

# NUMĂRĂTOARE BINARE SINCRONE

## **ARGUMENT**

Sistemele de radiocomunicații sunt sisteme care multă vreme au fost prin excelență, sisteme analogice. Deși, încă de la începuturile radiocomunicațiilor, se poate vorbi de reprezentarea digitală a informației, prin utilizarea transmisiilor telegrafice în cod Morse, totuși prelucrarea semnalelor radio a fost până nu de mult un domeniu exclusiv analogic.

Utilizarea teleimprimatoarelor, apoi a sintetizoarelor de frecvență, a deschis un câmp tot mai larg de aplicare pentru circuitele digitale în echipamentele de radiocomunicații. A urmat apoi digitalizarea elementelor de control a echipamentelor, mai întâi utilizându-se logică cablată și apoi logică programată (microcalculatoare). Asistăm astăzi, odată cu dezvoltarea vitezei și specializării circuitelor integrate numerice, la o penetrație tot mai profundă a acestora în însăși zonele de prelucrare a semnalelor radio, care până nu demult erau exclusiv analogice.

În lucrarea de față, se face o prezentare generală a circuitelor numerice și a unor aplicații importante, în echipamentele de radiocomunicații moderne.

## **CUPRINS**

<b><i>Argument.....</i></b>	<b><i>pag.3</i></b>
<b><i>Funcții Logice.....</i></b>	<b><i>pag.5</i></b>
<b><i>Elemente de bază ale circuitelor logice.....</i></b>	<b><i>pag.9</i></b>
<b><i>Cap.1-Circuite Basculante Bistabile-CBB.....</i></b>	<b><i>pag.17</i></b>
<b><i>CBB de tip S-R.....</i></b>	<b><i>pag.18</i></b>
<b><i>CBB de tip J-K.....</i></b>	<b><i>pag.23</i></b>
<b><i>CBB de tip D.....</i></b>	<b><i>pag.26</i></b>
<b><i>Cap.2-Numărătoare.....</i></b>	<b><i>pag.28</i></b>
<b><i>Cap.3-Numărătoare binare sincrone.....</i></b>	<b><i>pag.41</i></b>
<b><i>Bibliografie.....</i></b>	<b><i>pag.48</i></b>

# 1. Circuite numerice în radiocomunicații

## 1.1. FUNCȚII LOGICE

Modul de lucru al circuitelor digitale este studiat cu ajutorul *algebrei Boole* (introdusă de *George Boole* în jurul anului 1850 și aplicată de *Claude Shannon* în 1938 la funcțiile logice binare). Această algebră operează cu sistemul de numerație binar, simbolurile folosite fiind 0 și 1. Variabilele logice pot lua una din cele două valori (0 sau 1), iar operatorii fundamentali sunt **ȘI**, **SAU** și **NU**. În definirea operatorilor logici vom nota cu  $X, Y, Z$  variabilele logice.

- Operatorul logic **ȘI** (notat cu  $\cdot$  semn care uneori poate lipsi) se scrie  $X \cdot Y = Z$  și semnifică: dacă  $X=1$  **ȘI**  $Y=1$  atunci  $Z=1$ ; altfel  $Z=0$
- Operatorul logic **SAU** (notat cu  $+$ ) se scrie  $X + Y = Z$  și semnifică: dacă  $X=1$  **SAU**  $Y=1$  atunci  $Z=1$ ; altfel  $Z=0$
- Operatorul logic **NU** (notat cu o bară deasupra variabilei a căreia i se aplică) se scrie  $\bar{X} = Y$ . Dacă  $X=1$  atunci  $Y=0$ ; altfel  $Y=1$ .

Acești operatori logici se utilizează în schemele logice, sub forma simbolurilor grafice date în figura 1. Simbolurile din figura 1a și 1b se mai numesc porți logice **ȘI** respectiv **SAU** iar simbolul din figura 1c se numește inversor. De menționat că porțile pot avea și mai mult de două intrări.

Pornind de la definițiile de mai sus s-au dedus *teoremele algebrei Boole* date în tabelul 1.

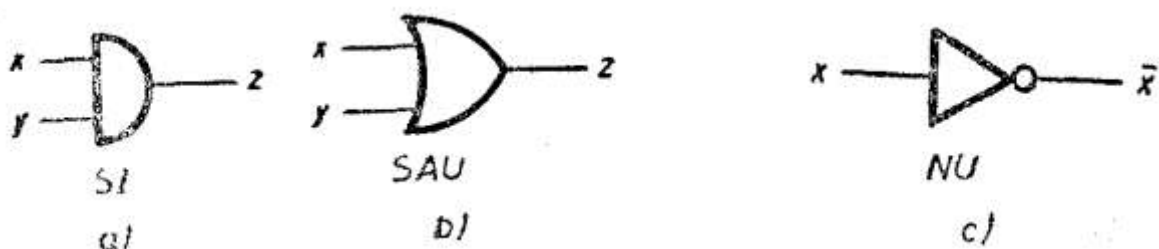


Fig. 1. Principalele porți logice.

Cu ajutorul operatorilor **ȘI**, **SAU** și **NU** se obțin funcțiile logice binare care au domeniul de definiție și al valorilor mulțimea  $\{0,1\}$ . Circuitele pentru care expresia funcției logice depinde numai de variabilele de intrare se numesc *circuite combinaționale*.

Pentru studiul acestui tip de funcții logice se folosesc tabelele de adevăr, care cuprind toate combinațiile posibile ale variabilelor componente. De exemplu, în figura 2 se dă expresia unei funcții logice, reprezentarea grafică și tabela de adevăr.

Combinațiile de variabile în stare normală sau negată care dau, în tabela de adevăr, valoarea 1 pentru funcția logică se numesc *mintermeni*. Expresia funcției logice, sub forma canonică disjunctivă, se obține aplicând operatorul **SAU** acestor *mintermeni*. Pentru exemplul din figura 2 rezultă expresia funcției logice:

$$F = XY\bar{Z} + \bar{X}\bar{Y}Z + X\bar{Y}Z + \bar{X}YZ + ZYZ \quad (1)$$

Teoremele *algebrei Boole*

Tabelul 1

1	$x \cdot 0 = 0$	15	$xy + xz = x(y + z)$
2	$x + 0 = x$	16	$(x + y)(x + z) = x + yz$
3	$x \cdot 1 = x$	17	$\overline{xyz} = \bar{x} + \bar{y} + \bar{z}$
4	$x + 1 = 1$	18	$\overline{x + y + z} = \bar{x} \cdot \bar{y} \cdot \bar{z}$
5	$x \cdot x = x$	19	$x(x + y) = x$
6	$x + x = x$	20	$x + xy = x$
7	$x \cdot \bar{x} = 0$	21	$x(\bar{x} + y) = xy$
8	$x + \bar{x} = 1$	22	$x + \bar{x}y = x + y$
9	$\overline{\bar{x}} = x$	23	$xy + \bar{x}y = y$
10	$\overline{\bar{\bar{x}}} = \bar{x}$	24	$(x + y)(\bar{x} + y) = y$
11	$xy = yx$	25	$(x + y)(\bar{x} + z) = xz + \bar{x}y$
12	$x + y = y + x$	26	$(x + z)(\bar{x} + y) = xy + \bar{x}z$
13	$xyz = (xy)z = x(yz)$	27	$(x + y)(\bar{x} + z)(y + z) = (x + y)(\bar{x} + z)$
14	$x + y + z = (x + y) + z = x + (y + z)$	28	$xy + yz + \bar{x}z = xy + \bar{x}z$

Trecerea de la această expresie, la expresia mai simplă din figura 2 se face prin simplificarea funcției.

În practica proiectării circuitelor digitale combinaționale se pornește de la tabela de adevăr (dictată de datele de proiectare), se deduce expresia funcției logice, se simplifică această expresie și se organizează sub o formă care permite implementarea cu circuite integrate.

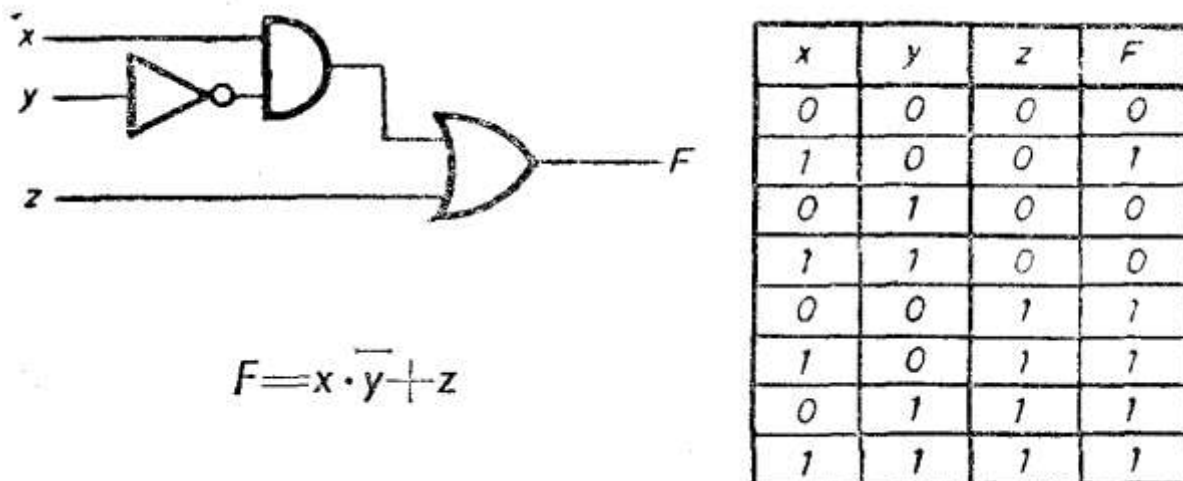


Fig. 2. Exemplu de reprezentare simbolică și cu tabelă de adevăr pentru o funcție logică.

**Simplificarea funcțiilor logice.** Simplificarea funcțiilor logice se poate face fie utilizând teoremele din tabelul 1, fie utilizând diverse metode grafice (diagramele *Venn*, *Veitch*, *Karnaugh*) sau metode special destinate (*Quine—McClusky*).

Pentru necesități obișnuite (mai puțin de 5 variabile) cea mai adecvată metodă de simplificare este cea a diagramelor *Karnaugh*, în care se face uz de *teorema 23* din tabelul 1:

$$XY + \bar{X}Y = Y.$$

Tocmai utilizarea acestei teoreme face necesară ordonarea căsuțelor în diagrame, astfel încât să nu avem, la trecerea de la un rând la altul sau de la o coloană la alta decât o singură schimbare a uneia dintre variabile. Diagrama se obține punând în fiecare căsuță corespunzătoare 0 sau 1 în funcție de conținutul tabelului de adevăr a funcției de minimizat.

În figura 3 se dă diagrama *Karnaugh* a funcției dată de expresia (1).

Se observă încercuirea pozițiilor cu valoarea 1. Aceste încercuiri pot cuprinde un număr  $2^n$  (2, 4, 8 etc.), de căsuțe adiacente ale diagramei, evidențiind eliminarea uneia sau mai multor variabile. Se constată că o încercuire de două căsuțe elimină o variabilă, de 4 căsuțe două variabile, având ca regulă: o încercuire de  $2^n$  căsuțe elimină  $n$  variabile.

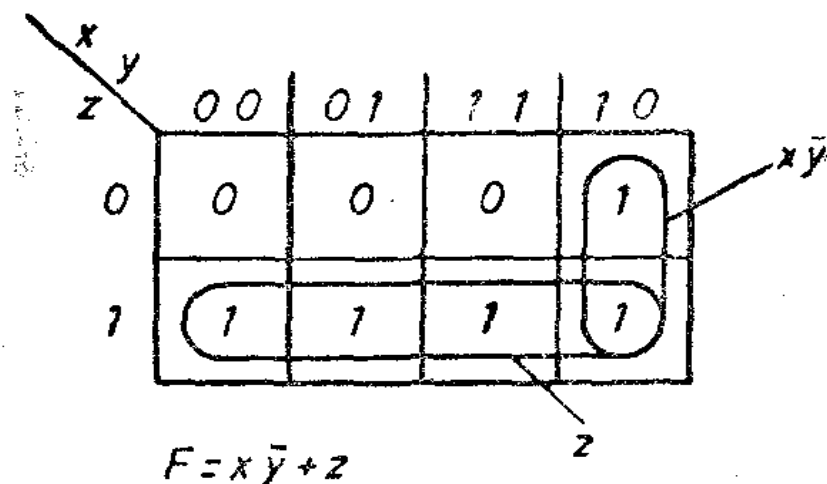
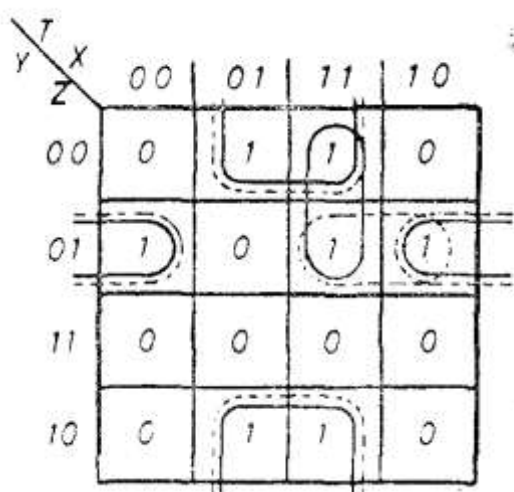


Fig. 3. Exemplu de *diagramă Karnaugh* pentru trei variabile.

Trebuie adăugat că în diagramă, căsuțele aflate la extremele rândurilor sau coloanelor se consideră adiacente și pot participa la o încercuire de eliminare, în figura 4 se dă un exemplu de minimizare a unei funcții de patru variabile. De remarcat că simplificarea nu este unică, în practică alegând expresia cea mai convenabilă pentru implementare.

În anume situații practice nu interesează valoarea funcției decât pentru o parte din combinațiile variabilelor de intrare, unele combinații neavând sens. În aceste cazuri valorile care se introduc în diagramă sunt indiferente (termeni redondanți), și se notează cu X. Acești termeni pot fi considerați 1 și incluși în încercuirile de eliminare, pentru a mări suprafața încercuirii cât mai mult și deci a elimina cât mai multe variabile. În cazul lucrului ulterior cu *diagrama Karnaugh* se va putea urmări această facilitate.



Expresia nesimplificată

$$F = \bar{T}\bar{X}\bar{Y}\bar{Z} + \bar{T}X\bar{Y}\bar{Z} + \bar{T}X\bar{Y}Z + \bar{T}X\bar{Y}\bar{Z} + \\ + TX\bar{Y}Z + TX\bar{Y}\bar{Z} + T\bar{X}YZ$$

Expresia simplificată cu linii continue

$$F = \bar{X}Z + X\bar{Y}Z + T\bar{X}\bar{Y}$$

Expresia simplificată cu linii întrerupte

$$F = \bar{X}Z + X\bar{Y}Z + T\bar{Y}Z$$

Fig. 4. Exemplu de *diagramă Karnaugh* pentru patru variabile.

Pentru funcții cu mai mult de cinci variabile utilizarea *diagramei Karnaugh* devine dificilă și se preferă metode de simplificare ce pot face uz de calculator. O astfel de metodă este metoda *Quine-McClusky*.

## 1.2. ELEMENTELE DE BAZĂ ALE CIRCUITELOR LOGICE

Realizarea practică a funcțiilor logice se poate face în diverse tehnici: cu comutatoare (relee), cu circuite pneumatice sau circuite electronice. Ultima posibilitate este cea de care ne vom ocupa având în vedere că în radiocomunicații constituie singura modalitate uzitată.

### 1.2.1. PORȚI LOGICE

Operatorii logici, ca și unele funcții logice simple capătă după cum am văzut (în figura 1) forme grafice specifice și sunt tratate ca entități distincte în circuitele digitale. Aceste entități poartă numele de *porți logice*.

Pentru realizarea porților logice se folosesc circuite care lucrează atât la intrare cât și la ieșire cu două nivele de tensiune. Alocarea acestor nivele de tensiune stărilor logice 0 sau 1, determină tipul de logică utilizată.

Astfel, dacă nivelul superior (*S*) de tensiune se consideră starea logică 1 se lucrează în logică pozitivă, iar dacă nivelul inferior (*I*) de tensiune se consideră starea logică 1 se lucrează în logică negativă. De exemplu, în fig. 5 se arată realizarea cu același circuit a două funcții logice diferite în funcție de tipul de logică adoptat.

Deși în general se lucrează în logică pozitivă producătorii de circuite digitale specifică funcționarea acestora cu nivelele de tensiune (*S* și *I*) rămânând la latitudinea utilizatorului să aleagă tipul de logică folosit. În cele ce urmează vom considera numai lucrul în logică pozitivă și se vor utiliza numai simbolurile 0 și 1.

Pe lângă porțile logice din figura 1 mai se utilizează și porțile logice ale căror simboluri grafice și tabele de adevăr se dau în figura 6.

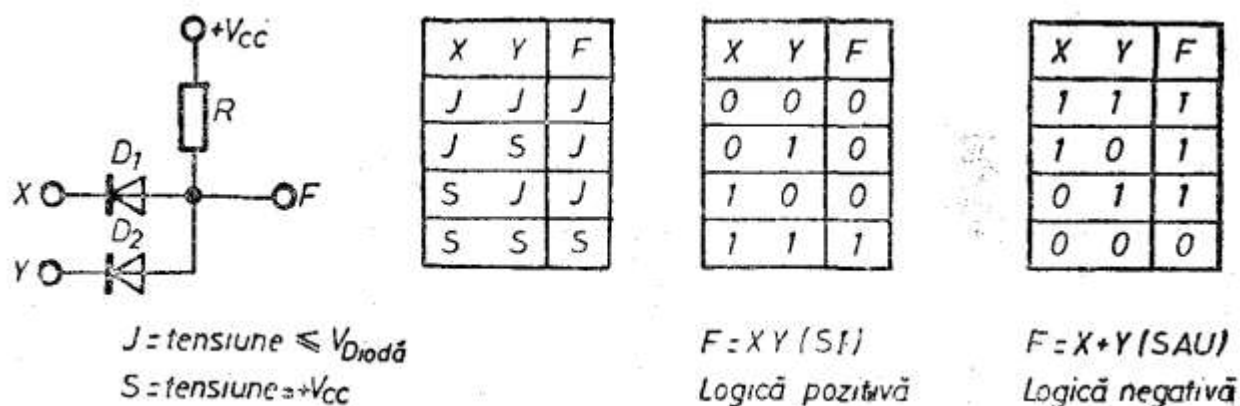


Fig. 5. Poartă logică cu diode și tabelele de adevăr în logică pozitivă și negativă.



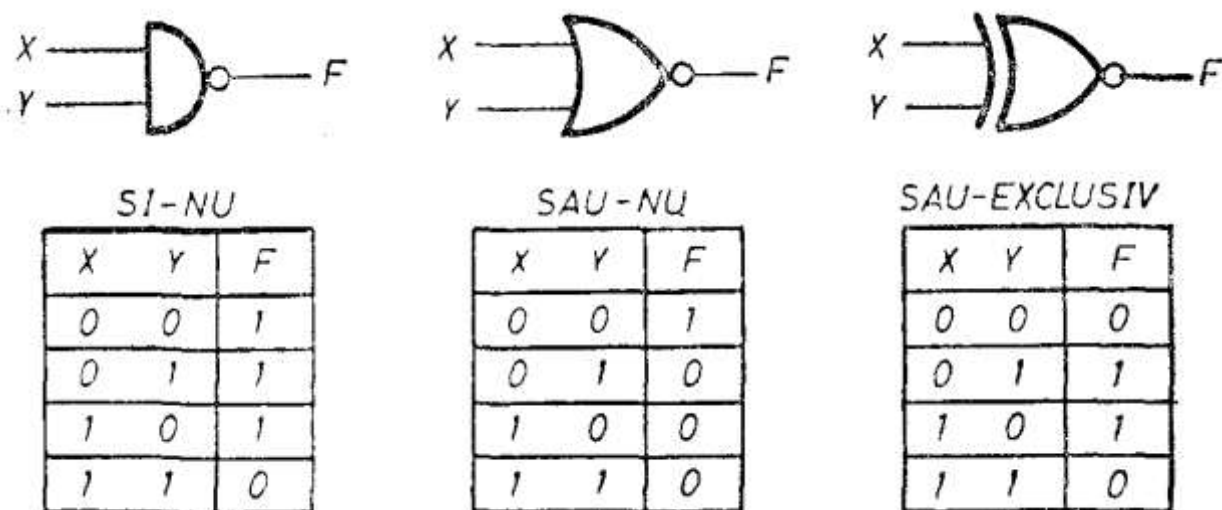


Fig. 6. Porțile **SI-NU**, **SAU-NU** și **SAU EXCLUSIV**.  
Reprezentare simbolică și tabele de adevăr.

Cu ajutorul teoremelor 17 sau 18 din tabelul 1, teoremele lui *De Morgan*, orice funcție logică se poate implementa cu ajutorul funcțiilor **SAU — NU** și **SI — NU** la care se adaugă inversoare (execută negarea).

De exemplu funcția logică din figura 2 se poate transforma astfel:

$$F = X\bar{Y} + Z = \overline{\overline{X\bar{Y} + Z}} = \overline{(\overline{X\bar{Y}}) \cdot \overline{Z}} \quad (2)$$

$$F = X\bar{Y} + Z = \overline{(\overline{X\bar{Y}}) \cdot \overline{Z}} = \overline{(\overline{X \cdot Y}) + Z} = \overline{(\overline{X} + \overline{Y}) + Z} = \overline{(\overline{X} + \overline{Y})} \cdot \overline{Z} \quad (3)$$

În figura 7a se dă implementarea funcției conformă cu (2) cu porți logice **SI—NU** și inversoare, iar în figura 7b cu forma (3) cu porți logice **SAU—NU** și inversoare.

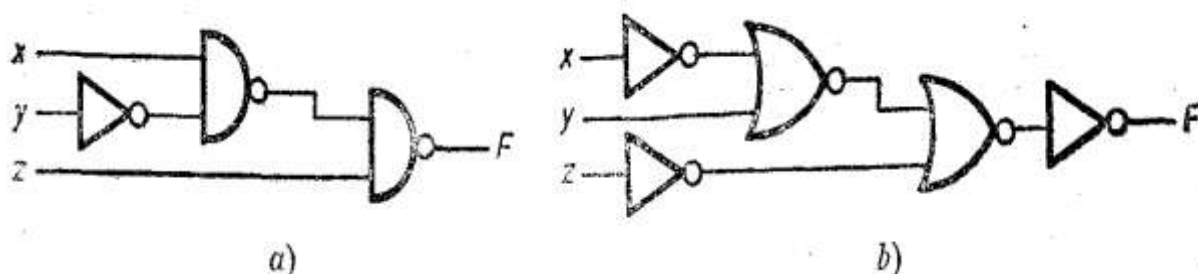


Fig. 7. Exemplu de implementare a funcției logice  $F = \overline{X\bar{Y}} + Z$  cu porți **SI-NU** și cu porți **SAU-NU**.

Pe lângă circuitele combinaționale, realizate cu porți logice, există circuitele logice secvențiale, ale căror stări, la un moment oarecare de timp, depind atât de valoarea variabilelor de la intrare, cât și de starea logică anterioară a circuitelor (circuite cu memorie). Circuitele logice secvențiale se realizează cu ajutorul circuitelor bistabile de diverse tipuri.

## 1.2.2 CIRCUITE BISTABILE

Cel mai simplu circuit bistabil, este bistabilul  $RS$ , care se poate obține în două moduri după cum se observă în figura 8.

Denumirea bistabilului  $RS$  provine de la denumirile intrărilor  $S$  (*set*=trece  $Q$  în 1) și  $R$  (*reset*=trece  $Q$  în 0). În varianta de implementare cu porți  $SI-NU$ , din figura 8a, intrările  $S$  și  $R$  acționează asupra bistabilului când sunt în starea 0. Acest lucru se vede și pe simbolul grafic al bistabilului unde la intrările  $R, S$  sunt puse cercurile ce semnifică o negare (semnal activ în 0). În varianta de implementare cu porți  $SAU-NU$ , din figura 8b), intrările  $R$  și  $S$  sunt active în starea 1. La ieșirea din stările nepermise și trecerea în stările de memorare,  $R=S=1$  pentru figura 8a) și  $R=S=0$  pentru figura 8b), bistabilul se va așeza într-o stare necontrolată, în funcție de viteza fiecăreia din porțile componente.

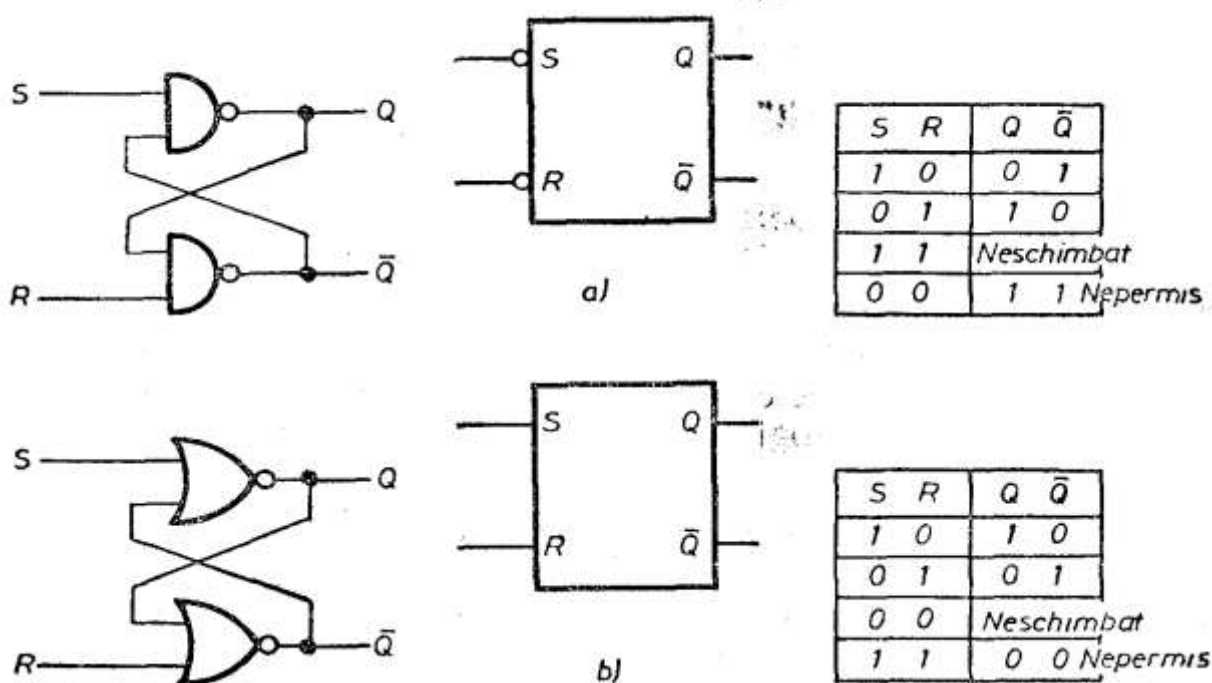


Fig. 8. Bistabile  $RS$ .

Adăugind elemente suplimentare de logică, s-au obținut îmbunătățiri ale acestui bistabil. În primul rând, a fost necesară o sincronizare a activității intrărilor, cu un impuls extern, numit impuls de tact, apoi datorită necesității ca la unele bistabile să se lege ieșirile înapoi la intrări, a fost introdusă o izolare suplimentară între intrări și ieșiri.

Acestea au fost obținute prin utilizarea unui bistabil suplimentar în serie cu cel inițial și porți de intrare sincronizate cu impulsul de tact. A fost obținut în final un bistabil „master-slave”. Schema logică a unui astfel de bistabil *RS*, implementată cu porți *SI—NU* și simbolul grafic al bistabilului sunt date în figura 9a și respectiv 9b.

În figura 9 se observă cei doi bistabili „Master” și „Slave”, fiecare având la intrare porți pentru sincronizarea intrărilor cu impulsul de tact (*T*). Bistabilul „Slave” primește impulsul de tact negat. Din acest motiv, când impulsul de tact trece în 1 informația se încarcă de pe intrări în „Master” având „Slave” blocat apoi când impulsul de tact redevine 0 informația din „Master” trece în „Slave” și deci și la ieșirile *Q* și  $\bar{Q}$ .

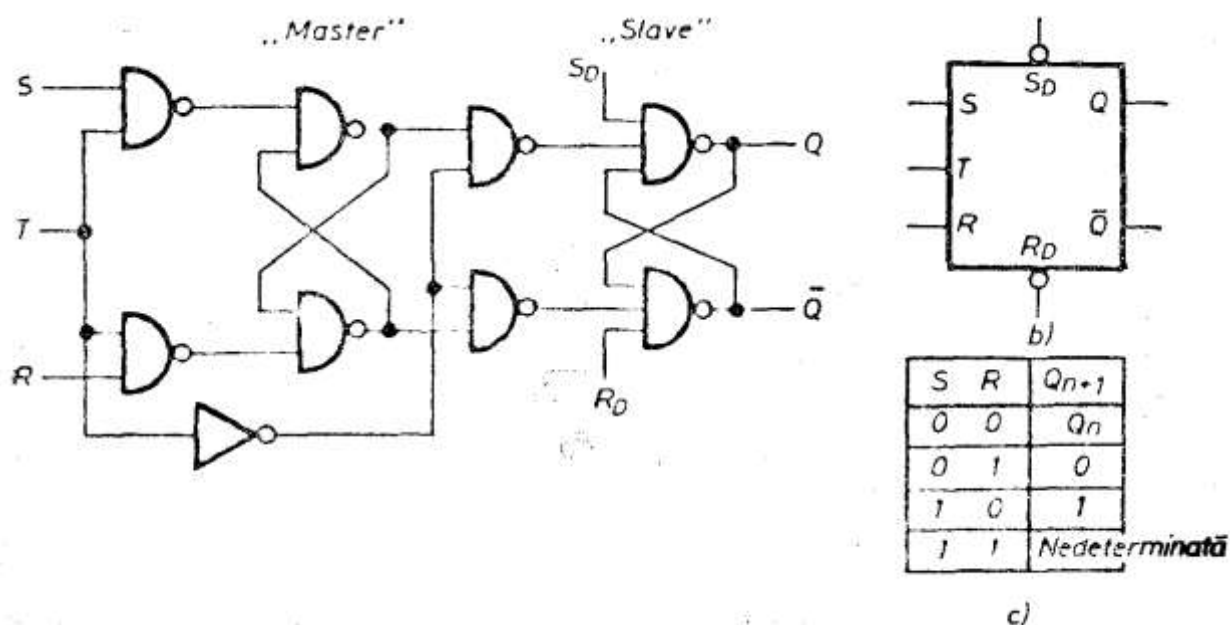


Fig. 9. Bistabil *RS* „master-slave”.

Intrările  $S_D$  și  $R_D$  sunt active în 0 și acționează direct în bistabilul „Slave”, folosindu-se pentru aducerea acestuia într-o stare dorită, *asincron* (indiferent de tact).

Notând cu  $Q_n$  și  $Q_{n+1}$  starea lui *Q* înainte și respectiv după impulsul de tact se poate scrie tabela de tranziție a bistabilului *RS* „master-slave”. Aceasta este dată în figura 9c (pentru  $S_D=R_D=1$ ).

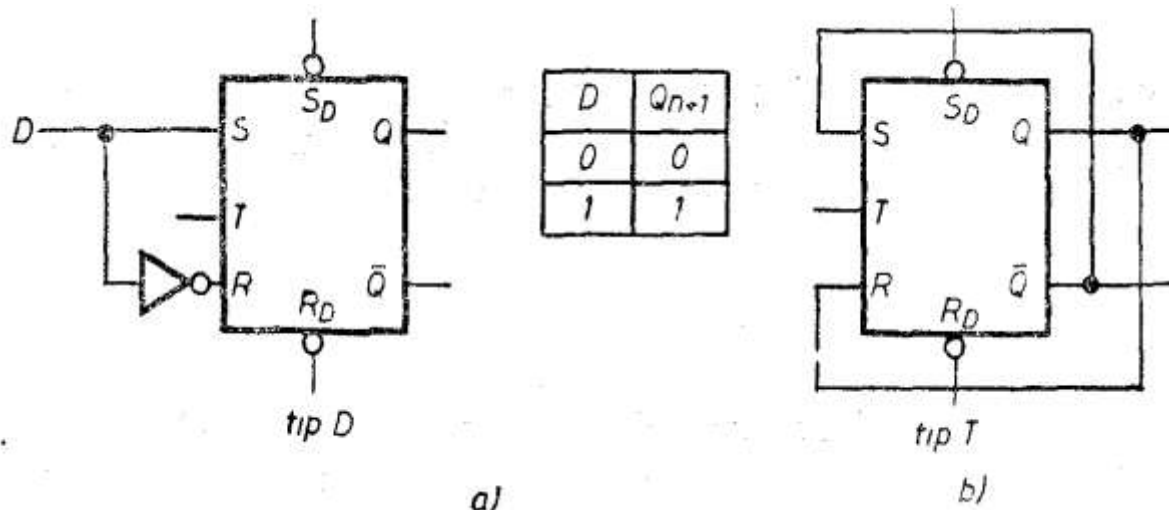


Fig. 10. Transformarea bistabilului  $RS$  în bistabili  $D$  sau  $T$ .

Din bistabilul  $RS$  se poate obține bistabilul  $D$  conform cu figura 10a. Tot din bistabilul  $RS$  se poate obține bistabilul de tip  $T$ , care schimbă starea ieșirilor după fiecare impuls de tact, putând fi considerat un divizor binar. Schema de conexiuni este dată în figura 10b.

Bistabilul cel mai complex și cel mai versatil este cel de tip  $JK$ . Schema logică este asemănătoare cu a bistabilului  $RS$ , având în plus reacții de la ieșiri la intrări. Această schemă și tabela de tranziție sunt date în figura 11. Și din bistabilul  $JK$  se pot obține bistabilii  $D$  și  $T$  ca în figura 12.

Trebuie precizat că schemele logice utilizate în circuitele integrate actuale, pot diferi sensibil de cele prezentate până acum. Printre alte deosebiri, cea mai importantă poate fi considerată acționarea multor bistabili pe frontul pozitiv al impulsului de tact. Această deosebire nu conduce însă la nici un fel de probleme în aplicații.

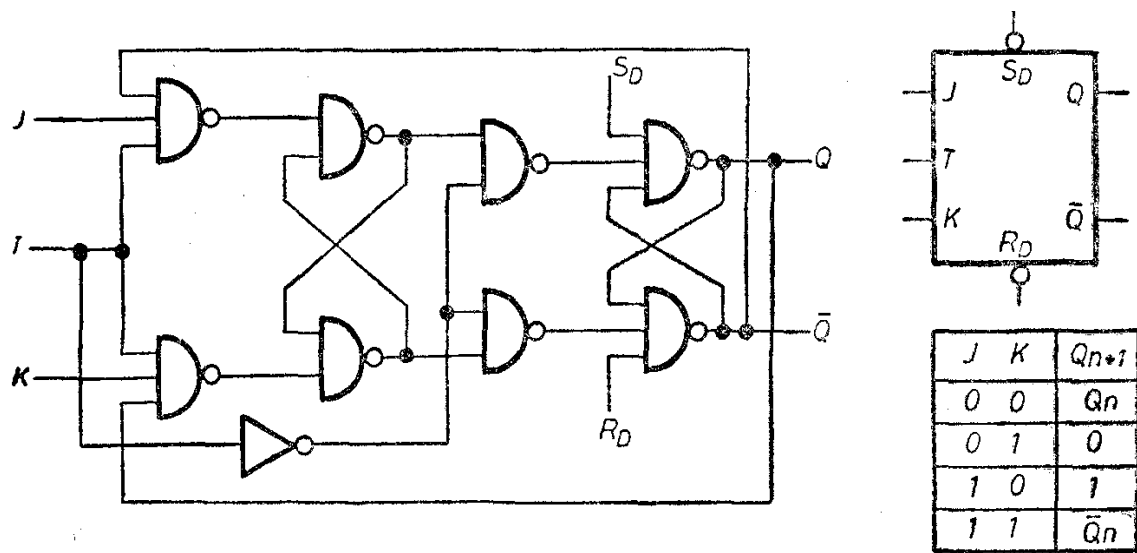


Fig. 11. Bistabil  $JK$  „master-slave”.

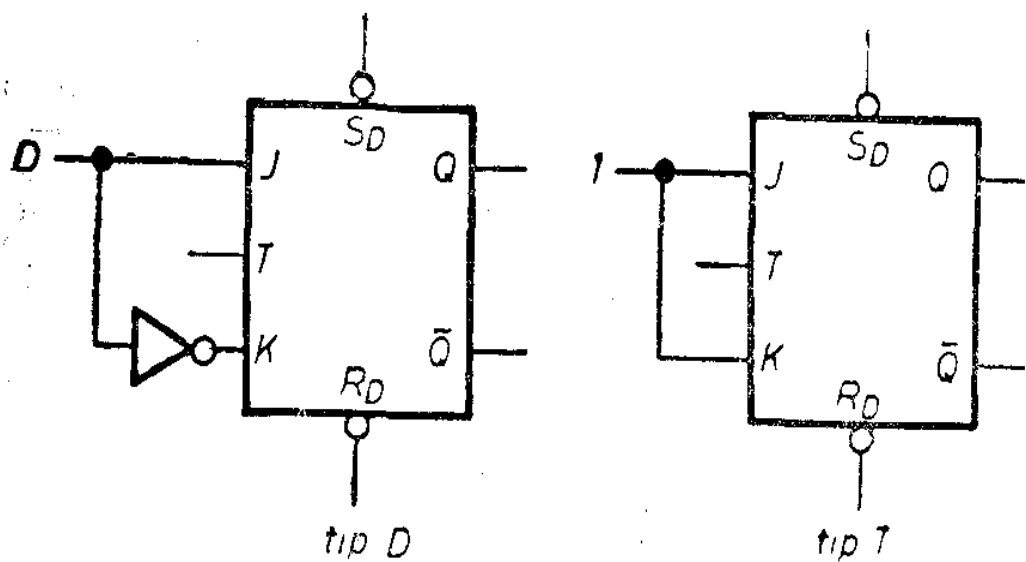


Fig. 12. Transformarea bistabilului  $JK$  în bistabili  $D$  sau  $T$ .

### 1.2.3. CARACTERISTICI DE TIMP

Aceste caracteristici se referă la capacitatea circuitelor digitale de a răspunde unor schimbări ale intrărilor sau ale impulsului de tact. În cazul porților se poate vorbi numai despre așa-numitul timp de propagare între schimbarea unei intrări și apariția schimbării corespunzătoare la ieșire. Acești timpi pot fi diferiți la trecerea ieșirii din starea „Jos” în starea „Sus” și invers. După cum se vede în figura 13.

Pentru bistabili se definesc mai multe mărimi temporale caracteristice:

- timpii de propagare ( $t_p$ ) (între frontul activ al tactului și ieșiri)
- timpii de stabilire ( $t_s$ ) (între momentul stabilirii intrărilor și frontul activ al tactului)
- timpii de reținere ( $t_r$ ) (între frontul activ al tactului și momentul în care se acceptă schimbarea intrărilor).

Acești timpi sunt ilustrați în figura 14.

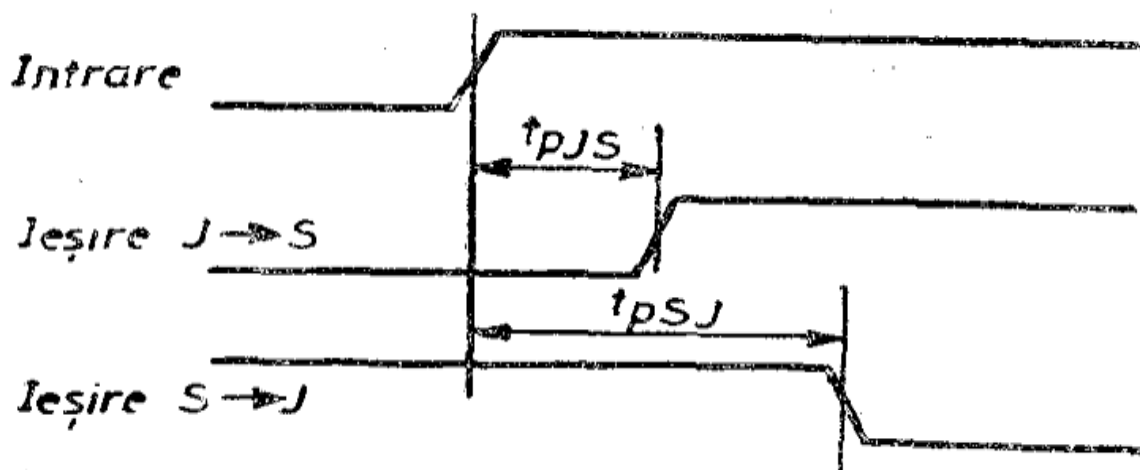


Fig. 13. Reprezentarea timpilor de propagare pentru porți logice.

Pe lângă acești timpi se mai definesc frecvența maximă și durata minimă a impulsurilor de tact.

Producătorii de circuite integrate digitale furnizează date despre toți acești timpi în gama temperaturilor de funcționare.

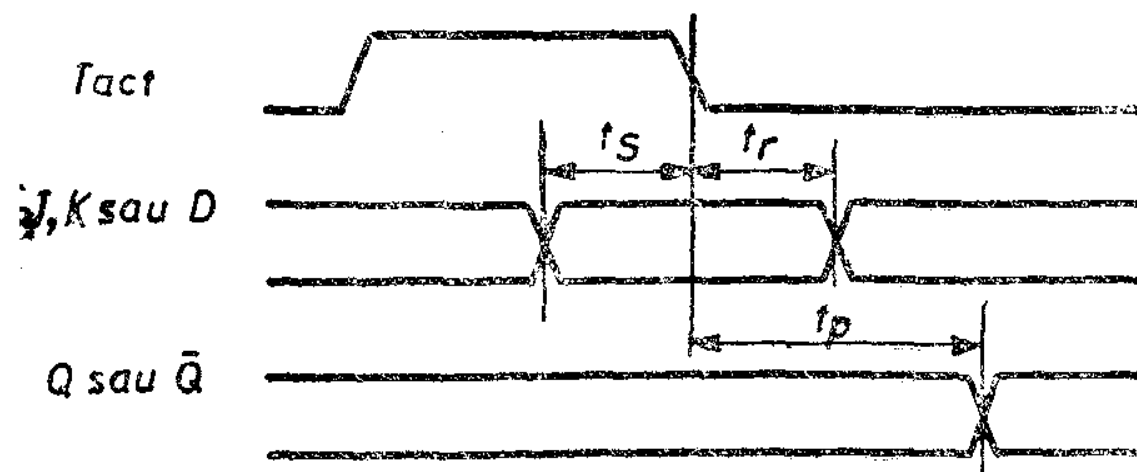


Fig. 14. Reprezentarea timpilor de propagare, de stabilire și reținere pentru bistabili.

# CAPITOLUL 1

## CIRCUITE BASCULANTE BISTABILE (CBB)

### 1. GENERALITĂȚI

Se numesc *circuite basculante bistabile* (prescurtat CBB sau bistabil), circuitele care au două stări stabile, trecerea dintr-o stare în cealaltă făcându-se numai la aplicarea unei comenzi din exterior.

Caracteristica lor principală este că ele au memorie. Aceasta înseamnă că, din examinarea semnalelor de ieșire, se poate deduce ultima comandă primită de circuit.

Datorită proprietăților lor susmenționate, își găsesc numeroase aplicații cum ar fi de exemplu: realizarea numărătoarelor, a registrelor, a memoriilor RAM etc.

Se disting următoarele tipuri de circuite bistabile:

— ***S-R***;

— ***J-K***;

— ***D***.

După natura funcționării lor ele se împart în circuite asincrone și sincrone, în cele ce urmează vom analiza în detaliu funcționarea acestor circuite.



## 2. CBB DE TIP S-R

### 2.1. CBB de tip S-R sincron

Pentru a justifica necesitatea unor **comenzi sincrone** este necesar să se examineze circuitul din fig. 15.

Se constată astfel că semnalul de comandă care se aplică pe borna  $S$  a **CBB** este:  $S=A \cdot B$ . Să considerăm că există următoarea situație  $Q_n=0$ ,  $R_n=0$ ,  $A_n=1$ ,  $B_n=0$  și deci  $S_n=0$  și că în funcționarea normală a circuitului trebuie să apară o schimbare a stărilor intrărilor  $A$  și  $B$  care însă să nu influențeze starea bistabilului. Trecerea din starea  $A_n=1$ ,  $B_n=0$  în starea  $A_{n+1}=0$ ,  $B_{n+1}=1$  se poate face fie prin comutarea semnalului de pe intrarea  $A_n$  înaintea celui de pe intrarea  $B_n$  (fig.15 b), fie prin comutarea  $B_n$  înaintea celui de pe intrarea  $A_n$  (fig. 15 c).

În primul caz starea bistabilului nu se modifică  $Q_{n+1}=Q_n=0$ , pe când în cel de-al doilea caz, datorită faptului că semnalul  $A_n$  a comutat în urma semnalului de pe  $B_n$  pe intrarea  $S$  a bistabilului apare un impuls parazit care îl trece în starea  $Q_{n+1}=1$ .

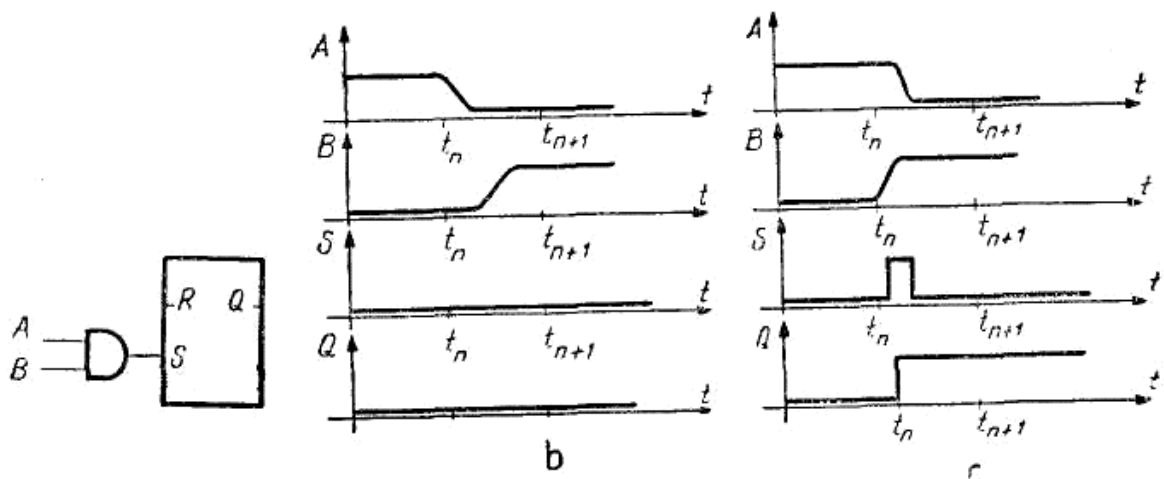


Fig. 15. Funcționarea bistabilului **S-R asincron** cu comandă aplicată pe intrarea  $S$  de la ieșirea unei porți **SI**.

Această schimbare, nedorită, a stării bistabilului poate avea implicații greu previzibile asupra funcționării ansamblului din care face parte. Cum în practică este greu de controlat succesiunea tranzițiilor, este necesar să existe un circuit care să execute comenzile numai după ce acestea s-au stabilizat la valorile lor corecte, evitându-se astfel executarea unor comenzi greșite datorate impulsurilor parazite ce pot să apară în timpul tranzițiilor. Un astfel de circuit este **CBB de tip S-R sincron**. Un **CBB de tip sincron** va avea două (una) intrări de date, o intrare de tact și două ieșiri. Informația se transmite spre bistabilul propriu-zis numai la sosirea impulsului de tact (fig.16 a și 16 b).

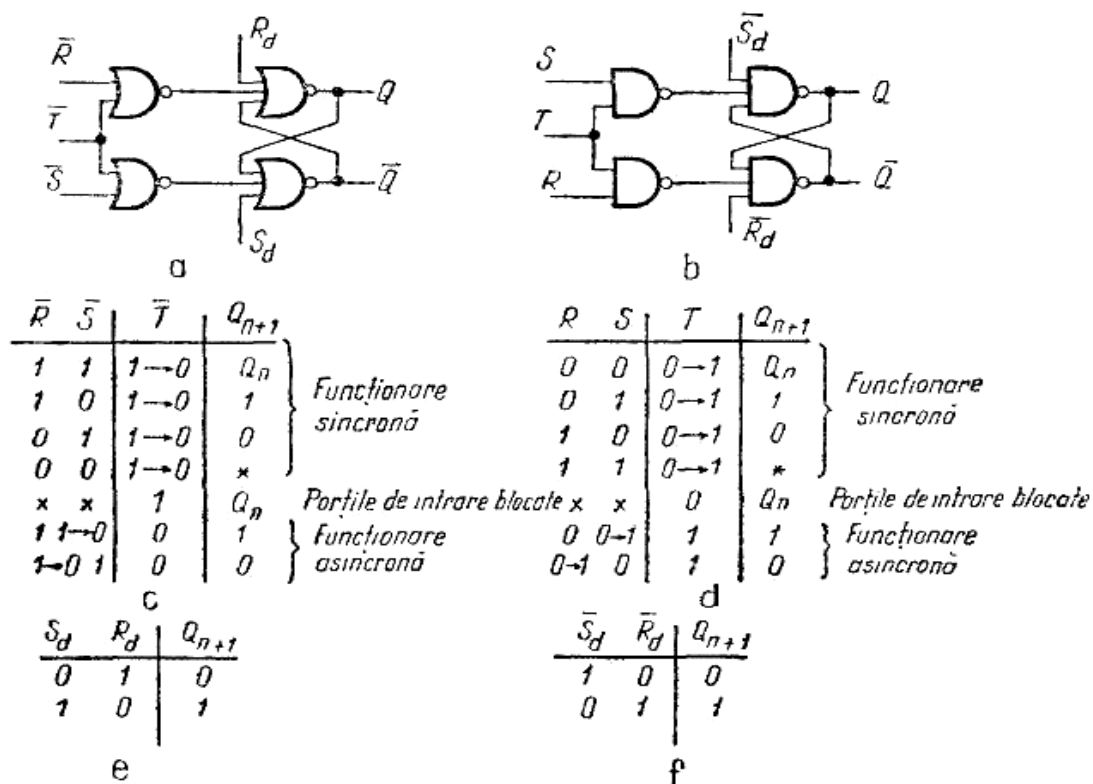


Fig. 16 **CBB de tip S-R sincron**- a) cu porți SAU-NU; b) cu porți ȘI-NU; c) și d) tabelele de adevăr pentru circuitele de la punctele a și b; e) și f) tabelele de adevăr pentru intrările asincrone ale circuitelor de la punctele a și b.

Pe durata cât porțile de intrare sunt deschise, circuitul funcționează asincron. Pentru caracterizarea acestei situații se poate introduce noțiunea de transparență în raport cu intrările de date. De exemplu, pentru bistabilul din fig.16 b pentru  $T=1$ , orice modificări ale intrărilor de date se reproduc la ieșire, momentele de tranziție fiind determinate numai de modificările intrărilor.

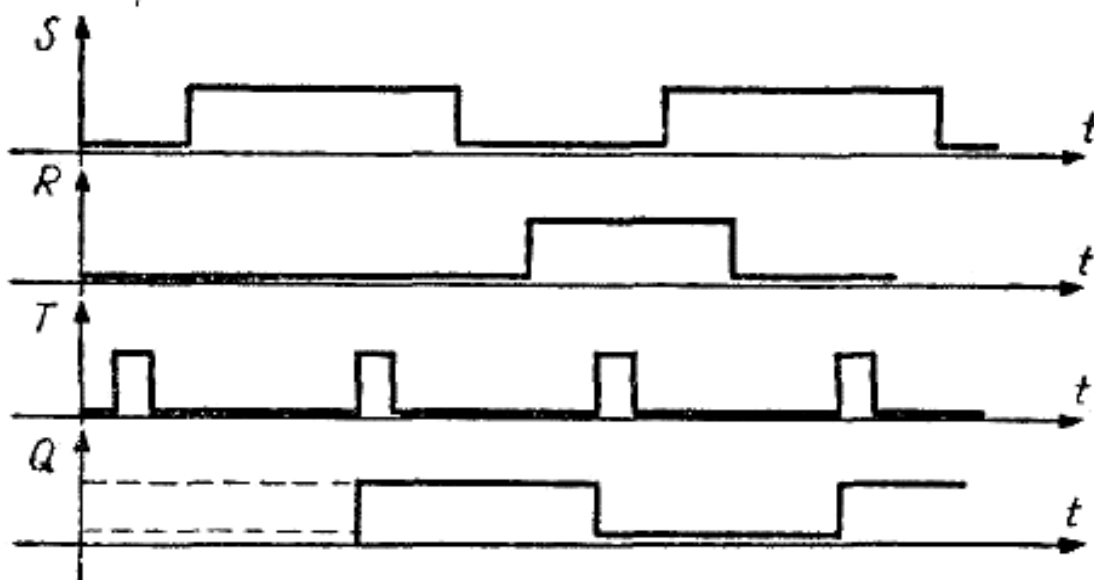


Fig. 17. Diagrame ce ilustrează funcționarea circuitului *S-R sincron*.

Pentru asigurarea funcționării sincrone trebuie evitate schimbările intrărilor de date pe durata cât porțile de intrare sunt deschise. Comenzile într-un asemenea circuit se vor executa în ritmul impulsurilor de tact. La fiecare impuls se execută câte un pas în procesul prelucrării informației, ceea ce justifică denumirea de impulsuri de ceas (fig. 17).

În afara intrărilor sincrone la aceste bistabile se introduc și una sau două intrări asincrone. Aceste intrări servesc la aducerea la 0 a bistabilului ( $R_d$  respectiv  $\bar{R}_d$ ), sau la aducerea lui în starea 1 ( $S_d$  respectiv  $\bar{S}_d$ ). Apariția unor comenzi pe aceste intrări se execută independent de prezența tactului. Din acest motiv, intrările acestea pot fi considerate prioritare în raport cu celelalte. Toate **bistabilele sincrone** sunt prevăzute cu intrare de aducere la zero (numită *Reset*), unele din ele au și o intrare, de aducere în starea 1 a bistabilului (această intrare se numește *Preset*).

## 2.2. CBB de tip S-R sincron cuplat în curent alternativ

Schema acestui circuit este prezentată în fig. 18.

Schema s-a obținut prin adăugarea porților  $P_5$ ,  $P_4$  la schema unui **CBB** de tip **S-R asincron** cuplat în curent alternativ.

Posibilitățile de folosire a circuitului sunt următoarele:

1. **CBB** de tip **S-R asincron** dacă sunt folosite intrările directe  $S_d$ ,  $R_d$ .
2. **CBB** de tip **S-R asincron** cu cuplaj în curent alternativ dacă sunt ținute deschise porțile de intrare  $P_5$ ,  $P_4$  punând  $\overline{CK} = 0$ .
3. **CBB** de tip **S-R sincron** cu cuplaj în curent alternativ.

Va fi examinat cazul 3, deoarece celelalte două nu formează subiectul prezentului proiect.

Dacă impulsul de tact  $\overline{CK}$  are o tranziție din 1 în 0, atunci în funcție de stările  $\overline{S}_D$  și  $\overline{R}_D$ , această tranziție se transmite la ieșirea porților  $P_5$  și  $P_4$ , determinând apariția unor impulsuri negative pe intrările  $R'$  sau  $S'$ , ceea ce nu are nici un efect asupra bistabilului.

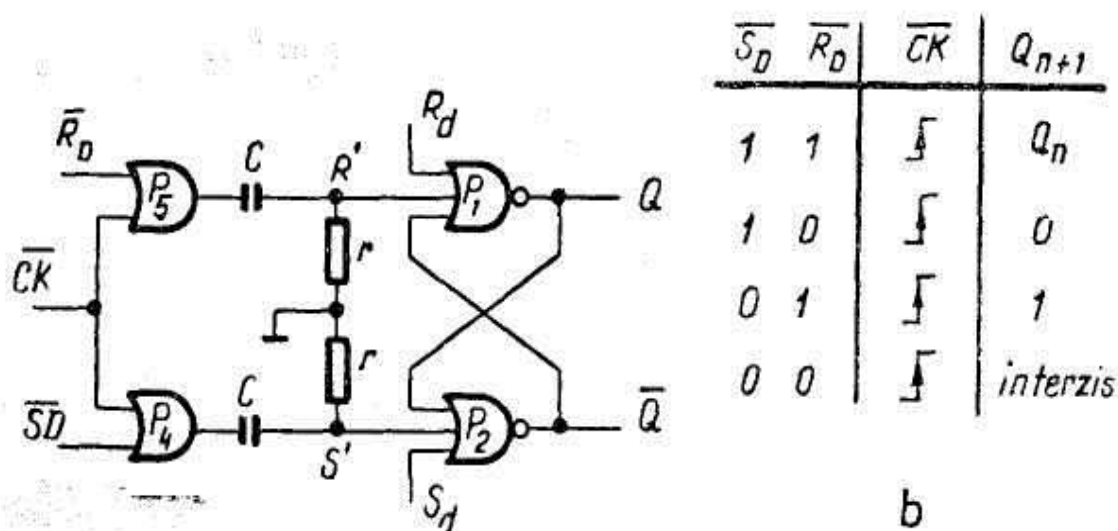


Fig. 18. **CBB** de tip **S-R sincron** cuplat în ca.

În cazul în care impulsul de tact trece din 0 în 1, atunci această tranziție, în funcție de comenzile  $\overline{S}_D$  și  $\overline{R}_D$  apare pe ieșirea uneia din porțile  $P_5$  sau  $P_4$ , ceea ce duce la apariția unor tranziții pozitive pe intrările  $R'$  sau  $S'$ . Acest fapt va avea ca efect bascularea circuitului.

De exemplu dacă  $\overline{R}_D=1$  și  $\overline{S}_D=0$ , atunci poarta  $P_5$  este blocată, iar  $P_4$  deschisă. În acest caz tranziția  $0 \rightarrow 1$  a impulsului de tact se transmite numai prin poarta  $P_4$ , ieșirea lui  $P_5$  rămânând neschimbată. Tranziția de la ieșirea porții  $P_4$  determină apariția unui impuls pozitiv pe intrarea  $S'$  ceea ce va determina trecerea bistabilului în starea  $Q=1$ . În mod similar se pot analiza și celelalte situații din tabelul din fig. 18 b.

În legătură cu funcționarea *sincronă* se pot face următoarele observații :

1. Având în vedere că basculările se produc la tranzițiile  $0 \rightarrow 1$  ale impulsului de tact, atunci când porțile de intrare  $P_5$ ,  $P_4$  trec din starea deschisă în cea închisă, este posibilă conectarea în cascadă a unor astfel de bistabile (funcționarea din acest punct de vedere este similară cu aceea a bistabilului *Master-Slave*).

2. Pentru declanșarea circuitului este necesar ca frontul impulsului de tact să aibă o durată mai mică decât o anumită valoare limită.

3. Modificările semnalelor de comandă  $\overline{S}_D$  și  $\overline{R}_D$  trebuie făcute numai în timpul cât porțile de intrare  $P_5$  și  $P_4$  sunt blocate.

### 3. CBB de tip J-K

#### 3.1 CBB de tip J-K sincron

Schema acestui circuit se obține din cea a **CCB** de tip **J-K asincron**, prin introducerea unei borne de tact, care să controleze porțile de intrare (fig. 19 a).

Tabelul de adevăr al circuitului este reprezentat în fig. 19 b. Este indicat să fie examinată mai în detaliu situația  $J_n = K_n = 1$ . Se presupune că există starea  $Q = 1$ , caz în care, cele două ieșiri  $Q$  și  $\bar{Q}$  asigură deschiderea porții  $P_1$  și blocarea porții  $P_2$ . Atâta timp cât  $CK = 0$  semnalele  $S'$  și  $R'$  vor fi în zero logic ( $S' = R' = 0$ ). Dacă pe borna de tact ( $CK$ ) semnalul trece din 0 în 1, pe ieșirea porții  $P_1$  se va obține semnalul  $R' = 1$ , iar pe ieșirea porții  $P_2$  semnalul  $S' = 0$ , ceea ce determină trecerea circuitului în starea  $Q = 0$ .

O dată cu trecerea circuitului în această stare, poarta  $P_1$  se va bloca și poarta  $P_2$  se va deschide. Dacă între timp impulsul de tact nu a dispărut ( $CK = 1$ ) pe ieșirea porților apar următoarele semnale:  $R' = 0$  și  $S' = 1$ . Această comandă va determina trecerea circuitului în starea  $Q = 1$  ș.a.m.d.

În concluzie, pe durata tactului ( $CK = 1$ ) și cu comanda  $J_n = K_n = 1$ , circuitul trece singur dintr-o stare în alta (oscilează).

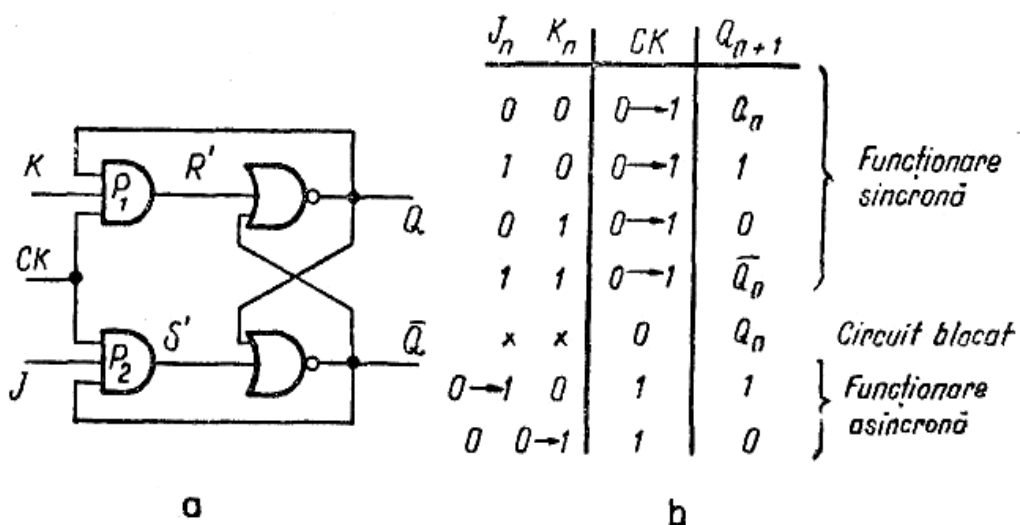


Fig. 19. CBB de tip J-K sincron: a) schema logică; b) tabelul de adevăr.

### 3.1. CBB de tip J-K sincron pe front

Există o gamă largă de circuite basculante sincronice pe front. Diversitatea este determinată de tehnologia în care se realizează circuitul, precum și de frontul pe care basculează.

Pentru exemplificare, se va examina schema unui **CBB** de tip **J-K sincron pe front negativ** (trecere  $1 \rightarrow 0$ ) realizat în tehnologia **TTL**. Schema circuitului este prezentată în fig. 20.

Funcționarea circuitului se bazează pe întârzierile interne ale porților. Notând cu  $t_p$  timpul de propagare printr-o poartă, atunci schema va funcționa numai dacă întârzierea porților  $P_1, P_2$  este de cel puțin  $4 t_p$ .

Se va examina funcționarea circuitului pentru comanda  $J=1, K=0$  presupunând că inițial circuitul se află în starea  $Q=0, \bar{Q}=1$  (fig. 20 b). Pentru  $CK=0$ , se obțin următoarele stări logice:  $X=1, Y=1, P=1, R=0, U=0$  și  $V=0$ .

La trecerea impulsului de tact din 0 în 1, în circuit se produc următoarele schimbări:

— după o întârziere egală cu  $t_p$ , se schimbă ieșirea porții  $P_4$  ( $R$  trece din 0 în 1) iar după o întârziere  $4 t_p$  (caracteristică porții  $P_1$ ) semnalul  $X$  trece din 1 în 0.

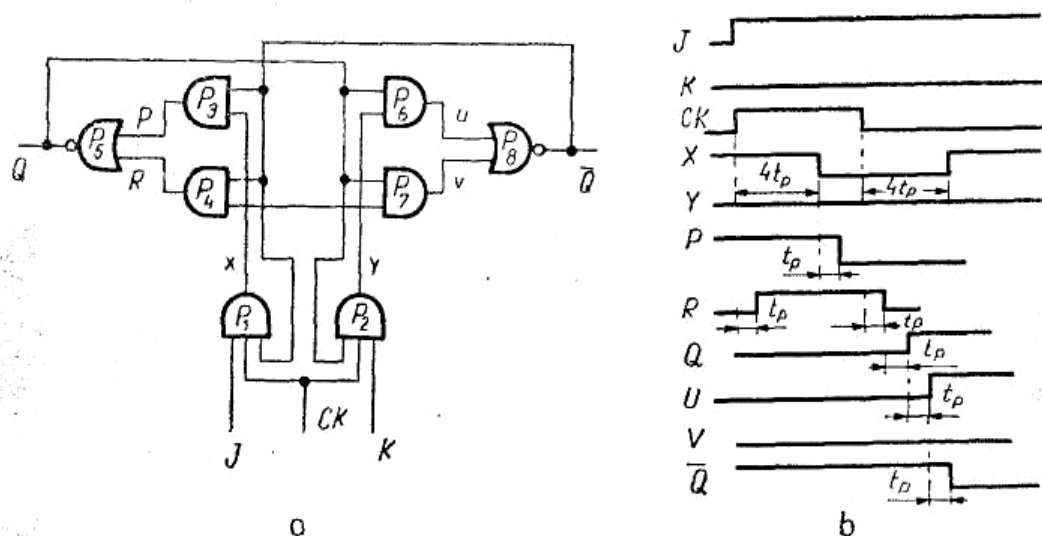


Fig. 20. **CBB** de tipul **J-K sincron pe front**: a) schema logică; b) diagrame care ilustrează funcționarea.

— apariția saltului  $1 \rightarrow 0$  pe ieșirea  $X$  determină modificarea semnalului de la ieșirea porții  $P_3$ , iar semnalul  $P$  trece din  $1$  în  $0$  cu o întârziere  $t_p$ .

Trecerea impulsului de tact din  $1$  în  $0$  antrenează după sine următoarele modificări de semnale:

— cu o întârziere  $t_p$ , la ieșirea porții  $P_4$  apare o tranziție  $1 \rightarrow 0$ , tranziție care se propagă prin poarta  $P_5$  și după o nouă întârziere  $t_p$  determină apariția unui salt  $1 \rightarrow 0$  pe ieșirea acesteia ( $Q$  trece în starea  $1$ );

— după o nouă întârziere  $t_p$  apare o tranziție  $0 \rightarrow 1$  pe ieșirea porții  $P_6$  după care cu o întârziere  $t_p$  se modifică ieșirea porții  $P_8$  ( $\bar{Q}$  trece în starea  $0$ );

— după trecerea unui timp egal cu  $4t_p$  de la apariția trecerii  $1 \rightarrow 0$  a impulsului de tact, se schimbă ieșirea porții  $P_1$  (semnalul  $X$  trece din  $0$  în  $1$ ).

Dacă schimbarea semnalului  $X$  s-ar produce mai repede decât  $4 t_p$ , acest lucru ar duce la apariția unei tranziții  $0 \rightarrow 1$  pe ieșirea porții  $P_3$ , ceea ce ar menține în continuare semnalul  $Q$  pe zero și deci, bascularea circuitului nu s-ar mai produce.

După această schimbare a semnalului de la ieșirea porții  $P_1$  nivelele tuturor semnalelor rămân constante până la apariția unui nou impuls de tact.



## 4. CBB DE TIP D

### 4.1. CBB de tip D sincron

Dacă la o celulă binară  $S$ - $R$  sincronă se introduce un inversor în scopul de a se realiza  $D=S=\bar{R}$ , se obține celula binară numită cu *zăvorâre* ( $D$ -Latch) prezentată în fig. 21 a.

Circuitul prezintă caracteristica principală că pe toată durata tactului ( $CK=1$ ) ieșirea copiază intrarea. În momentul dispariției impulsului de tact (trecere  $1 \rightarrow 0$ ) porțile de intrare  $P_1$ ,  $P_2$  se închid și bistabilul menține la ieșirea sa starea din momentul dispariției impulsului de tact (vezi fig. 21 b).

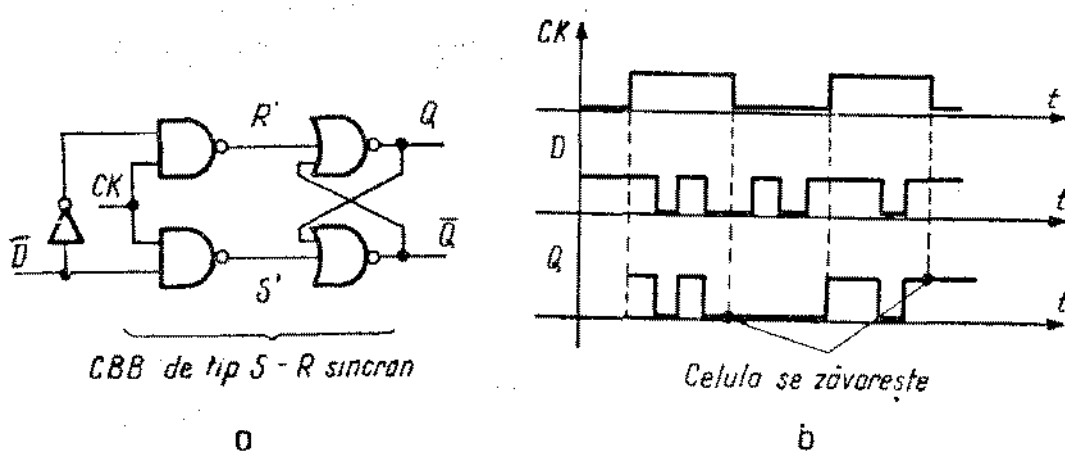


Fig.21. **CBB** de tip **D-Latch**: a) schema logică; b) diagrame de funcționare.

#### 4.4. CBB de tip *D* sincron pe front

O variantă de celulă binară de tip *D* sincronă pe front pozitiv (trecere  $0 \rightarrow 1$ ) este reprezentată în fig. 22.

Celula copiază la ieșirea  $Q$  ceea ce este pe intrarea  $D$  în momentul trecerii  $0 \rightarrow 1$  a impulsului de tact ( $CK$ ). Modificările intrării de date  $D$  cât timp  $CK=0$  sau  $CK=1$  nu au nici-o influență asupra semnalului de ieșire.

Pentru a urmări funcționarea circuitului se presupune pentru început că  $CK=0$  și  $D=0$ . În acest caz, ieșirile porților vor fi în următoarele stări:  $P_2$  în 1 ( $\bar{S}=1$ );  $P_3$  în 1 ( $\bar{R}=1$ );  $P_4$  în 1, iar  $P_1$  în 0 (stări subliniate în fig. 22).

Având în vedere că  $\bar{S}=\bar{R}=1$ , starea bistabilului de bază realizat cu porțile  $P_5$  și  $P_6$ , nu se schimbă, adică  $Q_{n+1}=Q_n$ . Să presupunem acum că în acest moment apare o tranziție  $0 \rightarrow 1$  a impulsului de tact: având în vedere că poarta  $P_2$  este blocată de ieșirea porții  $P_1$  înseamnă că modificarea impulsului de tact nu afectează semnalul  $\bar{S}$  care rămâne  $\bar{S}=1$ . În schimb, semnalul de ieșire al porții  $P_3$  se modifică, trecând din 1 în 0, deci pe intrarea  $\bar{R}$  ia naștere o tranziție  $1 \rightarrow 0$  care comandă trecerea bistabilului în starea  $Q=0$ .

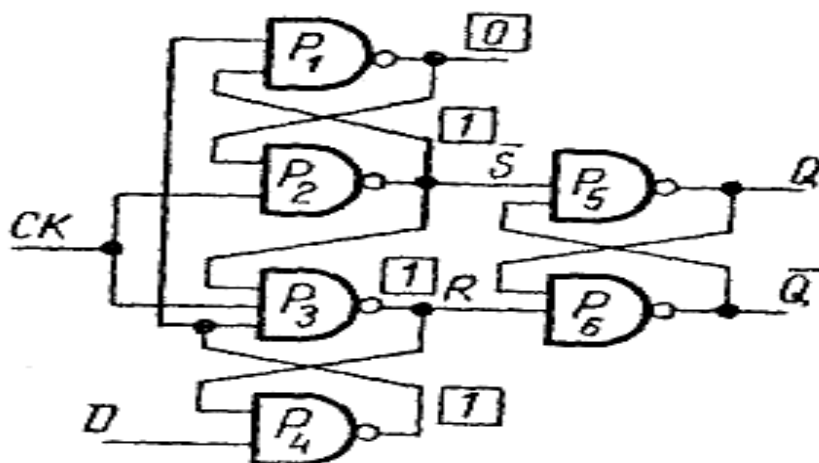


Fig. 22. CBB de tip *D* sincron pe front.

Același semnal al porții  $P_3$  va bloca poarta  $P_4$  astfel încât orice modificări ale semnalului de pe intrarea  $D$  nu vor afecta starea bistabilului de bază.

Practic, din această categorie de bistabile se fabrică curent celulele ***D-Latch*** și cele ***sincrone pe front***. Celula ***D Master-Slave*** se poate obține foarte ușor din celula ***J-K Master-Slave*** care se fabrică în mod curent.

## CAPITOLUL 2

### NUMĂRĂTOARE

#### 2.1. GENERALITĂȚI, DEFINIȚII

Pentru echipamentele de radiocomunicații numărătoare și registrele de deplasare, realizate în general cu circuite integrate, au o importanță deosebită, legată mai ales de utilizarea lor în sintetizoarele de frecvență, în circuitele de adaptare automată a antenei etc.

##### 2.1.1. CODURI BINARE

Circuitele digitale cu cele două stări, 0 și 1, pot fi folosite pentru reprezentarea numerelor în orice bază de numerație. Pentru a face această reprezentare este necesară adoptarea unui cod binar, care reprezintă legea de transformare, între număr și reprezentarea sa binară. În genere legea de codificare se prezintă sub forma ponderilor alocate pentru fiecare bit din reprezentarea binară. De exemplu, utilizând un cod binar în care biții au ponderile 8 ( $2^3$ ), 4 ( $2^2$ ), 2 ( $2^1$ ) și 1 ( $2^0$ ) numărul 9 (mai mic de 16) se poate scrie: 1001 :

$$9 = 1.8 + 0.4 + 0.2 + 1.1$$

Codul binar cu aceste ponderi se numește codul binar natural și este dat în tabelul 2.1.

Pe lângă acest cod prezintă interes și alte coduri, care datorită unor proprietăți speciale sunt preferate în anume aplicații. Tot în tabelul 2.1 se dă codul binar reflectat (codurile 7 cu 8, 6 cu 9, ... 0 cu 15 au diferit numai primul bit) de tip *Gray*.

Codul *Gray* este un cod progresiv (în care de la o stare la alta nu se schimbă decât un singur bit). În acest fel, posibilitatea apariției unor stări intermediare incorecte, datorate vitezei de comutare diferită a elementelor este exclusă.

Cât privește codurile zecimale (**ZCB** — *zecimal codificat binar*), care de departe sunt cele mai des întâlnite, în tabelul 2.2 sunt date cele mai frecvent utilizate.

*Tabelul 2.1*  
Coduri binare

<i>Echivalent zecimal</i>	<i>Cod binar natural</i>	<i>Cod Gray</i>
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 1
3	0 0 1 1	0 0 1 0
4	0 1 0 0	0 1 1 0
5	0 1 0 1	0 1 1 1
6	0 1 1 0	0 1 0 1
7	0 1 1 1	0 1 0 0
8	1 0 0 0	1 1 0 0
9	1 0 0 1	1 1 0 1
10	1 0 1 0	1 1 1 1
11	1 0 1 1	1 1 1 0
12	1 1 0 0	1 0 1 0
13	1 1 0 1	1 0 1 1
14	1 1 1 0	1 0 0 1
15	1 1 1 1	1 0 0 0

Pe lângă codul zecimal codificat binar natural, cu ponderile 8, 4, 2, 1 se dau în tabelul 2.2 și alte coduri: *biquinar* (5, 4, 2, 1), (2, 4, 2, 1), exces 3 și complement față de 9. Codul exces 3 se obține din codul binar natural (vezi tabelul 2.1) din care se renunță la primele 3 și ultimele 3 stări. Codul complement față de 9 al unui număr  $N$  (mai mic de 10) se obține codificând binar natural (8, 4, 2, 1) numărul  $9-N$ .

Bineînțeles că pentru codificarea unor numere zecimale de mai multe cifre se utilizează aceeași codificare pentru fiecare cifră separat.

*Tabelul 2.2*  
Coduri zecimale

Echivalent zecimal	8421	5421	2421	Exces 3	Complement față de 9
0	0000	0000	0000	0011	1001
1	0001	0001	0001	0100	1000
2	0010	0010	0010	0101	0111
3	0011	0011	0011	0110	0110
4	0100	0100	0100	0111	0101
5	0101	1000	1011	1000	0100
6	0110	1001	1100	1001	0011
7	0111	1010	1101	1010	0010
8	1000	1011	1110	1011	0001
9	1001	1100	1111	1100	0000

Conversia de la un cod la altul, în practică, se realizează cu circuite integrate specializate sau cu circuite combinaționale cu porți logice integrate. Un convertor pentru o cifră de la cod zecimal codificat binar natural la codul complement față de 9 este dat în figura 2.1. Acest convertor se poate utiliza pentru oricâte cifre are numărul care trebuie convertit.

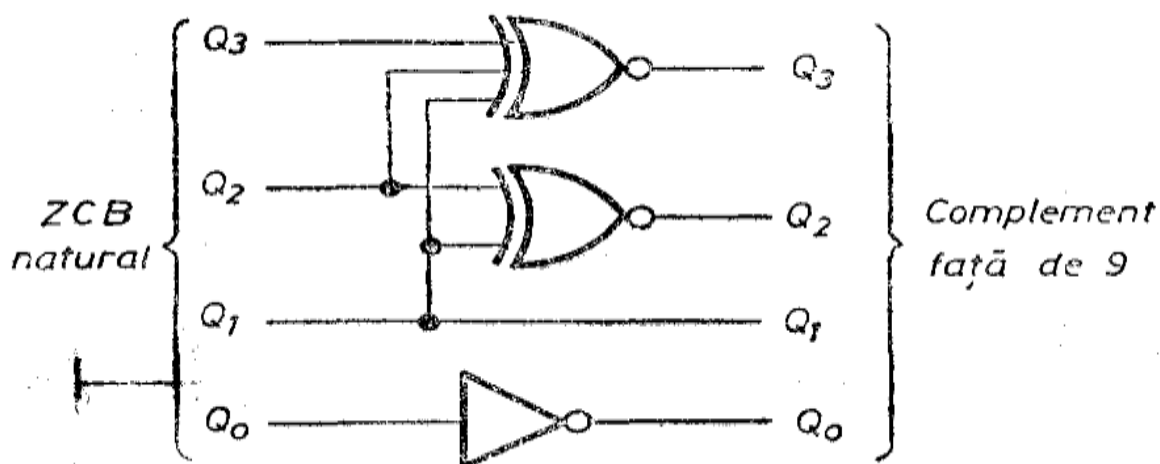


Fig. 23. Conversia cod **ZCB** natural la cod Complement față de 9.

i  
Un numărător este un circuit electronic care numără impulsurile aplicate la intrarea sa. Aceste circuite pot fi clasificate după mai multe criterii.

A. După modul în care își modifică conținutul există:

— *numărătoare directe* caracterizate prin faptul că își cresc conținutul cu câte o unitate la fiecare impuls aplicat la intrare;

— *numărătoare inverse* la care conținutul scade cu câte o unitate la fiecare impuls aplicat la intrare;

— *numărătoare reversibile* care numără în sens direct sau invers în funcție de o comandă aplicată din exterior.

B. După modul de funcționare există:

— *numărătoare asincrone*. Un astfel de numărător se caracterizează prin faptul că celulele binare din care este constituit numărătorul nu comută simultan sub acțiunea unui impuls de tact comun aplicat tuturor celulelor;

— *numărătoare sincrone*. În cazul unui astfel de numărător toate celulele binare din care este constituit numărătorul comută simultan sub acțiunea unui impuls de tact aplicat tuturor celulelor.

Înainte de începerea unei tratări sistematice sunt necesare câteva precizări utile:

1. *Numărătoarele electronice* se realizează cu celule binare de tip *T*. Proprietatea esențială a acestei celule este aceea că realizează o divizare cu 2.

Prin interconectarea adecvată a „*n*” astfel de celule se va obține schema unui numărător care poate fi privit și ca un circuit secvențial cu un număr de stări distincte. Fiecărei stări îi vom putea asocia câte un cuvânt de cod binar de lungime *n*, reprezentând conținutul celor *n* celule binare pentru starea dată a numărătorului. În consecință, codul în care numără un numărător va fi dat de succesiunea cuvintelor de cod binare asociate stărilor numărătorului. Alegerea codului în care va număra numărătorul este dictată în general de aplicația ce urmează a i se da numărătorului.

Cele mai frecvent utilizate coduri sunt: codul binar natural, codul binar reflectat, coduri **BCD** etc.

Așadar, vom putea clasifica numărătoare și din acest punct de vedere în numărătoare binare, zecimale etc.

2. Numărul stărilor distincte posibile ale unui numărător format din  $n$  celule binare este  $2^n$ . De multe ori însă din cele  $2^n$  stări posibile se sar un număr de  $k$  stări rezultând un numărător cu  $p=2^n-k$  stări distincte. Deci, din punct de vedere matematic operația realizată de un numărător este o operație *modulo*  $2^n$  sau *modulo*  $p$ .

3. Vom defini capacitatea unui numărător ca fiind egală cu numărul stărilor distincte pe care le are.

4. Revenirea numărătorului în starea inițială (de obicei asociată cuvântului de cod  $000 \dots 0$ ) este însoțită de apariția unui impuls (tranziție activă) pe ieșirea acestuia.

Vom defini factorul de divizare al numărătorului prin raportul dintre numărul impulsurilor de la intrare și numărul impulsurilor de la ieșire.

Numărătoarele sunt construite, utilizând drept elemente de bază diversele tipuri de bistabili prezentați anterior. Rolul acestora constă în numărarea diverselor evenimente apărute în schema în care sunt utilizate.

Starea la un moment dat a numărătorului poate fi decodată, ea reprezentând un anume cod. În acest gen de aplicații avem de-a face cu numărătoare propriu-zise. Dacă se utilizează numai una dintre ieșirile numărătorului unde frecvența este de  $N$  ori mai mică decât frecvența impulsurilor de tact, avem de-a face cu un divizor de frecvență.

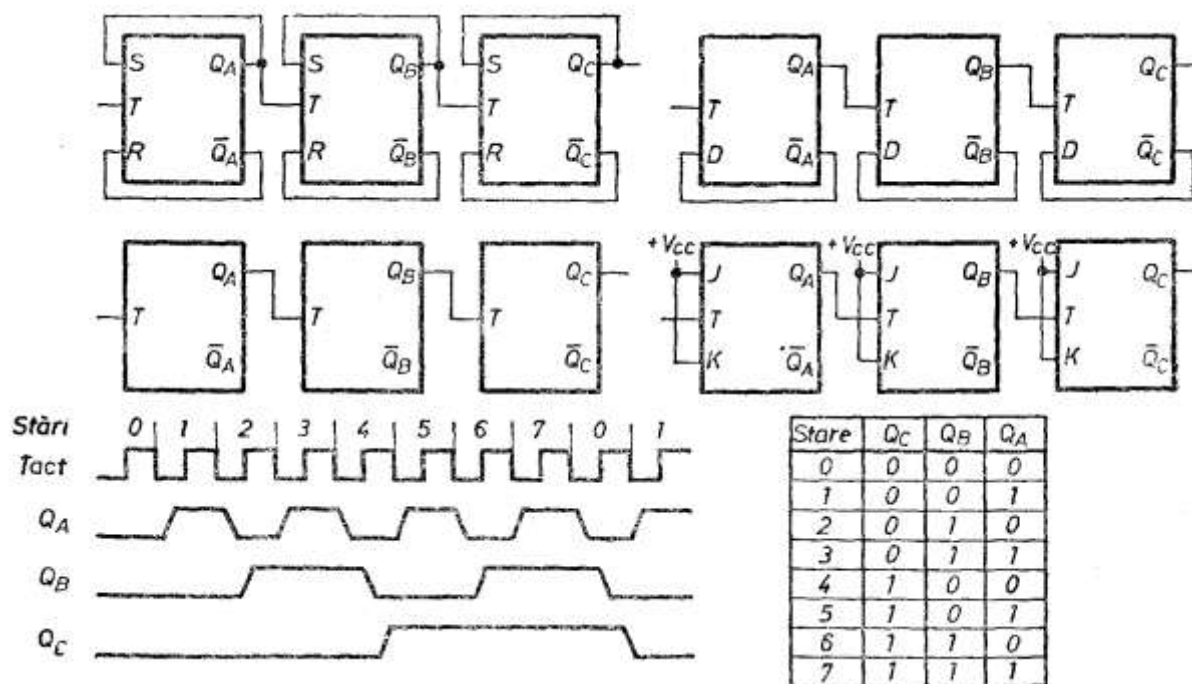


Fig. 24. Exemple de numărătoare asincrone cu bistabili **RS**, **T**, **D** și **JK**.

**Numărătoare binare asincrone.** În aceste numărătoare, stările bistabililor care le compun nu se schimbă sincron, în funcție de impulsurile de tact aplicate la intrare, ci pe rând, de la bistabilul care reprezintă bitul cel mai puțin semnificativ, către biții cu ponderi mai mari. Această modalitate de schimbare a stărilor, este rezultatul faptului că impulsurile de tact pentru bistabilii cu ponderi mai mari, sunt obținute de la ieșirile bistabililor cu ponderile imediat inferioare. În figura 24 sunt date mai multe modalități de realizare, pentru un numărător binar asincron cu trei biți, utilizând diversele tipuri de bistabili. În această figură se consideră că avem bistabili de tip „*master-slave*” indiferent dacă este vorba de bistabilii *RS*, *T*, *D* sau *JK*, și deci tranzițiile de la 1 la 0 pe intrările de tact (*T*) determină bascularea. Formele de undă și tabela de stări a numărătoarelor sunt date de asemenea în figura 24. Bitul cel mai semnificativ este *C* iar cel mai nesemnificativ este *A*. Codul de numărare este cel binar natural.

Se observă că formele de undă pentru ieșirile  $Q_A$ ,  $Q_B$  și  $Q_C$  sunt întârziate progresiv unele față de altele, ajungându-se ca în cazul unor numărătoare cu mulți biți întârzierea totală de la intrare la ieșire să fie chiar câteva perioade de tact, mai ales dacă frecvența de tact este suficient de apropiată de frecvența maximă de lucru a bistabililor componenți.

O problemă care derivă din cele prezentate mai sus, este legată de posibilitatea de a decoda diverse stări ale numărătorului. Astfel, dacă se dorește să se obțină un semnal la trecerea numărătorului prin starea 4 se va folosi o poartă cu trei intrări prezentată în figura 25.

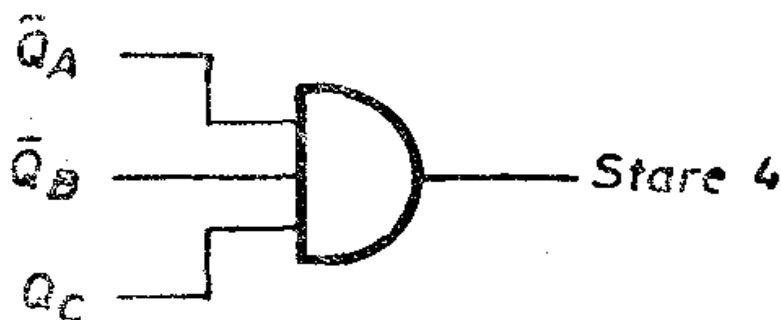


Fig. 25. *Decodificator pentru starea 4.*

Studiind formele de undă din figura 24, se poate observa că, în afară de semnalul obținut la trecerea prin starea 4, se va obține, datorită întârzierilor, un semnal scurt și la trecerea din starea 5 în starea 6 ( $Q_A=Q_B=0$  și  $Q_C=1$ ).





*Tabelul 2.3*  
*Modurile de comandă pentru numărătorul 7490*

<i>R0(1)</i>	<i>R0(2)</i>	<i>R9(1)</i>	<i>R9(2)</i>	<i>Stare numărător</i>
1	1	0	X	0 0 0 0
1	1	X	0	0 0 0 0
X	X	1	1	1 0 0 1
X	0	X	0	Numără
0	X	0	X	Numără
0	X	X	0	Numără
X	0	0	X	Numără

Trebuie precizat că utilizarea acestei tehnici de realizare a numărătoarelor prin *N asincrone*, trebuie făcută cu circumspecție, existând posibilitatea unei funcționări eronate. Datorită întârzierilor, după cum am văzut, este posibil ca starea finală să se decodeze greșit. Se poate, de asemenea, ca odată decodată corect starea finală și aplicat impulsul de readucere în starea inițială (pe intrările  $R_D$  sau  $S_D$  ale bistabililor) unul dintre bistabili să-și modifice starea, înainte ca ceilalți să aibă timpul să o facă, fapt ce conduce la dispariția impulsului de readucere în starea inițială și deci numărătorul va reveni într-o stare incorectă, cu rezultatul că raportul de divizare nu va fi cel dorit.

**Tabelul 2.4**  
**Numărătoare cu N realizate cu numărătorul 7490**

N	Configurație	Observații
2		Secțiunea de divizor cu 5 neutilizată
3		ZCB pe QA și QB
4		ZCB pe QA și QB
5		ZCB pe QB, QC și QD
6		ZCB pe QA, QB și QC
7		Resetare în 9, nu rezultă cod ZCB
8		ZCB pe QA, QB, QC și QD
9		ZCB pe QA, QB, QC și QD
10		ZCB pe QA, QB, QC și QD
10		Cod biquinar pe QB, QC, QD, QA sau factor 1/2 pe QA

Notă : I = intrare tact  
 0 = ieșire divizor cu N

**Numărătoare binare sincrone.** În acest tip de numărătoare impulsurile de tact se aplică tuturor bistabililor componenți, obținând astfel comutarea sincronă a tuturor bistabililor care schimbă starea. Aceste numărătoare au fost introduse pentru a spori viteza de lucru, prin înlăturarea dificultăților legate de propagarea stărilor în numărătoarele asincrone. Aceasta s-a obținut însă complicând schemele logice.

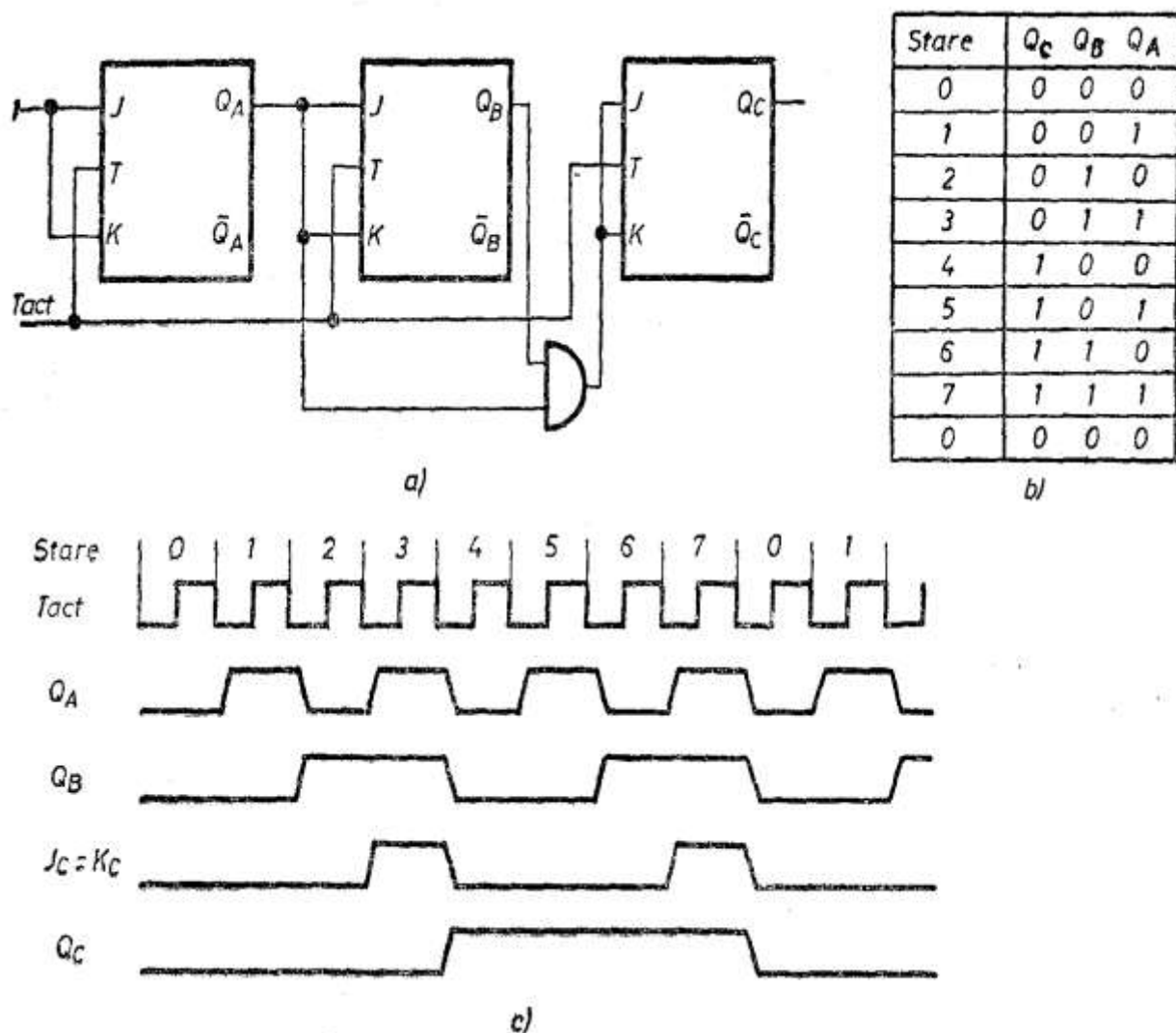


Fig. 27. Numărător binar sincron de trei biți.

Bistabilii cei mai folosiți pentru realizarea numărătoarelor sincrone sunt cei de tip **JK**, dar pot fi utilizați și bistabili de tip **D**, **T** sau **RS**.

Schema unui numărător binar sincron este dată în figura 27.a. Din tabela de stări dată în fig. 27.b, se observă că bascularea unui bistabil are loc în momentul când toți bistabilii anteriori lui sunt în starea 1. Acest lucru se observă atât în schemă cât și pe formele de undă din fig. 27.c.

**Numărătoare cu  $N$  sincrone.** Spre deosebire de numărătoarele asincrone, în cazul numărătoarelor sincrone, nu există o deosebire de principiu între numărătoarele cu  $N$  și cele *binare*. Pentru oricare tip de numărător sincron condițiile de basculare, a bistabililor componenți, sunt realizate cu circuite cu porți logice, care prelucrează starea curentă a numărătorului.

Vom exemplifica proiectarea unui astfel de numărător cu  $N=10$  lucrând în cod zecimal codificat binar natural. În tabelul 2.5.a se dă tabelul de tranziție pentru bistabilul **JK**, iar în tabelul 2.5.b se dau stările numărătorului, împreună cu stările necesare la intrările **J** și **K** pentru a obține tranzițiile de la o stare la alta, conform cu tabelul 2.5.a. În figura 28.a se dau diagramele *Karnaugh* pentru fiecare din intrările **J** și **K** ale bistabililor iar schema logică este dată în figura 28.b.

*Tabelul 2.5*

Stările și semnalele de intrare pentru un numărător zecimal

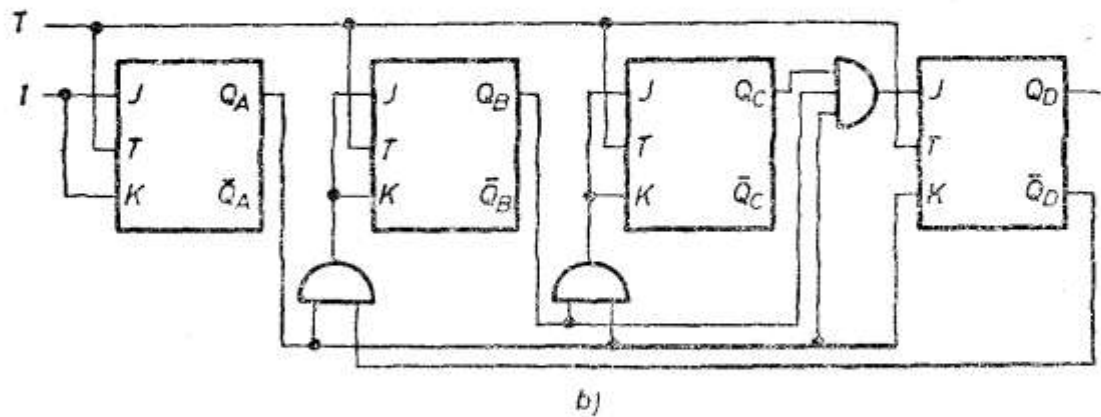
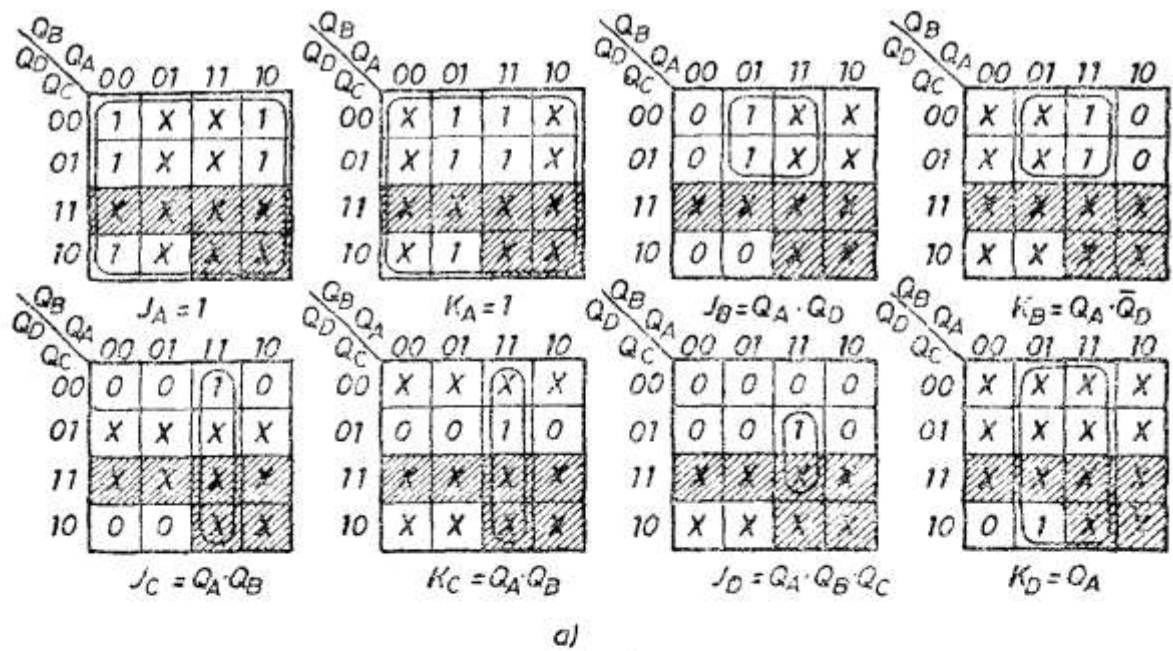
Stare	$Q_D$	$Q_C$	$Q_B$	$Q_A$	$J_D$	$K_D$	$J_C$	$K_C$	$J_B$	$K_B$	$J_A$	$K_A$
0	0	0	0	0	0	X	0	X	0	X	1	X
1	0	0	0	1	0	X	0	X	1	X	X	1
2	0	0	1	0	0	X	0	X	X	0	1	X
3	0	0	1	1	0	X	1	X	X	1	X	1
4	0	1	0	0	0	X	X	0	0	X	1	X
5	0	1	0	1	0	X	X	0	1	X	X	1
6	0	1	1	0	0	X	X	0	X	0	1	X
7	0	1	1	1	1	X	X	1	X	1	X	1
8	1	0	0	0	X	0	0	X	0	X	1	X
9	1	0	0	1	X	1	0	X	0	X	X	1
0	0	0	0	0								

a)

$Q_n \rightarrow Q_{n+1}$	J	K
0 $\rightarrow$ 0	0	X
0 $\rightarrow$ 1	1	X
1 $\rightarrow$ 0	X	1
1 $\rightarrow$ 1	X	0

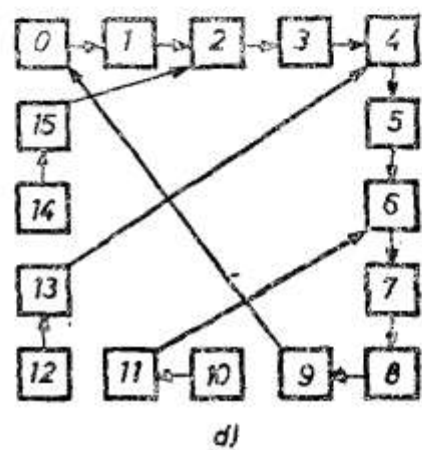
b)

Odată determinată schema logică, proiectarea nu este încheiată. Având în vedere că din cele 16 stări posibile ale numărătorului se utilizează numai 10, există 6 stări interzise (hașurate pe diagramele *Karnaugh*). Numărătorul poate ajunge în una din aceste stări, fie la pornire, fie datorită unei interferențe externe. Este necesar ca în această situație, numărătorul să revină la ciclul de numărare corect fără intervenție externă. Pentru acest motiv, se verifică comportarea numărătorului odată ajuns în una din cele 6 stări interzise. Pentru numărătorul dat în figura 28.b verificarea aceasta se face în tabelul din figura 28.c, unde se analizează care este starea următoare pentru fiecare din cele 6 stări interzise. În figura 28.d este dată diagrama de evoluție a stărilor numărătorului, realizată pe baza tabelului din figura 28.c, în care se poate urmări cu ușurință evoluția numărătorului și faptul că nu există un ciclu parazit de numărare.



c)

Stare interzisă	Stare următoare	$J_D$	$K_D$	$J_C$	$K_C$	$J_B$	$K_B$	$J_A$	$K_A$
1 0 1 0	1 0 1 1	0	0	0	0	0	0	1	1
1 0 1 1	0 1 1 0	0	1	1	1	0	0	1	1
1 1 0 0	1 1 0 1	0	0	0	0	0	0	1	1
1 1 0 1	0 1 0 0	0	1	0	0	0	0	1	1
1 1 1 0	1 1 1 1	0	0	0	0	0	0	1	1
1 1 1 1	0 0 1 0	1	1	1	1	0	0	1	1



**Numărătoare reversibile.** în unele aplicații (circuite de calcul, circuite de implementare a acordului cvasicontinuu etc.) sunt foarte utile numărătoarele reversibile, care pot funcționa după un cod dat, parcurgând stările fie în sens crescător (*în sus*), fie în sens descrescător (*în jos*), în funcție de comenzile externe aplicate.

Proiectarea unor astfel de numărătoare este foarte asemănătoare cu proiectarea numărătoarelor sincrone simple prezentate mai înainte, folosindu-se, de asemenea, circuite combinaționale pentru realizarea condițiilor de comutație pe intrările **J** și **K** ale bistabililor.

Astfel de numărătoare, indiferent de codul utilizat, au în general o schemă complicată, iar realizarea lor cu bistabili și porți separate este neeconomică. Din fericire, astfel de numărătoare cu 4 biți, se realizează în mod curent sub formă de circuite integrate în diversele familii logice. Se pot cita în familia logică **TTL** circuitele 74168, 74192 (*decade reversibile*), 74169, 74193 (*binare reversibile*), în familia logică **CMOS** circuitele 4029 (*binar sau decadic reversibil*) 4510, 40192 (*decade reversibile*) 4516, 40193 (*binare reversibile*) și în familia logică **ECL** circuitele 10136 (*binar reversibil*) și 10137 (*decadic reversibil*). Unele din circuitele de mai sus, au intrări de tact diferite pentru cele două sensuri de numărare (74192, 74193, 40192, 40193) iar altele, au o singură intrare de tact și o altă intrare, care comandă sensul de numărare (74168, 74169, 4029, 4510, 4516, 10136 10137).

Tabelul 2.6  
Modurile de comandă pentru numărătorul **4029 CMOS**

$P_L$	$\frac{BIN/}{DEC}$	$\frac{UP/}{DN}$	$\overline{CE}$	$CP$	Mod
1	X	X	X	X	Încărcare
0	X	X	1	X	Blocat
0	0	0	0	↑	Numără în jos cu 10
0	0	1	0	↑	Numără în sus cu 10
0	1	0	0	↑	Numără în jos cu 16
0	1	1	0	↑	Numără în sus cu 16

Notă: ↑ reprezintă frontul pozitiv al tactului

În plus, toate aceste numărătoare mai au și facilitatea de a putea fi programate, adică la o comandă externă aplicată circuitului integrat, stările prezentate pe intrările de date, sunt transferate în bistabilii numărătorului. Acest transfer poate avea loc asincron (direct pe intrările  $S_D$  și  $R_D$ ) sau sincron (la venirea impulsului de tact) (v. ex. tab. 2.6).

## CAPITOLUL 3

### NUMĂRĂTOARE BINARE SINCRONE

La aceste numărătoare toate celulele binare componente ale acestora comută simultan sub acțiunea unui impuls de tact comun aplicat tuturor celulelor.

#### 3.1. Numărător binar sincron de tip serie

Schema acestui numărător împreună cu tabelul de adevăr pe care îl realizează sunt reprezentate în fig. 23.

Din examinarea celulei de tip *T* cunoaștem că această celulă basculează dacă are intrările pe nivel unu ( $J=K=1$ ) și i se aplică un impuls de tact.

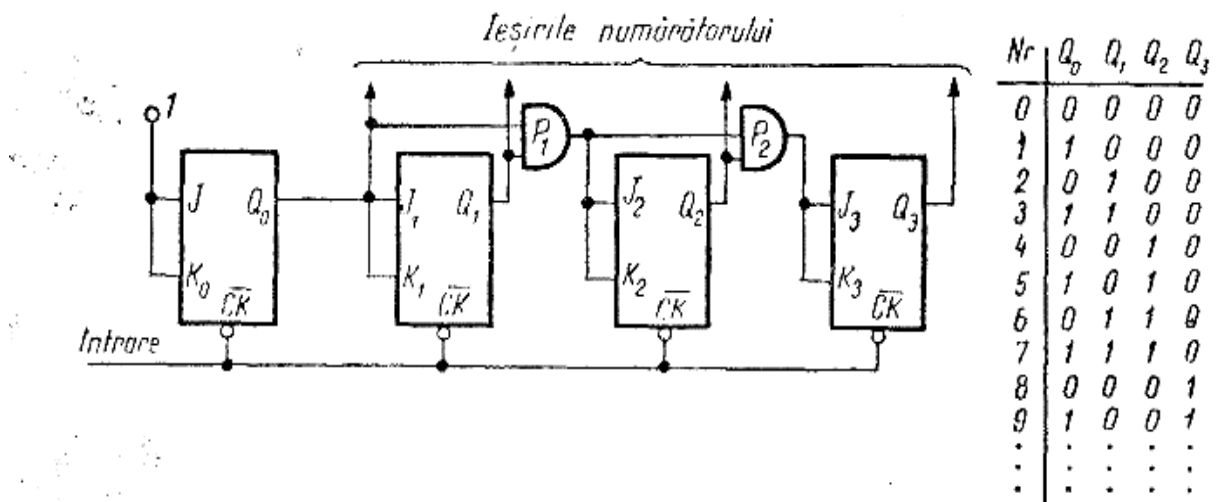


Fig. 29. Numărător binar sincron serie.



În schema din figura 29 bornele de tact ale bistabililor au fost legate împreună formând intrarea numărătorului; iar intrările  $J$  și  $K$  ale fiecărui bistabil sunt comandate cu ajutorul unor porți  $\text{SI}$  în conformitate cu tabelul de adevăr. Astfel:

—  $CBB_0$  trebuie să basculeze la fiecare impuls aplicat la intrare (vezi coloana  $Q_0$  din tabel), în consecință vom pune intrările sale la nivel logic unu ( $J_0=K_0=1$ );

—  $CBB_1$  basculează din două în două impulsuri aplicate la intrare (vezi coloana  $Q_1$  din tabel), adică numai atunci când  $Q_0=1$ . În consecință, vom lega intrările  $J_1=K_1$  la ieșirea  $Q_0$ .

—  $CBB_2$  basculează din patru în patru impulsuri aplicate la intrare (vezi coloana  $Q_2$  din tabel), adică atunci când atât  $Q_0$  cât și  $Q_1$  sunt în starea 1. Această comandă este asigurată de poarta  $P_1$  la ieșirea căreia au fost legate intrările  $J_2$  și  $K_2$ .

După un raționament similar rezultă că  $CBB_3$  va bascula când  $Q_0=Q_1=Q_2=1$ , comandă asigurată de poarta  $P_2$ .

În legătură cu schema acestui numărător putem face următoarele observații:

1. Se recomandă ca citirea conținutului numărătorului să se facă și în acest caz cu ajutorul unui impuls de *STROBE* deoarece pot apărea erori de decodare, deși funcționarea este sincronă.

Erorile de decodare apar datorită duratelor finite de comutare directă și inversă a bistabililor, precum și variației acestor timpi de la un bistabil la altul. Spre deosebire de cazul numărătorului binar asincron intervalul de timp pe care pot apărea aceste erori este constant și independent de numărul bistabililor interconectate.

2. Pentru calculul frecvenței maxime de lucru a numărătorului vom analiza în detaliu trecerea acestuia din starea 0111 în starea 1111.

Inițial avem:  $Q_0=0$ ,  $Q_1=Q_2=Q_3=1$ . Deoarece  $Q_0=0$  avem:  $J_1=K_1=0$ , poarta  $P_1$  blocată ( $J_2=K_2=0$ ) și poarta  $P_2$  blocată ( $J_3=K_3=0$ ).

La aplicarea unui impuls pe intrare se întâmplă următoarele:

— basculează la început  $CBB_0$ . Cu o întârziere specifică bistabilului  $t_{pdCBB_0}$  se schimbă starea logică a ieșirii acestuia din 0 în 1.

— trecerea lui  $Q_0$  din 0 în 1 asigură:  $J_1=K_1=0$  și cu o întârziere specifică porții  $\text{SI}$  ( $t_{pdSI}$ ) determină schimbarea stării de ieșire a porții  $P_1$  din 0 în 1, ceea ce face ca să avem și  $J_2=K_2=1$ .

— apariția semnalului 1 la ieșirea porții  $P_1$  cu o nouă întârziere  $t_{pdSI}$  și va determina apariția pe ieșirea porții  $P_2$  a stării 1, ceea ce asigură comenzile  $J_3=K_3=1$ .

S-a ajuns astfel în situația  $Q_0=Q_1=Q_2=Q_3=1$  porțile  $P_1$ ,  $P_2$  deschise și pe intrările  $J$  și  $K$  ale tuturor bistabililor să avem 1.

Aplicarea unui nou impuls la intrare, va determina bascularea tuturor bistabililor și deci, trecerea numărătorului în starea  $Q_0=Q_1=Q_2=Q_3=0$ . Intervalul de timp dintre cele două impulsuri aplicate la intrare trebuie să fie mai mare decât suma timpilor de propagare la care să se adauge și un eventual timp necesar strobării ( $T_s$ ).

$$T > T_{min} = t_{pdCBB_0} + 2t_{pdSI} + T_s \quad (3.1)$$

În cazul unui numărător format din  $n$  celule vom avea:

$$T > T_{min} = t_{pdCBB_0} + (n-2)t_{pdSI} + T_s \quad (3.2)$$

sau

$$f < f_{max} = \frac{1}{t_{pdCBB_0} + (n-2)t_{pdSI} + T_s} \quad (3.3)$$

Având în vedere faptul că  $t_{pdSI} < t_{pdCBB}$ , acest tip de numărător sincron va lucra la frecvențe mai mari decât numărătoarele asincrone. Creșterea vitezei de lucru a numărătorului sincron s-a obținut pe seama creșterii complexității schemei acestuia (s-au introdus porți **SI** între celulele de numărare).

### 3.2. Numărător binar sincron de tip paralel

O mărire suplimentară a vitezei de lucru a numărătorului sincron se poate obține dacă porțile **SI** dintre celule nu se mai leagă în cascadă ca în schema de tip serie (v. fig. 29), ci fiecare poartă **SI** este cuplată direct la ieșirile bistabililor care condiționează deschiderea (v. fig. 30).

În cazul acestei structuri, la stabilirea expresiei frecvenței maxime de lucru, va trebui să ținem seama că toate porțile **SI** comută simultan. Cu această observație rezultă următoarea expresie a frecvenței maxime:

$$f_{max} \leq \frac{1}{t_{pdCBB} + t_{pdSI} + T_s} \quad (3.4)$$

Acest tip de numărător este cel mai rapid dintre toate cele examinate până acum. Dezavantajul schemei constă în faptul că porțile **SI** consecutive au câte o intrare în plus iar fiecare poartă de comandă nou introdusă mărește cu câte o unitate gradul de încărcare al bistabililor (de exemplu ieșirea  $Q_0$  este legată la  $J_1=K_1$ ,  $P_1$  și  $P_2$ ;  $Q_1$  la  $P_1$  și  $P_2$  iar  $Q_2$  la  $P_2$ ).

Este important de reținut faptul că fiecare încărcare suplimentară a bistabililor mărește timpul lor de basculare și în consecință va reduce frecvența de lucru a numărătorului.

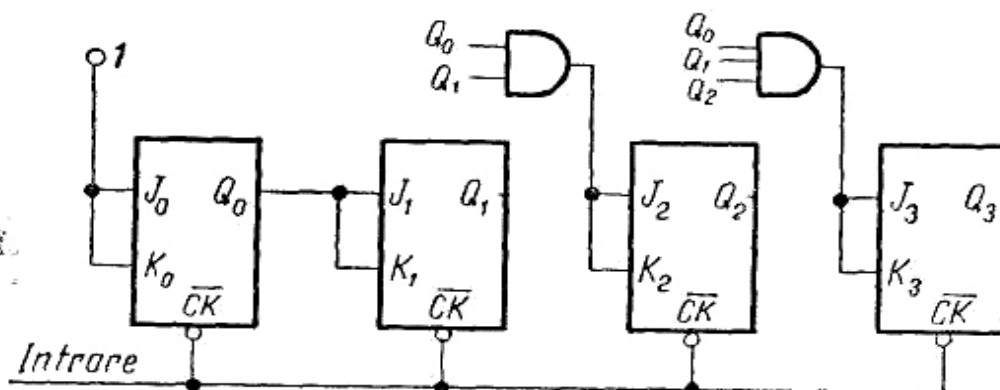


Fig. 30. Numărător binar sincron paralel

### 3.3. Numărător binar sincron reversibil

Numărătoarele binare sincrone se fabrică în general sub formă de numărătoare reversibile. Interconectarea celulelor în vederea obținerii schemei de numărător sincron reversibil se face ca în fig. 31.

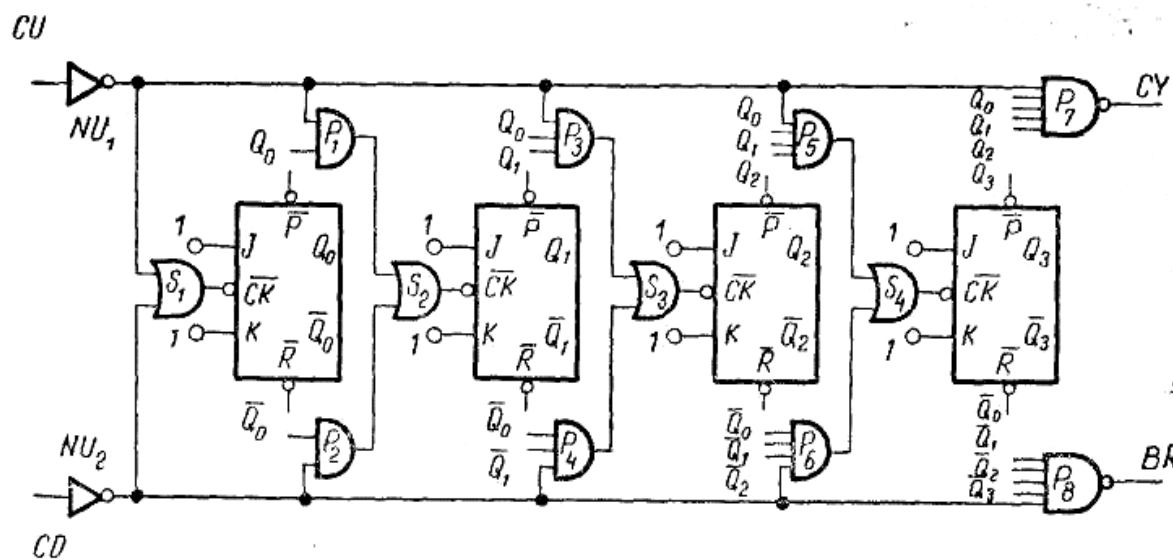


Fig. 31. Schema numărătorului binar sincron reversibil.

Față de schemele din fig. 29 și 30 apare următoarea modificare: intrările  $J$  și  $K$  ale tuturor celulelor sunt la nivel  $1$  iar impulsul de tact este dirijat prin intermediul unor porți spre celulele ce trebuie să basculeze conform tabeli de funcționare. Bascularea celulelor binare se produce la tranziția  $0 \rightarrow 1$  a semnalului de pe una din intrările de tact: **CU** (*COUNT UP*) pentru numărare directă respectiv **CD** (*COUNT DOWN*) pentru numărare inversă.

Sensul de numărare este determinat de intrarea activată în timp ce cealaltă intrare este în stare  $1$ .

De exemplu să considerăm cazul numărării directe: — pe borna **CD** se aplică semnal  $1$ . După negarea semnalului de către circuitul  $NU_2$  se obține semnal  $0$ , care blochează porțile  $P_2$ ,  $P_4$  și  $P_6$ . Porțile fiind de tipul **SI** ele se blochează pe semnal de ieșire  $0$  și ca urmare porțile de tip **SAU**:  $S_1$ ,  $S_2$ ,  $S_3$  și  $S_4$  vor fi deschise.

—Impulsurile ce urmează a fi numărate se aplică pe borna **CU**, sunt negate de către circuitul  $NU_1$  și aplicate porților  $P_1$ ,  $P_3$  și  $P_5$ . Înaintea aplicării primului impuls starea celulelor este următoarea:  $Q_0=Q_1=Q_2=Q_3=0$ , și deci porțile  $P_1$ ,  $P_3$  și  $P_5$  vor fi blocate.

—Tranziția  $0 \rightarrow 1$  a primului impuls aplicat pe intrare, inversată de circuitul  $NU_1$  se aplică primului bistabil prin poarta deschisă  $S_1$ . Prima celulă basculează, trecând în starea  $Q_0=1$ . Al doilea impuls aplicat la intrare se va dirija atât spre primul bistabil prin  $S_1$  cât și spre cel de al doilea prin poarta  $P_1$  (deschisă de  $Q_0=1$ ) și  $S_2$ . În consecință, starea numărătorului va fi  $Q_0=0$ ,  $Q_1=1$ ,  $Q_2=0$  și  $Q_3=0$  ș.a.m.d.

Numărătorul mai este prevăzut cu două porți suplimentare de tip **SI-NU**  $P_7$ ,  $P_8$  pentru generarea comenzilor:

— **CARRY** ( $CY$ ), când numărătorul se umple

$$Q_0=Q_1=Q_2=Q_3=1$$

— **BORROW** ( $BR$ ), când numărătorul se golește

$$Q_0=Q_1=Q_2=Q_3=0$$

Pentru a oferi o flexibilitate maximă schemei, celulele de bază s-au ales de tipul *J-K Master-Slave* cu intrări prioritare de preselectie și de ștergere. Folosind aceste intrări într-o conexiune de tipul *D* sincronă putem realiza *încărcarea paralelă* a celulelor numărătorului (v. fig.32).

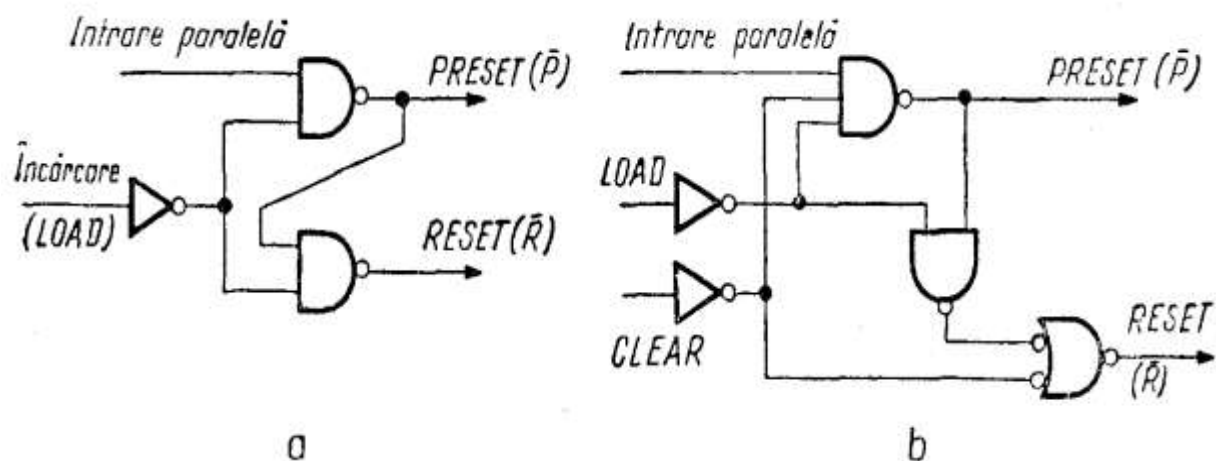


Fig. 32. Schema încărcării paralele și a ștergerii.

Operația de încărcare este independentă de intrările de tact și se realizează când intrarea de încărcare (*LOAD*) este în 0. Circuitul mai este prevăzut și cu o intrare prioritară de ștergere (*CLEAR*), care forțează ieșirea în starea  $Q=0$  atunci când pe această bornă se aplică nivel 1.

Schema bloc completă a numărătorului binar sincron reversibil este reprezentată în fig. 33.

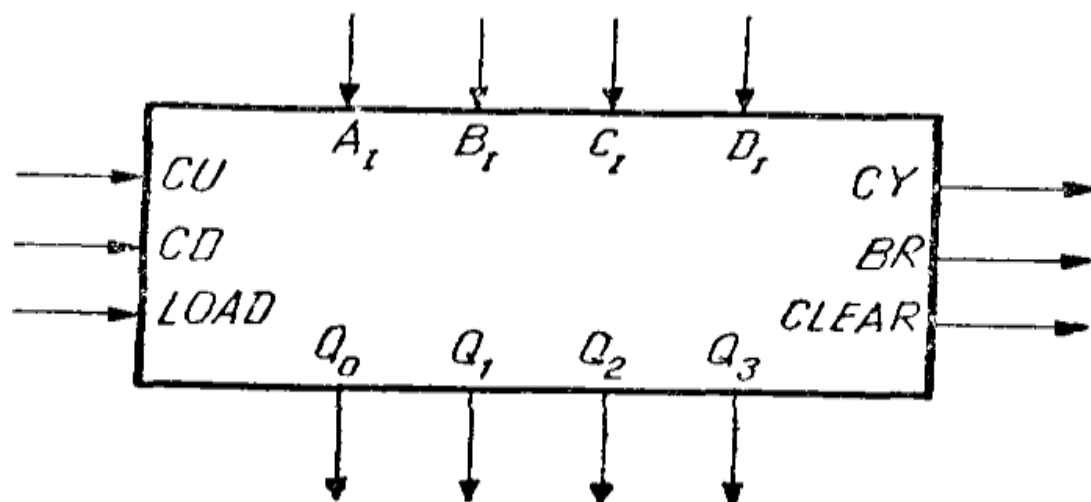


Fig. 33. Schema bloc a numărătorului sincron reversibil.

În încheierea capitolului de numărătoare facem următoarele observații:

— numărarea reprezintă o operație de importanță majoră în aproape orice sistem numeric de măsurare, comandă și control;

— numărătoarele prezentate în acest capitol au fost realizate cu **CBB** de tip **J-K**. Trebuie menționat însă faptul că funcția de numărare mai poate fi implementată și cu **CBB** de tip **D**. Pentru aceasta celulele de tip D vor fi interconectate într-o schemă de registru de deplasare cu reacție sau pot fi cuplate cu sumatoare elementare obținându-se în acest din urmă caz schema unui acumulator;

— într-un număr mare de aplicații este necesar să se determine o funcție matematică cum ar fi: o putere, logaritmul, rădăcina pătrată, etc. a numărului de impulsuri generate într-un proces tehnic. Asemenea funcții pot fi ușor implementate cu schema utilizând numărătoare și acumulatori.

## ***BIBLIOGRAFIE***

- 1.De la poarta TTL la MICROPROCESOR I.Sztojanov,E.Borcoci,s.a.***
- 2.Manualul Inginerului Electronist vol.III Edmon Nicolau,s.a.***
- 3.Numărătoare Electronice R.M.M.Oberman prof.Universitatea Tehnică Delf***
- 4.Componente și Circuite Electronice-manual ptr. Cls.a XI-a și a XII-a Theodor Dănilă, monica Ionescu-Vaida***