

Cursul nr. 10

C10.1 Numărătoare asincrone

În circuitele numărătoare asincrone celulele flip-flop nu sunt declanșate de un semnal de clock comun. Primul flip-flop este comandat de un semnal de clock extern iar circuitele flip-flop următoare sunt comandate fiecare de ieșirea Q sau Q negat a circuitului flip-flop anterior. În figura 10.1 este propus spre analiză un numărător asincron pe 2 biți.

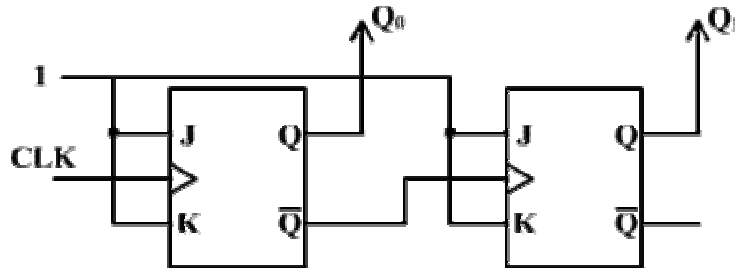


Fig. 10.1 Numărător asincron pe 2 biți

Numărătorul constă din două circuite flip-flop și are doi biți de ieșire (Q_0 și Q_1); circuitul este un numărător pe două nivele. Semnalul clock aplicat la intrare nu declanșează ambele celule flip-flop, de aceea este un asincron. Circuitele flip-flop sunt active pe frontul pozitiv al semnalului de clock. Intrările J și k sunt conectate împreună la 1 logic, astfel bistabilele vor comuta alternativ dintr-o stare în cealaltă.

Diagrama de timp asociată circuitului din figura 10.1 este prezentată în figura 10.2 (presupunem că starea inițială a ieșirilor este 0, $Q_0 = Q_1 = 0$).

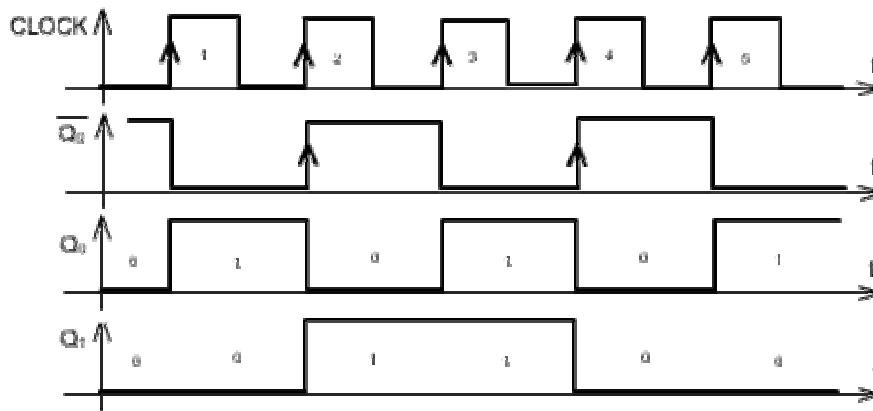


Fig. 10.2 Diagrama de timp pentru numărătorul din figura 10.1

Secvența de numărare este 00, 01, 10, 11, 00, 01, ... unde Q_0 este LSB (*Last Significant bit*). Circuitul este un numărător direct modulo 4. Direct se referă la ordinea de numărare - numărare crescătoare.

Semnalul clock aplicat din exterior este conectat doar la intrarea de sincronizare a primului flip-flop. Primul flip-flop va schimba starea pe fiecare front crescător al impulsului de clock, iar al doilea flip-flop va schimba starea doar atunci când ieșirea \bar{Q} a primului flip-flop va face o tranziție din 0 în 1. Din cauza întârzierilor de propagare prin flip-flop inerente, tranzițiile ieșirii \bar{Q} nu vor avea niciodată loc în același moment cu tranzițiile semnalului clock aplicat la intrare, prin urmare, circuitele flip-flop nu vor fi declanșate sincron, ci acestea vor opera în mod asincron. Pentru ca

explicarea funcționării să fie mai clară, în diagramele de timp prezentate în figura 10.2 tranzițiile pentru Q_0 , Q_2 și CLK sunt considerate simultane, deși numărătorul este un numărător asincron.

Numărătoarele asincrone sunt cunoscute și ca numărătoare cu propagare deoarece efectul semnalului clock aplicat primei celule nu este resimțit imediat de circuitul flip-flop de la nivelul următor, fapt datorat întârzierilor de propagare. Efectul se semnalului clock de intrare se propagă din celulă în celulă până ce ultima celulă din structură "simte" efectul. Propagarea poate fi asemănată cu cea a valurilor pe apă.

Din analiza formelor de undă se poate observa că ieșirea Q_0 este ieșirea mai puțin semnificativă (LSB) și că frecvența semnalului la această ieșire este egală cu $1/2$ din frecvența semnalului clock. Se poate observa în continuare că ieșirea Q_1 este ieșirea mai semnificativă (MSB) și frecvența acestui semnal este $1/4$ din frecvența semnalului clock de la intrare.

Numărătorul asincron pe doi biți din figura 10.1 poate avea patru stări distincte, fiecare din ele corespunde unei valori de numărare. Prin extensie, un numărător format cu n celule flip-flop poate avea 2^n stări. Numărul de stări distincte posibile este cunoscut ca **număr modulo**. Un numărător pe 2 biți poate fi un numărător modulo-4.

Un numărător modulo- n poate fi referit și ca **divizor cu n** deoarece la ieșirea circuitului flip-flop cel mai îndepărtat de intrarea clock se produce un puls la fiecare n pulsuri ale semnalului clock aplicat primei celule flip-flop. Numărătorul exemplificat în figura 10.1 este un divizor cu 4.

În figura 10.3 este prezentat un numărător asincron pe 3 biți și diagrama de semnal asociată unui ciclu de numărare. Ieșirile sunt $Q_0Q_1Q_2$ și circuitele flip-flop sunt active pe front negativ. Circuitul funcționează la fel ca și numărătorul asincron pe 3 biți, cu excepția că acest circuit are 8 stări datorită prezenței celui de-al treilea flip-flop.

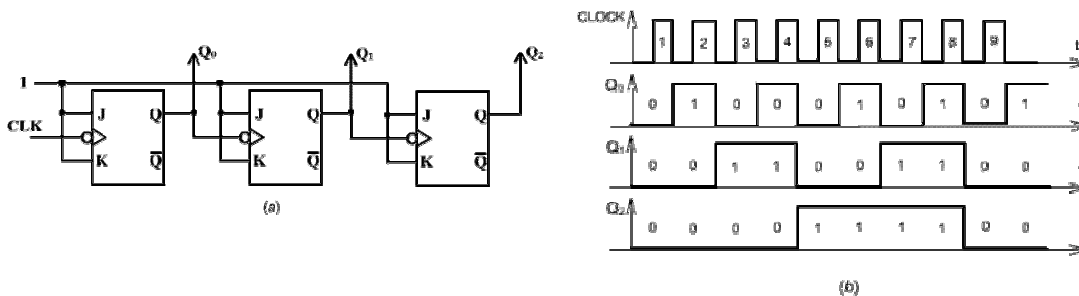


Fig. 10.3 Numărător asincron pe 3 biți: (a) Circuit logic; (b) Diagramă de semnal

Numărătoarele binare prezentate până în acest punct au 2^n stări. Există și numărătoare care au mai puțin de 2^n stări; acestea sunt proiectate să aibă un număr de stări diferit de 2^n . Aceste numărătoare sunt numărătoare cu secvență de numărare trunchiată. Trunchierea secvenței de numărare se obține prin forțarea numărătorului să reia secvența de la început înainte de a parcurge toate stările. Un numărător cu secvență trunchiată des întâlnit este numărătorul modulo 10. Un numărător cu 10 stări este numit **numărător decadic**. Circuitul prezentat în figura 10.4 este un numărător decadic.

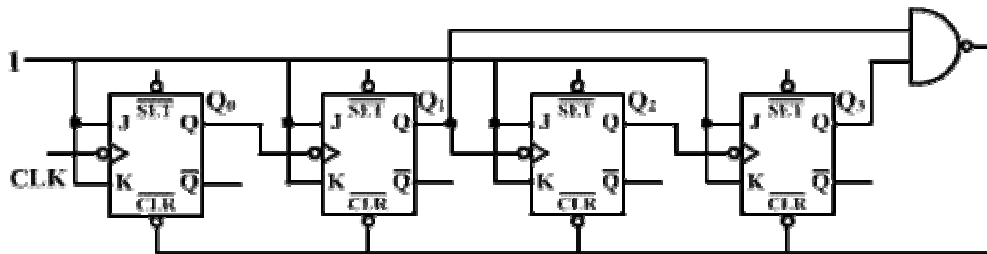


Fig. 10.4 Numărător decadic

Când numărătorul ajunge la zece (1010 - binar), toate circuitele flip-flop sunt șterse.

Observați că pentru decodificarea cifrei 10 sunt utilizate doar ieșirile Q_1 și Q_3 . Acest mod de decodificare este numit decodificare parțială. Acest lucru este posibil deoarece prima stare după 0 care are Q_1 și Q_3 în 1 simultan este starea ce corespunde numărului zece

În anumite aplicații, există necesitatea de a număra atât crescător cât și descrescător; acest numărător este numit **numărător reversibil**. Circuitul din figura 10.5 prezintă o implementare posibilă a unui numărător pe trei biți reversibil. Circuitul numără crescător sau descrescător, în funcție de starea semnalelor aplicate la o intrările de comandă UP și DOWN.

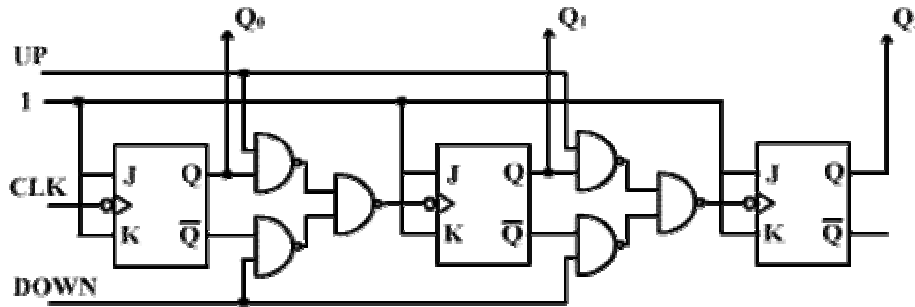


Fig. 10.5 Numărător pe 3 biți reversibil

Când intrarea UP este 1 și DOWN este 0, circuitul NAND dintre FF_0 și FF_1 va permite propagarea ieșirii neinvertate Q a celulei FF_0 spre intrarea clock a celulei FF_1 . În mod asemănător, trecerea ieșirii Q a celulei FF_1 spre intrarea clock a următoarei celule va fi controlată prin poarta NAND dintre FF_1 și FF_2 . Prin această condiționare a aplicării semnalelor la intrările clock numărătorul va număra crescător.

Când intrarea UP este 0 și DOWN este 1, se va permite trecerea ieșirilor invertate ale celulelor FF_0 și FF_1 către intrările clock ale celulelor FF_1 , respectiv FF_2 . Dacă starea inițială a celulei flip-flop este 0, atunci numărătorul va parcurge, pe măsură ce se aplică pulsuri la intrarea clock, secvența redată în tabelul 10.1 dintre FF_0 și FF_2 .

Tabelul 10.1

Q_2	Q_1	Q_0
0	0	0
1	1	1
1	1	0
1	0	1
1	0	0
0	1	1
0	1	0
0	0	1

Este de reținut faptul că un numărător reversibil asincron este mai lent decât un numărător direct (crescător) sau un numărător invers (descrescător) din cauza întârzierilor suplimentare introduse de circuitele NAND.

C10.2 Numărătoare sincrone

La numărătoarele sincrone, toate intrările clock ale tuturor celulelor flip-flop sunt conectate împreună și sunt declanșate de același semnal. În acest fel toate circuitele flip-flop schimbă starea simultan (în paralel). Prin urmare, în acest tip de numărătoare nu sunt întârzieri cumulative datorită propagării efectului prin toate celulele.

Numărătoarele sincrone pot fi și ele proiectate să numere direct sau invers; în plus acestea pot fi proiectate să parcurgă secvențe de numere neconsecutive. Circuitul din figura 10.6 este numărător sincron pe 3 biți.

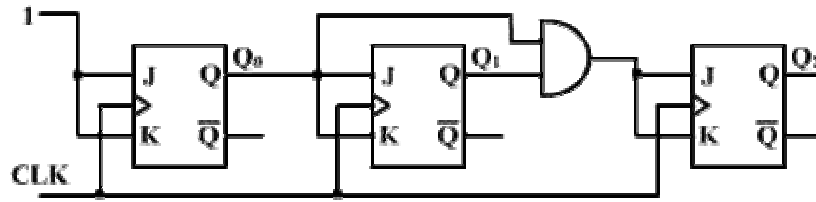


Fig. 10.6 Numărător sincron pe 3 biți

Intrările J și K ale celulei FF₀ sunt conectate la 1 permanent. Intrările J și K ale celulei FF₁ sunt comandate de ieșirea Q a celulei FF₀; intrările J și K ale celulei FF₂ sunt comandate de ieșirea unei porți AND care operează ieșirile Q ale celulelor FF₀ și FF₁.

După al treilea impuls de clock, ambele ieșiri ale FF₀ și FF₁ sunt 1 și frontul pozitiv al celui de-al patrulea clock va determina celula FF₂ să comute datorită prezenței porții AND. Secvența de numărare a numărătorului din figura 10.6 este redată în tabelul 10.2.

Tabelul 10.2

Puls de clock	Q ₂	Q ₁	Q ₀
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Principalul avantaj al unui numărător sincron este faptul că nu există întârzieri cumulative deoarece toate circuitele flip-flop sunt declanșate simultan. Din acest motiv frecvența de numărare posibilă cu un numărător sincron va fi multă mai mare decât cea posibilă cu un numărător asincron.

Ca și în cazul unui numărător decadic asincron, un numărător decadic sincron va număra de la 0 la 9 după care va relua ciclul de la 0. Reluarea ciclului este posibilă prin forțarea comutării în starea 0000 atunci când este atinsă starea 1010. Circuitul din figura 10.7 realizează o astfel de numărare cu secvență trunchiată.

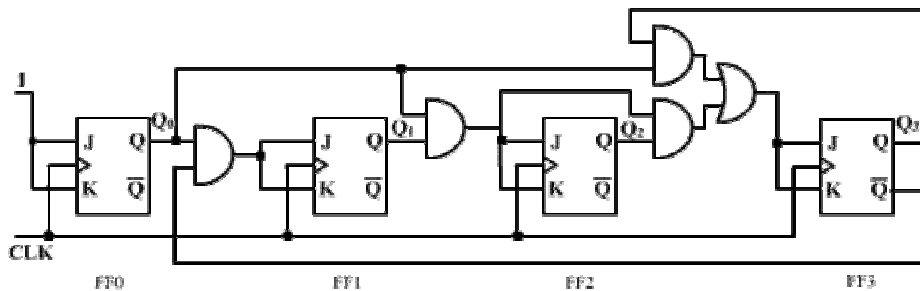


Fig. 10.7 Numărător decadic sincron

Secvența de numărare a circuitului din figura 10.7 este redată în tabelul 10.3.

Tabelul 10.3

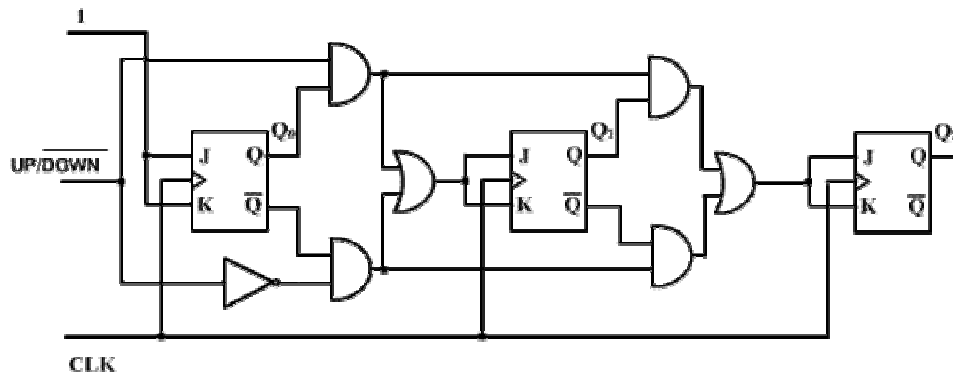
Puls de clock	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Urmărind secvența din tabelul 10.3 pot fi notate următoarele observații:

- Q₀ comută la fiecare puls de clock
- Q₁ se modifică la pulsul de clock ce urmează de fiecare dată când Q₀=1 și Q₃=0
- Q₂ se modifică la pulsul de clock ce urmează de fiecare dată când Q₀=Q₁=1
- Q₃ se modifică la pulsul de clock ce urmează de fiecare dată când Q₀=Q₁=Q₂=1 și Q₀=Q₃=1

Circuitul este implementat cu porți Si și SAU așa cum este prezentat în figura 10.7

În figura 10.8 este prezentat un circuit numărător reversibil pe 3 biți. Asemănător cu numărătorul reversibil asincron, un numărător sincron reversibil este prevăzut cu o intrare de comandă up/down.

**Fig. 10.8** Numărător sincron reversibil pe 3 biți

În tabelul 10.4 este redată secvența de numărare pentru circuitul din figura 10.8.

Tabelul 10.4

	Q ₃	Q ₂	Q ₁	
	0	0	0	UP/DOWN=0
	0	0	1	
	0	1	0	
	0	1	1	
↑ UP/DOWN=1	1	0	0	
	1	0	1	
	1	1	0	
	1	1	1	

Analizând secvența din tabelul 10.4 se poate constata că:

- atât în numărare directă cât și în numărare inversă, Q_0 comută la fiecare puls de clock
- în numărare directă, Q_1 comută doar la primul puls de clock după ce $Q_0=1$
- în numărare inversă, Q_1 comută doar la primul puls de clock după ce $Q_0=0$
- în numărare directă, Q_2 Q_1 comută doar la primul puls de clock după ce $Q_0=Q_1=0$
- în numărare inversă, Q_2 comută doar la primul puls de clock după ce $Q_0=Q_1=0$

Circuitul este implementat cu porți AND, OR și NOT interconectate ca în figura 10.8.

C10.3 Registre paralele

Registrele paralele elemente de bază utilizate pentru a memora cuvinte binare. Fiecare bit al unui cuvânt este depozitat într-o celulă flip-flop. Atât intrările cât și ieșirile sunt paralele, toți biții de date apar la ieșirile paralele imediat după ce a avut loc citirea simultană a intrărilor paralele. Circuitul din figura 10.9 este un registru paralel de 4 biți construit cu celule flip-flop D.

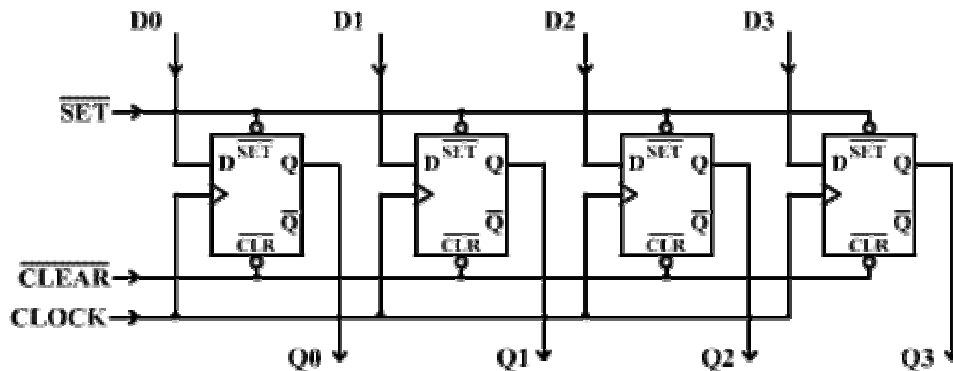


Fig. 10.9 Registru paralel pe 4 biți

Intrările D ale fiecărui flip-flop constituie intrările paralele; ieșirile Q ale fiecărui flip-flop constituie ieșirile paralele. De îndată ce registrul primește front activ de clock, toate datele de la intrările D sunt înscrise în celulele bistabile și apar la ieșirile Q corespunzătoare simultan. Ieșirile pot fi resetate în mod asincron dacă se aplică un 1 la intrarea CLEAR.

Caracteristica reprezentativă a acestor registre este faptul că toate datele sunt memorate și transferate la ieșire sincron.

EXEMPLUL 10.1

Registrele sunt implementate pentru cuvinte de 4 biți sau pentru cuvinte de 8 biți. Există registre care dispun de intrări asincrone de ștergere, ca cel din figura 10.9, și registre care dispun de intrare de validare (ENABLE). Circuitul integrat 74x377 este un registru cu intrare de validare. Simbolul logic al circuitului integrat și schema logică pentru calea unui bit sunt prezentate în figura 10.10.

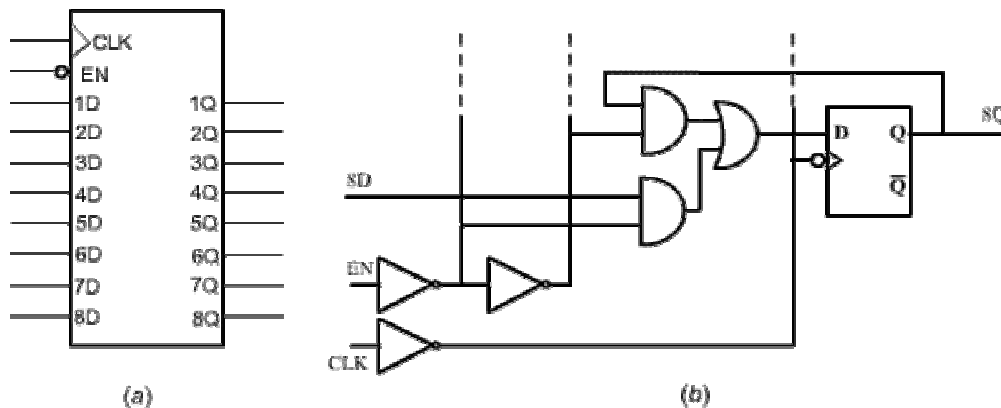


Fig. 10.10 74x377: (a) Simbol; (b) schemă parțială

Pulsul de clock este aplicat tuturor celor 8 celule flip-flop. Când intrarea EN (ENABLE) este 0, intrarea 8D este conectată la intrarea circuitului flip-flop prin rețeaua de porți AND, OR. Primul front crescător al semnalului clock va transfera intrarea 8D la ieșirea 8Q (același lucru se întâmplă la fiecare din cele 8 circuite flip-flop).

Dacă intrarea EN este 1, nu au loc modificări ale stării circuitelor flip-flop deoarece ieșirea Q este adusă la intrarea D prin rețeaua de porți AND, OR.

C10.4 Registre seriale și combinații ale acestora

Un registru capabil să deplaseze la un moment dat informația cu o poziție este numit **registru de deplasare**. Registrele seriale sunt registre de deplasare.

Configurația logică a unui registru de deplasare constă dintr-un șir de circuite flip-flop conectate în cascadă; ieșirea unui flip-flop este conectată la intrarea celulei flip-flop vecine. În registrele de deplasare operația are loc sincron; toate circuitele flip-flop sunt declanșate de un același semnal de clock. Mișcările de bază posibile într-un registru de deplasare pe 4 biți sunt prezentate în figura 10.11.

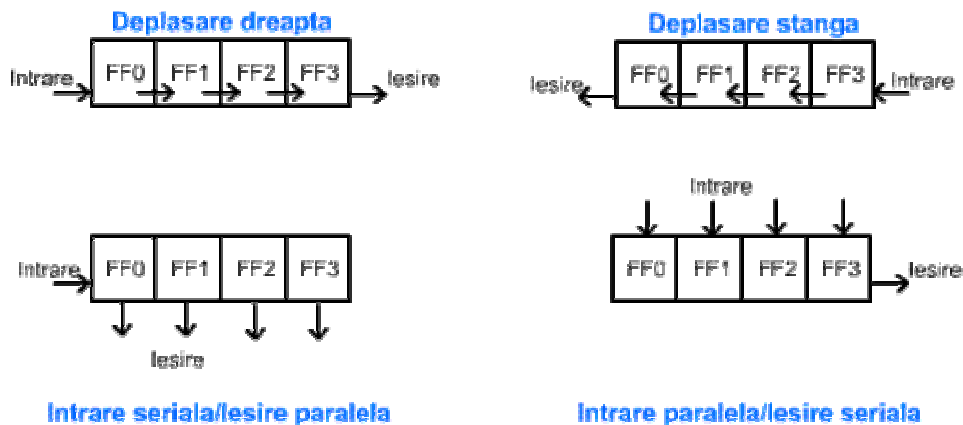


Fig. 10.11 Mișcări de bază într-un registru de deplasare pe 4 biți

Un registru de deplasare pe 4 biți poate fi construit utilizând 4 celule flip-flop D după exemplul prezentat în figura 10.12.

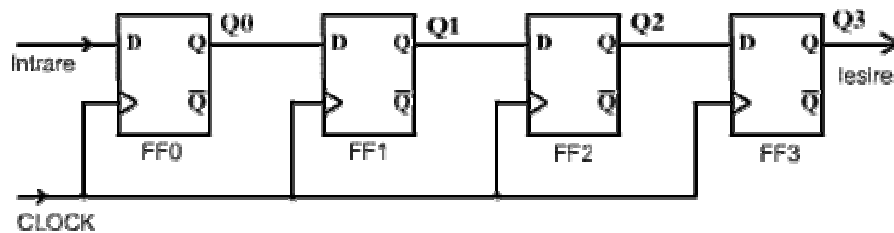


Fig. 10.12 Registru serial pe 4 biți

Să presupunem că în stare inițială circuitele flip-flop sunt resetate ($Q_0=Q_1=Q_2=Q_3=0$) și că bitul de intrare este aplicat celei mai din stânga celule. Înainte de primul front crescător al semnalului clock, intrarea de date este conectată la 1 și coboară în 0 în timpul primei perioade al semnalului clock. Diagrama de timp a circuitului este arătată în figura 10.13.

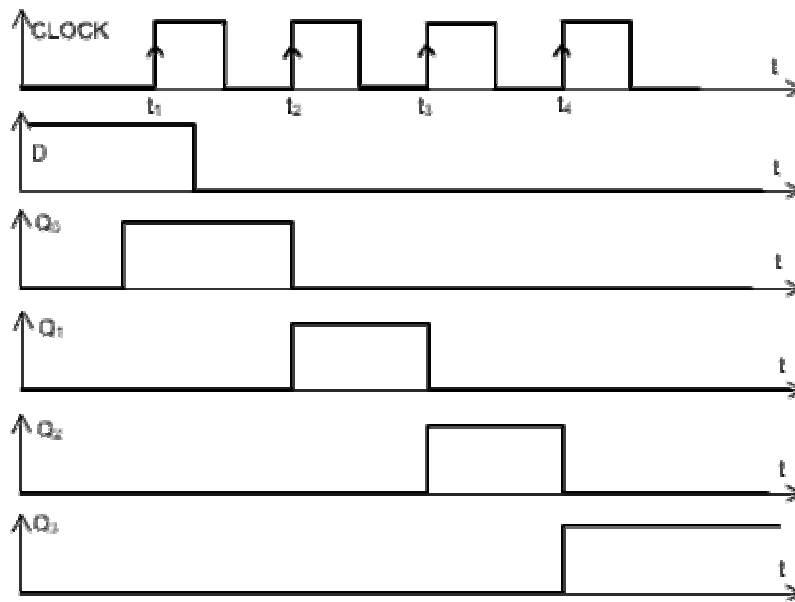


Fig. 10.13 Diagrama de timp pentru circuitul din figura 10.12

Deși întârzierile de propagare nu sunt reprezentate în diagramă, acestea trebuie totuși considerate pentru a obține rezultatul dorit.

La fiecare moment de timp începând cu t_1 , o deplasare la dreapta a informației corespunde unei schimbări de stare care este echivalentă cu o operație de împărțire la 2. Dacă se operează în ordine cronologică inversă (de la t_4 la t_1), deplasarea la stânga va fi echivalentă cu o operație de înmulțire cu 3.

Un registru de deplasare bidirecțional care este capabil să deplaseze informația la dreapta (Shift Right) sau la stânga (Shift Left), va fi capabil să proceseze înmulțiri cu 2 sau împărțiri la 2. Registrele de deplasare pot procesa informația prin deplasarea ei urmată de livrarea la ieșire în această formă.

Registru cu Intrare serială - Ieșire paralelă

În cazul acestor registre datele sunt înscrise serial, în același mod ca în cazul registrelor seriale. Diferă modul în care biții de date sunt scoși la ieșire. Datele sunt disponibile la ieșire simultan, din toate celulele, de îndată ce informația a fost preluată în celulele flip-flop. Un registru cu intrare serială și ieșire paralelă este prezentat în figura 10.14.

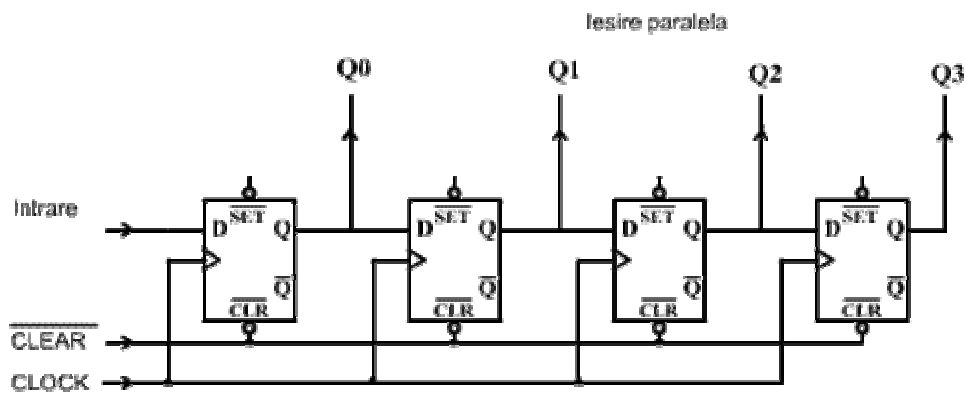


Fig. 10.14 Registru de 4 biți cu intrare serială și ieșire paralelă

În animația de mai sus puteți urmări cum numărul binar 1001 este deplasat și cum este accesibil la ieșirile Q ale registrului.

Registru cu intrare paralelă - ieșire serială

Un registru de deplasare pe 4 biți cu intrare paralelă și ieșire serială este prezentat în figura 10.15. Circuitul utilizează celule D și porți NAND pentru a înscris datele.

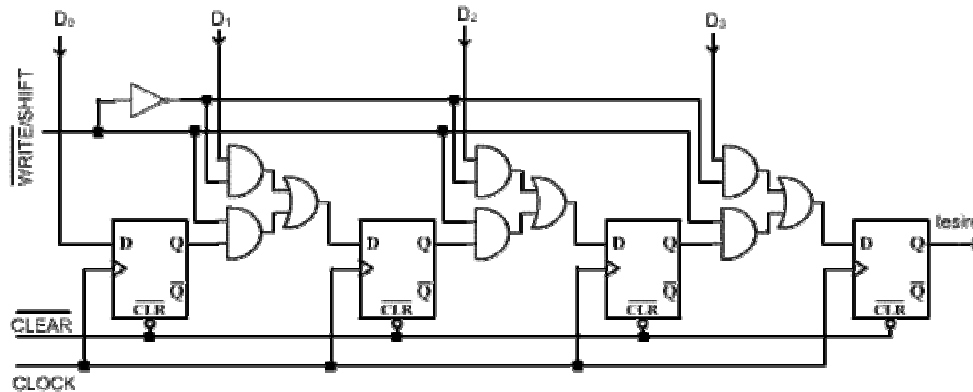


Fig. 10.15 Registru cu intrare paralelă și ieșire serială pe 4 biți

D_0 , D_1 , D_2 și D_3 reprezintă intrările paralele, iar D_0 este cel mai semnificativ bit și D_3 este cel mai puțin semnificativ bit. Pentru a înscris date în registru este necesar ca linia de comandă de mod ($nWrite/Shift$) să fie 0; datele sunt înscrise în flip-flop pe frontul activ al semnalului de clock. Informația înscrisă în celulele flip-flop poate fi deplasată dacă linia de control de mod este 1. La fiecare front activ al semnalului de clock registrul va procesa o deplasare la dreapta.

Registre de deplasare bidirecționale

Registrele de deplasare prezentate până acum pot opera doar deplasarea la dreapta a informației. Fiecare deplasare la dreapta are efectul unei împărțiri la 2. Dacă se schimbă sensul de deplasare, la stânga, aceasta are efectul unei înmulțiri cu 2 a informației. Cu o rețea de porți logice interconectate convenabil este posibil ca un același registru serial să poată procesa atât deplasări la dreapta cât și deplasări la stânga. Un astfel de registru dispune de o linie de control numită $nLeft/Right$. Registrul poate opera oricare din combinațiile posibile amintite (intrare serială - ieșire serială, intrare serială - ieșire paralelă, intrare paralelă - ieșire serială).

EXEMPLUL 10.2

Registrele sunt întâlnite în multe aplicații cum ar fi: stocarea temporară de cuvinte binare, executarea de conversii serie-paralel, pentru a produce întârzieri de timp, pentru a opera înmulțiri sau împărțiri cu 2^n , sau pentru a simplifica logica combinațională.

Un exemplu de registru care poate să facă toate operațiile amintite este circuitul integrat 74x194. Schema logică a circuitului este prezentată în figura 10.16.

D , C , B , A sunt intrările paralele și Q_D , Q_C , Q_B , Q_A sunt ieșirile paralele. L_{IN} și R_{IN} sunt intrările seriale pentru cazul în care se efectuează o deplasare la stânga, respectiv la dreapta. Intrarea

$CLEAR$ șterge simultan toate celulele flip-flop. Prin intrarea CLK se aplică semnal de clock tuturor celulelor simultan. S_1 și S_0 sunt decodificate în interiorul circuitului prin porți ȘI. În funcție de combinația aplicată la intrările S_1 și S_0 , registrul va executa diferite operații, așa cum este arătat în tabelul 10.5.

Tabel 10.5

S_1	S_0	Operație
0	0	Păstrează starea
0	1	Deplasare dreapta
1	0	Deplasare stânga
1	1	Încărcare paralelă

Când registrul efectuează deplasare la dreapta, ieșirea serială este Q_D ; când registrul efectuează deplasare la stânga, ieșirea serială este Q_A .

Circuitele SAU permit transferarea către intrările celulelor flip-flop a semnalelor potrivite operației selectate prin combinația aplicată la liniile de comandă S_1 și S_0 .

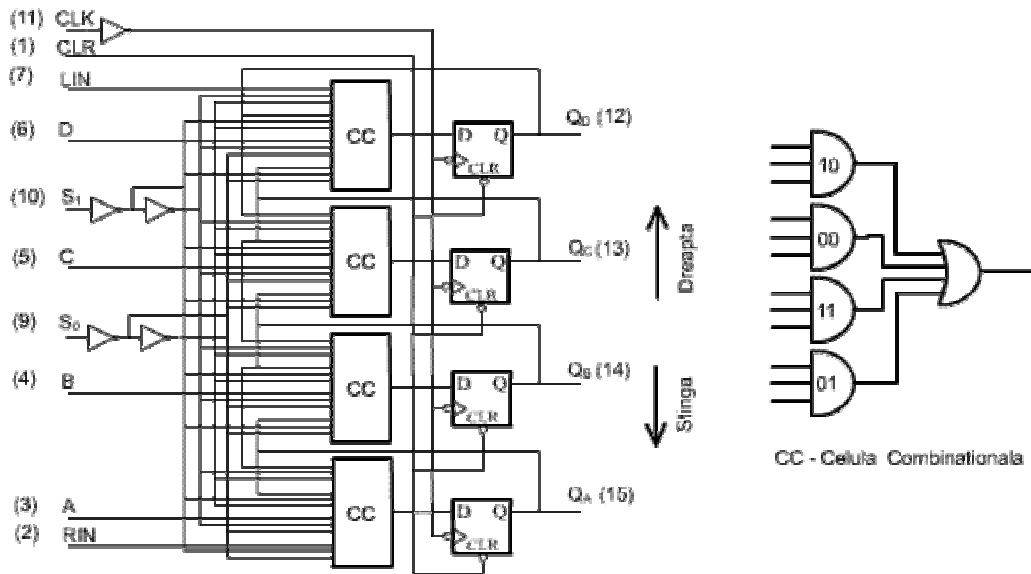


Fig. 10.16 Registru universal 74x194