

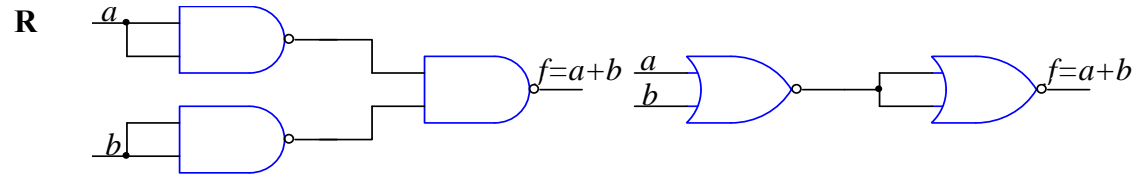
# **DISCIPLINE FUNDAMENTALE**

**-TST-ID-**

**CIRCUITE ELECTRONICE FUNDAMENTALE**  
**ANUL 2, SEMESTRUL 3**

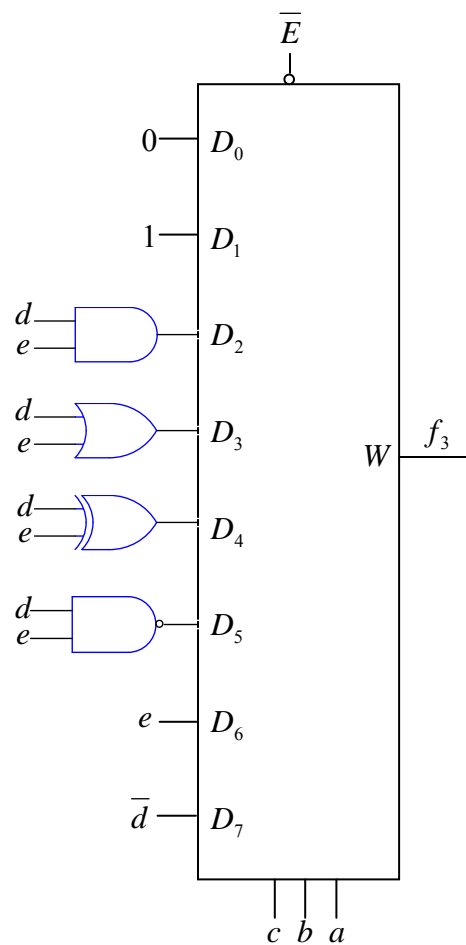
1. Reprezentati o functie logica SAU cu 2 intrari folosind doar:

- a) porti SI-NU cu 2 intrari;
- b) porti SAU-NU cu 2 intrari



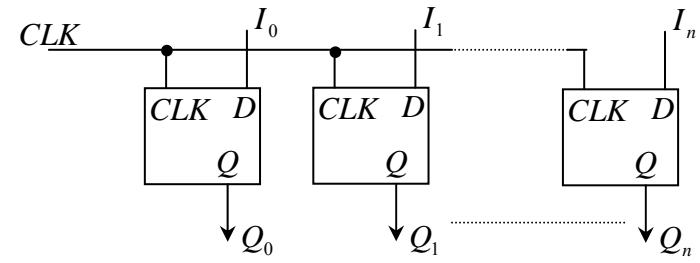
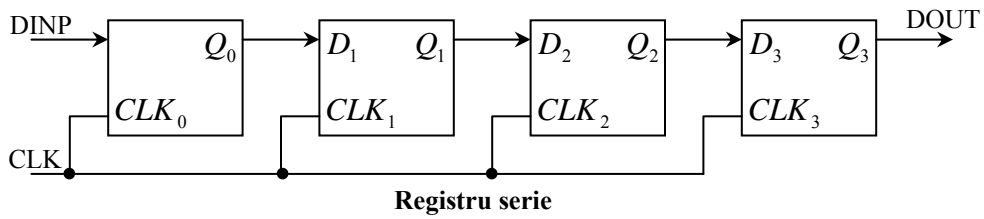
2. Sa se reprezinte funcția logica definită mai jos cu multiplexor cu 3 intrări de adresă si porti logice:

| R        |          |          |          |          |                      |
|----------|----------|----------|----------|----------|----------------------|
| <i>a</i> | <i>b</i> | <i>c</i> | <i>d</i> | <i>e</i> | <i>f<sub>1</sub></i> |
| 0        | 0        | 0        | 0        | 0        | 0                    |
| 0        | 0        | 0        | 0        | 1        | 0                    |
| 0        | 0        | 0        | 1        | 0        | 0                    |
| 0        | 0        | 0        | 1        | 1        | 0                    |
| 0        | 0        | 1        | 0        | 0        | 1                    |
| 0        | 0        | 1        | 0        | 1        | 1                    |
| 0        | 0        | 1        | 1        | 0        | 1                    |
| 0        | 0        | 1        | 1        | 1        | 1                    |
| 0        | 1        | 0        | 0        | 0        | 0                    |
| 0        | 1        | 0        | 0        | 1        | 0                    |
| 0        | 1        | 0        | 1        | 0        | 0                    |
| 0        | 1        | 0        | 1        | 1        | 1                    |
| 0        | 1        | 1        | 0        | 0        | 0                    |
| 0        | 1        | 1        | 0        | 1        | 1                    |
| 0        | 1        | 1        | 1        | 0        | 1                    |
| 0        | 1        | 1        | 1        | 1        | 1                    |
| 1        | 0        | 0        | 0        | 0        | 0                    |
| 1        | 0        | 0        | 0        | 1        | 1                    |
| 1        | 0        | 0        | 1        | 0        | 1                    |
| 1        | 0        | 0        | 1        | 1        | 0                    |
| 1        | 0        | 1        | 0        | 0        | 1                    |
| 1        | 0        | 1        | 0        | 1        | 1                    |
| 1        | 0        | 1        | 1        | 0        | 1                    |
| 1        | 0        | 1        | 1        | 1        | 0                    |
| 1        | 1        | 0        | 0        | 0        | 0                    |
| 1        | 1        | 0        | 0        | 1        | 1                    |
| 1        | 1        | 0        | 1        | 0        | 0                    |
| 1        | 1        | 0        | 1        | 1        | 1                    |
| 1        | 1        | 1        | 0        | 0        | 1                    |
| 1        | 1        | 1        | 0        | 1        | 1                    |
| 1        | 1        | 1        | 1        | 0        | 0                    |
| 1        | 1        | 1        | 1        | 1        | 0                    |



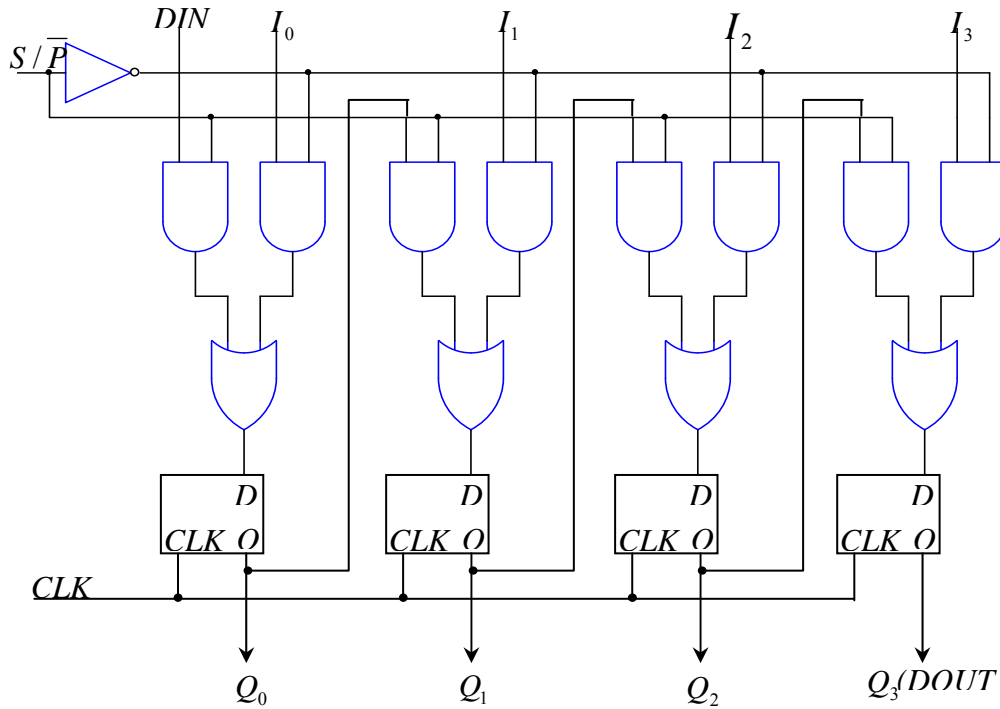
**3. Reprezentati schemele pentru un registru serie si unul paralel pe 4 biti**

R:



**4. Reprezentati schema pentru un registru serie-paralel pe 4 biti**

R:



**5. Completați mai jos numele circuitelor care corespund definițiilor:**

a) Circuitul logic combinațional care asigură, direct sau indirect, însumarea a două numere binare cu câte un bit fără a lua în considerare transportul de la bitul cu ponderea imediat inferioară este denumit

.....  
R: *semisumator*

b) .....este un circuit logic combinational, integrat pe scară medie, obținut printr-o extensie de tip paralel a unor porți logice și are “n” intrări de adresare și  $2^n$  iesiri. Din punct de vedere funcțional activează una din cele  $2^n$  ieșiri ale sale, funcție de codul aplicat la cele “n” intrări.

R: *decodificatorul*

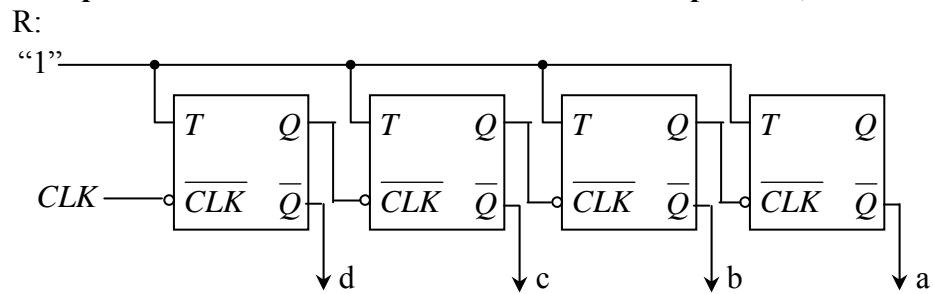
c) ..... se definește ca fiind circuitul logic combinațional care asigură, direct sau indirect, efectuarea însumării a două numere binare ținând cont de un eventual transport inițial.

R: *sumatorul*

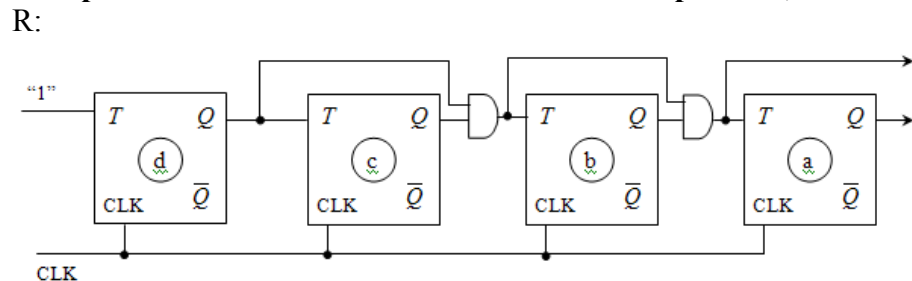
d) Din punct de vedere funcțional ..... este un selector ce conectează la ieșire intrarea adresată. Circuitul are în general  $2^n$  intrări de date și “n” intrări de adresă. Valoarea ieșirii este determinată de valoarea intrării selectate prin adresă.

R: *multiplexorul*

**6. Reprezentați schema unui numarator asincron pe 4 biti, folosind bistabile T.**



**7. Reprezentați schema unui numarator sincron pe 4 biti, folosind bistabile T**



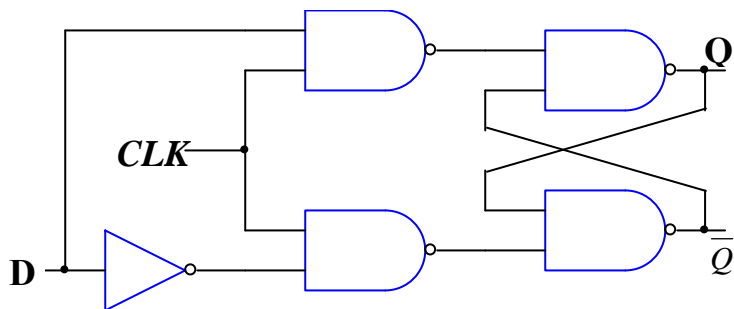
8. Completati tabellele de functionare ale urmatoarelor bistabile: JK, RS.

| J | K | ST.PREZ. | ST.VIIT.  |
|---|---|----------|-----------|
|   |   | $Q_n$    | $Q_{n+1}$ |
| 0 | 0 | $Q_n$    |           |
| 0 | 1 | $Q_n$    |           |
| 1 | 0 | $Q_n$    |           |
| 1 | 1 | $Q_n$    |           |

| $\bar{S}$ | $\bar{R}$ | ST.PREZ. | ST.VIIT.  |
|-----------|-----------|----------|-----------|
|           |           | $Q_n$    | $Q_{n+1}$ |
| 0         | 0         | $Q_n$    |           |
| 0         | 1         | $Q_n$    |           |
| 1         | 0         | $Q_n$    |           |
| 1         | 1         | $Q_n$    |           |

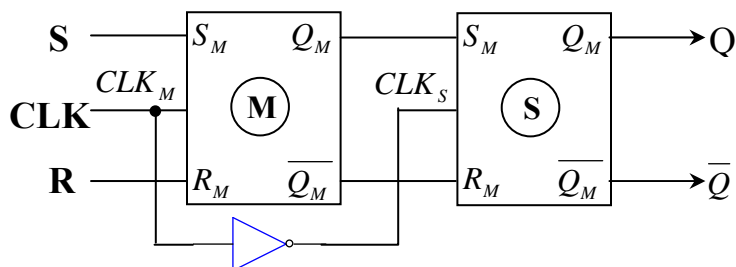
9. Reprezentati schema cu porti logice pentru un bistabil RS sincron

R:



10. Reprezentati schema pentru o structura de tip Master-Slave

R:



**CIRCUITE INTEGRATE DIGITALE**  
**ANUL 2, SEMESTRUL 3**



# 1. De cine depinde puterea disipată de un circuit digital CMOS?

## Manual pagina 62, subcapitol Puterea disipată de o poartă CMOS

### Capitolul 4

#### Familia de circuite integrate digitale CMOS

##### Curenții de intrare

$$I_{iLMax} = I_{iHMax} = 0 \text{ (0,1 ... 1 } \mu\text{A)}.$$

##### Factorul de brașament

Datorită valorii mici a curentului de intrare (sub 1  $\mu\text{A}$ ), valoarea factorului de brașament  $N = N_L = N_H$  este foarte mare (pentru curenți maximi de ieșire de câțiva miliamperi). Cele mai multe circuite logice din familia CMOS se fabrică cu un curent de ieșire  $I_0 = 3...4 \text{ mA}$ , deci au factorul de brașament foarte mare în regim static.

În practică factorul de brașament este limitat de valoarea  $C_p$  a cărei componentă principală este  $\Sigma C_i$ . Creșterea  $C_p$  duce la înrăutățirea comportamentului dinamic al circuitului ( $C_i = 5 - 15 \text{ pF}$ ). În concluzie, factorul de brașament se limitează din cauza funcționării în regim dinamic la o valoare maximă de 50.

##### Curentul de alimentare

Curentul de alimentare în regim static este neglijabil ( $\mu\text{A}$ ) iar în regim dinamic depinde de frecvență,  $C_p$  și  $V_{DD}$  (vezi relația 4.1).

##### Puterea disipată de o poartă CMOS

Puterea medie este specificată pentru un semnal dreptunghiular cu factor de umplere 50% aplicat la intrarea circuitului.  $P_D$  este specificată în foile de catalog ale diverșilor producători. Studiind graficul din figura 4.16 se observă că la frecvențe de până la circa 1 MHz, un circuit CMOS disipă o putere mai mică decât unul TTL LS; peste această limită, mai avantajoase sunt circuitele LS.

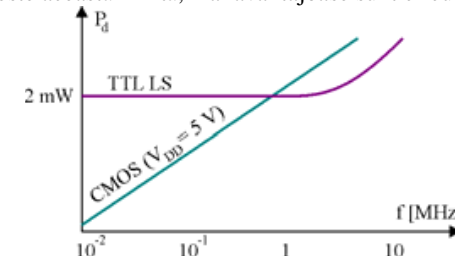


Figura 4.16. Comparație între puterea disipată de un circuit CMOS și unul TTL LS.

##### Timpul de propagare

Timpul de propagare se definește similar cu cel de la circuitele TTL. În acest caz  $U_L = 0$  și  $U_H = V_{DD}$ . Punctele de măsură sunt specificate tot la 50% din nivelul  $U_H$ . În cazul seriei 4000,  $t_{pHL}$  și  $t_{pLH}$  sunt egale, iar  $t_p = 40 \dots 100 \text{ ns}$  (depinde de tensiunea de alimentare, fabricant, etc).

## 2. Decodificator. Rol, functionare, tabel de adevar.

### Manual pagina 71-73, subcapitol Decodificatorul

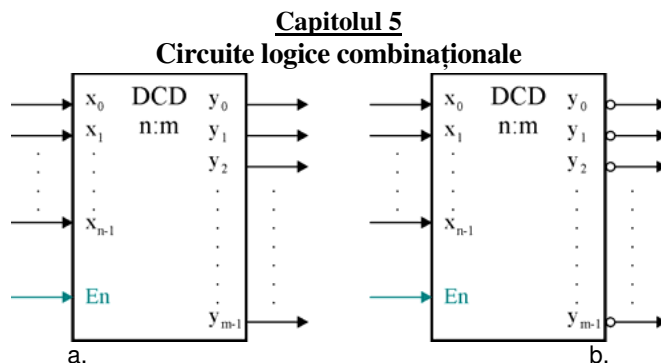
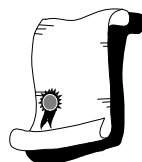
#### Capitolul 5 Circuite logice combinaționale

##### Subiecte

- 5.0. Introducere
- 5.1. Decodificatorul
- 5.2. Demultiplexorul
- 5.3. Multiplexorul
- 5.4. Comparatorul numeric

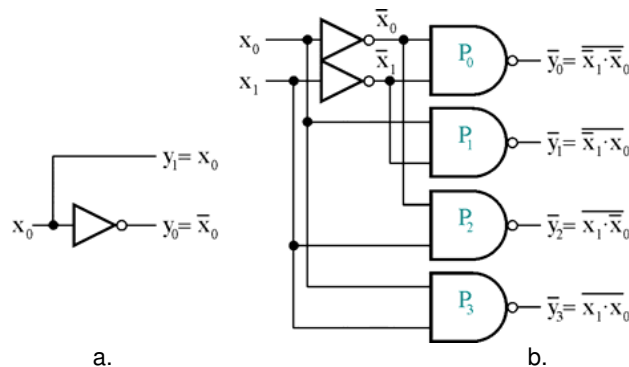
##### Evaluare:

1. Răspunsuri la întrebările finale
2. Discuție pe tema: "Utilizarea decodificatorului ca demultiplexor"



**Figura 5.1.** Schema bloc pentru un decodificator  $n:m$  cu validare,  
a – ieșirile active SUS, b – ieșirile active JOS.

Cel mai simplu DCD are o intrare și o ieșire, fiind realizat cu un inversor (figura 5.2). Un DCD 2:4 necesită 4 porți ȘI-NU și două inversoare, ieșirile fiind active JOS.

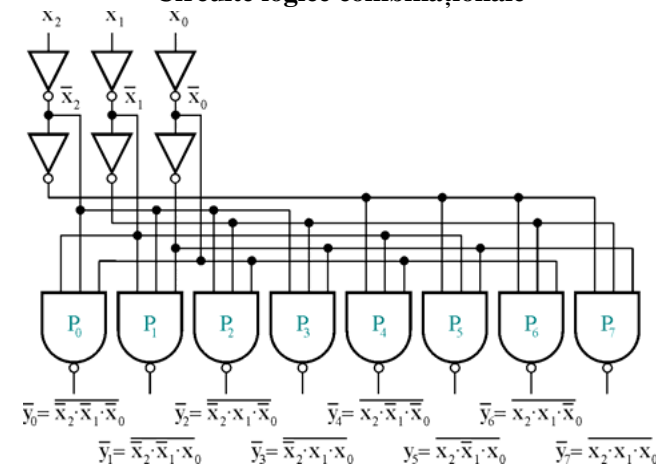


**Figura 5.2.** DCD simple – schema electrică,  
a – DCD 1:2, ieșiri active SUS, b – DCD 2:4, ieșiri active JOS.

Schema electrică pentru un DCD 3:8 necesită 8 porți ȘI-NU cu câte 3 intrări (figura 5.3). Intrările se aplică prin perechi de inversoare pentru a asigura ca fiecare intrare să reprezinte o singură sarcină (TTL).

Schema se poate completa cu un circuit de validare (figura 5.4). Dacă circuitul nu este validat, toate ieșirile DCD sunt în starea 1. Pentru validare este necesar ca  $E_2 = 1, E_{1A} = E_{2B} = 0$ .

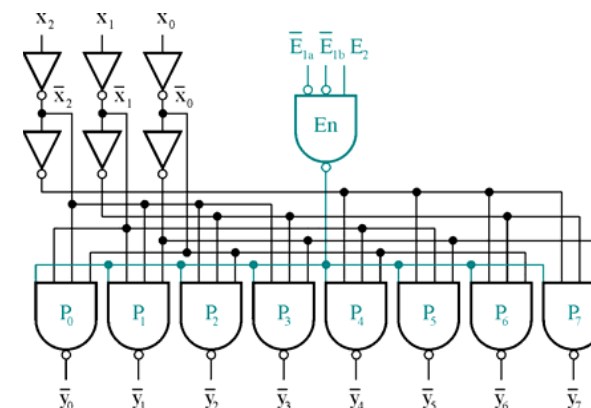
#### Capitolul 5 Circuite logice combinaționale



**Figura 5.3.** Structura unui DCD 3:8.

Fiecărei ieșiri îi corespunde un circuit ȘI-NU, ceea ce face ca ieșirile circuitului să fie active pe 0. Acest lucru înseamnă că ieșirea activată este pe 0 iar toate celelalte ieșiri sunt pe 1. De exemplu: pentru  $x_0 = 1, x_1 = 0, x_2 = 1$ , ieșirea  $\bar{y}_5 = E_2 \cdot \bar{E}_{1a} \cdot \bar{E}_{1b} \cdot (x_2 \cdot \bar{x}_1 \cdot x_0)$  este pe 0 și toate celelalte sunt 1.

Decodificatorul din figura 5.4 realizat în tehnologie TTL (74LS138) este foarte răspândit în aplicații datorită versatilității oferite de validarea multiplă.



**Figura 5.4.** Un DCD 3:8 foarte răspândit, 74LS138.

### 3. Regulile de utilizare ale circuitelor integrate CMOS.

## Manual pagina 57-60, subcapitol Reguli de utilizare ale circuitelor CMOS

1/2

#### Capitolul 4

#### Familia de circuite integrate digitale CMOS

$$P_{d2} = f C_p V_{DD}^2; C_p = C_0 + C_{con} + \sum_{k=1}^N C_{ik}, \text{ unde } C_0 \text{ se dă în}$$

catalog,  $C_{con}$  reprezintă capacitatea conexiunilor și  $C_{ik}$  este dat în catalog pentru fiecare intrare (valorile tipice fiind cuprinse între 5 și 15 pF).

- Explicați importanța zonei de conducție simultană din figura 4.4.
- Explicați comportamentul inversorului CMOS în funcție de tensiunea de alimentare, figura 4.9.
- Prezentați componentele puterii disipate de inversorul CMOS.



#### 4.3. Reguli de utilizare a circuitelor CMOS

1. Nici o intrare a unui circuit logic CMOS nu se lasă flotantă, ci se conectează la un potențial bine stabilit:  $U_L$  sau  $U_H$  în funcție de tipul circuitului.
  - a. O posibilitate de conectare pentru porțile ȘI-NU, respectiv ȘI este polarizarea cu o tensiune  $V_{DD}$ , în acest caz rezistența  $R_p$  utilizată la circuitele TTL nu mai este necesară.
  - b. La circuitele SAU, respectiv SAU-NU polarizarea se realizează prin legare directă la masă (figura 4.10).
  - c. Intrările nefolosite se pot lega la alte intrări folosite (figura 4.11), cu dezavantajul legat de multiplicarea capacității de intrare  $C_i$  (crește proporțional și curentul de intrare, dar rămâne la o valoare neglijabilă).

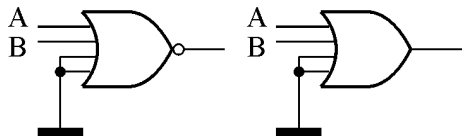


Figura 4.10. Pentru porțile SAU-NU, SAU, intrările nefolosite se conectează la masă sau  $U_L$ .

#### Capitolul 4

#### Familia de circuite integrate digitale CMOS

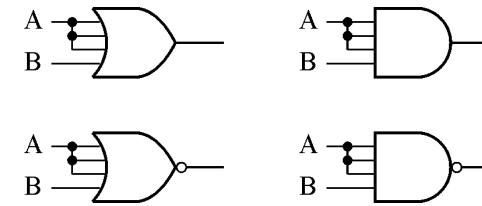


Figura 4.11. Indiferent de tipul porții, intrările nefolosite se pot lega la alte intrări.

2. Intrările porților nefolosite pot fi conectate ori la masă, ori la  $V_{DD}$ , puterea consumată fiind aceeași (neglijabilă).
3. Este interzisă interconectarea ieșirilor a două sau mai multe circuite logice, dacă există posibilitatea ca aceste ieșiri să ajungă la niveluri logice diferite. În figura 4.12 este prezentată o situație în care ieșirile pot fi interconectate – legând în paralel atât intrările cât și ieșirile unor porți din aceeași capsulă.

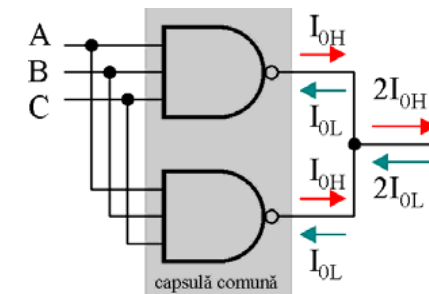


Figura 4.12. Posibilitate de interconectare a ieșirilor a două circuite CMOS.

4. Niciodată ieșirile circuitelor logice nu se conectează direct la masă sau  $V_{DD}$ .
5. Cerințele de decuplare ale circuitelor integrate CMOS sunt mult diminuate față de omoloagele TTL datorită consumului de curent mai redus. Un singur condensator de decuplare de 100 nF la fiecare rând de 10 – 15 circuite CMOS și un condensator electrolitic de 10 ... 100  $\mu$ F pentru întreaga placă sunt de obicei suficiente.

### 3. Regulile de utilizare ale circuitelor integrate CMOS.

## Manual pagina 57-60, subcapitol Reguli de utilizare ale circuitelor CMOS

2/2

#### Capitolul 4

#### Familia de circuite integrate digitale CMOS

6. Există cerințe speciale referitor la manipularea sau stocarea acestor circuite derivate din dorința de a minimiza efectele descărcărilor electrostatice (ESD – *electrostatic discharge*).

Toate circuitele electronice sunt susceptibile la distrugere datorită descărcărilor electrostatice. Corpul omenesc se poate ușor încărca electrostatic la potențiale de peste 30.000 V, prin simpla deplasare pe un covor, purtarea unui plover sau mângâierea unei pisici. Prin simpla atingere a unui circuit electronic sarcinile astfel stocate sunt în contact direct cu circuitul. Tranzistoarele și circuitele integrate CMOS sunt în primul rând sensibile la sarcini electrostatice datorită impedanței mari de intrare și a stratului subțire de dioxid de siliciu care se poate astfel ușor străpunge. Rezultatul străpungerii este ireversibil și circuitul sau dispozitivul este distrus.

Producătorii de dispozitive, circuite și echipamente electronice acordă problemelor ESD o atenție sporită. Chiar dacă marea majoritate a circuitelor MOS moderne au rețele de protecție formate din rezistoare și diode (asemănătoare celor din figura 4.3), următoarele măsuri de prevedere sunt general valabile:

- Circuitele integrate MOS se păstrează în fișele speciale antistatice, în folii de aluminiu sau materiale speciale conductoare. Aceasta conduce la egalizarea potențialelor tuturor pinilor și prin urmare nu pot apărea tensiuni periculoase între pini.
- După extragerea circuitului din materialul antistatic, acesta se va monta imediat pe placa de circuit imprimat. Se va evita atingerea pinilor cu mâna.
- În echipament intrările nefolosite ale circuitelor MOS nu se lasă neconectate, deoarece acestea tind să acumuleze sarcini electrostatice.
- La transport conectorii plăcilor se scurtcircuitează, iar plăcile se transportă în folii antistatice conductoare. Se evită atingerea conectorilor cu mâna.
- La lipire operatorul folosește o brățară specială metalică legată la pământare prin intermediul unei rezistențe de  $1M\Omega$  pentru a descărca eventualele sarcini electrostatice. Rezistența elimină riscul electrocutării dacă din accident sunt atinse puncte aflate la un potențial ridicat.
- Operatorul uman va purta un echipament adecvat (de exemplu o pereche de accesorii conductoare peste pantofi pentru a micșora rezistența de contact la pământ).



- **Explicați precauțiile suplimentare de utilizare a circuitelor CMOS față de cele TTL.**
- **Comentați din experiența proprie 5 măsuri ESD proprii unei producții de echipamente electronice moderne.**

#### Capitolul 4

#### Familia de circuite integrate digitale CMOS

- g. Șasiul tuturor echipamentelor, vârful letconului sau stației de lipit se conectează la pământare pentru a preveni acumularea de sarcini electrostatice.

#### 4.4. Parametrii circuitelor CMOS din seria 4000

Niveluri de tensiune garantate (pentru  $V_{DD} = 5 V$ )

$U_{iLMax} = 1,5 V$ ;  $U_{oLMax} = 0,05 V$ ;  $U_{iHMax} = 3,5 V$ ;  $U_{oHmin} = 4,95 V$ . Aceste valori sunt utile pentru a putea determina marginea de zgomot.

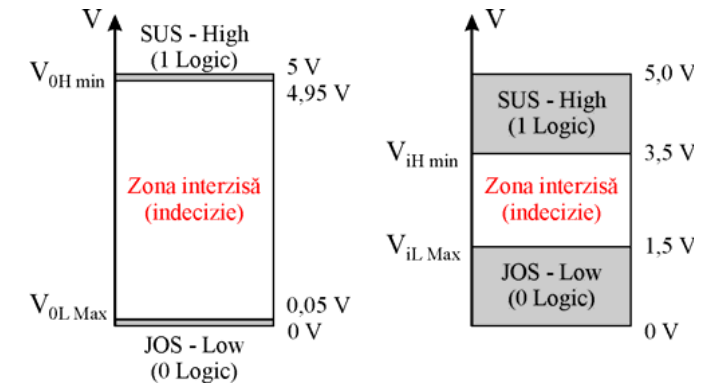


Figura 4.13. Niveluri de tensiune pentru seria CMOS 4000.

Tabelul 4.1

Nivelurile de tensiune pentru seria CMOS 4000, alimentare la 5 V

| Tensiunea | min [V]               | tipic [V] | maxim [V]             |
|-----------|-----------------------|-----------|-----------------------|
| $V_{OH}$  |                       | 4,95      |                       |
| $V_{OL}$  |                       | 0,05      |                       |
| $V_{IH}$  | $70\% V_{DD} = 3,5 V$ |           |                       |
| $V_{IL}$  |                       |           | $70\% V_{DD} = 1,5 V$ |

27 - 2016

# 4. Numarator binar asincron: schema, diagrame de functionare.

## Manual pagina 117-118, subcapitol Numarator binar asincron direct

### Capitolul 7

#### Registre și numărătoare

Inițializarea numărătorului se face de obicei prin intermediul semnalului de ștergere (*Reset* sau *Master Reset*), activ SUS sau JOS (nMR). Ștergerea se poate face *asincron*, dacă survine independent de starea semnalului de tact și de îndată ce semnalul MR este activ sau *sincron*, în care ștergerea se face numai după frontul activ al semnalului de tact (crescător sau descrescător).

Anumite numărătoare poate fi inițializate în orice stare dacă sunt prevăzute cu posibilitatea încărcării paralele, folosind o linie adițională notată LD (LOAD), activă SUS sau JOS (nLD). Încărcarea se poate face *asincron*, dacă survine îndată ce semnalul LD este activ sau *sincron*, în care încărcarea se face numai după frontul activ al semnalului de tact (crescător sau descrescător).

#### 7.4. Numărătoare asincrone

##### 7.4.1. Numărătorul asincron binar direct

Pentru  $n = 4$  bistabile numărul stărilor distincte (modulul numărătorului) binar este  $m = 2^4 = 16$ . Impulsurile de tact se aplică primului bistabil, următoarele bistabile având fiecare ca semnal de tact ieșirea  $Q$  a bistabilului anterior. Bistabilele funcționează în regim de divizor de frecvență. Divizarea frecvenței de tact depinde de poziția în numărător a bistabilului la ieșirea căruia se culege semnalul.

În figura 7.25 s-a reprezentat un ciclu complet de funcționare și parțial începutul celui de-al doilea ciclu. Ieșirile numărătorului evoluează în sens crescător (direct), cu fiecare impuls de tact aplicat valoarea la ieșire crește cu o unitate. Numărătorul prezentat este modulo 16 (are 4 bistabile). Cel de-al 16-lea impuls de tact încheie ciclul, el aducând numărătorul pe zero. Cel de-al 17-lea este primul impuls de tact din cel de-al doilea ciclu.

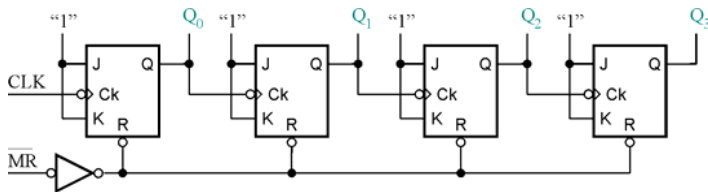


Figura 7.24. Numărător binar asincron direct – schema electrică.

La un moment dat codul binar de ieșire corespunde numărului de impulsuri de tact aplicate în ciclul respectiv, în aceasta constând

### Capitolul 7

#### Registre și numărătoare

practic funcția de numărare. Citind ieșirile după cel de-al 11-lea impuls de tact, rezultă  $Q_3Q_2Q_1Q_0 = 1011$ , care este tocmai corespondentul în binar al numărului zecimal 11.

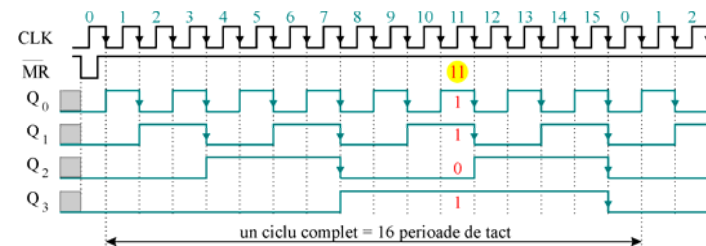


Figura 7.25. Numărător binar asincron direct – diagrame de semnal.

Pentru extinderea capacității de numărare se pot conecta mai multe numărătoare în cascadă prin conectarea ieșirii  $Q_3$  la intrarea de tact a următorului numărător.

##### 7.4.2. Numărătorul asincron binar invers

În anumite aplicații este necesară utilizarea unor numărătoare care să poată număra și în sens invers, adică numărătorul să își micșoreze conținutul cu câte o unitate la fiecare impuls de tact. În acest scop semnalul de tact a bistabilului următor nu se mai culege de la ieșirea  $Q$  a bistabilului anterior, ci de la ieșirea  $nQ$ .

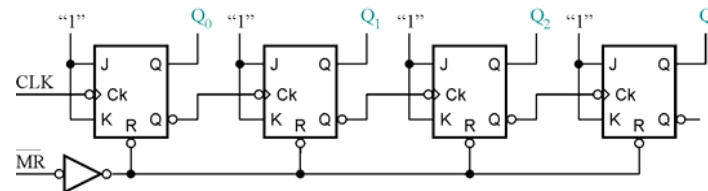


Figura 7.26. Numărător binar asincron invers – schema electrică.

Când  $Q$  trece din 1 în 0,  $nQ$  trece din 0 în 1, (bistabilul următor nu comută), dar când  $Q$  trece din 0 în 1,  $nQ$  trece din 1 în 0 și determină comutarea bistabilului următor. Acest lucru poate fi verificat în tabelul 7.3.

5. Ce este un divizor programabil de frecvență?

Orice numărător se comportă ca un divizor de frecvență. Un divizor *programabil* adaugă facilitatea modificării modulului de numărare, de obicei utilizând numărătoare cu posibilitatea de încărcare și conectând ieșirea de transport la intrarea de încărcare. Raportul de divizare se alege setând corespunzător presetarea de încărcare (detaliați și exemplificați).

# 6. Cum se poate obține un divizor de frecvență cu 2 dintr-un bistabil D?

## Manual pagina 100-101, subcapitol Bistabilul de tip T

### Capitolul 6 Circuite basculante bistabile

#### 6.4. Bistabilul de tip T

Bistabilul T (*toggle*) se caracterizează prin faptul ca el este forțat să funcționeze doar în două situații ce corespund la două linii ale tabelului 6.3;

$$J_n = K_n = 1$$

$$J_n = K_n = 0$$

Intrarea T a unui astfel de bistabil se obține prin interconectarea intrărilor J și K. Pentru realizarea bistabilului de tip T se folosesc numai bistabile JK-MS.

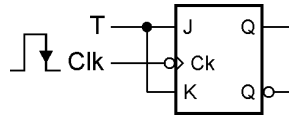


Figura 6.25. Bistabil T din JK.

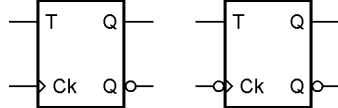


Figura 6.26. Bistabil T - simbol.

La funcționarea secvențială:

- dacă T este permanent egal cu 1  $\Rightarrow Q_{n+1} = \overline{Q}_n$ ;
- dacă T este permanent egal cu 0  $\Rightarrow Q_{n+1} = Q_n$

Ecuția caracteristică se deduce din:

$$Q_n = J_n \overline{Q}_n + \overline{K}_n Q_n$$

$$Q_{n+1} = T_n \overline{Q}_n + \overline{T}_n Q_n$$

Tabelul 6.6

Funcționarea bistabilului T

| $T_n$ | $Q_{n+1}$        |
|-------|------------------|
| 0     | $Q_n$            |
| 1     | $\overline{Q}_n$ |

### Capitolul 6 Circuite basculante bistabile

Dacă T este permanent 1,  $Q_{n+1} = \overline{Q}_n$ , bistabilul basculează la fiecare impuls de tact și se poate folosi ca divizor de frecvență a impulsurilor de tact raportat la ieșirea  $Q_n$

$$f_Q = f_{CK}/2$$

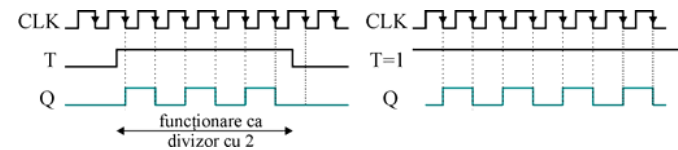


Figura 6.27. Funcționarea bistabilului T (stânga) și ca divizor de frecvență cu 2 (dreapta).

- Explicați cum se poate transforma un bistabil D în unul T. Dar invers?
- Care este principala aplicație a bistabilului T ?



#### ÎNTREBĂRI FINALE

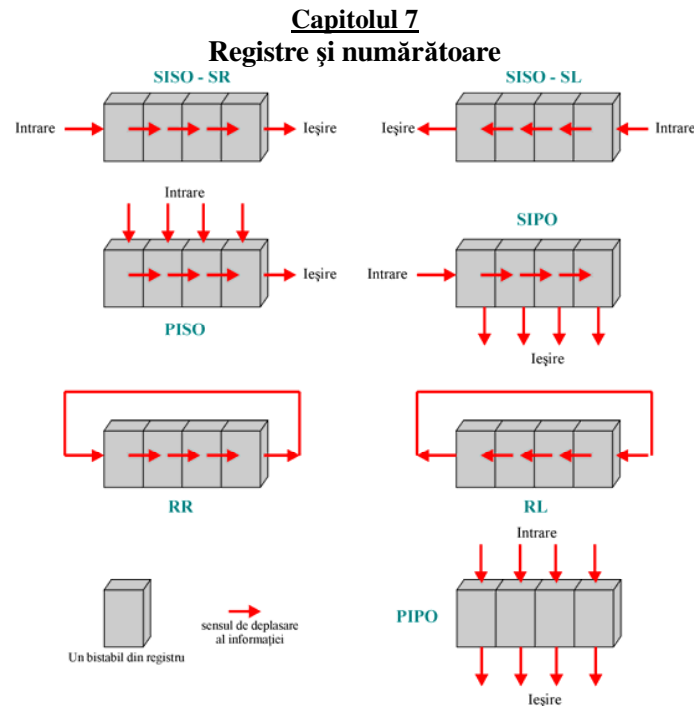
1. Pentru toate tipurile de bistabile studiate, alcătuiți un tabel care să conțină ecuația caracteristică și variante de implementare TTL și CMOS.
2. Ce se întâmplă la cascada a două bistabile de tip T cu T=1?
3. Căutați foaia de catalog și descrieți funcționarea unui latch octal.

#### REZUMAT

- Principalele tipuri de bistabile sunt RS, JK, D și T.
- Intrările de tip RS prezintă combinația interzisă R=S=1 pentru toate tipurile de bistabile.
- Bistabilele pot fi convertite dintr-un tip în alt tip.
- Datele și tactul trebuie să respecte restricțiile temporale impuse de timpul de stabilire și de menținere.



## 7. De câte bistabile este nevoie pentru a implementa un numărător modulo 2008? log22008, cu explicare.



**Figura 7.1.** Registre de deplasare și memorare – principiu de funcționare.

**Numărătoarele** sunt circuite care evoluează periodic (ciclic) între anumite stări. Numărul stărilor distincte dintr-un ciclu se numește modulul numărătorului și se notează cu  $m$ . Numărătoarele în inel sau Johnson, realizate cu registre de deplasare formate din bistabile D, studiate anterior aveau modulul  $m = n$  respectiv  $m = 2n$ ; ( $n$  era numărul de bistabile a registrului). În acest caz  $m \leq 2^n$ .

Se pune problema obținerii cu același număr de bistabile  $n$  a unui număr cât mai mare de stări distincte în cadrul unui ciclu, adică a măririi modulului  $m$ . Numărul  $n$  de bistabile necesare pentru realizarea unui numărător modulo  $m$  este  $n \geq \log_2 m$ .

### 7.1. Registre elementare

#### 7.1.1. Registrul SISO

Acest tip de registru respectă structura din figura 7.2, formată din  $n$  bistabile de tip D. Intrarea D a bistabilului  $k + 1$  este conectată la ieșirea Q a bistabilului  $k$ . O intrare asincronă activă pe '0' notată  $nMR$



## 8. Ce este o linie partajată?

### Manual pagina 41, subcapitol Circuite cu trei stări din familia TTL, figura 3.16 – Linia partajată, cu explicații.

#### Capitolul 3

#### Familia de circuite integrate digitale TTL

Curentului de ieșire al circuitului este egal cu curentul rezidual al tranzistorului  $T_4$  respectiv  $T_3$ . Pentru seria standard  $I_{OZM} = \pm 40\mu A$ .

Circuitul cu trei stări se fabrică de cele mai multe ori ca inversor cu 3 stări, operator neinvertor cu 3 stări și poartă ȘI-NU cu 3 stări. Reprezentarea simbolică pentru circuitele cu 3 stări este cea din figura 3.15.

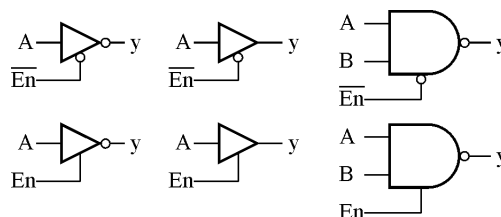


Figura 3.15. Variante de fabricare pentru circuitele cu trei stări.

Avantajul major al acestor circuite este posibilitatea interconectării ieșirilor, legătură care se numește *linie partajată*. În sistemele de calcul mai multe astfel de linii partajate sunt grupate în *magistrale* de semnal: de date, de adrese, de control sau combinații ale acestora.

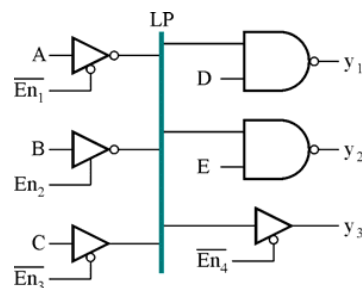


Figura 3.16. Linia partajată.

La linia partajata (figura 3.16) se pot conecta numai *ieșirile* unor circuite cu 3 stări și orice combinație de intrări de circuite logice.

# 9. Ce tip de bistabile stau la baza realizării numărătoarelor?

## Manual pagina 116, subcapitol Clasificarea numărătoarelor

### Capitolul 7

#### Registre și numărătoare

și  $m$  = numărul stărilor distincte ale circuitului, în cadrul ciclului de funcționare  $m$  purtând și denumirea de modulul numărătorului.

Este posibilă numărarea impulsurilor de tact deoarece fiecărui impuls de tact îi corespunde un cod binar, urmărind valorile de la ieșirile  $Q$ . Astfel de circuite pot fi utilizate pentru comanda succesivă întrețesută a  $n$  elemente actuator.



- Din experiența proprie prezentați utilizări pentru conversia serie paralel și paralel serie a informației.
- Precizați minim două aplicații pentru întârzierea cu circuite numerice a informației.
- Căutați în bibliografie și conspectați schema unui numărător în inel.

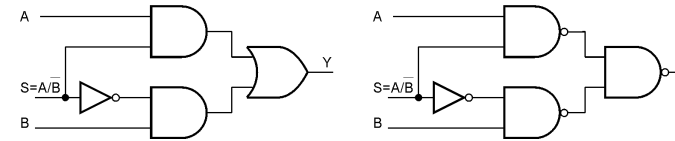
#### 7.3. Clasificarea numărătoarelor

1. După modul de aplicare a impulsurilor de tact
  - asincrone – tactul se aplică numai bistabilului celui mai puțin semnificativ, următoarele bistabile au semnalul de tact provenit de la ieșirea  $Q$  sau  $/Q$  a bistabilului precedent;
  - sincrone – impulsul de tact se aplică simultan tuturor bistabilelor.
2. După modul
  - Binare  $m = 2^n$ ;
  - Zecimale sau decadice  $m = 10$ ;
  - Modulo  $p \neq 2^n$ .
3. După sensul de numărare
  - directe – acestea numără într-un singur sens în sens direct adică crescător;
  - inverse - acestea numără în sens descrescător
  - reversibile - numără în ambele sensuri adică atât în sens direct cât și în sens invers.

Un numărător care evoluează ciclic prin exact 10 stări se numește zecimal sau decadic. Dacă cele 10 stări sunt 0, 1, 2, ..., 9 atunci el se mai numește numărător BCD (*Binary Coded Decimal*).

Bistabilele utilizate în construcția numărătoarelor sunt de tip T realizate de obicei din bistabile JK sau D-MS, cu  $T = 1$  permanent sau uneori cu validarea accesibilă în exterior.

**Capitolul 7**  
**Registre și numărătoare**

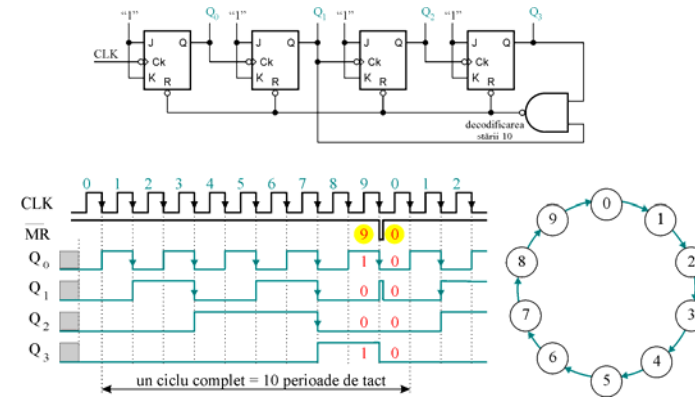


**Figura 7.29.** Variante de realizare a multiplexorului 2:1.

**10. Care este principiul de realizare a numărătoarelor/divizoarelor modulo n?**

Manual pagina 120, subcapitol Numărătoare asincrone zecimale, se alege o poartă ȘI-NU care să decodifice corespunzător starea binară pentru n; pentru memorarea impulsului de ștergere se poate eventual adăuga un latch cu porți (setat de poarta ȘI-NU și șters de semnalul de tact negat)

**7.4.4. Numărătoare asincrone zecimale**



**Figura 7.30.** Numărător asincron zecimal direct: schemă electrică, diagrame de semnal, diagrama de stări.

Numărătorul zecimal din figura 7.30 este des întâlnit în aplicațiile practice. El se bazează pe structura numărătorului asincron binar din figura 7.24 la care se adaugă un circuit de decodificare a stării 10, format dintr-o poartă ȘI-NU. Când numărătorul ajunge în starea 10, ieșirea porții ȘI-NU trece în 0 logic, determinând trecerea numărătorului în starea 0, după care ciclul de funcționare se repetă.

**7.4.5. Determinarea frecvenței maxime de operare pentru numărătoarele asincrone**

Principalul avantaj al numărătoarelor sincrone îl constituie simplitatea arhitecturii, aspect contrabalansat de dezavantajul major al unei frecvențe maxime de operare reduse, datorată propagării succesive a semnalului de tact. Întârzierea produsă de un bistabil este egală cu  $t_{pCLR \rightarrow Q}$ . Pentru stabilirea frecvenței, maxime de operare

**CIRCUITE INTEGRATE ANALOGICE**  
**ANUL 2, SEMESTRUL 4**

### **1. Ce se înțelege prin echilibrarea unui AO si care este scopul corectiei caracteristicii sale de frecventa?**

Asigurarea lui 0V la iesirea amplificatorului (folosind o retea rezistiva alimentata de la +/-E si conectata la anumiti pini ai integratului) atunci cand intrarile sunt conectate la masa.

Asigurarea stabilitatii amplificatorului pentru orice amplificare cu reactie.

### **2. Clasificați și exemplificați erorile AO.**

Erori de regim static sau de curent continuu (cauzate de offseturi sau decalaje initiale de tensiune si current si de derivatele lor termice)

Erori de regim dinamic (cauzate de banda de frecventa limitata a amplificatorului si de zgomotele proprii ale acestuia)

Erori cauzate de idealizarea functiei de transfer (cauzate de amplificarea finita a AO)

### **3. Care sunt proprietățile AO ideal?**

- amplificare de tensiune infinită,
- rezistență de intrare diferențială infinită,
- rezistență de ieșire nulă,
- curent de polarizare (intrare) nul,
- bandă de frecvență foarte largă (astfel încât nu intervine în funcționarea circuitului),
- decalaje inițiale, derivate, zgomot nule,
- factor de rejecție a semnalului comun infinit,
- factor de rejecție a variației tensiunilor de alimentare infinit.

#### 4. Specificați și prezentați câțiva parametri ai AO care caracterizează funcționarea sa în regim dinamic.

- **amplificarea de tensiune**, fără reacție, la semnal mare, în condiții de  $\pm E$  și  $R_S$  precizate. Valoarea amplificării este în mod obișnuit 100.000...300.000;
- **banda de frecvență la amplificarea unitară**, ce reprezintă frecvența de tăiere a axei logf de către caracteristica de frecvență a amplificatorului fără reacție corectat (sau frecvența de tăiere a amplificatorului cu reacție în regim de repetor, când  $A_{ur}=1$ , respectiv când  $20 \log A_{ur}=0$ );
- **viteza maximă de creștere a tensiunii de ieșire**, „slew-rate”, notată SR, pentru semnal mare. La unele amplificatoare (cu corecție externă) se dă viteza maximă realizabilă pentru diferite corecții (care se aleg în funcție de amplificarea cu reacție dorită).

Pentru ca un semnal sinusoidal cu anumită amplitudine să sufere distorsiuni mici - 1% - la trecerea prin amplificator, trebuie ca mărimea SR să aibă o valoare:

$$SR \geq 2\pi f_{\max} (u_{em})_{\max},$$

iar pentru distorsiuni mai mici, coeficientul 2 se înlocuiește cu unul mai mare (3...4 pentru 0,5% sau chiar 8...10 pentru distorsiuni neglijabile). Deseori se dă în catalog caracteristica  $(u_{em})_{\max} = F(f_{\max})$  rezultată din relația de mai sus, pentru semnal sinusoidal cu distorsiuni 1% și o anumită corecție (deci o anumită viteză SR), (fig.2.9). Abaterea de la forma de variație hiperbolică este datorată atingerii excursiei maxime de tensiune la ieșirea AO impusă de alimentare și sarcină.

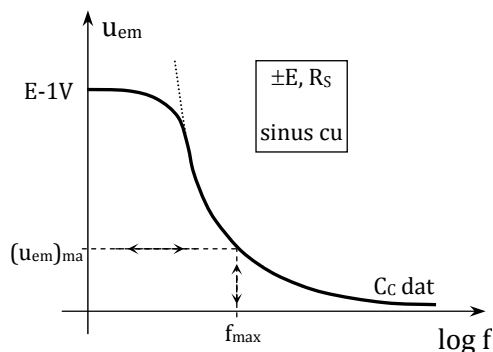


Fig. 2.9. Amplitudinea maximă a semnalului sinusoidal de la ieșirea AO în funcție de frecvență, în condițiile în care sunt precizate  $\delta$

## 5. Amplificatorul de masura (clasic) cu 3 amplificatoare operationale.

Schema clasică de amplificator de instrumentație se realizează cu 3 AO distincte, din care primele două trebuie să fie de precizie, sau se poate găsi sub formă de circuit integrat monolitic la care se atașează din exterior  $R_A$ . Simetria circuitului de intrare duce la o creștere a factorului CMRR global.

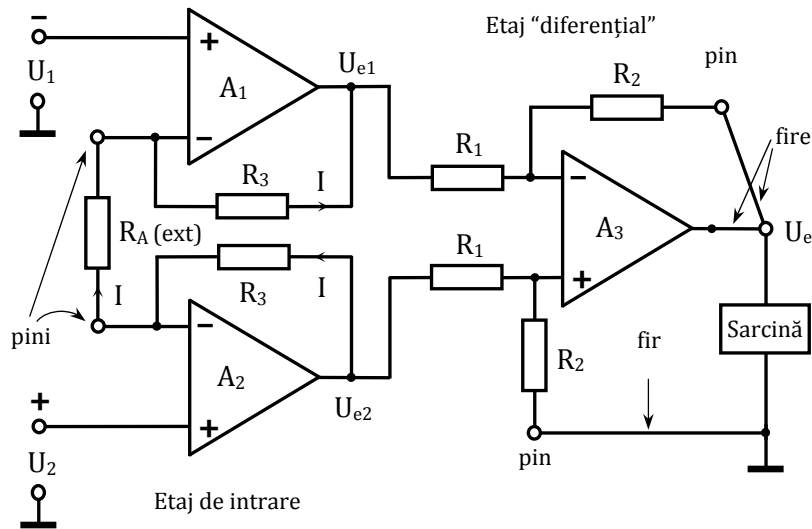


Fig. 3.23. Amplificator de măsură clasic

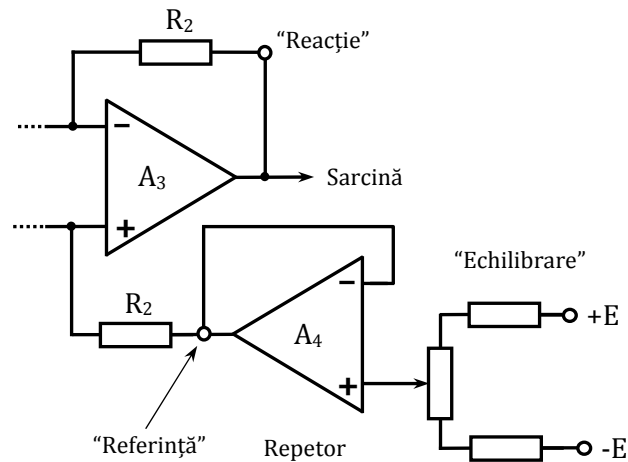


Fig. 3.24. Realizarea echilibrării la amplificatorul de măsură

Relația tensiunii de ieșire se stabilește ținând cont că amplificatorul realizat cu  $A_3$  este diferențial, iar amplificatoarele cu  $A_1$  și  $A_2$  sunt neinversoare, fiecare utilizând rezistența  $R_A$  care impune amplificarea (și poate fi deci programabilă):

$$U_e = (U_{e2} - U_{e1}) \frac{R_2}{R_1} = I(R_A + 2R_3) \frac{R_2}{R_1} = \frac{U_2 - U_1}{R_A} (R_A + 2R_3) \frac{R_2}{R_1} =$$

$$= (U_2 - U_1) \left( 1 + \frac{2R_3}{R_A} \right) \frac{R_2}{R_1} = A_{ur} (U_2 - U_1)$$

Deci amplificatorul este „diferențial” și având la ambele intrări rezistență foarte mare – este un amplificator de „instrumentație”.

Un astfel de amplificator monolitic prezintă pini pentru intrările – și +, pini pentru conectarea unei rezistențe  $R_A$  (notați „Amplificare”), precum și un pin numit „Reacție” și un pin numit „Referință” (marcați în fig.3.23). Aceștia din urmă permit eliminarea efectelor nedorite ale firelor lungi spre sarcină (ambii pini se leagă prin fire separate direct pe bornele sarcinii), iar pinul „Referință” mai permite introducerea unui circuit de echilibrare (fig.3.24). Se cunosc soluții speciale pentru folosirea amplificatorului de instrumentație cu fire lungi la intrare și (sau) ieșire [3]. În cazul de față, **circuitul de echilibrare**, folosind un AO repetor, nu introduce rezistență în serie cu  $R_2$  la pinul „Referință”, deci nu produce erori în amplificarea totală.

## 6. Amplificator inversor cu AO. Schema, expresia amplificarii si conditia de minimizare a erorilor statice. (§3.1)

Amplificatorul inversor (fig.3.1).

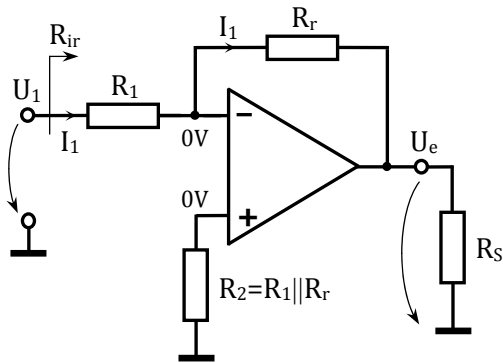


Fig. 3.1. Amplificator inversor cu AO

Amplificarea cu reacție ideală a acestui circuit este:

$$A_{ur} = \frac{U_e}{U_1} = \frac{-I_1 R_r}{I_1 R_1} = -\frac{R_r}{R_1}$$

și poate fi făcută de orice valoare. **Rezistența de intrare**  $R_{ir}$  „văzută” de sursa  $U_1$  este aproximativ egală cu  $R_1$  și este de valoare relativ redusă ( $n \times 10 \text{ K}\Omega$ ) din cauza reacției negative de tip paralel-paralel. Pentru a se lucra cu  $R_1$  de valoare mare trebuie folosit un amplificator cu  $R_i$  foarte mare. Rezistența de ieșire este neglijabilă datorită reacției negative cu configurație paralel la ieșire.



## 7. Amplificator neinvertor cu AO. Schema, expresia amplificării și condiția de minimizare a erorilor statice. (§3.1)

### Amplificator neinvertor (fig.3.2).

Amplificarea de tensiune cu reacție este:

$$A_{ur} = \frac{U_e}{U_2} = \frac{U_e}{U_e \frac{R_1}{R_1 + R_r}} = 1 + \frac{R_r}{R_1}$$

și poate fi doar **supraunitară** pentru acest circuit. Rezistența de intrare „văzută” de sursa  $U_2$  este foarte mare, datorită reacției negative de tipul paralel-serie. Totuși ea este limitată la valoarea rezistenței de intrare pentru semnal comun care a fost ignorată față de rezistența de intrare diferențială până acum. La amplificatoarele uzuale rezistența de intrare pentru semnal comun are o valoare de ordinul  $n \times 10 \text{ M}\Omega$ .

Pentru realizarea unei amplificări de tensiune **subunitare** se poate utiliza un divizor de tensiune la intrarea + dar în acest caz rezistența de intrare coboară la o valoare obișnuită ( $n \times 10 \text{ K}\Omega$ ), (fig.3.3). Pentru acest circuit se poate scrie tensiunea de ieșire:

$$U_e = U' \left( 1 + \frac{R_r}{R_1} \right) = U_2 \frac{R_3}{R_2 + R_3} \left( 1 + \frac{R_r}{R_1} \right)$$

și acum amplificarea lui  $U_2$  poate fi făcută subunitară.

Rezistența de intrare devine însă relativ redusă:  $R_{ir} \cong R_2 + R_3$   
Pentru dimensionarea divizorului se vor utiliza condițiile:

- realizarea unei divizări impuse de relația de mai sus;
- realizarea unei erori minime prin egalitatea rezistențelor echivalente de la cele două intrări.

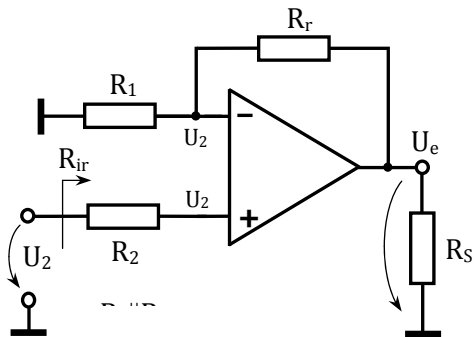


Fig. 3.2. Amplificator neinvertor cu AO

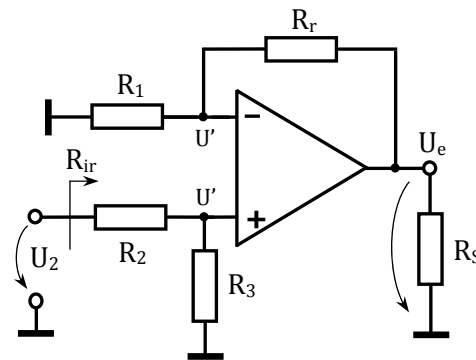


Fig. 3.3. Amplificator neinvertor cu divizor

**8. Care este elementul din structura unui redresor de precizie cu AO care permite creșterea cu cel puțin un ordin de mărime a frecvenței maxime de lucru? De ce?**

Dioda antisaturatie (conectata între intrarea inversoare și ieșirea AO) menține o reacție negativă locală la AO și evită intrarea sa în saturatie atunci când dioda redresoare se blochează (reducând de circa 10 ori saltul de tensiune pe care trebuie să îl facă ieșirea AO în acel moment).

**9. Care e deosebirea funcțională dintre detectorul de amplitudine și detectorul de vârf?**

Detectorul de amplitudine urmărește în sens crescător sau descrescător variațiile de amplitudine ale semnalului de intrare, pe când detectorul de vârf reține cea mai mare valoare de tensiune de intrare atinsă pe un interval de timp dat.

**10. Ce rol poate avea un AO într-o schemă de stabilizator de tensiune?**

Amplificator de eroare - amplifică abaterea unei fracțiuni prestabilite din tensiunea de ieșire față de o fracțiune prestabilă a tensiunii de referință interne a stabilizatorului în scopul menținerii constante a tensiunii de ieșire a stabilizatorului.

# **SEMNALE ȘI SISTEME**

**ANUL 2, SEMESTRUL 4**

### 1. Care este legătura între impulsul unitar și treapta unitară în timp discret?

Pag. 102:

Semnalul discret treaptă unitate, notat  $\sigma[n]$  este definit de relația:

$$\sigma[n] = \begin{cases} 1 & \text{pentru } n \geq 0 \\ 0 & \text{pentru } n < 0 \end{cases} \quad (5.13.)$$

Între cele două semnale există relațiile:

$$\delta[n] = \sigma[n] - \sigma[n-1]$$

$$\sigma[n] = \delta[n] + \delta[n-1] + \dots + \delta[n-k] + \dots$$

### 2. Definiți energia și puterea semnalului în timp discret.

Pag. 101:

Se numește *energie a semnalului*,  $E_x$ , respectiv *putere medie a semnalului*,  $P_x$ , cantitățile definite mai jos:

$$E_x = \sum_{n=-\infty}^{\infty} |x[n]|^2 ; P_x = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2 \quad (5.9.)$$

Se pot defini și în acest caz semnale de energie finită, semnale de putere medie finită, iar interpretarea fizică a acestor noțiuni este aceeași ca și în cazul continuu.

### 3. Enunțati Teorema lui Parseval.

Pag. 101:

Pentru semnale periodice discrete este valabilă relația lui Parseval care permite evaluarea puterii semnalului periodic discret fie prin cunoașterea eșantioanelor  $x[n]$  fie prin cunoașterea coeficienților spectrali  $a_k$ :

$$\frac{1}{N} \sum_{n=\langle N \rangle} |x[n]|^2 = \sum_{k=\langle N \rangle} |a_k|^2 \quad (5.36.)$$

### 4. Ce se intelege prin modulație exponentială (unghiulară) ?

Pag.191 :

Semnalul purtător este în acest caz un semnal sinusoidal de forma:

$$x_p(t) = A_p \cos(\omega_p t + \varphi_p) = \text{Re}\{x_c(t)\} \quad (7.42.)$$

unde:

$$x_c(t) = A_p \cdot e^{j(\omega_p t + \varphi_p)} \quad (7.43.)$$

**Definiție:** Modulația de frecvență și fază modifică exponentul (unghiul) purtătoarei, motiv pentru care se mai numesc și modulații exponențiale sau unghiulare

### 5. Proprietăți ale funcției răspuns la impuls pentru sistemele discrete, liniare și invariante în timp.

Pag.140-141:

Funcția răspuns la impuls caracterizează complet comportarea oricărui SLIT. În consecință proprietățile de stabilitate și cauzalitate vor impune funcției  $h[n]$  anumite restricții.

Astfel, stabilitatea SLIT este asigurată dacă și numai dacă  $h[n]$  este o funcție absolut sumabilă, adică:

$$\sum_{n=-\infty}^{\infty} |h[n]| < \infty \quad (6.10.)$$

Cauzalitatea SLIT este asigurată dacă:

$$h[n] = 0 \text{ pt. } n < 0 \quad (6.11.)$$

Suma de convoluție permite analiza și descrierea sistemelor interconectate, prin funcția pondere echivalentă.

Suma de convoluție este comutativă. Acest lucru implică egalitatea:

$$x[n] * h[n] = h[n] * x[n] \quad (6.12.)$$

Suma de convoluție este distributivă:

$$x[n](h_1[n] * h_2[n]) = x[n] * h_1[n] + x[n] * h_2[n] \quad (6.13.)$$

**6. Ce se înțelege prin filtru FIR? Dați un exemplu.**

Pag.145:

Sunt caracterizate de o relație intrare-ieșire de forma:

$$y[n] = b_0x[n] + b_1x[n-1] + \dots + b_kx[n-k] + \dots + b_Mx[n-M] = \sum_{k=0}^M b_kx[n-k] \quad (6.16.)$$

Se pune problema determinării lui  $h[n]$ , răspunsul la impuls al filtrului.

Pentru aceasta admitem că  $x[n] = \delta[n]$ . În consecință,  $y[n] = h[n]$  și are expresia:

$$h[n] = \sum_{k=0}^M b_k \cdot \delta[n-k] \quad (6.17.)$$

sau în forma echivalentă:

$$h[n] = \begin{cases} b_n, & 0 \leq n \leq M \\ 0, & \text{în rest} \end{cases} \quad (6.18.)$$

Sistemul descris de ecuația cu diferențe finite:

$$y[n] = 2x[n] - \frac{1}{2}x[n-1] + x[n-2]$$

Este un filtru FIR.

## 7. Ce se înțelege prin filtru IIR?

Pag. 149:

Sunt caracterizate de o relație intrare-ieșire de forma:

$$\sum_{k=0}^N a_k \cdot y[n-k] = \sum_{k=0}^M b_k \cdot x[n-k] ; a_0 \neq 0 , N > M \quad (6.19.)$$

unde  $a_k$  și  $b_k$  sunt constante reale sau complexe.  $N$  reprezintă în acest caz ordinul filtrului. Relația (6.19.) se poate scrie sub forma:

$$y[n] = \sum_{k=0}^M \frac{b_k}{a_0} x[n-k] - \sum_{k=1}^N \frac{a_k}{a_0} y[n-k] \quad (6.20.)$$

Relația (6.20.) arată că eșantionul curent al răspunsului depinde de eșantionul curent al intrării, precum și de  $M$  eșantioane anterioare ale intrării și  $N$  eșantioane anterioare ale răspunsului. Datorită acestei ultime particularități filtrele cu răspuns infinit la impuls se mai numesc și *filtre recursive*. Relația (6.19.) sau echivalenta acesteia (6.20.) nu caracterizează complet SDLIT. Sunt necesare informații în plus, referitoare la cauzalitatea sistemului și la condițiile inițiale.



### 8. Definiți transformata Fourier pentru semnalele in timp discret.

Pag.120

Prin definiție, transformata Fourier a unui semnal discret aperiodic  $x[n]$ , este:

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] \cdot e^{-j\Omega n} \quad (5.37.)$$

Această sumă nu converge pentru orice tip de semnal discret.

Spre exemplu: pentru  $x[n] = \sigma[n]$  suma nu e convergentă. De asemenea ea nu converge pentru semnalele exponențiale complexe discrete neatenuate.

Existența transformatei Fourier  $X(\Omega)$  implică următoarea condiție necesară și suficientă:

$$|X(\Omega)| = \left| \sum_{n=-\infty}^{\infty} x[n] \cdot e^{-j\Omega n} \right| \leq \sum_{n=-\infty}^{\infty} |x[n]| < \infty \quad (5.38.)$$

Potrivit condiției (5.37.),  $x[n]$  trebuie să aparțină clasei semnalelor de modul sumabil. Acestea sunt incluse în clasa semnalelor de energie finită (de pătrat sumabil), datorită inegalității:

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 \leq \left[ \sum_{n=-\infty}^{\infty} |x[n]| \right]^2 < \infty \quad (5.39.)$$

### 9. Ce reprezintă răspunsul armonic al unui filtru discret ?

Pag.155

Fie un SDLIT cu funcția pondere  $h[n]$  la intrarea căruia se aplică semnalul exponențial:

$$x[n] = e^{j\Omega_0 n} ; n \in Z \quad (6.25.)$$

Conform relației (6.8.) răspunsul  $y[n]$  al sistemului va fi:

$$\begin{aligned} y[n] &= x[n] * h[n] = \sum_{k=-\infty}^{\infty} h[k] \cdot x[n-k] = \sum_{k=-\infty}^{\infty} h[k] \cdot e^{j\Omega_0(n-k)} = \\ &= e^{j\Omega_0 n} \sum_{k=-\infty}^{\infty} h[k] \cdot e^{-j\Omega_0 k} \end{aligned} \quad (6.26.)$$

Notăm: 
$$\sum_{k=-\infty}^{\infty} h[k] \cdot e^{-j\Omega k} = H(\Omega) \quad (6.27.)$$

o funcție ce depinde de frecvență și care reprezintă transformata Fourier a răspunsului la impuls  $h[n]$ .

În aceste condiții (6.26.) devine:

$$y[n] = e^{j\Omega_0 n} \cdot H(\Omega_0) \quad (6.28.)$$

Relația (6.28.) constituie metoda armonică de determinare a răspunsului unui SDLIT.

Metoda armonică permite găsirea cu ușurință a răspunsului la orice semnal de intrare ce poate fi exprimat printr-o sumă (finită sau infinită) de semnale exponențiale complexe, cum ar fi semnalele armonice, semnalele periodice.

Funcția  $H(\Omega)$  depinde numai de filtrul discret. Ținând cont de caracterul complex al variabilei funcției putem scrie:

$$H(\Omega) = |H(\Omega)| \cdot e^{j \arg\{H(\Omega)\}} \quad (6.29.)$$

În consecință (6.28.) devine:

$$y[n] = |H(\Omega_0)| \cdot e^{j(\Omega_0 n + \arg\{H(\Omega_0)\})} \quad (6.30.)$$

**10. Cum se calculeaza randamentul in cazul modulatiei de amplitudine: P+2BL ?**

Pag.177

Randamentul sau eficienta transmisiei se apreciaza cu o relatie de forma:

$$\eta = \frac{P_{2BL}}{P} \quad (7.17.)$$

unde prin  $P_{2BL}$  s-a notat puterea în cele două benzi laterale, iar P este dat de (7.16.):

$$\eta = \frac{\frac{m^2 A_p^2}{4}}{\frac{A_p^2}{2} \left(1 + \frac{m^2}{2}\right)} = \frac{m^2}{2 + m^2} \quad (7.18.)$$

Întrucât m poate lua valoarea maximă 1 rezultă un randament maxim:

$$\eta = \frac{1}{3} = 0,33 \quad (7.19.)$$

**SISTEME DE PRELUCRARE NUMERICĂ CU PROCESOARE**  
**ANUL 2, SEMESTRUL 4**

# 1. Care este rolul magistralelor de adrese (MA), de date si control pentru un SPNP ?

## Daca MA are 16 linii, care este dimensiunea de memorie care poate fi adresata ?

### Raspunsul este in Capitolul 1/ Subcapitol 1.1./Pagina 1,2

## In general, pentru un SPNP cu N linii de adresa, numarul de locatii de memorie adresabile este 2N.

### CAPITOLUL 1

#### PRINCIPII GENERALE PRIVIND STRUCTURA SI FUNCTIONAREA UNUI SISTEM DE PRELUCRARE NUMERICA

##### 1.1. Structura generala a unui sistem de prelucrare numerica ( SPN )

Structura generala a unui SPN este prezentata in figura 1.1\_1.

Componenta esentiala a unui SPN este *unitatea aritmetica si logica* ( UAL ) cu rolul de a realiza operatii aritmetice si logice cum sunt : adunare, scadere, inmultire, incrementare, decrementare, SI, SAU, SAU-exclusiv etc. UAL este un circuit combinational, de unde rezulta necesitatea unor registre de memorare temporara a operanzilor si a rezultatului unei operatii.

*Registrul cu indicatorii de conditii* contine biti ( numiti si *fanoane* ) care se pozitioneaza pe "0" sau "1" in functie de rezultatul unei operatii efectuate de UAL. Cele mai reprezentative tipuri de fanoane sunt :

- Fanionul Z ( de zero ). Indica daca rezultatul unei operatii este nul sau nenul ( Z=1 pentru rezultat nul )
- Fanionul S ( de semn ). Indica daca rezultatul unei operatii este negativ sau pozitiv ( S=1 pentru rezultat negativ )
- Fanionul C ( de transport ). Indica daca in operatie a aparut sau nu transport din rangul cel mai semnificativ ( C=1 daca a aparut transport )
- Fanionul V ( de depasire ). Indica daca rezultatul operatiei este sau nu afectat de o eroare de depasire a capacitatii de operare a UAL( V=1 daca este eroare de depasire )
- Fanionul P ( de paritate ). Indica daca rezultatul operatiei precedente are un numar impar respectiv par de biti cu valoarea "1" ( P=1 pentru numar par de biti cu valoarea "1" )

Legatura intre componentele unui SPN se realizeaza prin semnale electrice grupate pe trei magistrale.

*Magistrala de adrese* selecteaza in mod unic o locatie de memorie, un port sau un registru, printr-un cuvint binar numit *adresa*.

Pe *magistrala de date* are loc transferul de informatie utila ( de date ) intre componentele SPN.

Semnalele *magistralei de control* realizeaza sincronizarea transferului pe

magistrala de date si precizeaza tipul operatiei : citire / scriere, lucru cu memoria / cu porturile.

*Registrul numarator de program* ( PC – Program Counter sau IP – Instruction Pointer ) are rolul de a indica adresa de la care urmeaza a se citi urmatoarea instructiune. Acest registru functioneaza, de regula, in regim de numarare ( incrementare ) si doar in cazul instructiilor de salt, apel de subrutina si revenire din subrutina in regim de incarcare paralela.

*Registrul de instructiuni* ( IR – Instruction Register ) pastreaza codul operatiei care face obiectul instructiunii. Are extensii in care se pastreaza informatii conexe codului operatiei cum ar fi informatiile privind locul in care se afla operanzii sau unde trebuie depus rezultatul.

*Circuitele de decodificare* au rolul de a decodifica codul operatiei memorat in IR.

*Unitatea de control si sincronizare* are rolul de a genera ansamblul semnalelor necesare pentru executia completa a instructiunii decodificate.

*Generatorul de tact* are rolul de a furniza un semnal de pilotare temporal cu care se sincronizeaza intreaga activitate a unitatii de control si in consecinta a intregului SPN.

Funcția de prelucrare numerica este realizata de catre sistem prin executia secventiala a unor instructiuni depuse in *memoria de program* si adresate prin registrul PC.

*Memoria de date* se utilizeaza pentru memorarea datelor cu care se lucreaza si care pot fi constante sau variabile pe parcursul executiei programului.

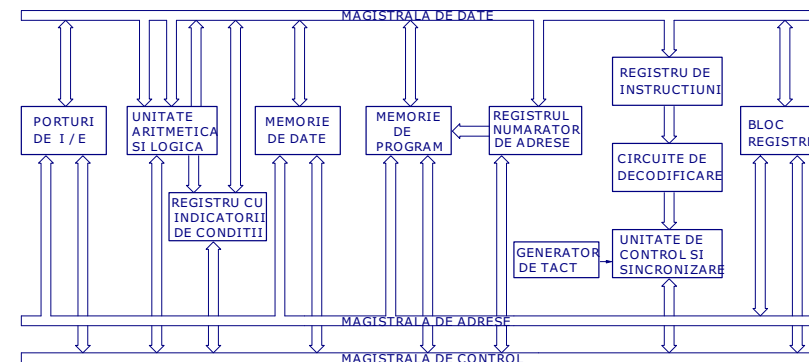


Fig. 1.1\_1. Structura generala a unui sistem de prelucrare numerica

## 2. Ce se intelege prin capabilitatea de intrerupere pentru un SPNP. Ce inseamna tratarea intreruperilor prin vectorizare ? Raspunsul este in Capitolul 1/ Subcapitol 1.2./ Pagina 3,4

*Blocul de registre* este specific fiecarui SPN si contine atit registre generale de lucru cit si registre dedicate ( speciale ) prin structura sistemului. Astfel exista registre dedicate care se utilizeaza ca surse cu operanzi sau ca destinatii ale rezultatelor pentru anumite operatii.

Un registru special este registrul *acumulator* notat cu A sau ACC utilizat ca sursa si destinatie in multe operatii aritmetice, logice si de transfer.

Un registru dedicat care exista in toate tipurile de SPN este registrul *indicator al virfului de stiva*( SP ) care pastreaza pointerul la locatiile de memorie gestionate dupa principiul stivei – “ultimul intrat ,primul ieseit “ ( LIFO – Last In, First Out ).

Transferul de date intre SPN si echipamentele de intrare / iesire ( I/E ) se realizeaza prin *porturi de I/E*.

Funcția de prelucrare numerică este realizată de către sistem prin executia secventială a unor instructiuni care implementeaza operatii aritmetice, logice, si de transfer. O instructiune este definita prin 1 – 4 cuvinte de 8 sau 16 biti care contin codul operatiei de executat, operanzii sau adresele operanzilor si adresa destinatiei.

Rularea unei instructii are 2 etape :

1. Extragerea sau aducerea( fetch ) codului operatiei si informatiilor privind locul operanzilor si/sau rezultatului
2. Executia propriu-zisa a instructiunii

*Etapa 1* cuprinde un ciclu de citire din memorie identic pentru toate instructiile. In cadrul acestui ciclu se citește locatia cu adresa indicata de PC si continutul ei se aduce in IR.

*Etapa 2* cuprinde unul sau mai multe cicluri de citire respectiv scriere din / in memorie, specifice fiecărei instructii. Intreaga succesiune de activare / dezactivare a semnalelor necesare este asigurata de unitatea de control si sincronizare.

In general, realizarea unui SPN se bazeaza pe utilizarea unui circuit integrat de tip *microprocesor, microcontroler sau procesor de semnal*. De exemplu, un microprocesor Z80 nu contine din structura prezentata memoria interna si porturile I/E. Un microcontroler 8051 contine insa toate componentele din structura unui SPN.

### 1.2. Capabilitatea de intreruperi

Conlucrarea dintre SPN si echipamentele periferice ar fi mai eficienta daca acesta ar avea capabilitatea de a executa, in momente de timp necunoscute apriori, cind anumite evenimente exterioare o cer, secvente de instructii adecvate acestora, neincluse in programul in rulare ci stocate in zone de memorie prestabilite. D.p.d.v. logic aceste secvente de instructii apar ca si cum ar fi inserate in programul in rulare.

Aceasta capabilitate a fost denumita conceptual *intrerupere* bazat pe faptul ca ea conduce la intreruperea temporara a rularii programului care detine controlul asupra SPN. Astfel, executia programului de baza, numit program principal poate fi intrerupta de o cerere de intrerupere efectuata de un port asociat unui echipament periferic de I/E. Acceptarea cererii, determina, dupa terminarea instructiunii in curs de executie, apelul unei subrutine de tratare a intreruperii specifica sursei a carei cerere de intrerupere a fost acceptata. Dupa executia acesteia, SPN revine la executia programului principal incepind cu instructiunea urmatoare celei dupa care s-a acceptat intreruperea. Pentru executia corecta a programului principal, se impune ca la inceputul si la sfirsitul subrutinei de intrerupere sa se salveze, respectiv sa se restabileasca starea programului intrerupt. De exemplu, operatiile de salvare si restaurare vizeaza continuturile registrelor interne cu care opereaza si programul principal si subrutina de intrerupere. Salvarea se poate face, de exemplu, in memoria stiva cu instructiunea PUSH, iar restaurarea din stiva cu instructiunea POP.

Sursele de cereri de intrerupere pot fi *nemascabile* sau *mascabile*. O sursa se numeste nemascabila daca nu exista posibilitatea de blocare prin program a acceptarii de catre SPN a cererilor de intrerupere corespunzatoare sursei. In cazul surselor mascabile exista posibilitatea de validare sau invalidare prin program a acceptarii de catre SPN a cererilor de intrerupere corespunzatoare. Validarea sau invalidarea se poate realiza global, pentru toate sursele de cereri de intrerupere mascabile sau individual, pentru fiecare sursa.

Deoarece, la un moment dat, un SPN poate deservi o singura cerere de intrerupere, prin rularea subrutinei de intrerupere corespunzatoare, rezulta necesitatea stabilirii unei ordini de prioritate a surselor de cereri de intrerupere, ordine bazata pe criteriile functionale si realizata atit prin structura sistemului cit si prin programare.

In general intreruperile sunt tratate prin *vectorizare* care consta in urmatoarele : se atribuie cite un numar ( *index* ) fiecărei cereri de intrerupere care trebuie sa duca la executia unei subrutine specifice. Dupa o anumita regula, cind o anumita intrerupere este acceptata de SPN acesta foloseste indexul pentru a calcula adresa la care se gaseste prima instructiune din subrutina care trebuie executata. In acest scop se creaza in memorie o asa numita *tabela de vectori de intrerupere*, fiecare vector reprezentind adresa la care se afla prima instructiune din cadrul unei subrutine de tratare a intreruperii. Deoarece pentru fiecare intrerupere sunt necesare 2 locatii de memorie in tabela ( adrese pe 2 octeti ), adresa la care se afla vectorul corespunzator unei intreruperi se calculeaza inmultind indexul acesteia cu 2. Tratarea unei intreruperi decurge astfel :

- Un dispozitiv activeaza cererea de intrerupere
- La sfirsitul instructiei in executie, cind in registrul PC se afla adresa urmatoarei instructii, microprocesorul anunta ca a acceptat cererea de intrerupere

# 3. Cum este organizata memoria unui microcontroler 80C552 ?

## Raspunsul este in Capitolul 3/ Subcapitol 3.1.2./ Pagina 33,34,35

*P0.0-0.7* - este un port de I/O pe 8 biti cu iesiri de tip "colector in gol". In timpul accesarii memoriei externe liniile portului au functii de magistrala de adrese/date, multiplexata în timp

*P1.0-1.7* - este un port de I/O pe 8 biti . Liniile portului pot avea si functii alternative

*P2.0-2.7* - este un port de I/O pe 8 biti . In timpul accesarii memoriei externe, liniile portului au functii de magistrala de adrese continind octetul mai semnificativ al adresei.

*P3.0-3.7* - este un port de I/O pe 8 biti . Liniile portului pot avea si functii alternative

*P4.0-4.7* - este un port de I/O pe 8 biti . Liniile portului pot avea si functii alternative

*P5.0-5.7* - este un port de intrare pe 8 biti . Liniile portului pot avea si functia alternativa de intrari pentru convertorul analog-numeric

*RST* - intrare/iesire RESET - ca intrare un nivel "1" realizeaza initializarea circuitului; ca iesire genereaza un impuls la depasirea capacitatii timer T3 ( se realizeaza si un reset intern )

*XTAL1, XTAL2* - intre cei doi pini se conecteaza cristalul de cuarț pentru oscilatorul intern sau la XTAL1 se aduce semnalul de tact extern cind se utilizeza un oscilator extern

*V<sub>SS</sub>* - masa digitala

*PSEN* - *Program Store Enable* - iesire "0" activa; reprezintă strobul de citire din memoria program externa si este activ în fiecare ciclu masina de extragere a codului operatiei . *PSEN*/ nu este activat la accesarea memoriei de date externa

*ALE* - *Address Latch Enable* - iesire "1" activa; semnal folosit pentru memorarea octetului mai puțin semnificativ al adresei, într-un latch extern, pentru adresarea memoriei externe

*EA* - *External Access* - intrare pentru circuit; cind valoarea adusa la pin este "0" logic procesorul executa instructiile din memoria program externa, daca valoarea adusa este "1" logic se executa instructiile din memoria program interna atit timp cit numarul de program este mai mic decit 8192 ( vezi organizarea memoriei )

*AV<sub>REF-</sub>* - tensiunea de referinta "low" pentru convertorul analog- numeric

*AV<sub>REF+</sub>* - tensiunea de referinta "high" pentru convertorul analog- numeric

*AV<sub>SS</sub>* - masa analogica

*AV<sub>DD</sub>* - tensiunea de alimentare analogica

### 3.1.2. Organizarea memoriei

Microcontrolerul 80C552 are spatiul de adrese separat pentru memoria program ( MP ) si pentru memoria de date ( MD ). Organizarea memoriei este prezentata in figura 3.1.2\_1.

MP este de 64 Kocteti, intreg spatiul fiind extern circuitului, situatie in care pinul *EA*/ este conectat, prin "strapare" la "0" logic. MP este de tip ROM ea putind fi numai citita.

MD este de tip RAM si este împartita în doua blocuri: interna si externa. In interiorul circuitului se gasesc 256 octeti iar în exterior este disponibil un spațiu de 64 de Kocteti.

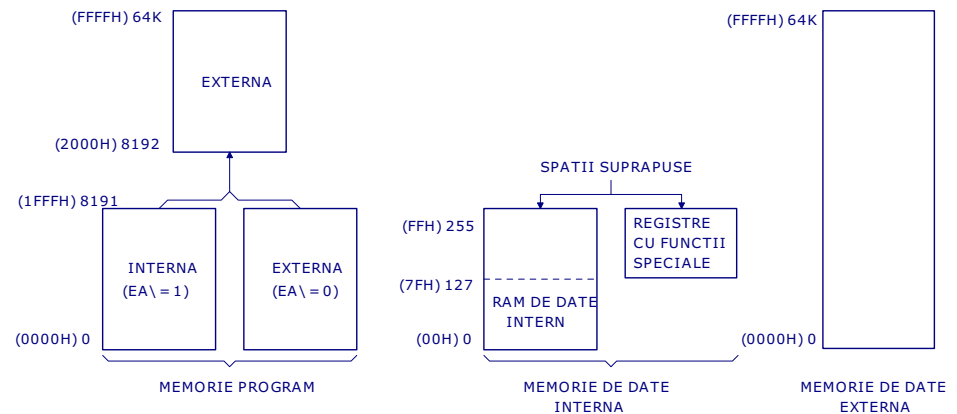


Fig. 3.1.2\_1. Organizarea memoriei

MD internă ( figura 3.1.2\_2 ) este împartita in trei zone distincte:

- zona de adrese 00H - 7FH (128 octeți "low") - poate fi adresata direct și indirect.
- zona de adrese 80H - FFH (128 octeți "high") - poate fi adresata numai indirect prin registru.
- zona de adrese 80H - FFH (spatiul registrelor cu functii speciale ) - este rezervata pentru SFR-uri (porturi, PSW-ul, timere, SP-ul, acumulator etc.), si poate fi adresata numai direct. Ultimele doua zone, desi au aceeasi adresa, sunt separate fizic.

Cind o instructie adreseaza o locatie internă cu adresa mai mare decit 7FH, CPU stie daca accesul este la octetii 128 superiori sau la SFR prin modul de adresare utilizat in instructie. Instructiile care utilizeaza adresare directa acceseaza spatiul SFR. De exemplu:

```
MOV 0A0H,data
```

acceseaza spatiul SFR, locatia 0A0H (care este P2).

Instructiile care folosesc adresare indirecta, acceseaza octetii 128 mai semnificativi. De exemplu:

```
MOV @R0,data
```

### 3. Cum este organizata memoria unui microcontroler 80C552 ?

Raspunsul este in Capitolul 3/ Subcapitol 3.1.2./ Pagina 33,34,35

unde R0 contine valoarea 0A0H acceseaza octetul de date la adresa 0A0H si nu portul P2.

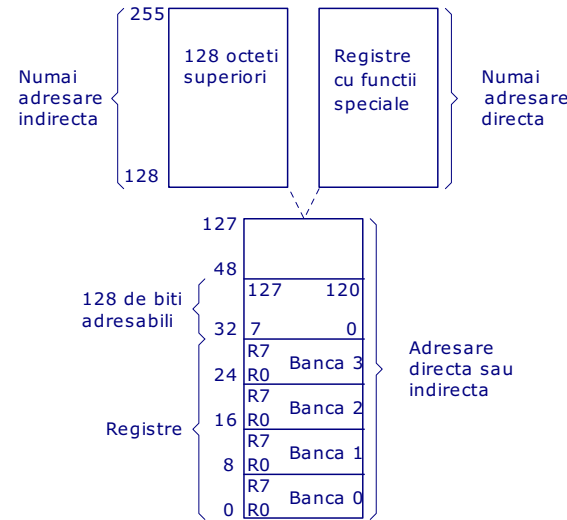


Fig. 3.1.2 2. Organizarea memoriei interne de date

Zona de adrese 00H – 7FH a RAM-ului intern este impartita astfel :

- zona pentru registrii R0-R7, de 32 octeti (de la 00H la 1FH), împartita la rindul ei în 4 "banci" a câte 8 octeti, fiecare octet reprezentind un registru general R0-R7. Numai o singura banca poate fi activa la un moment dat, ea fiind selectata prin program cu ajutorul bitilor RS1 și RS0 din registrul de stare PSW ( figura 3.1.2\_3). După o operatie de reset este activa banca 0 și SP este initializat la 07H.
- zona de lucru pentru procesorul boolean considerata ca memorie adresabila pe bit de la 20H pînă la 2FH (128 de adrese de bit).
- zona de lucru generala de la 30H pina la 7FH.

#### PSW

|    |    |    |     |     |    |   |   |
|----|----|----|-----|-----|----|---|---|
| CY | AC | F0 | RS1 | RS0 | OV | - | P |
|----|----|----|-----|-----|----|---|---|

unde: CY - fanionul "carry"; poate fi setat/ resetat hard sau soft

AC - fanion "carry" auxiliar ( pentru operatii BCD )

F0 - fanion ce poate fi definit de catre utilizator

RS1 RS0 - se folosesc la stabilirea grupei de registre folosita

0 0 - se selecteaza BANK 0

0 1 - se selecteaza BANK 1



## 4. Care este rolul registrelor cu functii speciale (SFR) pentru programarea unui microcontroller?

### Raspunsul este in Capitolul 3/ Subcapitol 3.1.3./ Pagina 36

CAP.3. MICROCONTROLLERUL 80C552

- 1 0 - se selecteaza BANK 2
- 1 1 - se selecteaza BANK 3
- OV - fanion depasire; este setat/ resetat hard daca se constata depasire de capacitate
- P - fanion de paritate; este setat/ resetat hard

**Fig. 3.1.2\_3.** Registru de stare PSW

Pentru conectarea memoriei externe, adresele necesare selectiei sunt obtinute de la porturile P0 și P2. Liniile portului P0 au functii de magistrala de adrese/date multiplexate în timp. In momentul în care pe liniile P0.0 - P0.7 se afla adrese (A0-A7) microcontrolerul activeaza semnalul ALE care este strobul de validare a adreselor, si este folosit pentru memorarea adreselor într-un latch de adresa. Liniile portului P2 vor conține adresele A8-A15.

Pentru accesarea MP microcontrolerul activeaza semnalul PSEN/ (Program Strobe Enable) care are rolul de strob de citire. Accesarea MD în ciclurile de citire respectiv scriere se face prin folosirea semnalelor RD/ si WR/ (semnale obtinute de la pinii P3.7 și P3.6 ai portului P3) ca stoburi de citire respectiv scriere din / in MD. In cazul accesarii MD interna semnalele RD/ si WR/ nu se activeaza.

Accesarea MP se face în cicluri de extragere a codului instructiunii utilizand adrese pe 16 biti generate prin intermediul registrului PC. Pentru accesarea MD externe se pot utiliza adrese pe 16 biti, folosind registrul DPTR , sau adrese pe 8 biti, folosind adresarea indirecta prin registru .

#### 3.1.3. Registre cu functii speciale ( SFR )

Registrele SFR includ porturile, numaratoarele, registre pentru controlul interfetelor, etc. Acestea pot fi accesate doar prin adresare directa, adresele, care pot fi atit de octet cit si de bit sunt cuprinse între 80H si FFH.

Tabelul 3.1.3\_1 prezinta registrele SFR .

Se observa ca nu toate adresele sunt ocupate. Adresele neocupate nu sunt implementate, astfel incit citirea lor va returna valori aleatoare, iar scrierea lor nu va avea nici un efect. De asemenea, 16 registre, IP1, B, IEN1, ACC, S1CON, PSW, TM21R, P4, IP0, P3, IEN0, P2, S0CON, P1, TCON, P0 pot fi accesate la nivel de bit, adresa celui mai putin semnificativ bit fiind adresa registrului, iar adresa celui mai semnificativ bit fiind adresa registrului + 7.

# 5. Un timer poate fi programat ca numarator sau temporizator. Care este deosebirea intre cele doua notiuni ?

## Raspunsul este in Capitolul 3/ Subcapitol 3.3.1./ Pagina 47

CAP.3. MICROCONTROLLERUL 80C552

### 3.3. Sistemul de numarare/temporizare al microcontrollerului 80C552

#### 3.3.1. Preliminarii

Microcontrollerul 8x552 contine trei numaratoare/temporizatoare de 16 biti denumite Timer 0 - T0 , Timer 1 - T1, Timer 2 - T2 și un temporizator de control, de 8 biti, Timer 3 - T3 .

In principiu, un numarator/temporizator pentru microcontrollerul 8x552 este format dintr-un registru de numarare (pe 8 sau 16 biti) care se incrementeaza la fiecare impuls de actionare extern sau intern (ciclu masina).

In momentul depasirii capacitatii registrului de numarare, se seteaza un fanion de intrerupere, se genereaza o intrerupere ( daca intreruperile sunt validate) si se face un salt in rutina de tratare a intreruperii.

Numararea este declansata explicit prin control hard sau soft si porneste din zero sau dintr-o valoare care se incarca prealabil in registrul de numarare.

In modul de lucru *numarator* ("*counter*"), continutul registrului este incrementat la fiecare tranzitie din "1" în "0" (front negativ) a semnalului de intrare. Deoarece sunt necesare două cicluri masina (24 perioade de tact) pentru recunoasterea unei tranzitii, frecventa maxima de numarare este de 1/24 din frecventa de tact.

In modul de lucru *temporizator* ("*timer*"), continutul registrului este incrementat la fiecare ciclu masina executat. Astfel, deoarece un ciclu masina contine 12 perioade de tact, frecventa maxima de numarare este de 1/12 din frecventa de tact.

#### 3.3.2. Timer 0 si Timer1

Resursele hard ale T0, respectiv T1, sunt urmatoarele:

- 2 registre de numarare pe 8 biti, TL0 si TH0 (pentru T0), respectiv TL1 si TH1 (pentru T1), care pot fi utilizate concatenate, pentru a obtine un registru pe 16 biti
- 1 registru pe 8 biti pentru programarea modului de lucru, TMOD ("Timer MODE") - fig.3.3.2\_1
- 1 registru pe 8 biti pentru control, TCON ("Timer CONtrol") - fig.3.3.2\_2

Registrul TMOD este comun ambelor timere; cei patru biti mai semnificativi programeaza T1, iar ceilalti T0.

Bitii M0, M1 selecteaza unul din cele patru moduri de operare. Modurile 0,1 și 2 sunt identice pentru ambele timere, iar modul 3 este diferit.

Bitul C/T selecteaza intre numarator sau temporizator.

Bitul GATE selecteaza modul de validare (de pornire) a numararii: validare hard (GATE=1) sau validare soft (GATE=0).

## 6. Care este rolul unui timer de tip “watchdog” intr-un sistem cu microcontroler? Raspunsul este in Capitolul 3/ Subcapitol 3.3.4. / Pagina 56

CAP.3. MICROCONTROLERUL 80C552

depasire capacitate registru pe 16 biti ( T2 ) prin bitul T2OV ( TM2IR.7 ). Se poate programa ca una sau amindoua din depasirile de capacitate sa genereze intreruperi prin bitii T2IS0 ( TM2CON.6 ) si T2IS1 ( TM2CON.7 ). Ambele intreruperi au acelasi vector de adresa.

Toate fanioanele de intrerupere asociate lui T2 trebuie resetate prin program.

### 3.3.4. Timer 3 ( T3 )

Microcontrolerul 80C552 include un timer de control ("watchdog") alcatuit dintr-un divizor de frecventa pe 11 biti si un temporizator de 8 biti ( T3 ). Schema bloc este prezentata in figura 3.3.4\_1.

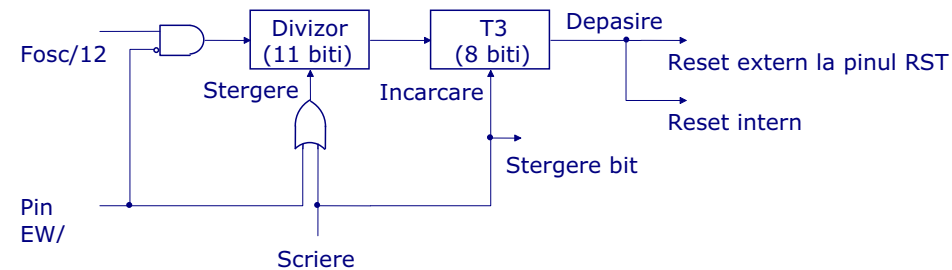


Fig. 3.3.4\_1. Schema bloc pentru Timer 3

Frecventa cu care este incrementat T3 rezulta din formula:

$$f_{\text{timer}} = \frac{f_{\text{osc}}}{12 * 2048}$$

La depasirea capacitatii registrului T3, microcontrolerul este resetat si este generat si un impuls la pinul RST. Pentru a preveni acest lucru, registrul T3 trebuie reincarcat periodic de programul care ruleaza. Astfel, daca procesorul functioneaza anormal ( hard sau soft ) programul care rulează nu efectueaza la timp reincarcarea T3, fapt ce duce la generarea unui reset intern, care impiedeca astfel o functionare anormala a microcontrolerului.

T3 poate fi reincarcat doar dacă fanionul WLE ( PCON.4 ) a fost setat prin soft. In momentul reincarcarii T3 fanionul este automat resetat. Intervalul de timp dupa care poate apare automat resetul depinde de valoarea care se incarca în numarator si poate varia, de exemplu, între 2 ms și 0.5 s pentru frecventa de oscilatie de 12 Mhz.

T3 este controlat extern prin semnalul /EW : "0" logic valideaza timerul si invalideaza modul "Power-down", iar "1" logic face actiunea inversa. In modul "Idle" timerul ramine activ.

## **7. Programarea si functionarea unei resurse a microcontrolerului se poate face cu sau fara intreruperi. Care este diferenta dintre cele 2 abordari ?**

*Intr-o aplicatie, programarea unei resurse a microcontrolerului se poate realiza, in principiu, in doua moduri:*

- *fara intreruperi, prin interogare ( “polling” ), testind un fanion de stare pentru a determina momentul producerii unor evenimente (de ex. sfirsit conversie analog-numeric, receptie seriala a unui caracter).*
- *prin intreruperi. In acest caz, programatorul trebuie sa scrie o subrutina de tratare a intreruperii unde se ajunge automat (prin mecanisme hard) atunci cind se produce evenimentul.*

*Precizarile anterioare sunt valabile pentru sistemul de numarare/temporizare, convertor analog-numeric, interfata seriala .*

## 8. Care este rolul modurilor speciale de reducere a puterii consumate pentru un microcontroler ?

### Raspunsul este in Capitolul 3/ Subcapitol 3.4.4./ Pagina 69

CAP.3. MICROCONTROLERUL 80C552

generat de T3 ( 3 cicluri masina ) va activa tranzistorul de “pull – up” care “trage” la “1” logic linia RST.

Dupa reset, registrele SFR iau valori bine determinate, iar continutul RAM-ului intern nu este afectat.

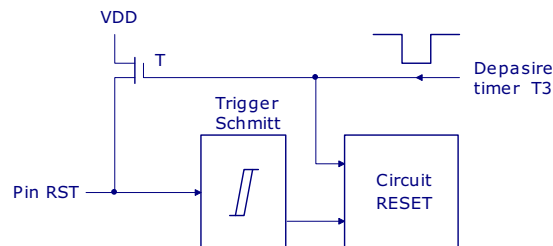


Fig. 3.4.3\_1. Circuitul pentru RESET

Resetarea procesorului se poate obtine si la punerea sub tensiune, obtinandu-se asa numitul *reset automat* . In acest caz, se leaga pinul RST la VDD prin intermediul unui condensator de 2,2  $\mu$ F. Deoarece impulsul scurt generat de T3 nu poate descarca acest condensator, schema poate fi utilizata doar daca nu se are in vedere validarea “watchdog”. In caz contrar, trebuie utilizat un circuit separat pentru reset la punerea sub tensiune.

#### 3.4.4. Modurile Idle si Power down

80C552 dispune de doua moduri speciale de reducere a puterii consumate, *Idle* si *Power down*, activate prin setarea unor biti din registrul PCON( figura 3.4.4\_1 ).

Modul Idle, activat prin setarea bitului IDL ( PCON.0 ), permite ca intreruperile, porturile seriale si Timerele 0 si 1, Timer 3 sa continue sa functioneze in timp ce procesorul, Timer T2, PWM0, PWM1 si ADC sunt oprite.

Odata instalat modul Idle, starea procesorului , continutul RAM si toate registrele SFR ramin intacte. Exista doua metode de a iesi din starea Idle:

- activarea oricarei intreruperi validate va duce la terminarea acestui mod; dupa procesarea intreruperii (dupa executia instructiunii RETI), se revine la urmatoarea instructie de executat, dupa cea care a dus la instalarea modului Idle; fanioanele GF0 si GF1 se pot folosi pentru a sti daca intreruperea a fost receptionata in timpul executiei normale sau in timpul modului Idle;
- prin reset hard extern sau prin reset intern cauzat de depasirea capacitatii lui T3. Deoarece tactul este validat, resetul hard trebuie sa fie activ cel putin pe durata a doua cicluri de masina.

# 9. Care sunt caracteristicile unui semnal generat pe iesirea PWM, care pot fi modificate prin programarea registrilor microcontrolerului 80C552 ?

## Raspunsul este in Capitolul 3/ Subcapitol 3.4.1./ Pagina 65

### 3.4. Iesiri PWM. Bloc conversie analog - numerica Circuitele de tact si de reset. Modurile Idle si Power down

#### 3.4.1. Iesiri PWM

Microcontrolerul 80C552 contine doua canale ( iesiri ) pe care se pot genera impulsuri cu frecventa si factor de umplere programat ( "pulse width modulation outputs" ). Schema bloc este prezentata in figura 3.4.1\_1

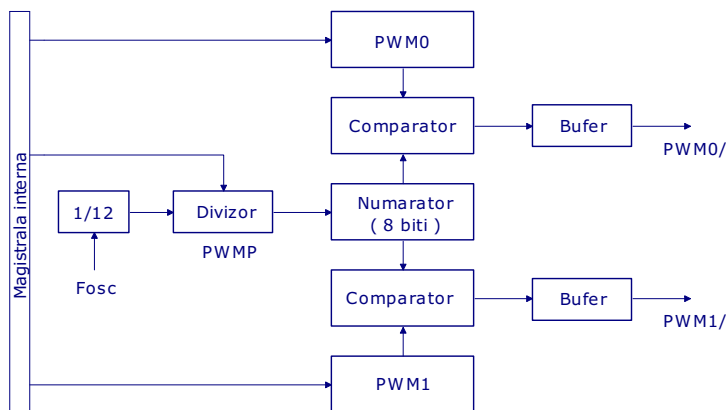


Fig. 3.4.1\_1. Schema bloc pentru iesiri PWM

Resursele hard sunt urmatoarele:

- Registrul PWMP pe 8 biti, comun ambelor canale, cu rol de divizor de frecventa
- un registru de numarare pe 8 biti
- pentru fiecare canal, un registru asociat PWM0, respectiv PWM1, un comparator pe 8 biti si un bufer pentru formarea semnalelor de iesire

Frecventa de tact (Fosc) divizata succesiv cu 12 si cu registru PWMP actioneaza numaratorul comun celor doua canale care numara modulo 255 ( deci intre 0 si 254 inclusiv ).

Valoarea acestuia este comparata permanent cu continutul celor doua registre PWM0 si PWM1; daca continutul registrului este mai mare decit continutul numaratorului iesirea corespunzatoare a canalului este pusa pe zero, iar daca este mai mic sau egal, iesirea corespunzatoare este pusa pe unu. Astfel registrele PWMi determina factorul de umplere al semnalelor generate in domeniul 0 - 255/255

programabil in incrementi de 1/255. Frecventa semnalelor obtinute la cele doua iesiri este data de relatia:

$$f_{PWM} = \frac{f_{osc}}{2 * (1+PWMP) * 255}$$

iar valoarea PWMP pentru o frecventa dorita  $f_{PWM}$  este:

$$PWMP = \frac{f_{osc}}{f_{PWM} * 2 * 255} - 1$$

De exemplu, pentru frecventa de oscilatie de 12MHz rezulta  $f_{PWM}$  intre 92 Hz si 23,5 KHz.

Asa cum am mentionat, PWMP are rol de registru de divizare, factorul de divizare fiind dat de relatia:

$$\text{factor de divizare} = (PWMP) + 1$$

Daca registrele PWMi se incarca cu 00H sau FFH iesirile PWMi pot fi mentinute la un nivel constant de "1" respectiv "0".

#### 3.4.2. Bloc conversie analog - numerica

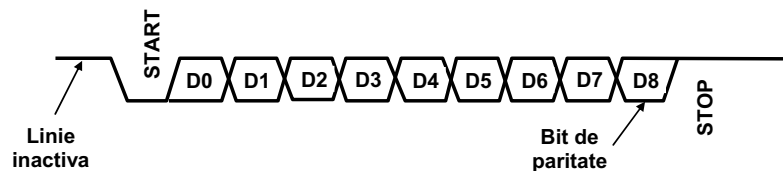
Blocul de conversie analog - numerica ( BCAN ) contine un multiplexor analogic pe 8 canale si un convertor analog - numeric cu rezolutia de 10 biti. Conversia necesita 50 de cicluri masina, adica 50 microsecunde la frecventa de tact de 12 MHz. Schema bloc este prezentata in figura 3.4.2\_1.

BCAN este controlat prin registrul ADCON ( figura 3.4.2\_2 ).

Selectia, pentru conversie, a uneia din cele 8 intrari analogice ( pe portul de intrare P5 ), se face prin bitii AADR2, AADR1 si AADR0 din registrul ADCON. Modificarea acestor biti se poate face doar cind bitii ADCI ( ADCON.4 ) si ADCS ( ADCON.3 ) sunt la "0" logic.

Startul pentru o conversie analog - numerica se poate face extern, de la pinul STADC, daca bitul ADEX = 1 ( ADCON.5 ) sau intern, prin program. In ambele cazuri trebuie setat, prin program, bitul ADCS ( ADCON.3 ), iar logica interna asigura ca acest bit sa ramina pe "1" atit timp cit se executa conversia. La sfirsitul acesteia, ADCS este resetat iar fanionul de intrerupere ADCI, setat. ADCS nu poate fi sters iar ADCI setat prin program ( ci numai prin hard ).

**10. Figura reprezinta diagrama pentru schimbul de date prin interfata seriala in modul serial asincron. Precizati secventa de biti transmisi pe linia seriala pentru transferul caracterului 41H, pentru 8biti/caracter, paritate para, un bit de STOP.**



*Raspunsul este in Capitolul 3/ Subcapitol 3.5.1./ Pagina 73*

*Caracterul 41H(adica in hexazecimal) se transforma in binar, iar D0-D7 sunt bitii acestei reprezentari binare.*

*Paritate para inseamna ca bitul D8, trebuie ales astfel incit intreaga reprezentare (D0-D8) trebuie sa aiba un numar par de biti de 1.*

### 3.5. Interfata seriala UART a microcontrolerului 80C552

#### 3.5.1. Preliminarii

Transferul serial al datelor între două echipamente se realizează sub forma bit după bit pe o singură linie fizică.

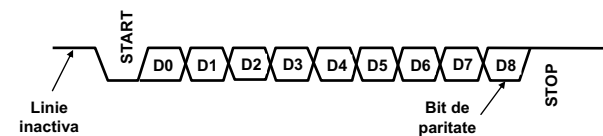
După direcția de deplasare a datelor, transferurile seriale se împart astfel:

- simplex – transfer unidirecțional, de la un transmitor la un receptor, pe o singură linie de semnal (plus linia de masă corespunzătoare).
- semi-duplex (half-duplex) – transfer bidirecțional desfășurat la momente de timp diferite pe o singură linie de semnal.
- duplex (full duplex) – transfer bidirecțional, care poate avea loc simultan și necesită două linii de semnal.

După modul de sincronizare între transmitor și receptor, există două modalități de transfer: asincron și sincron.

Transferul asincron numit și transfer pe caracter (fig. 3.5.1\_1), începe cu un bit de START ("0" logic) folosit pentru sincronizarea cu receptorul. Urmează bitii caracterului, începând cu cel mai puțin semnificativ (5, 6, 7 sau 8 biti de date pe caracter). Bitul (optional) de paritate are rolul de a detecta o eroare de transfer singulară sau un număr impar de erori. Transferul se încheie cu 1, 1½ sau 2 biti de STOP ("1" logic) prevăzuți pentru a asigura o separare între două caractere consecutive. De asemenea, procesorul poate prelua caracterul asamblat de interfata serială pentru a evita suprascrierea acestuia.

Transferul sincron numit și transfer pe bloc (sau mesaj) începe cu unul până la cinci caractere speciale cu rol de sincronizare. Urmează bitii caracterului care alcătuiesc blocul și care sunt transmiși și recepționați sincronizat cu un tact generat de transmitor. Este necesară, în general, o linie de tact separată, dar există și tehnici care permit combinarea datelor și a tactului pe aceeași linie.



**Fig. 3.5.1\_1.** Structura unui caracter în mod asincron