

CAPITOLUL 5

Circuite logice secvențiale

Circuitele logice combinaționale, fig. 4.1, sunt considerate *sisteme digitale de ordin zero*, având ca element reprezentativ poarta logică elementară. Ele sunt *circuite fără memorie* și se caracterizează prin faptul că semnalele de ieșire sunt combinații logice ale semnalelor de intrare, relația 4.1, existând numai atâta timp cât semnalele de intrare există.

La circuitele logice secvențiale (c.l.s.), considerate *sisteme de ordin ≥ 1* , starea ieșirilor depinde nu numai de starea actuală a intrărilor, dar și de stările anterioare ale circuitului. Din acest motiv, se spune că *circuitele logice secvențiale sunt circuite cu memorie*.

Schema bloc a unui circuit logic secvențial este prezentată în fig. 5.1, în care am notat cu x_1, x_2, \dots, x_n intrările principale, cu y_1, y_2, \dots, y_m – ieșirile principale, cu q_1, q_2, \dots, q_i – stările interne prezente ale circuitului și cu q_1', q_2', \dots, q_i' - stările interne următoare ale acestuia.

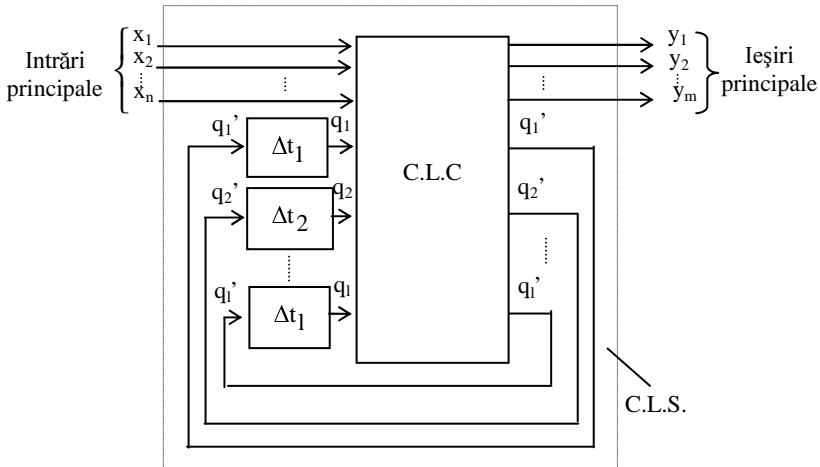


Fig. 5.1. Schema bloc a unui circuit logic secvențial

Expresiile ieșirilor și stărilor următoare ale unui circuit logic secvențial în funcție de intrări și stările prezente pot fi scrise astfel:

$$\begin{aligned} y_k &= y_k(x_1, x_2, \dots, x_n, q_1, q_2, \dots, q_l); \\ q_i' &= q_i'(x_1, x_2, \dots, x_n, q_1, q_2, \dots, q_l). \end{aligned} \quad (5.1)$$

În această formă, relațiile 5.1 definesc un *automat Mealy*.

În cazul în care y_k nu depinde decât de intrările x_1, x_2, \dots, x_n , spunem că relațiile 5.1 astfel modificate, definesc un *automat de tip Moore*.

Stările următoare q_i' devin prezente după un interval de timp determinat de întârzierile $\Delta t_1, \Delta t_2, \dots, \Delta t_l$, special introduse în circuit.

Dacă $\Delta t_1 \neq \Delta t_2 \neq \dots \neq \Delta t_l$, spunem că *c.l.s. este de tip asincron*, iar dacă $\Delta t_1 = \Delta t_2 = \dots = \Delta t_l = \Delta t$, deci modificarea stărilor are loc după un același interval de timp, Δt , la comanda unui impuls de tact, spunem că *c.l.s. este de tip sincron*.

Se observă că trecerea de la sisteme de ordinul zero (c.l.c.) la cele de ordin superior (c.l.s) se face prin introducerea unor reacții, care conferă ieșirilor circuitului o autonomie parțială, la limită – totală, față de intrări, deci calitatea de *memorie*.

5.1. Circuite basculante bistabile SR

Circuitele basculante bistabile SR (CBB-SR) se obțin prin introducerea unei reacții într-un sistem elementar de ordin zero. Sistemul astfel obținut este de ordin 1.

CBB-SR pot fi realizate în varianta asincronă, sincronă sau "Master-Slave" (stăpân-sclav).

5.1.1. Circuitul basculant bistabil SR asincron

Circuitul basculant bistabil SR asincron, cunoscut - datorită proprietăților sale de a memora - și sub denumirea de *latch* (zăvor), poate fi realizat cu NOR-uri sau cu NAND-uri.

5.1.1.1. Circuitul basculant bistabil SR asincron realizat cu NOR-uri

Circuitul basculant bistabil SR asincron realizat cu NOR-uri prezintă schema din fig. 5.2 și *tabelul de tranziție* – tab. 5.1, în care s-a notat cu indice n - valoarea logică prezentă și cu $n+1$ - valoarea logică viitoare.

Expresia ieșirii Q a circuitului poate fi obținută din schema din fig. 5.2, astfel:

$$\overline{Q_{n+1}} = S_n + \overline{\left(\overline{R_n + Q_n} \right)} = \overline{S_n + R_n} \cdot Q_n. \quad (5.2)$$

Eliminând negația în ambii membri ai relației 5.2, obținem:

$$Q_{n+1} = S_n + \overline{R_n} \cdot Q_n. \quad (5.3)$$

Un alt mod de a obține expresia 5.3 îl reprezintă utilizarea diagramei VK din fig. 5.3, în locațiile căreia au fost trecute valorile logice ale lui Q_{n+1} .

Completarea locațiilor diagramei s-a făcut ținând seama de tabelul de tranziție, tab. 5.1, astfel:

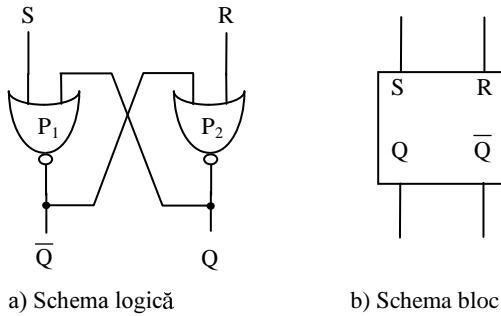


Fig. 5.2. CBB-SR asincron, varianta NOR

Tab.5.1. Tabel de tranziție al CBB-SR asincron, varianta NOR

S_n	R_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	x

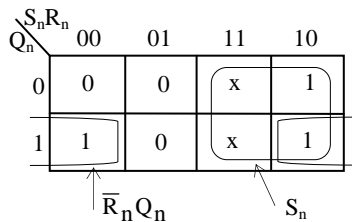


Fig. 5.3. Diagrama VK pentru CBB-SR asincron - varianta NOR

- pentru $S_n R_n = 00$, $Q_{n+1} = Q_n$ (prima linie a tabelului de tranziție), deci valorile logice ale lui Q_n se trec în coloana $S_n R_n = 00$ a diagramei VK;
- pentru $S_n R_n = 01(10)$, $Q_{n+1} = 0(1)$ indiferent de valorile lui Q_n și locațiile din coloana a doua (a patra) a diagramei VK se completează cu 0(1).
- pentru $S_n R_n = 11$, ieșirile celor două porți sunt forțate simultan în 0 logic, deci s-ar ajunge la situația inadmisibilă în care:

$$Q_{n+1} = \bar{Q}_{n+1} = 0. \tag{5.4}$$

Din acest motiv combinația de intrare $S_n R_n = 11$ este interzisă (de obicei prin logică suplimentară) iar în locațiile corespunzătoare ale tab. 5.1 și diagramei VK din fig. 5.3, se pune semnul "x", specific locațiilor în care funcția este nedefinită.

În urma minimizării, se obține relația 5.3.

Denumirile S (SET) și R (RESET) ale intrărilor latch-ului SR asincron provin din limba engleză și au semnificațiile: *înscriere*, respectiv *ștergere*.

Într-adevăr, observăm că pentru $S_n R_n = 10$, intrarea de înscriere S_n este activată și în memoria elementară se înscrie 1 logic, deci $Q_{n+1} = 1$.

Similar, pentru $S_n R_n = 01$, intrarea de ștergere R_n este activată și memoria este ștearsă: $Q_{n+1} = 0$.

Relația 5.3 se verifică cu ușurință pentru primele 3 linii ale tab. 5.1.

5.1.1.2. Circuitul basculant bistabil SR asincron realizat cu NAND-uri

Schema circuitului basculant bistabil SR asincron realizat cu NAND-uri este prezentată în fig. 5.4, iar tabelul de tranziție este tab. 5.2.

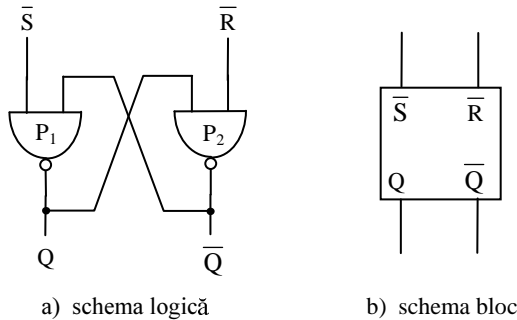


Fig.5.4. CBB-SR asincron, varianta NAND

Tab.5.2. Tabel de tranziție al CBB-SR asincron, varianta NAND

\bar{S}_n	\bar{R}_n	Q_{n+1}
1	1	Q_n
1	0	0
0	1	1
0	0	x

Pe schema din fig. 5.4 putem scrie:

$$Q_{n+1} = \bar{S}_n \cdot \left(\overline{\bar{R}_n \cdot Q_n} \right) = S_n + \bar{R}_n \cdot Q_n,$$

relație identică cu rel. 5.3, obținută în cazul circuitului basculant bistabil SR realizat cu NOR-uri.

Aceeași relație se obține și în urma minimizării funcției logice Q_{n+1} cu ajutorul diagramei VK din fig. 5.5.

$\bar{S}_n \bar{R}_n$	00	01	11	10
0	x	1	0	0
1	x	1	1	0

S_n $\bar{R}_n Q_n$

Fig. 5.5. Diagrama VK pentru CBB-SR asincron, varianta NAND

Indiferent de varianta de implementare adoptată, CBB-SR asincron prezintă următoarele deficiențe:

- aceleași semnale care indică modul *cum* (în care) trebuie să se facă înscrierea, dictează și momentul *când* trebuie să aibă loc aceasta;
- pentru anumite tranziții ale intrărilor circuitului, starea ieșirilor este imprevizibilă.

Exemplu: Tranziția 11→00 a intrărilor, poate aduce ieșirile Q, \bar{Q} ale CBB din fig. 5.2 în oricare din cele două stări posibile. Astfel, pentru $S_n R_n = 11$, ambele ieșiri vor fi forțate în 0, $Q = \bar{Q} = 0$, validând prin intermediul legăturilor de reacție porțile P_1, P_2 . Aplicând acum $S_n R_n = 00$ și admitând că poarta P_1 este mai rapidă, se va obține un 1 logic la ieșirea Q , ceea ce determină - prin reacție - un 0 logic la ieșirea Q . Evident, dacă aplicăm aceeași supoziție pentru poarta P_2 , valorile logice ale ieșirilor se inversează.

5.1.2. Circuitul basculant bistabil SR sincron

Circuitul basculant bistabil SR sincron se obține din cel asincron prin adăugarea a două porți, 3 și 4, validabile de un impuls de tact (fig. 5.6 și 5.7).

Funcționarea celor două CBB-SR sincrone fiind similară, ne vom limita la explicarea funcționării circuitului din fig. 5.6 a.

Observăm că pentru $\overline{CLK} = 1$, porțile 3 și 4 sunt inhibitate și orice modificare a lui \bar{S} și \bar{R} nu va afecta CBB-ul SR asincron format din porțile 1 și 2. Într-adevăr, pentru $\overline{CLK} = 1$, intrările acestuia vor fi $S_n R_n = 00$ și, conform primei linii din tab. 5.1, $Q_{n+1} = Q_n$ și ieșirile vor rămâne neschimbate.

Când $\overline{CLK} = 0$, porțile 3 și 4 vor fi validate și intrările \bar{S}, \bar{R} , transformate în SR, vor avea acces la intrările CBB-SR asincron, acționând conform tab. 5.1.

Pentru o funcționare sincronă a circuitului este necesar ca $\overline{\text{CLK}} = 0$, care dictează *când* să se execute comenzile date de intrările \overline{S} \overline{R} , să apară numai după ce acestea s-au stabilizat. Modificarea lui \overline{S} \overline{R} în intervalul de timp în care porțile de intrare 3 și 4 sunt deschise, conduce la o funcționare asincronă a circuitului. Din acest motiv, sunt necesare condiții restrictive pentru relația de timp dintre $\overline{\text{CLK}}$ și \overline{S} \overline{R} .

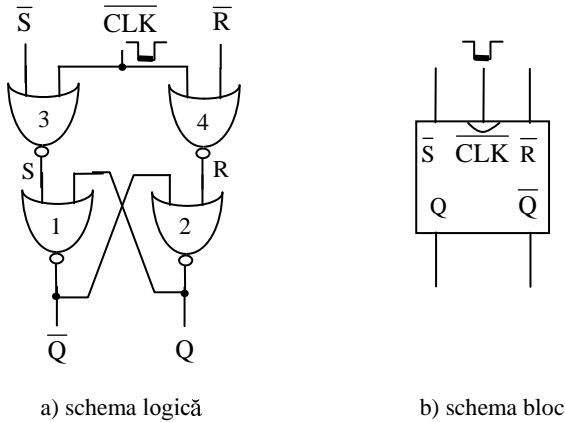


Fig. 5.6. CBB-SR sincron, varianta NOR

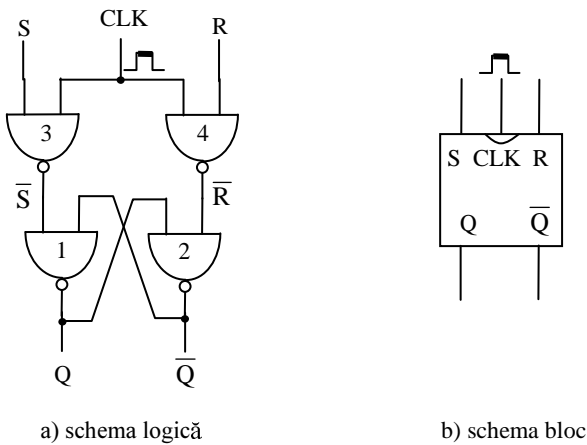


Fig. 5.7. CBB-SR sincron, varianta NAND

Circuitul din fig. 5.7 funcționează similar, impulsul de tact fiind de această dată activ pe palierul superior (1 logic) al impulsului de tact.

5.1.3. Circuitul basculant bistabil SR Master-Slave

După cum reiese din fig. 5.8, circuitul basculant bistabil SR Master-Slave reprezintă o extensie serie a bistabilului SR sincron implementat cu NAND-uri (v. fig. 5.7). Schema logică este prezentată în fig. 5.9 a, iar diagramele impulsurilor CLK și $\overline{\text{CLK}}$ - în fig. 5.9 b și c.

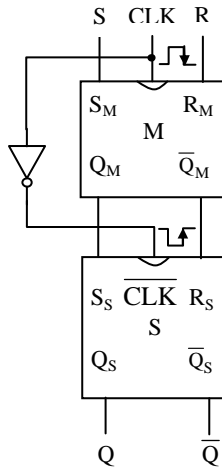


Fig. 5.8. CBB-SR-MS - Schema bloc

Funcționare

În intervalul (1)-(2), v. diagramele b și c din fig. 5.9, porțile de intrare (3M, 4M) și de transfer (3S, 4S) sunt blocate, iar MASTER-ul este izolat atât de intrări cât și de SLAVE.

În intervalul (2)-(3), CLK=1 și porțile 3M, 4M sunt validate, iar informația se înscrie în MASTER; porțile 3S, 4S fiind blocate ($\overline{\text{CLK}} = 0$), bistabilul SLAVE este în continuare izolat față de MASTER.

În intervalul (3)-(4) se repetă situația din intervalul (1)-(2) când MASTER-ul era izolat atât de intrări cât și de SLAVE.

În sfârșit, după momentul (4), porțile 3M, 4M sunt blocate (MASTER-ul izolat față de intrări) iar porțile 3S, 4S sunt validate și informația din MASTER se transferă în SLAVE.

Concluzionând, înscrierea informației în MASTER are loc înainte de momentul (3) (posibil chiar pe frontul descrescător al CLK), iar transferul ei în

SLAVE (și deci la ieșire) are loc după momentul (4) (deci pe același front descrescător al CLK).

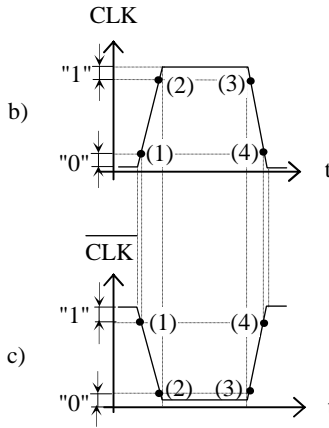
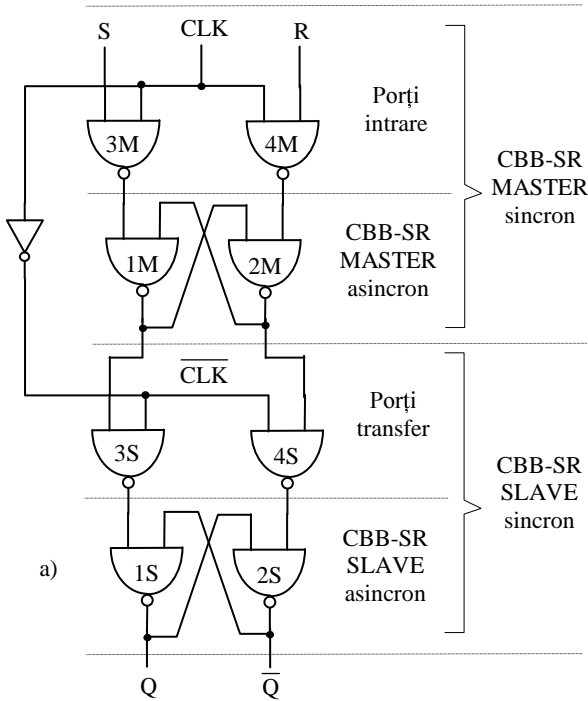


Fig. 5.9. CBB-SR-MS: a) schemă; b), c) diagrame

Prin urmare, pentru înscrierea fără erori a informației în CBB-SR-MS, este necesar ca aceasta să rămână stabilă la intrare un interval de timp în jurul intervalului (3)-(4).

Deși realizează o mult mai bună separație între *când* și *cum* trebuie să se modifice informația memorată, CBB-SR-MS nu elimină dezavantajul reprezentat de posibilitatea apariției tranzițiilor nedeterminate (v. tab. 5.1 și 5.2).

Evident, se pot construi CBB-SR-MS care să comute pe tranziția pozitivă a impulsului de tact.

5.2. Circuite basculante bistabile de tip D

Circuitele basculante bistabile de tip D pot fi realizate în varianta asincronă, sincronă și Master-Slave.

5.2.1. Circuitul basculant bistabil de tip D asincron

Circuitul basculant bistabil de tip D asincron, fig. 5.10, se obține dintr-un CBB-SR asincron (fig. 5.2, tab. 5.1 sau fig. 5.4, tab. 5.2), prin atașarea unui inversor în scopul eliminării stărilor nedeterminate.

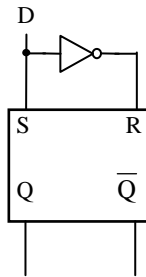


Fig. 5.10. Circuitul basculant bistabil de tip D asincron

Datorită inversorului, din tabelul 5.1 rămân numai liniile 2 și 3 pentru care $D_n = S_n = \bar{R}_n$, obținându-se tabelul 5.3.

Tab. 5.3. Tabelul de tranziție al CBB de tip D

$D_n = S_n = \bar{R}_n$	Q_n	Q_{n+1}
1	x	1
0	x	0

Deoarece repetă practic instantaneu la ieșire ceea ce i se aplică la intrare (v. tab. 5.3), circuitul nu prezintă interes practic.

5.2.2. Circuitul basculant bistabil de tip D sincron

Variantele de CBB tip D sincron prezentate în fig. 5.11 și 5.12 au fost obținute prin atașarea câte unui inversor circuitelor basculante bistabile SR sincrone din fig. 5.6 și 5.7.

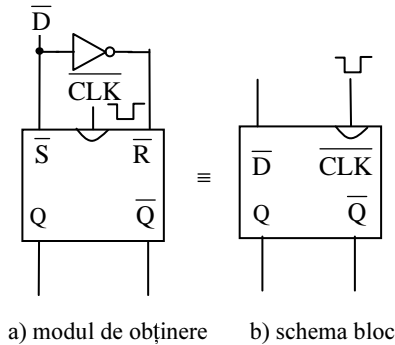


Fig. 5.11. CBB-D sincron comandat de palierul inferior al CLK

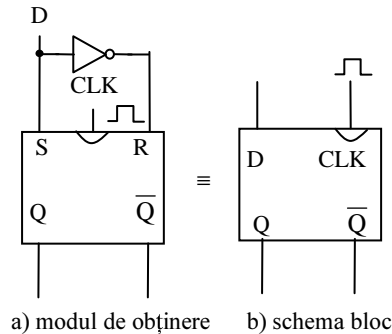


Fig. 5.12. CBB-D sincron comandat de palierul superior al CLK

Ca și în cazul CBB-SR sincron, pentru a realiza o comutare sincronizată de CLK, este necesar ca informația de la intrarea D să se modifice în afara palierului activ al impulsului de tact ($\overline{CLK} = 0$ pentru fig. 5.11 și $CLK=1$ pentru fig. 5.12), în timpul palierului respectiv aceasta trebuind să rămână stabilă. Apariția palierului activ al impulsului de CLK declanșază operațiunea de înscriere a informației în bistabil și permite citirea acesteia la ieșire.

Intervalul de timp scurs între momentul apariției informației la intrarea bistabilului și momentul în care aceasta poate fi citită la ieșire, reprezintă o *temporizare comandată prin CLK*. De fapt, denumirea de bistabil de tip D, provine din englezescul *DELAY=întârziere*.

În fig. 5.13 am reprezentat schema logică a unuia din cele două latch-uri de tip D a câte 2 biți fiecare, din structura circuitului integrat CDB 475, iar în tab. 5.4 - funcționarea latch-ului respectiv.

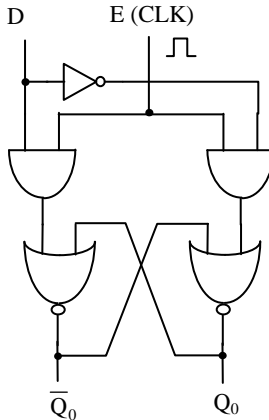


Fig. 5.13. Schema logică a latch-ului de tip D din structura CI - CDB 475

Tab. 5.4. Explicativ pentru funcționarea latch-ului de tip D din fig. 5.13

Mod operare	Intrări		Ieșiri	
	E_n	D_n	Q_{n+1}	\overline{Q}_{n+1}
Autorizare date	1	0	0	1
	1	1	1	0
Blocare date	0	x	Q_n	\overline{Q}_n

Bistabilul de tip D sincron are numeroase aplicații practice, dintre care amintim: latch-ul adresabil, memoria RAM, etc.

5.2.2.1. Latch-ul adresabil

Latch-ul adresabil, fig. 5.14, reprezintă o extensie paralel a circuitului basculant bistabil (latch-ului) de tip D sincron din fig. 5.11 și se compune din 8 astfel de circuite bistabile și un decodificator de adresă.

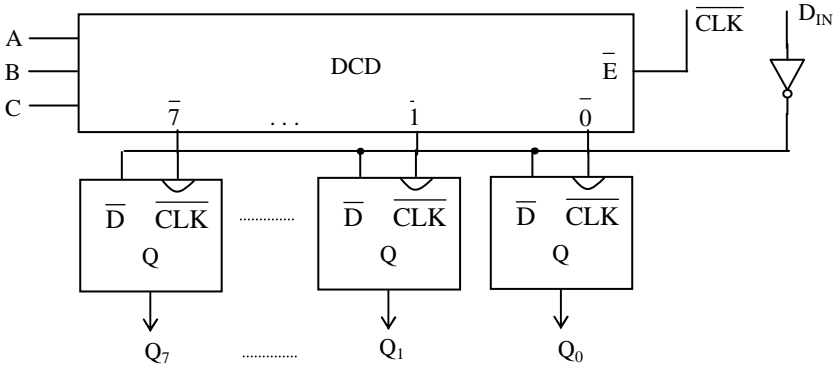


Fig. 5.14. Latch-ul adresabil

Datele de intrare D_{IN} sosesc într-o manieră serială, fiecare bit fiind distribuit la intrările \overline{D} ale celor 8 latch-uri sincrone. Combinația logică a liniilor de adresă A, B, C, activează una din liniile de ieșire ale decodificatorului, selectând astfel latch-ul în care urmează a fi înscrisă informația în timpul palierului activ al impulsului de CLK. Evident, următorul bit de informație va fi dirijat de către combinația logică a liniilor de adresă către un alt bistabil, ș.a.m.d.

Observăm că latch-ul adresabil este de fapt o memorie în care informația este înscrisă bit cu bit, putând însă fi citită integral la ieșirile celor 8 bistabile. Prin urmare, latch-ul adresabil poate fi privit și ca un convertor serie-paralel.

Latch-ul adresabil realizează o bună separație între *unde*, *când* și *cum* trebuie să se înscrie informația. Astfel, combinația logică a liniilor de adresă stabilește *unde* (în ce bistabil) urmează a fi înscrisă informația, impulsul CLK dictează momentul *când* să aibă loc înscrierea, iar valoarea logică a fiecărui bit din componența D_{IN} stabilește modul *cum* urmează să se modifice informația din bistabilul selectat.

5.2.2.2. Memoria RAM

Memoria RAM (Random Acces Memory = memoria cu acces aleator) prezintă schema din fig. 5.15 și poate fi obținută din latch-ul adresabil prin adăugarea la cele două niveluri (de decodificare și memorare) a unui al treilea nivel, de multiplexare.

Funcționarea memoriei RAM cuprinde două regimuri de lucru și anume: *înscrierea* și *citirea* informației.

Regimul de înscriere se realizează pentru $\overline{WE} = 0$ (Write Enable = autorizare de înscriere), situație în care decodificatorul este activat în timp ce multiplexorul este inhibat.

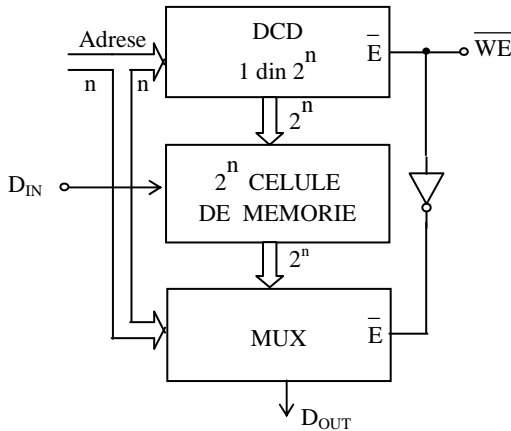


Fig. 5.15. Memoria RAM

Combinăția logică a celor n linii de adresă va activa una din cele 2^n linii de ieșire ale decodificatorului, selectând astfel una din cele 2^n celule de memorare în care urmează a se înscrie bitul de informație sosit pe linia de date D_{IN} .

După epuizarea tuturor celor 2^n combinații logice posibile ale liniilor de adresă, un număr de 2^n biți sosiți pe intrarea serială de date D_{IN} se vor afla deja înscrși în cele 2^n locații ale memoriei RAM.

Regimul de citire se realizează pentru $\overline{\text{WE}} = 1$, situație în care multiplexorul este activat, iar decodificatorul este inhibat.

Combinăția logică a liniilor de adresă va selecta locația de memorie al cărei conținut trebuie să aibă acces la ieșirea MUX-ului.

Putem astfel avea acces practic instantaneu la informația stocată în oricare din cele 2^n celule de memorie, cu condiția aplicării combinației logice corespunzătoare a liniilor de adresă.

Baleierea aleatoare (în orice ordine) a tuturor celor 2^n combinații de adresă, va permite o citire serială, într-o ordine oarecare, a conținutului tuturor celor 2^n locații de memorie.

5.2.3. Circuitul basculant bistabil D Master-Slave

Circuitul basculant bistabil D Master-Slave se obține, ca și omologul său în variantă SR, din două bistabile D sincrone conectate în cascadă și comandate în contratimp de impulsul de CLK.

În funcție de tipul de bistabile D sincrone din care este constituit, bistabilul D Master-Slave poate comuta fie pe frontul anterior, fie pe cel posterior al impulsului de CLK.

Dintre cele mai frecvente aplicații ale sale, menționăm registrele: registrul de deplasare serie, paralel, combinat, universal, etc.

5.2.3.1. Registrul de deplasare serie

Registrul de deplasare serie, fig. 5.16, este format din 4 bistabili de tip D Master-Slave.

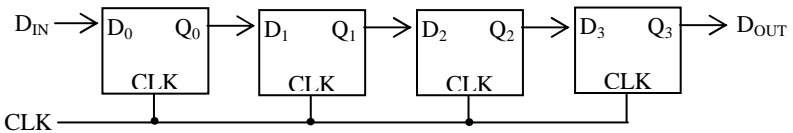


Fig. 5.16. Schema generală a unui registru de deplasare serie

În timpul funcționării, latch-urile de tip *master* sunt deschise simultan pentru $CLK=0$, cele de tip *slave* fiind închise. În timpul tranziției din 0 în 1 a semnalului de CLK, latch-urile *master* se blochează iar cele *slave* se deschid și primesc informația din *master*. Se remarcă faptul că în nici un moment nu există o cale deschisă între intrarea și ieșirea registrului.

Pe baza schemei din fig. 5.16 putem scrie următoarele relații:

$$D_{OUT}^n = Q_3^n = D_3^{n-1} = Q_2^{n-1} = D_2^{n-2} = Q_1^{n-2} = D_1^{n-3} = Q_0^{n-3} = D_{IN}^{n-4} \quad (5.5)$$

Se observă că informația D_{IN} ajunge la ieșirea registrului după 4 impulsuri de tact.

Registrele de deplasare pot fi construite atât în variante statice cât și în variante dinamice. În cazul structurilor dinamice va trebui impusă o frecvență minimă a semnalului de ceas pentru ca datele înscrise în celulele de memorare să se poată regenera sigur prin transferul în celulele următoare

Registrele de deplasare serie pot fi utilizate ca memorii cu acces serie (SAR - Serial Acces Memory/Register). Ele sunt construite pentru un număr foarte mare de biți, creșterea numărului de celule de memorare neavând nici un fel de implicații asupra numărului de conexiuni externe ale integratului.

5.2.3.2. Registrul paralel

Registrul paralel (de stocare, tampon) prezentat în fig. 5.17, este format din 4 bistabili de tip D acționați sincron de un tact comun.

În momentul aplicării tactului, cuvântul binar de 4 biți prezent la intrările I_0, I_1, I_2, I_3 , este înscris în cele 4 celule de memorie și poate fi citit la ieșirile Q_0, Q_1, Q_2, Q_3 .

Funcția principală a unui astfel de registru este aceea de a stoca temporar anumite configurații binare în scopul unui acces ușor la ele în vederea prelucrării.

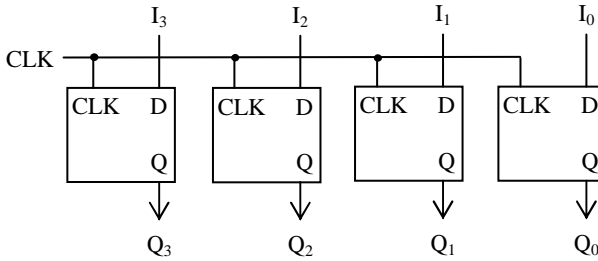


Fig. 5.17. Schema generală a unui registru paralel

Registru paralel este memoria zonelor de viteză maximă dintr-un sistem digital de prelucrare a datelor.

5.2.3.3. Registrul combinat

Cele două tipuri de registre tratate mai sus sunt utilizate în aplicații în care transferul datelor se face fie numai paralel, fie numai serie. Registrele combinate permit trecerea de la transferul paralel la cel serie și invers.

În fig. 5.18 prezentăm un registru combinat (paralel-serie sau serie-paralel) de 4 biți.

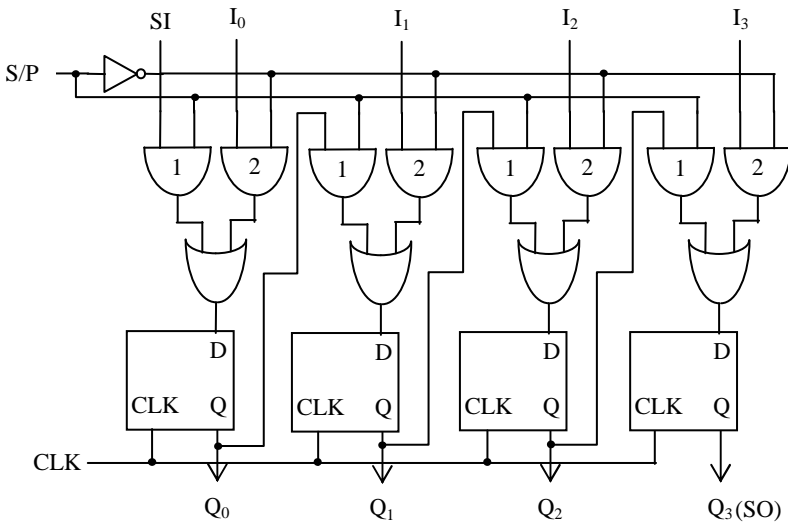


Fig. 5.18. Schema generală a unui registru combinat

Pentru $S/P = 0$, sunt validate porțile 2 și datele de intrare I_0, I_1, I_2, I_3 au acces la intrările celor 4 bistabile. Încărcarea paralel are loc în momentul aplicării impulsului de CLK.

Pentru $S/P = 1$ sunt validate porțile 1, astfel încât registrul realizează o deplasare serie a datelor de la stânga la dreapta, cu câte un bit pentru fiecare impuls de CLK.

Registrul poate funcționa ca un convertor paralel-serie, datele fiind introduse paralel la intrările I_0, I_1, I_2, I_3 și fiind extrase serie la ieșirea SO (Serial Output) a circuitului.

În regim de convertor serie-paralel, datele se introduc de o manieră serială la intrarea SI (Serial Input) și sunt extrase paralel la ieșirile Q_0, Q_1, Q_2, Q_3 .

5.2.3.4. Registrul universal bidirecțional de 4 biți

Registrul universal bidirecțional de 4 biți SN 74194, fig. 5.19, acoperă practic toate variantele de registre prezentate anterior.

Caracteristicile funcționale ale acestui tip de registru sunt prezentate în tabelul de funcționare - tab. 5.5.

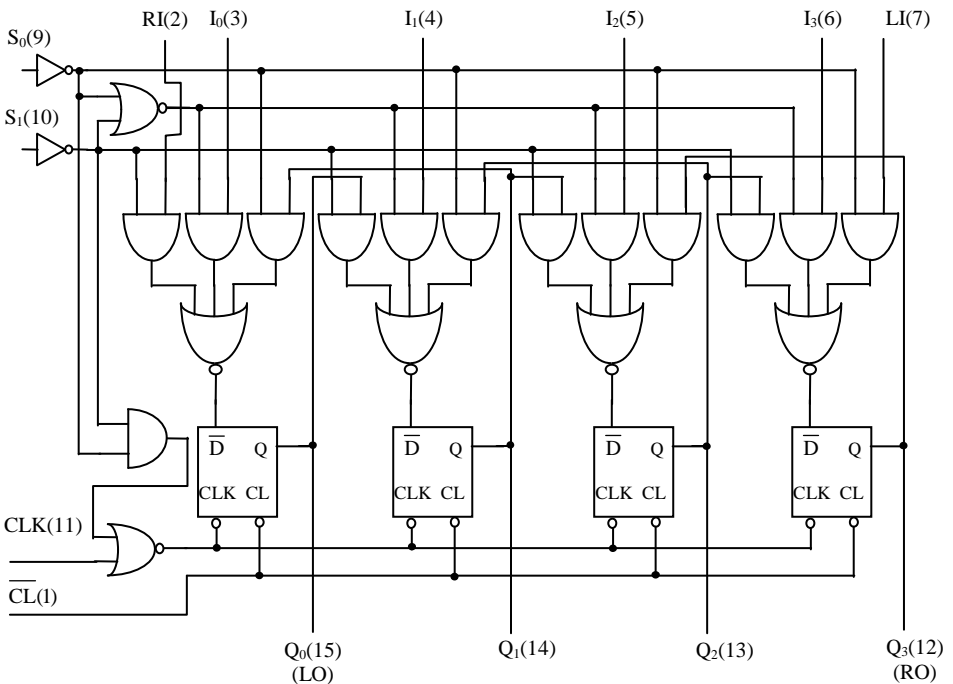


Fig. 5.19. Registrul universal bidirecțional de 4 biți (SN 74194)

Circuitul integrat SN 74194 prezintă comenzi logice speciale care-i sporesc domeniul de aplicabilitate. Funcționarea sincronă a circuitului este determinată de cele două intrări de selecție a modului de lucru, S_0 și S_1 . După cum rezultă și din tabelul de funcționare, datele pot fi introduse și deplasate de la stânga la dreapta, de la dreapta la stânga, sau introduse paralel, încărcând simultan în registru toți cei 4 biți. Dacă ambele intrări de selecție S_0 și S_1 sunt în stare "jos", datele existente în registru sunt păstrate. Terminalele RI (Right Input) și LI (Left Input) sunt intrări seriale pentru deplasarea la dreapta, respectiv la stânga a datelor și nu interferează în nici un fel cu operațiunea de încărcare paralel a datelor.

Intrările de selecție și de date trebuie să se stabilizeze cu un anumit interval de timp înaintea apariției frontului pozitiv al CLK, ele devenind active numai după acest moment.

Tab. 5.5. Tabelul de funcționare al registrului universal SN 74194

Mod de operare	Intrări							Ieșiri			
	CLK	\overline{CL}	S_1	S_0	RI	LI	In	Q_0	Q_1	Q_2	Q_3
Ștergere	X	L	X	X	X	X	X	L	L	L	L
Hold	X	H	l(b)	l(b)	X	X	X	q_0	q_1	q_2	q_3
Deplasare la stânga	↑	H	h	l(b)	X	l	X	q_1	q_2	q_3	L
	↑	H	h	l(b)	X	h	X	q_1	q_2	q_3	H
Deplasare la dreapta	↑	H	l(b)	h	l	X	X	L	q_0	q_1	q_2
	↑	H	l(b)	h	h	X	X	H	q_0	q_1	q_2
Încărcare paralel	↑	H	h	h	X	X	i_n	i_0	i_1	i_2	i_3

H = nivel de tensiune ridicat; h = idem, stabilit anterior tranziției L→H a CLK;

L = nivel de tensiune coborât; l = idem, stabilit anterior tranziției L→H a CLK;

$i_n(q_n)$ = stările intrărilor (sau ieșirilor) stabilite anterior tranziției L→H a CLK;

X = indiferent; ↑ = tranziție L→H a CLK.

Notă: (b) Tranziția H→L a intrărilor S_0 și S_1 trebuie să se producă numai în timp ce CLK este H pentru operații convenționale.

Circuitele basculante bistabile de tip SR și D fac parte din sistemele de ordinul I. Ne ocupăm în continuare de alte două tipuri de bistabile, T și JK, care, prezentând câte o reacție suplimentară, sunt considerate sisteme de ordinul II.

O secvență aplicată la intrarea unor astfel de sisteme, generează la ieșire un răspuns parțial corelat cu aceasta și puternic influențat de secvențele anterioare prin starea în care s-a aflat sistemul înainte de aplicarea secvenței.

În paralel cu evoluția ieșirilor evoluează și starea, autonomia parțială a ieșirilor față de intrări fiind conferită tocmai de această evoluție paralelă.

5.3. Circuite basculante bistabile de tip T

Circuitul basculant bistabil de tip T se obține dintr-un bistabil D prin introducerea unei reacții suplimentare ieșire-intrare, aplicată prin intermediul unui circuit logic combinațional elementar, fig. 5.20.

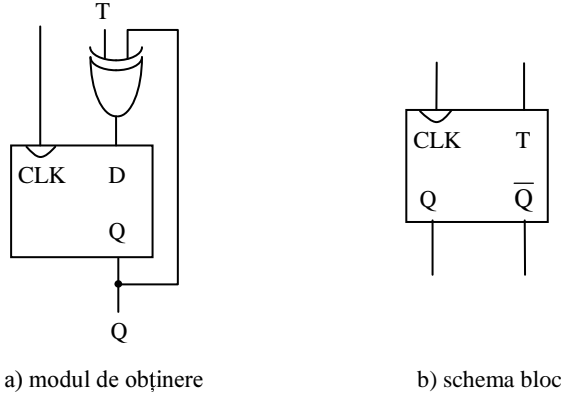


Fig. 5.20. Circuitul basculant bistabil de tip T sincron

Tab. 5.6. Tabelul de tranziție al circuitului basculant bistabil de tip T

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

Din tabelul de tranziție, tab. 5.6, se poate deduce expresia funcției de ieșire;

$$Q_{n+1} = Q_n \overline{T}_n + \overline{Q}_n T_n = Q_n \oplus T . \quad (5.6)$$

Bistabilul T din fig. 5.20 nu îndeplinește funcția de memorie propriu-zisă (cum este cazul bistabilelor SR și D), având un comportament definit atât de intrare cât și de starea în care se află. El este cel mai simplu sistem automat și este utilizat, spre exemplu, la construirea numărătoarelor asincrone.

5.4. Circuite basculante bistabile de tip JK

Reamintim faptul că bistabilul D a apărut ca urmare a necesității de a înlătura tranzițiile nedeterminate ale bistabilelor SR. Același efect de eliminare a tranzițiilor

nedeterminate se poate obține prin introducerea de reacții suplimentare în structurile SR.

2.5.1. Circuitul basculant bistabil JK asincron

Bistabilul JK asincron, fig. 5.21, poate fi obținut din bistabilul SR asincron prin introducerea unei reacții.

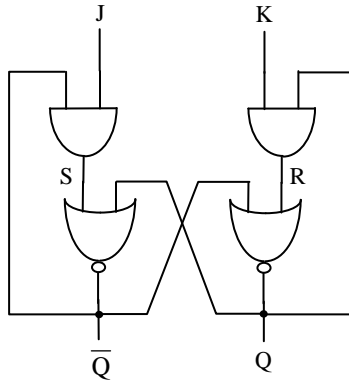


Fig. 5.21. Schema circuitului basculant bistabil JK asincron

Din fig. 5.21 se poate deduce succesiv funcția de ieșire a circuitului:

$$S_n = J_n \bar{Q}_n; \quad (5.7)$$

$$R_n = K_n Q_n; \quad (5.8)$$

$$\begin{aligned} Q_{n+1} &= \overline{K_n Q_n + (J_n \bar{Q}_n + Q_n)} = \overline{(K_n Q_n)(J_n \bar{Q}_n + Q_n)} = \\ &= (\bar{K}_n + \bar{Q}_n)(J_n \bar{Q}_n + Q_n) = \bar{K}_n J_n \bar{Q}_n + \bar{K}_n Q_n + J_n \bar{Q}_n; \end{aligned}$$

$$Q_{n+1} = J_n \bar{Q}_n + \bar{K}_n Q_n. \quad (5.9)$$

Ținând seama de rel. 5.9 și tabelul de tranziție al CBB-SR asincron, tab. 5.1, putem alcătui tab. 5.7.

Tab. 5.7. Tabelul de tranziție al CBB-JK asincron

J_n	K_n	S_n	R_n	Q_{n+1}
0	0	0	0	Q_n
0	1	0	Q_n	0
1	0	\bar{Q}_n	0	1
1	1	\bar{Q}_n	Q_n	\bar{Q}_n

Se observă că pentru $J_n=K_n=1$, se obține la ieșire $Q_{n+1}=\overline{Q_n}$, deci ieșirile oscilează permanent între 0 și 1 logic.

5.4.2. Circuitul basculant bistabil JK sincron

Schema CBB-JK sincron, fig. 5.22, se obține din cea precedentă prin introducerea unei borne suplimentare pentru tact iar tabelul de tranziție este tab. 5.8.

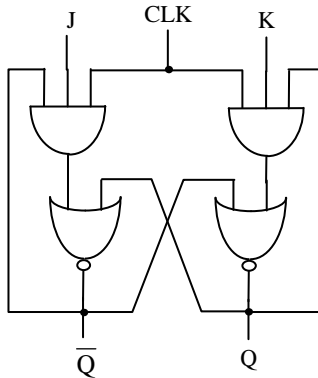


Fig. 5.22. Schema circuitului basculant bistabil JK sincron

Tab. 5.8. Tabelul de tranziție al circuitului basculant bistabil JK sincron

J_n	K_n	CLK	Q_{n+1}	
0	0	0→1	Q_n	Funcționare sincronă
1	0	0→1	1	
0	1	0→1	0	
1	1	0→1	$\overline{Q_n}$	
x	x	0	Q_n	Circuit blocat
0→1	0	1	1	Funcționare asincronă
0	0→1	1	0	

Se observă că prin legarea împreună a intrărilor J și K se obține un bistabil de tip T care, pentru $J_n=K_n=T_n=1$, basculează dintr-o stare în alta la comanda impulsului de CLK.

5.4.3. Circuitul basculant bistabil JK Master-Slave

Bistabilul JK-MS se obține prin conectarea în cascadă a două CBB-JK sincrone, transferul informației în secțiunea *slave* având loc pe frontul descrescător al impulsului de CLK. Tabelul de tranziție este tot tab. 5.7.

5.4.3.1. Numărătorul asincron

Numărătorul asincron, fig. 5.23, utilizează 4 circuite basculante bistabile JK Master-Slave, în regim de circuit basculant bistabil de tip T: $J_n=K_n=T_n=1$.

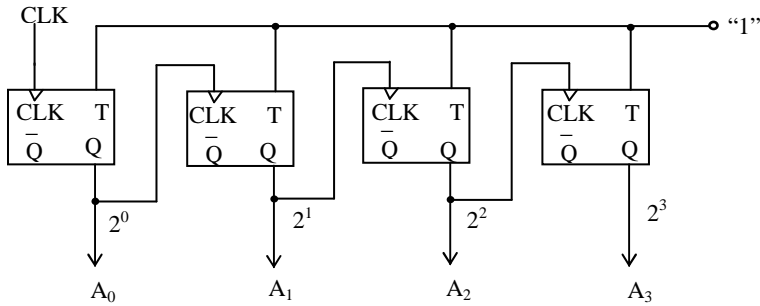


Fig. 5.23. Schema numărătorului asincron

Acest circuit se caracterizează prin faptul că impulsul de CLK nu acționează asupra tuturor bistabililor de tip T, ci numai asupra primului, ieșirile fiecărui bistabil fiind conectate la intrarea de CLK a bistabilului următor.

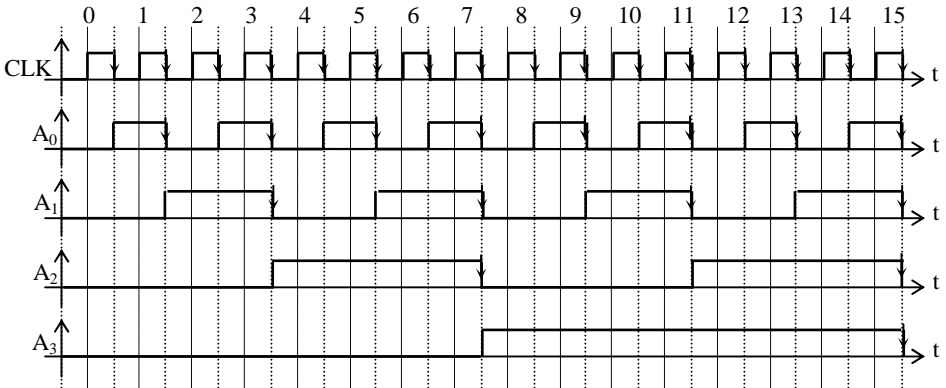


Fig. 5.24. Diagramele de semnal ale numărătorului asincron

În plus, toate intrările T ale bistabilelor fiind permanent conectate la 1 logic, valoarea logică a ieșirii fiecărui bistabil se modifică pe frontul negativ al impulsurilor primite pe intrarea de CLK, v. fig.5.24.

5.5. Conversia circuitelor basculante bistabile

În numeroase aplicații este necesară utilizarea unui anumit tip de CBB, practic fiind disponibil un altul. În aceste condiții, de mare ajutor sunt ecuațiile logice de legătură dintre diferite tipuri de bistabile, relații ce se pot obține pe baza tabelului comparativ, tab. 5.9.

Tab. 5.9. Tabel comparativ al diferitelor tipuri de CBB

		<i>Tip CBB</i>							
		<i>SR</i>		<i>D</i>		<i>T</i>		<i>JK</i>	
		$S_n R_n$	Q_{n+1}	D_n	Q_{n+1}	T_n	Q_{n+1}	$J_n K_n$	Q_{n+1}
Tabelul de adevăr		00	Q_n	0	0	0	Q_n	00	Q_n
		01	0					01	0
		10	1	1	1	1	\bar{Q}_n	10	1
		11	?					11	\bar{Q}_n
Ecuațiile logice	Q_{n+1}	$S_n + \bar{R}_n Q_n$		$D_n = S_n = \bar{R}_n$		$\bar{T}_n Q_n + T_n \bar{Q}_n$		$J_n \bar{Q}_n + \bar{K}_n Q_n$	
	\bar{Q}_{n+1}	$R_n + \bar{S}_n \bar{Q}_n$		$\bar{D}_n = \bar{S}_n = R_n$		$\bar{T}_n \bar{Q}_n + T_n Q_n$		$\bar{J}_n \bar{Q}_n + K_n Q_n$	

5.5.1. Conversia în T

Pentru realizarea conversiei JK→T sau D→T, trebuie găsită relația dintre intrarea T a bistabilului simulat și intrările JK sau D ale bistabilului disponibil - fig. 5.25.

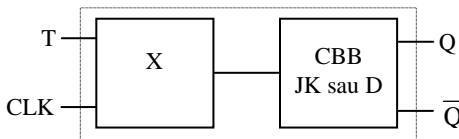


Fig. 5.25. Conversia în T: punerea problemei

Pentru aceasta se construiește tabelul ajutător 5.10 astfel: în primele două coloane se trec toate combinațiile logice posibile ale intrării (T_n) și stării (Q_n)

bistabilului simulat, în următoarele două coloane - valorile logice ale intrărilor J_nK_n și D_n , completate numai după trecerea în ultima coloană a valorilor logice ale ieșirii Q_{n+1} a bistabilului simulat.

Tab. 5.10. Explicativ pentru realizarea conversiilor în T

T_n	Q_n	J_nK_n	D_n	Q_{n+1}
0	0	0x	0	0
0	1	x0	1	1
1	0	1x	1	1
1	1	x1	0	0

Completarea cu valorile logice corespunzătoare a coloanelor J_nK_n și D_n se face pornind de la valorile logice ale stării prezente și viitoare (Q_n și Q_{n+1}), după o studiere atentă a tab. 5.9.

Astfel, situația $Q_n=0, Q_{n+1}=0$, se obține atunci când $J_nK_n=00$ sau 01 , deci $J_nK_n=0x$, unde prin "x" înțelegem "indiferent". $Q_n=1$ și $Q_{n+1}=1$ se obține când $J_nK_n=00$ sau 10 , deci $J_nK_n=x0$, ș.a.m.d.

Similar se procedează cu coloana lui D_n .

O dată completat tab. 5.10, se poate trece la sinteză, construind diagramele VK pentru funcțiile de ieșire J_n, K_n și D_n - fig. 5.26.

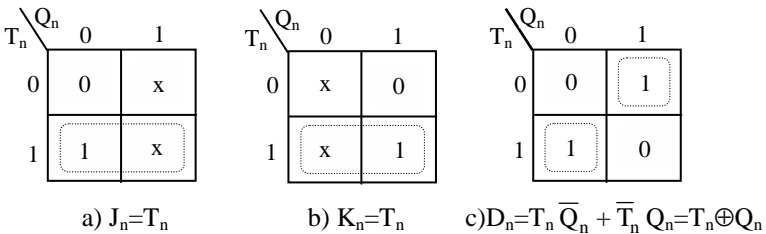


Fig. 5.26. Sinteza funcțiilor de ieșire ale blocului X din fig. 5.25

Cu aceste rezultate, schema generală din fig. 5.25 capătă aspectele concrete din fig. 5.27.

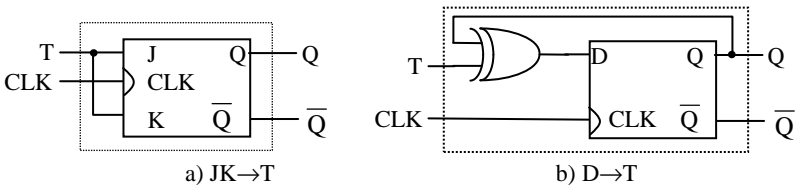


Fig. 5.27. Conversiile în T

5.5.2. Conversia în SR

Procedând similar obținem tab. 5.11 care permite implementarea circuitelor de conversie JK→SR și D→SR.

Tab. 5.11. Explicativ pentru realizarea conversiilor în RS

$S_n R_n$	Q_n	$J_n K_n$	D_n	Q_{n+1}
00	0	0x	0	0
00	1	x0	1	1
01	0	0x	0	0
01	1	x1	0	0
10	0	1x	1	1
10	1	x0	1	1
11	0	xx	x	0/0
11	1	xx	x	1/0

