF, a bitului de STOP în RB8 și activarea indicatorului RI se realizează doar dacă în momentul respectiv sunt îndeplinite condițiile:
  - RI = 0 și
  - SM2 = 0 sau SM2 = 1 și bitul STOP = 1.
- Dacă vreuna din aceste condiții nu este îndeplinită atunci data este ignorată și SBUF, RB8 și RI rămân nemodificate

# Sisteme Încorporate

## □ Modul 2

- Mod UART cu 11 biți/ caracter: 1 bit de START, 8 biți de date, 1 bit programabil și 1 bit de STOP;
- La transmisie, bitul al 9-lea este TB8 și poate fi programat iar la recepție, bitul al 9-lea este încărcat în RB8;
- Rata de transfer este:  
$$R = 2^{\text{SMOD}} \times f_{\text{OSC}} / 64;$$
- **Transmisia**: bit de START, 8 biți de date, bit 9 (TB8, care poate fi bitul de paritate dacă în prealabil acesta a fost încărcat în TB8), bit de STOP și activare a indicatorului TI;
- La **recepție**, după preluarea bitului de START și a biților de date, este preluat al 9-lea bit, care se încarcă în RB8, apoi este preluat bitul de STOP și se activează indicatorul RI; preluarea datei, încărcarea celui de-al 9-lea bit în RB8 și activarea indicatorului RI au loc dacă sunt îndeplinite, la momentul respectiv, condițiile:
  - $RI = 0$  și
  - $SM2 = 0$  sau  $SM2 = 1$  și bitul al 9-lea = 1;
- Dacă vreuna din aceste condiții nu este îndeplinită atunci data este ignorată și SBUF, RB8 și RI rămân nemodificate.

# Sisteme Încorporate

## □ Modul 3

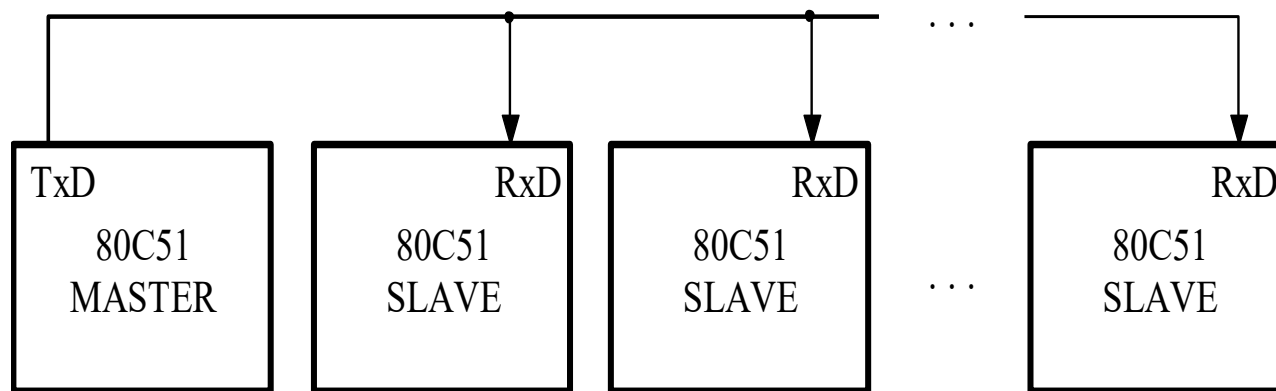
- Mod UART cu 11 biți/ caracter care diferă de modul 2 doar prin rata de transfer care aici depinde de  $f_{TIM}$ ; astfel:
  - $R = 2^{SMOD} \times f_{TIM} / 32$  dacă TIMER 1 lucrează în modul 0 sau 1 și
  - $R = 2^{SMOD} \times f_{osc} / 32 \times 12 \times (256 - (TH1))$  dacă TIMER 1 lucrează în modul 2.
- Ratele de transfer uzuale și obținerea lor din TIMER 1:

Rata de transfer	$f_{osc}$ [MHz]	SMOD	TIMER1		
			C/ /T	Mod	Const. de preîncărcare
Mod 0: max. 1,67 Mbiți/ s	20	X	X	X	X
Mod 2: max. 625 Kbiți/ s	20	1	X	X	X
Mod 1, 3: max104,2 Kbiți/ s	20	1	0	2	FFH
19,2 Kbiți/ s	11,059	1	0	2	FDH
9,6 Kbiți/ s	11,059	0	0	2	FDH
4,8 Kbiți/ s	11,059	0	0	2	FAH
2,4 Kbiți/ s	11,059	0	0	2	F4H
1,2 Kbiți/ s	11,059	0	0	2	E8H
110 biți/ s	12	0	0	1	D4H
110 biți/ s	6	0	0	2	72H

# Sisteme Încorporate

## Aplicație:

- Facilitatea de comunicare în sistem multiprocesor: se bazează pe existența celui de-al 9-lea bit de date în modurile 2 sau 3;
- Dacă rangul SM2 = 1 și bitul al 9-lea este 1 atunci data este preluată iar dacă bitul al 9-lea este 0 atunci data este neglijată;
- Fie un sistem multiprocesor, cu un microcontroler master și mai multe microcontrolere slave;
- Datele vor fi transmise doar de la master la slave.



# Sisteme Încorporate

- În faza inițială, toate microcontrolerele slave au  $SM2 = 1$  și așteaptă să fie selectate;
- Microcontrolerul master emite adresa microcontrolerului slave selectat; adresa diferă de date prin aceea că are rangul al 9-lea, TB8, egal cu 1;
- Toate microcontrolerele slave vor prelua adresa, își vor genera întreruperi și în rutinele de tratare vor verifica propria adresă cu cea sosită de la microcontrolerul master;
- Microcontrolerul slave selectat va modifica  $SM2$  deci doar la acesta va exista  $SM2 = 0$  iar la toate celelalte microcontrolere slave va exista  $SM2 = 1$ ; în continuare, microcontrolerul master emite datele, cu TB8 = 0;
- Doar acel microcontroler slave care va avea  $SM2 = 0$  va prelua datele iar toate celelalte microcontrolere slave le vor ignora întrucât la acestea  $SM2 = 1$ .
- După ce microcontrolerul master a emis toate datele, el va emite un cod de final, predeterminat, care va avea ca efect setarea lui  $SM2$  de către microcontrolerul slave care a fost selectat; în acest fel s-a revenit în starea inițială în care toate microcontrolerele slave au  $SM2 = 1$  și așteaptă să fie selectate.

## 3.3 Sistemul de întreruperi

- 5 surse pentru întreruperi:
  - 2 întreruperi externe generate la intrările /INT0 și /INT1 și
  - 3 întreruperi interne: 2 de la circuitele contoare/ temporizatoare și una de la întreruperea serială.
- Cererile de întrerupere setează indicatori care sunt ranguri din registrele TCON și SCON și care generează, de fapt, întreruperile;
- Astfel întreruperile externe setează indicatorii IE0 (TCON.1) și IE1 (TCON.3), întreruperile de la contoare setează rangurile TF0 (TCON.5) și TF1 (TCON.7) iar întreruperea serială setează fie rangul RI (SCON.0) fie rangul TI (SCON.1);
- RI și TI sunt intrări într-o poartă SAU a cărei ieșire este cererea de întrerupere de la interfața serială; pot fi controlați și prin program.
- Dacă se încarcă 1 în unul din acești indicatori, efectul va fi același ca și cel produs de cererea corespunzătoare hardware.

# Sisteme Încorporate

- Dacă există o cerere hardware în așteptare, care nu a fost încă tratată, ea poate fi anulată prin ștergerea indicatorului corespunzător.
- Întreruperile sunt controlate de 2 registre:
  - IE ("Interrupt Register") și
  - IP ("Priority Register");
  - Rolul registrului IE este acela de a valida/ invalida întreruperile; pentru a valida o întrerupere, bitul corespunzător trebuie să fie 1 iar pentru a o dezactiva, bitul corespunzător trebuie să fie 0; în plus, există un rang pentru dezactivarea generală a întreruperilor;
  - Rolul registrului IP este acela de a alocă priorități;
    - Există 2 nivele de prioritate: scăzut și ridicat; fiecărei surse i se poate alocă , în mod independent, oricare nivel de prioritate;
    - Fiecărei întreruperi îi corespunde un rang: 1 în rangul respectiv înseamnă prioritate ridicată iar 0 în rangul respectiv înseamnă prioritate scăzută;
    - Dacă apar simultan 2 cereri de întrerupere, de priorități diferite, va fi tratată cererea de prioritate mare;
    - Dacă apar simultan 2 sau mai multe cereri de aceeași prioritate, microcontrolerul le va lua în considerare într-o ordine prestabilită care este: IE0, TF0, IE1, TF1 și, ultima, RI + TI;
    - Rutina de tratare a unei întreruperi de prioritate scăzută poate fi întreruptă de o cerere de prioritate ridicată dar nu și invers; rutina de tratare a unei întreruperi nu poate fi întreruptă de o cerere de aceeași prioritate.



# Sisteme Încorporate

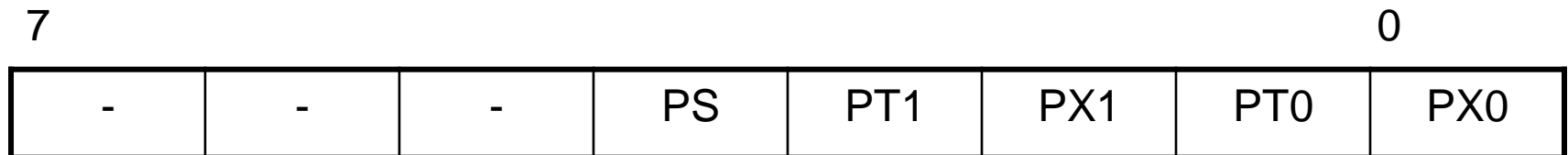
## □ Structura registrului IE:



- IE.0: EX0: bit de control pentru întreruperea externă 0.
- IE.1: ET0: bit de control pentru contorul/ temporizatorul 0.
- IE.2: EX1: bit de control pentru întreruperea externă 1.
- IE.3: ET1: bit de control pentru contorul/ temporizatorul 1.
- IE.4: ES: bit de control pentru întreruperea serială.
- IE.7: EA: bit de control general: dacă este 0 atunci o întrerupere nu poate fi tratată; dacă este 1, o întrerupere poate fi tratată dacă bitul de control corespunzător este 1.

# Sisteme Încorporate

## □ Structura registrului IP:



- IP.0: PX0: rang de prioritate pentru întreruperea externă 0.
- IP.1: PT0: rang de prioritate pentru întreruperea de la contorul/temporizatorul 0.
- IP.2: PX1: rang de prioritate pentru întreruperea externă 1.
- IP.3: PT1: rang de prioritate pentru întreruperea de la contorul/temporizatorul 1.
- IP.4: PS: rang de prioritate pentru întreruperea serială.

# Sisteme Încorporate

- Întreruperile sunt citite în timpul lui S5P2 al fiecărui ciclu mașină;
- Apoi are loc o testare a ceea ce s-a citit în timpul următorului ciclu mașină;
- Dacă vreunul din indicatorii de întrerupere a fost găsit activ și dacă întreruperea respectivă este validată atunci sistemul de întreruperi va genera o instrucțiune LCALL ("Long CALL") ADR, unde ADR depinde de tipul întreruperii, conform tab.:

Tip de întrerupere	Adresă
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
RI + TI	0023H

# Sisteme Încorporate

- Instrucțiunea LCALL nu se va genera în una din următoarele situații:
  - 1. o întrerupere de prioritate egală sau mai mare este deja în tratare;
  - 2. ciclul curent de testare nu este ultimul ciclu al instrucțiunii curente sau
  - 3. instrucțiunea curentă este RETI sau o instrucțiune ce scrie în registrele IE sau IP.
- **Condiția 2** asigură luarea în considerare a unei cereri de întrerupere doar la sfârșitul unei instrucțiuni iar **condiția 3** asigură luarea în considerare a unei cereri de întrerupere doar după ce se execută încă o instrucțiune după RETI sau după cea care scrie în registrele IE sau IP;
- Dacă există o singură sursă de întrerupere atunci timpul de răspuns este între 3 și 8 cicluri mașină.
- Instrucțiunea LCALL va depune conținutul lui PC în stivă și va încărca PC cu adresa de salt;
- În unele cazuri, are loc ștergerea, prin hardware, a indicatorului ce corespunde întreruperii tratate: indicatorii TF0, TF1 și IE0, IE1 dar numai dacă întreruperea externă este activă pe front;
- Indicatorul ce corespunde întreruperii seriale trebuie să fie șters doar prin program.

# Sisteme Încorporate

- Execuția rutinei de tratare a cererii de întrerupere se încheie la întâlnirea instrucțiunii RETI; încarcă în PC adresa de revenire și inițializează sistemul de întreruperi; rutina se poate încheia și cu instrucțiunea RET dar nu se va inițializa sistemul de întreruperi;
- Întreruperile externe pot fi programate, prin rangurile IT0, respectiv IT1 din registrul TCON, să fie active pe nivel sau pe front;
- O tranziție  $1 \rightarrow 0$  pe una din intrările /INT0 sau /INT1 va poziționa indicatorul IE0 sau IE1, dacă întreruperea este validată și dacă cererea are o durată minimă de 12 perioade de tact pe nivel 1 și 12 perioade de tact pe nivel 0;
- Această condiție rezultă din faptul că terminalele /INT0 și /INT1 sunt citite doar o singură dată în fiecare ciclu mașină;
- Dacă cererea externă este activă pe nivel, ea trebuie să rămână activă până la începerea execuției rutinei de tratare, după care trebuie dezactivată înainte de încheierea rutinei pentru a evita o nouă tratare a ei.

# Sisteme Încorporate

## Modurile de lucru cu consum redus

- 2 moduri: Idle și Power – Down;
- Utile în aplicații în care consumul este un factor critic;
- Instalarea lor se face prin program, acționând asupra a câte unui rang din registrul PCON ("Power Control Register"):

7							0
SMOD	-	-	-	GF1	GF0	PD	IDL

- PCON.0: IDL: 1 în acest rang instalează modul Idle.
- PCON.1: PD: 1 în acest rang instalează modul Power - Down.
- PCON.2, 3: GF0, GF1: ranguri pentru uz general.
- PCON.7: SMOD: utilizat la interfața serială.

# Sisteme Încorporate

## □ Modul Idle:

- Microcontrolerul intră în acest mod după activarea, de către o instrucțiune, a rangului IDL;
- Oscilatorul, interfața serială, contoarele/ temporizatoarele și sistemul de întreruperi continuă să funcționeze dar tactul nu mai ajunge la CPU. Întregul CPU își păstrează starea;
- ieșirea: prin întrerupere validată sau RST;

## □ Modul Power – Down:

- Este instalat de o instrucțiune ce activează rangul PD;
- Oscilatorul se oprește, ca urmare starea întregului microcontroler rămâne nemodificată (microcontrolerul "îngheață");
- Singura ieșire din acest mod este prin activarea intrării RST;
- În acest mod Vcc poate fi redus la 2 V; trebuie însă avut grijă ca reducerea să se facă după ce microcontrolerul a intrat în modul Power - Down iar activarea lui RST care provoacă ieșirea din modul Power - Down să se facă după ce Vcc a revenit la valoarea de lucru.

## 3.4 Porturile

- Este resursa direct utilizată în aplicațiile de timp real, de monitorizare, comandă și control;
- Microcontrolerul 8051 dispune de 4 porturi bidirectionale, pe 8 biți, notate cu P0, P1, P2 și P3;
- Fiecare port are un circuit de intrare, un registru intern notat tot cu P0, P1, P2 respectiv P3 și un etaj de ieșire;
- Porturile sunt de uz general, fiecare rang poate fi programat independent ca intrare sau ieșire și, în plus, porturile P0, P2 și P3 au și funcțiuni alternative;
- Astfel în ciclurile în care se face acces la memoria externă, liniile portului P0 constituie o magistrală multiplexată de adrese/ date, AD0 - 7, între jumătatea mai puțin semnificativă din magistrala de adrese și cea de date iar liniile portului P2 vor transfera jumătatea mai semnificativă a magistralei de adrese;



# Sisteme Încorporate

- Atunci când se aduce un cod de instrucțiune din memoria de program externă, liniile porturilor P0 și P2 vor fi comandate de PC iar atunci când se transferă o dată cu memoria de date externă liniile porturilor P0 și P2 vor fi comandate de registrul DPTR;
- Dacă adresa datei externe este pe 8 biți, atunci adresa se obține pe liniile portului P0 care va fi comandat de registrul R0 sau R1;
- Rangurile portului P3 au funcțiuni specifice individuale conform tab.:

Rang	Funcția specifică
P3.0	RxD: intrare serială
P3.1	TxD: ieșire serială
P3.2	/INT0: întrerupere externă 0
P3.3	/INT1: întrerupere externă 1
P3.4	T0: intrare de numărare pentru contorul 0
P3.5	T1: intrare de numărare pentru contorul 1
P3.6	/WR: semnal de comandă pentru scrierea în memoria de date externă
P3.7	/RD: semnal de comandă pentru citirea din memoria de date externă

## 3.5 Modul PWM (Pulse Width Modulator)

- Generează ieșiri modulate în durată;
- Facilitate utilă pentru comanda motoarelor de curent continuu;
- Implementat cu un contor/ temporizator pe 32 biți cu un divizor intern de 32 biți;
- Facilitatea PWM poate fi dezactivată;
- Perioada și durata pot fi programate;
- Se poate genera cerere de întrerupere la generarea duratei prescrise;
- Facilitate de control individual al fronturilor ridicător și coborâtor; utilă pentru comanda motoarelor de curent continuu multi – fază care cer 3 impulsuri cu posibilitate de poziționare distinctă a fronturilor ridicător și coborâtor ale fiecăruia;
- Posibilitate de a genera:
  - Ieșire activă la 1 la fiecare început de ciclu;
  - Ieșire activă la 1 sau la 0 în timpul ciclului;
- Registre care stabilesc durata semnalului activ și poziția sa;

# Sisteme Încorporate

## □ Un exemplu de modul PWM:

