
Universitatea Politehnica Timișoara

Facultatea de Automatică și Calculatoare

Proiectarea cu Microprocesoare

Anul 3 CTI

Titular: prof. dr. ing. Mircea POPA

Contact:
cab. B427
tel. 0256 403260
mircea.popa@upt.ro

Curs 1

Proiectarea cu Microprocesoare

■ Conținut:

- Introducere
- Unitatea centrală
- Conectarea memoriilor
- Conectarea porturilor de intrare/ ieșire
- Circuite specializate programabile
- DMA
- Sistemul de întreruperi
- Aplicații
- Microprocesoare moderne

Proiectarea cu Microprocesoare

■ Bibliografie:

- B. B. Brey, *The Intel Microprocessors: Architecture, Programming and Interfacing*; Prentice Hall, 2009
- J. L. Antonakos, *The Intel microprocesors family: hardware and software principles and applications*; Thomson Delmar, 2007
- M. Popa, *Proiectarea microsystemelor digitale*; Orizonturi Universitare, Timișoara, 2003
- M. Popa, *Sisteme cu microprocesoare*; Orizonturi Universitare, Timișoara, 2003

■ **Obiectivele disciplinei:**

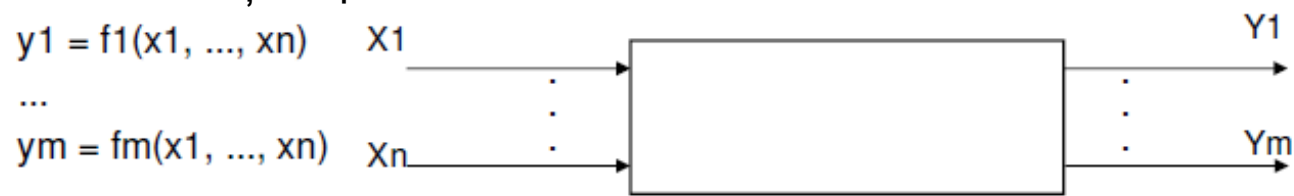
□ **Disciplina își propune:**

- cunoașterea definiției, caracteristicilor, structurii și funcționării unui microsistem cu microprocesor (digital);
- studiul problemelor tipice care apar la proiectarea unui microsistem digital bazat pe un microprocesor didactic precum și a soluțiilor tipice;
- studiul unor aplicații tipice;
- dobândirea de cunoștințe în vederea proiectării unui microsistem digital, bazat pe microprocesor, cu o structură tipică.

1. Introducere

1.1. Circuite logice

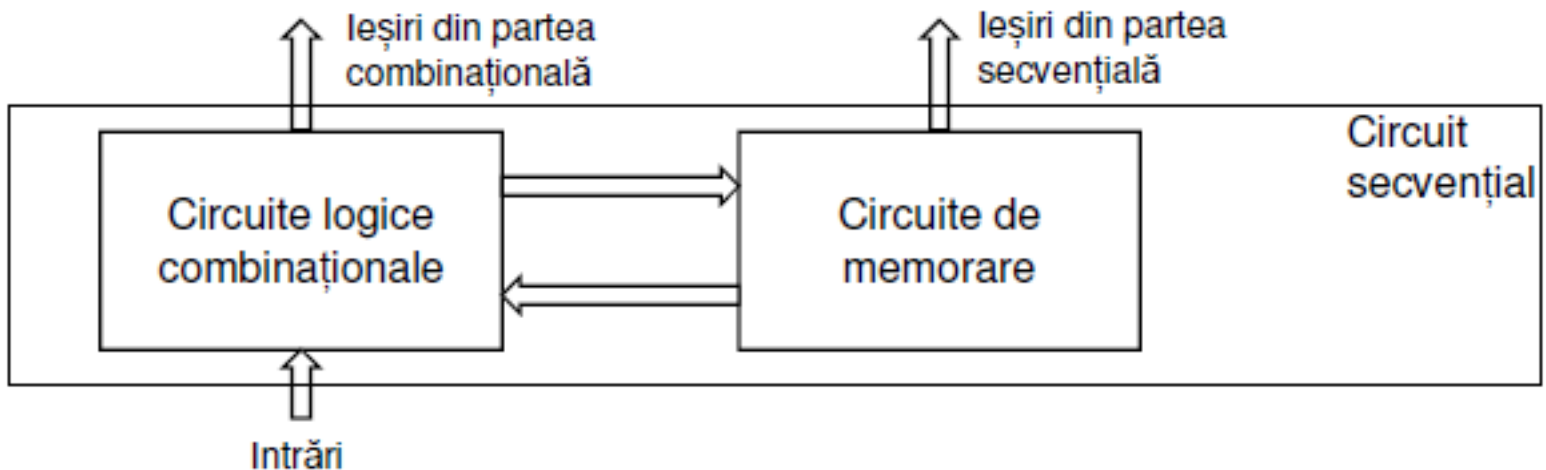
- Sunt circuite care lucrează cu nivele de tensiune, la intrare și ieșire, corespunzătoare celor 2 nivele logice: 0 și 1;
- Există 2 tipuri de circuite logice:
 - Combinaționale;
 - Secvențiale.
- Circuite combinaționale:
 - Sunt acele circuite logice care nu au elemente de memorie sau legături de reacție între ieșiri și intrări; ieșirile acestor circuite reacționează doar la modificarea intrărilor fără a ține seama de starea circuitului;
 - Circuitele combinaționale sunt descrise prin ecuații logice în care apar variabilele de intrare dar nu și timpul:



Proiectarea cu Microprocesoare

■ Circuite secvențiale

- Circuite la care ieșirile depind de intrări dar și de starea curentă a circuitului, care, la rândul ei, depinde de nivelul intrărilor la momente de timp anterioare; țin seama, în funcționarea lor, de istoria intrărilor; sunt circuite cu memorare fiindcă ieșirile depind de intrările la momente de timp anterioare;

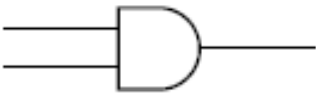

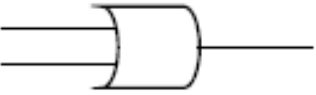

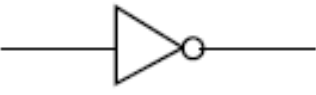
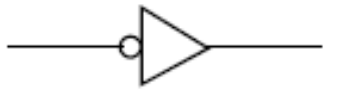


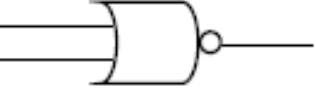

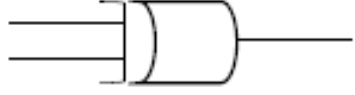


- Există 2 tipuri de circuite secvențiale:
 - Asincrone: la care ieșirile se modifică atunci când se modifică intrările, ținând seama și de starea circuitului;
 - Sincrone: la care ieșirile se pot modifica numai la momente de timp bine determinate, indicate prin intermediul unui semnal dedicat, numit tact (sau clock); forma unui semnal de tact:

Proiectarea cu Microprocesoare

1.1.1 Circuite combinaționale

■ Porți: circuite logice de bază:

Funcția circuitului	Reprezentare clasică	Reprezentare alternativă
ȘI		
SAU		
NU		
ȘI-NU		
SAU-NU		
SAU-EXCLUSIV		

Proiectarea cu Microprocesoare

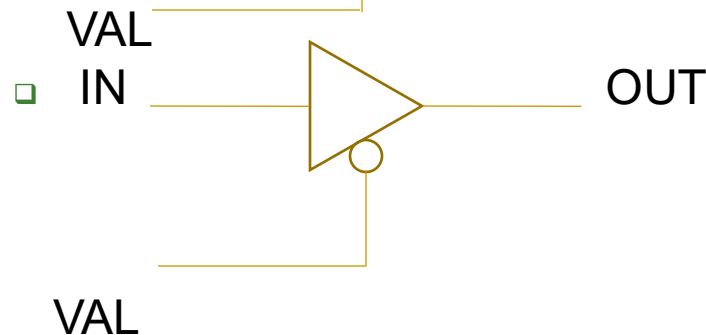
■ Porți cu 3 stări:

- ❑ Ieșirea: 1 logic, 0 logic, stare de înaltă impedanță;
- ❑ Starea de înaltă impedanță: circuitul nu încarcă linia, ca și cum nu ar exista;

❑ Simbol:



VAL = 1, OUT = IN
VAL = 0, OUT = a 3-a stare

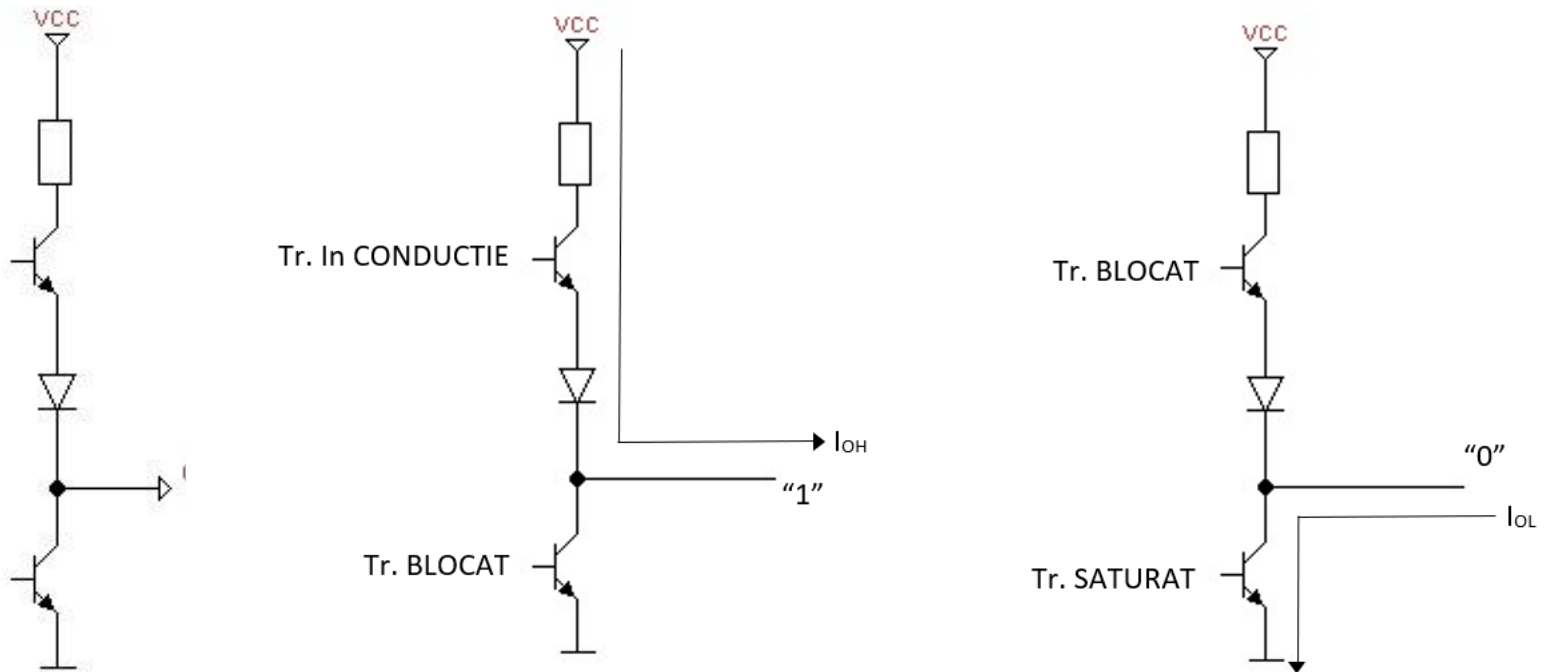


VAL = 1, OUT = a 3-a stare
VAL = 0, OUT = IN

- ❑ Avantaj: permite legarea ieșirilor a mai multor circuite împreună;

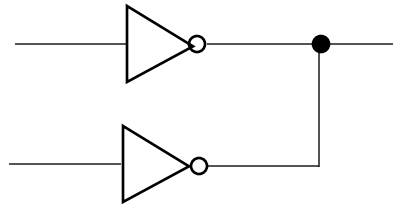
Proiectarea cu Microprocesoare

- ❑ Nu este permisă legarea ieșirilor a mai multor circuite logice care nu au facilitatea de a 3-a stare; de ce?



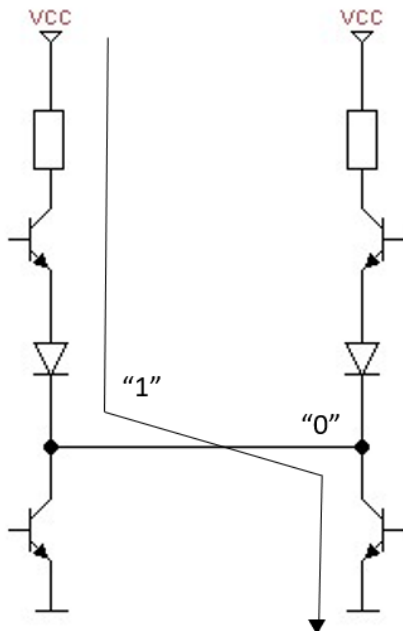
Proiectarea cu Microprocesoare

- Fie următoarea schemă:



(NU este permisă!)

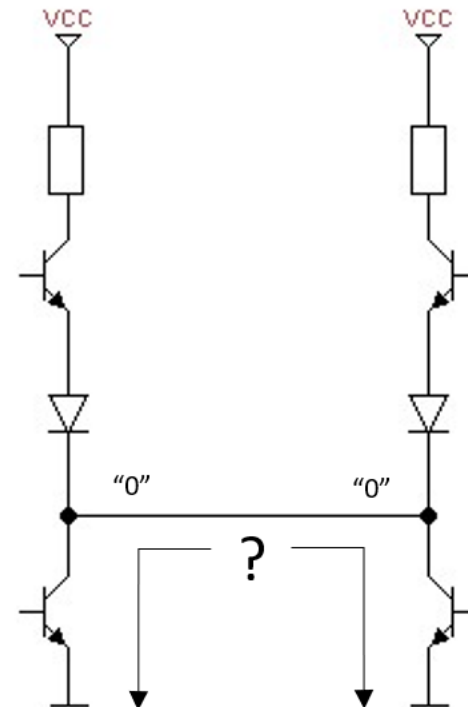
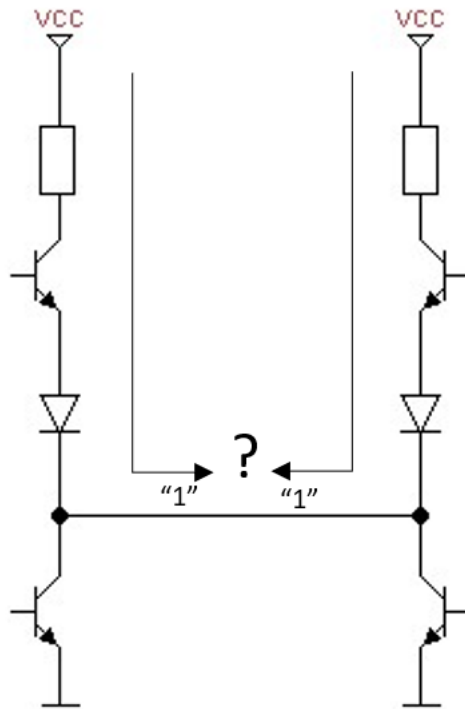
- Sunt posibile situațiile:



trece curent mare prin cele 2 etaje de ieșire;
este limitat doar de rezistența de sus ($130\ \Omega$);
urmarea: unul sau ambele circuite se încălzesc
și nu va/vor mai funcționa;

Proiectarea cu Microprocesoare

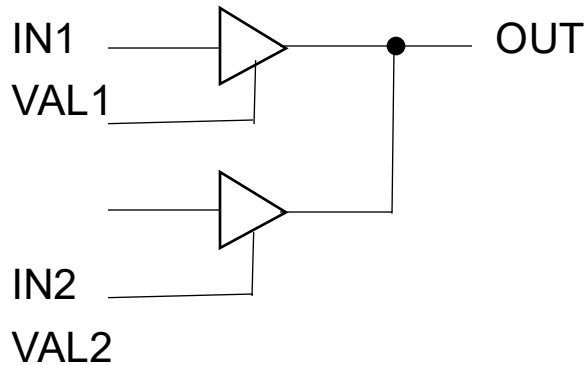
Sau:



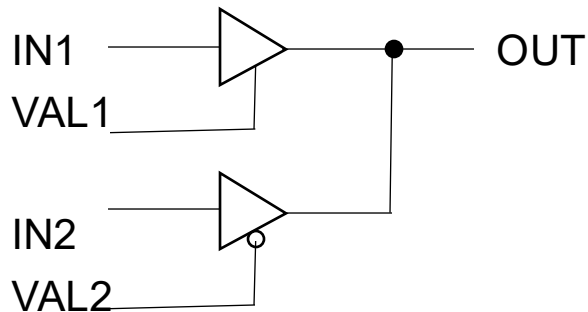
- nu trece curent, nu se strică circuitele;

Proiectarea cu Microprocesoare

- Fie schemele:



$VAL1 = 1, VAL2 = 0 \rightarrow OUT = IN1$
 $VAL1 = 0, VAL2 = 1 \rightarrow OUT = IN2$
 $VAL1 = 0, VAL2 = 0 \rightarrow OUT = \text{a 3-a stare}$
 $VAL1 = 1, VAL2 = 1 \rightarrow \text{interzis};$

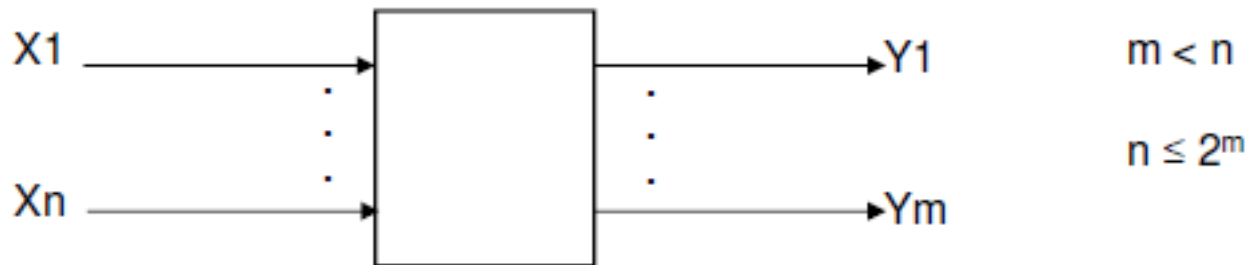


$VAL1 = 1, VAL2 = 0 \rightarrow \text{interzis}$
 $VAL1 = 0, VAL2 = 1 \rightarrow OUT = \text{a 3-a stare}$
 $VAL1 = 0, VAL2 = 0 \rightarrow OUT = IN2$
 $VAL1 = 1, VAL2 = 1 \rightarrow OUT = IN1$

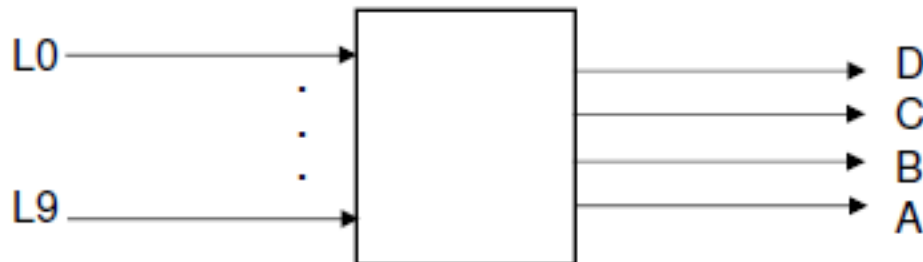
Proiectarea cu Microprocesoare

■ Codificatorul

- Este circuitul combinațional care transformă modul de reprezentare a informației din sistemul de numerație extern, accesibil, operatorului, în sistemul de numerație intern, binar, al unui sistem de calcul;
- Schema bloc:



- Ex.: un codificator zecimal – binar:



Proiectarea cu Microprocesoare

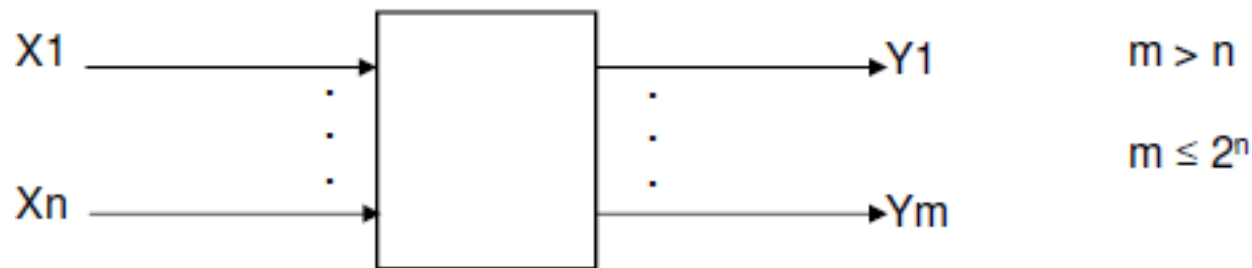
- Tabelul de adevăr:

L9	L8	L7	L6	L5	L4	L3	L2	L1	L0	D	C	B	A
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	1

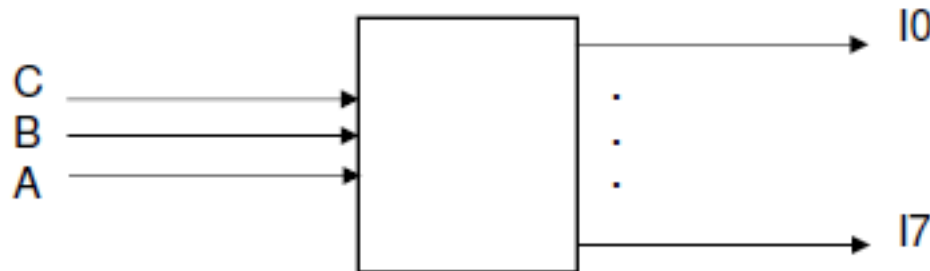
Proiectarea cu Microprocesoare

■ Decodificatorul

- Este circuitul combinațional care asociază fiecărei combinații de la intrare o ieșire, în sensul că acea ieșire este diferită de celelalte (activă la 1 în timp ce toate celelalte sunt active la 0 sau activă la 0 în timp ce toate celelalte sunt active la 1);
- Se folosește atunci când o anumită combinație de cod trebuie transformată în un semnal de comandă pe o anumită linie;
- Schema bloc:

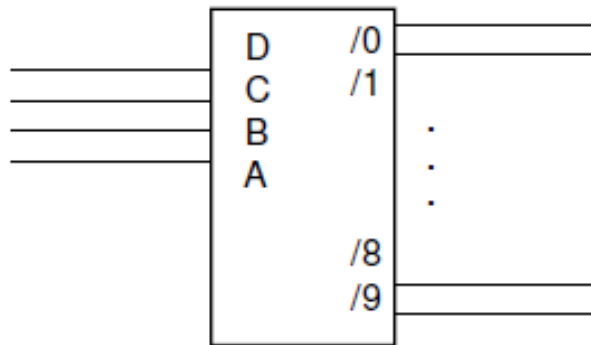


- Ex.: să se proiecteze un decodificator $3 \rightarrow 8$:



Proiectarea cu Microprocesoare

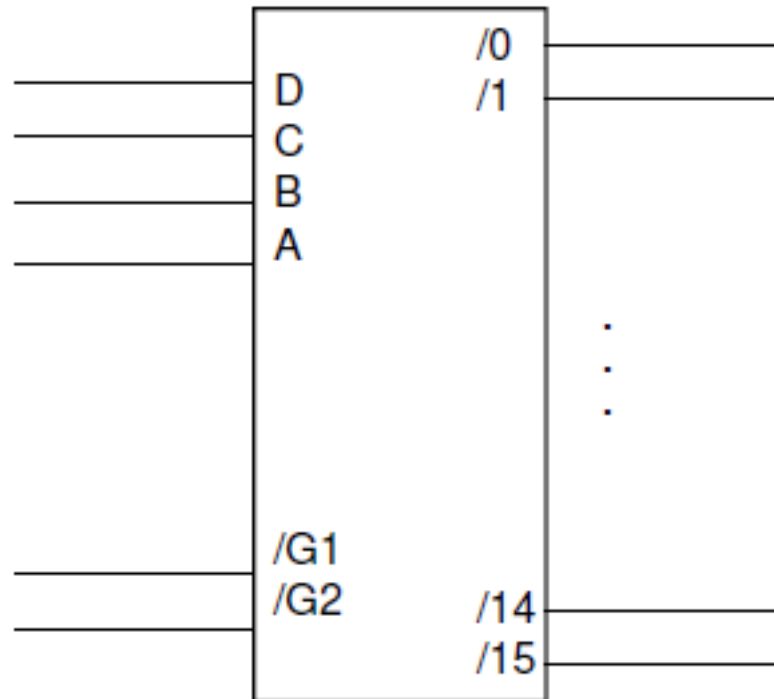
- Există mai multe decodificatoare integrate:
 - Au ieșiri active la 0 logic spre deosebire de cele proiectate și realizate cu porți care pot avea ieșiri active la 1 sau la 0
 - Majoritatea au și intrări de validare care trebuie să fie la 0 pentru ca circuitul să lucreze;
 - Pot fi combinate pentru a obține decodificatoare cu mai multe ieșiri; de ex. 2 decodificatoare integrate care au fiecare câte 8 ieșiri pot fi combinate pentru a obține un decodificator cu 16 ieșiri;
- Circuitul decodificator binar – zecimal 74x42:
 - Este un decodificator 4 → 10:



D	C	B	A	/9	/8	/7	/6	/5	/4	/3	/2	/1	/0
0	0	0	0	1	1	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	0	1
0	0	1	0	1	1	1	1	1	1	1	0	1	1
0	0	1	1	1	1	1	1	1	1	0	1	1	1
0	1	0	0	1	1	1	1	1	0	1	1	1	1
0	1	0	1	1	1	1	1	0	1	1	1	1	1
0	1	1	0	1	1	1	0	1	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1	1	1	1
1	0	0	0	1	0	1	1	1	1	1	1	1	1
1	0	0	1	0	1	1	1	1	1	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

Proiectarea cu Microprocesoare

- Circuitul decodificator binar – hexazecimal 74x154:
 - Este un decodificator $4 \rightarrow 16$;
 - /G1 și /G2 sunt intrări de validare;



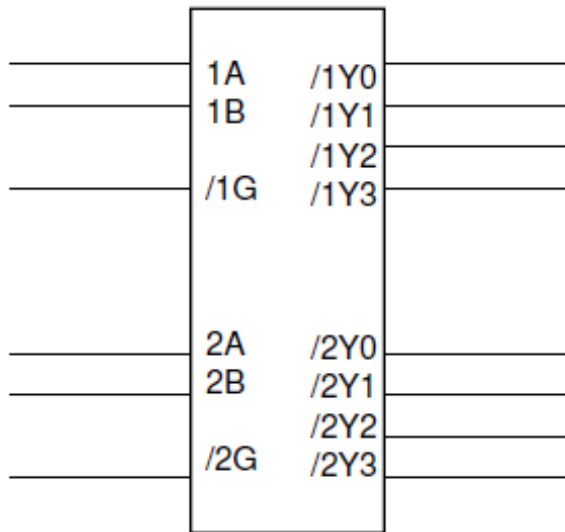
Proiectarea cu Microprocesoare

□ Tabelul de adevăr:

/G 1	/G 2	D	C	B	A	/15	/14	/13	/12	/11	/10	/9	/8	/7	/6	/5	/4	/3	/2	/1	/0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
0	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
0	0	1	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	x	x	x	x	x	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
x	1	x	x	x	x	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Proiectarea cu Microprocesoare

- Circuitul decodificator dual 74x139:
 - Este un decodificator $2 \times 2 \rightarrow 4$;
 - Cele 2 jumătăți sunt independente;
 - Fiecare jumătate are o intrare de validare;



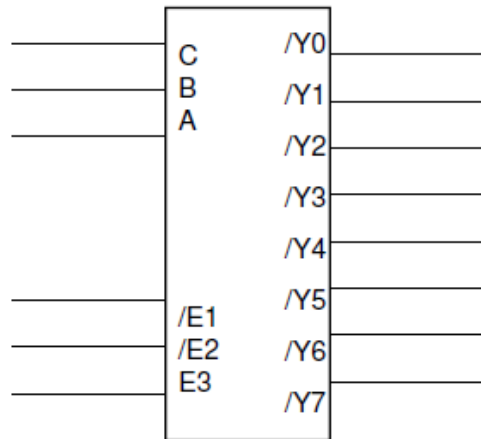
/iG	iB	iA	/iY3	/iY2	/iY1	/iY0
0	0	0	1	1	1	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	1
1	x	x	1	1	1	1

$i = 1, 2$

Proiectarea cu Microprocesoare

□ Circuitul decodificator 74x138:

- Este un decodificator 3 → 8:



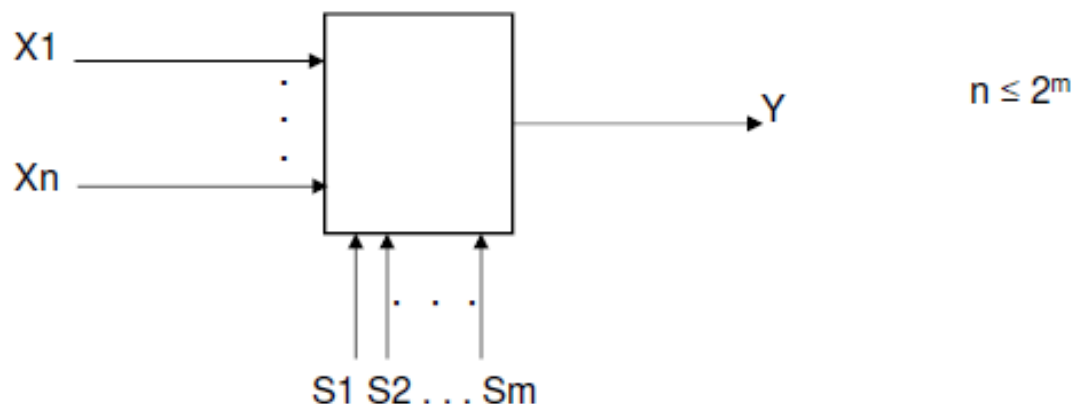
E3	/E2	/E1	C	B	A	/Y7	/Y6	/Y5	/Y4	/Y3	/Y2	/Y1	/Y0
1	0	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	0	0	1	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	0	0	1	1	1	0	1	1	1	1
1	0	0	1	0	1	1	1	0	1	1	1	1	1
1	0	0	1	1	0	1	0	1	1	1	1	1	1
1	0	0	1	1	1	0	1	1	1	1	1	1	1
0	x	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	x	1	1	1	1	1	1	1	1
x	x	1	x	x	x	1	1	1	1	1	1	1	1

- Să se proiecteze un decodificator 4 → 16 utilizând două circuite decodificator 74x138

Proiectarea cu Microprocesoare

■ Multiplexorul (Selectorul)

- Este un comutator digital; el conectează mai multe intrări la o unică ieșire în conformitate cu o anumită ordine dorită;
- Are mai multe intrări de date, mai multe intrări de selecție și o singură ieșire; combinația de pe liniile de selecție alege una dintre intrări care trece la ieșire;
- Schema bloc:

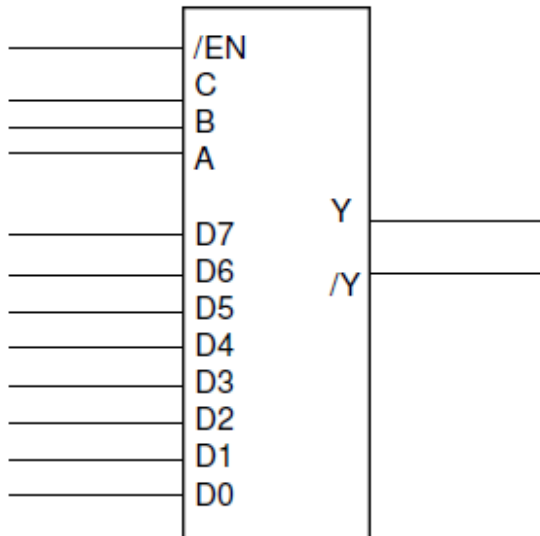


X_1, \dots, X_n sunt intrările de date și S_1, \dots, S_m sunt intrările de selecție

- Ex.: un multiplexor $4 \rightarrow 1$; va avea 4 intrări de date, o ieșire și va necesita 2 intrări de selecție;

Proiectarea cu Microprocesoare

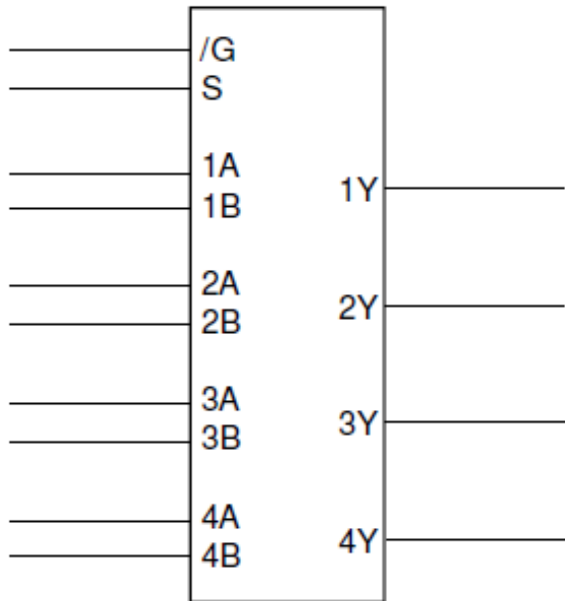
- Circuitul multiplexor 74x151:
 - Este un multiplexor 8 → 1;



/EN	C	B	A	Y
1	x	x	x	0
0	0	0	0	D0
0	0	0	1	D1
0	0	1	0	D2
0	0	1	1	D3
0	1	0	0	D4
0	1	0	1	D5
0	1	1	0	D6
0	1	1	1	D7

Proiectarea cu Microprocesoare

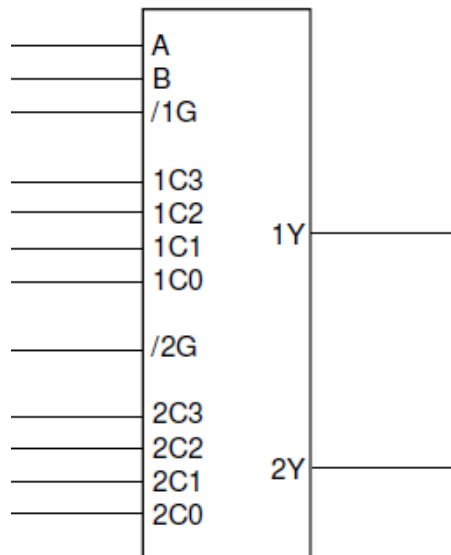
- ❑ Circuitul multiplexor 74x157:
 - Este un multiplexor quadruplu 4 x 2 → 1;
 - Are o singură intrare de validare;



/G	S	1Y	2Y	3Y	4Y
1	x	0	0	0	0
0	0	1A	2A	3A	4A
0	1	1B	2B	3B	4B

Proiectarea cu Microprocesoare

- Circuitul multiplexor 74x153:
 - Este un multiplexor dual 2 x 4 → 1;
 - Are o intrare de validare pentru fiecare secțiune;

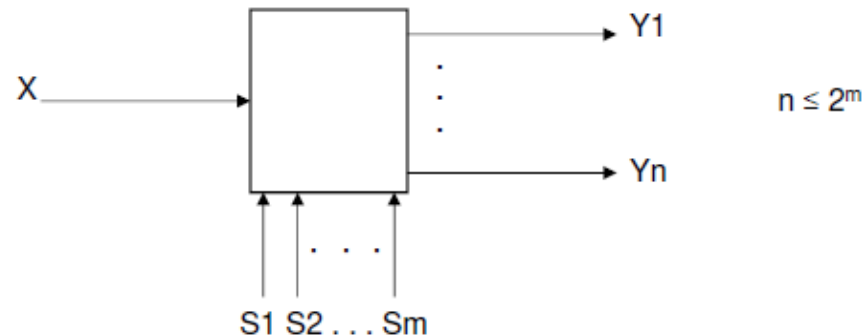


/1G	/2G	B	A	1Y	2Y
0	0	0	0	1C0	2C0
0	0	0	1	1C1	2C1
0	0	1	0	1C2	2C2
0	0	1	1	1C3	2C3
0	1	0	0	1C0	0
0	1	0	1	1C1	0
0	1	1	0	1C2	0
0	1	1	1	1C3	0
1	0	0	0	0	2C0
1	0	0	1	0	2C1
1	0	1	0	0	2C2
1	0	1	1	0	2C3
1	1	x	x	0	0

Proiectarea cu Microprocesoare

■ Demultiplexorul (Distribuatorul)

- Demultiplexorul este un comutator digital; el conectează o unică intrare la mai multe ieșiri în conformitate cu o anumită ordine dorită;
- Are o intrare de date, mai multe intrări de selecție și mai multe ieșiri; combinația de pe liniile de selecție alege ieșirea la care trece unica intrare;
- Schema bloc:



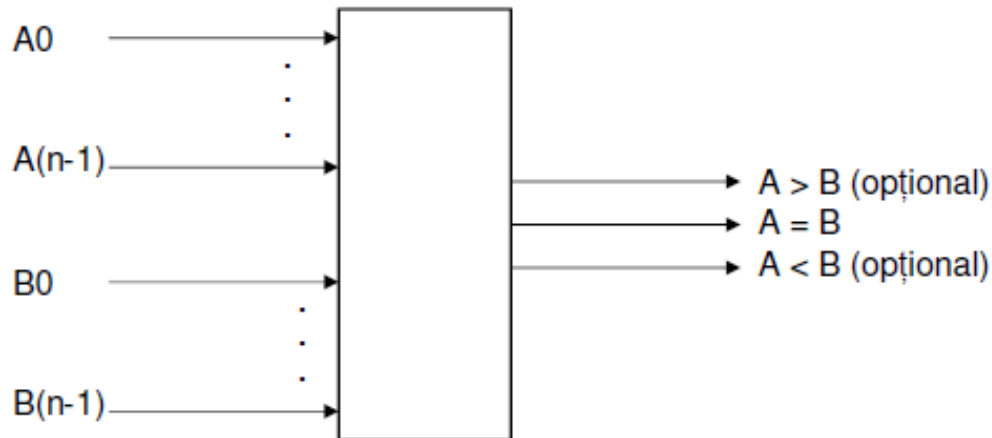
X este intrarea de date, Y_1, \dots, Y_n sunt ieșirile și S_1, \dots, S_m sunt intrările de selecție

- Ex.: un demultiplexor $1 \rightarrow 4$; va avea 1 intrare de date, 4 ieșiri și va necesita 2 intrări de selecție;
- Expandarea multiplexoarelor și demultiplexoarelor;

Proiectarea cu Microprocesoare

■ Comparatorul

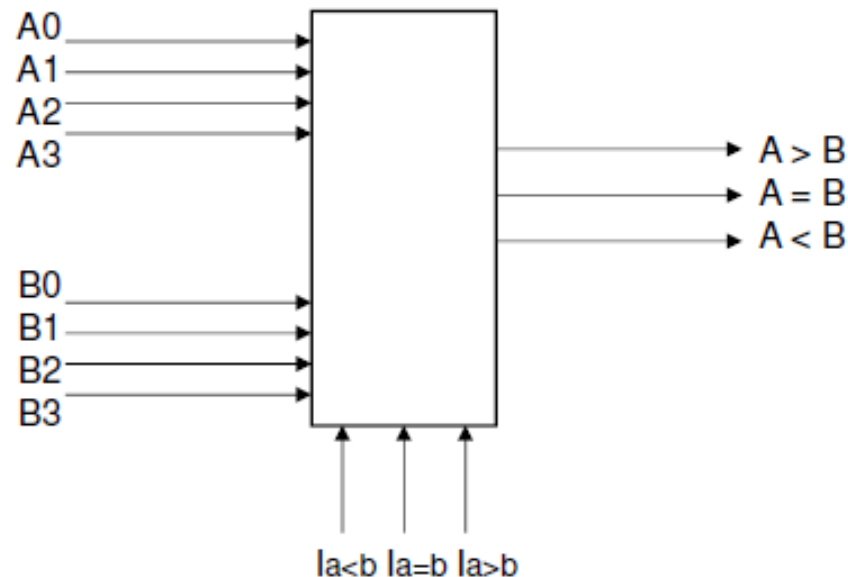
- Este circuitul combinațional care determină raportul existent între 2 numere binare, A și B, și indică una din următoarele situații: $A = B$, $A > B$ sau $A < B$;
- Este util în aplicații aritmetice sau de scanare a unei anumite configurații în un șir sau bloc;
- Schema bloc:



- Ex.: un comparator pe 4 ranguri care să indice egalitatea a 2 combinații de la intrările A0 – A3 și B0 – B3;
- Ex.: un comparator complet pe 4 ranguri; va avea ieșirile EGAL, A > B și A < B;

Proiectarea cu Microprocesoare

- ❑ Circuite integrate comparator:
 - Majoritatea au posibilitatea de conectare în cascadă astfel încât să se poată realiza comparatoare cu un număr mai mare de intrări decât numărul de intrări al unui circuit comparator;
 - Un asemenea circuit va avea intrări de comparat, intrări de cascaderare care permit conectarea circuitului în un lanț de comparatoare și ieșiri;
- ❑ Circuitul comparator 74x85
 - Este un comparator pe 4 ranguri, cu 3 intrări de cascaderare și 3 ieșiri;
 - Schema bloc:

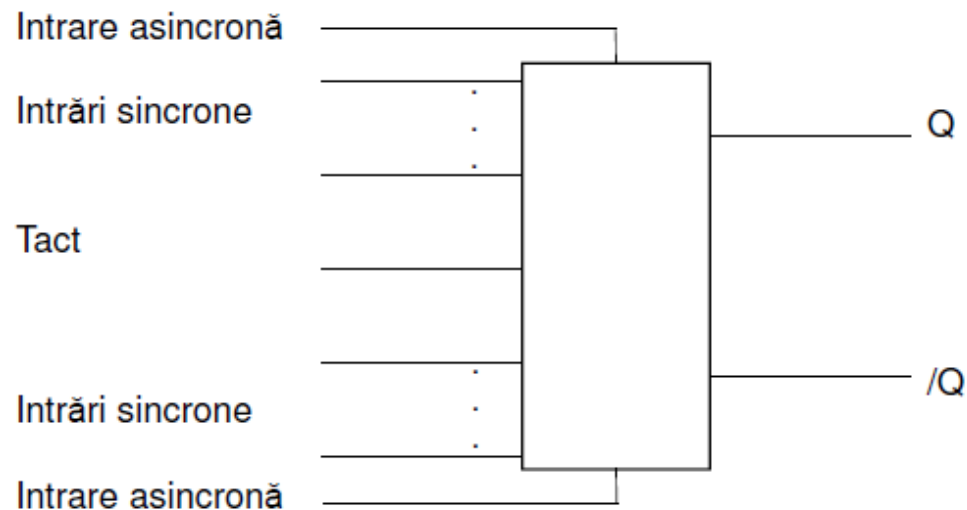


Proiectarea cu Microprocesoare

1.1.2 Circuite secvențiale

■ Bistabilul

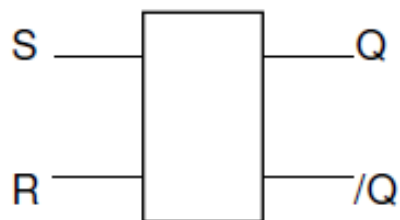
- Circuitul secvențial elementar;
- Simbolul general:



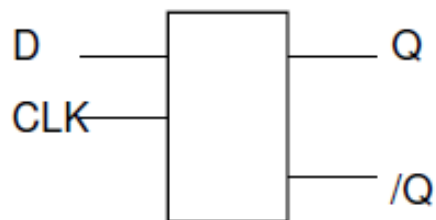
- Tipuri de bistabile:
 - Asincrone: RS;
 - Sincrone: RS, JK, JK-MS, D, T;

Proiectarea cu Microprocesoare

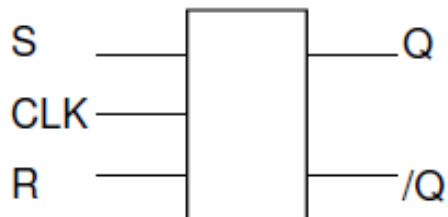
Simboluri clasice:



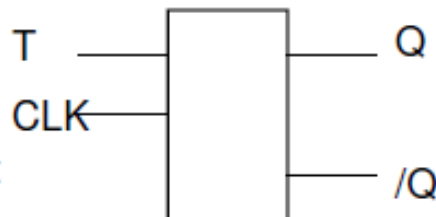
Bistabil RS asincron



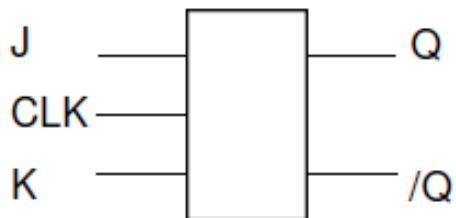
Bistabil D sincron
cu basculare pe nivel
1 al impulsului de tact



Bistabil RS sincron
cu basculare pe nivel
1 al impulsului de tact



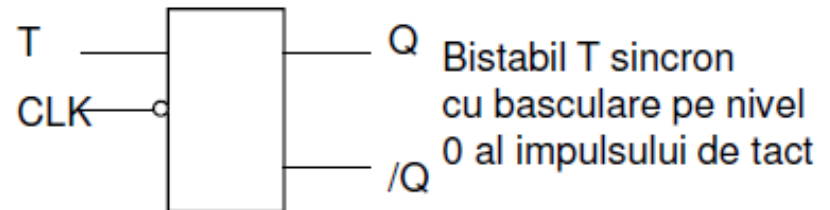
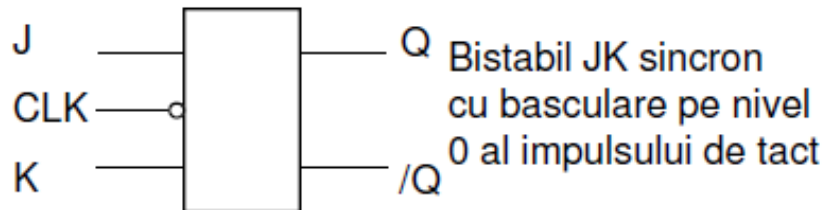
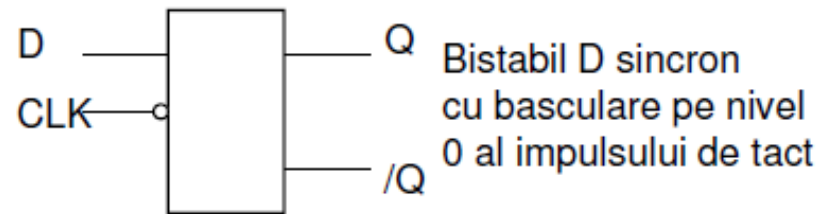
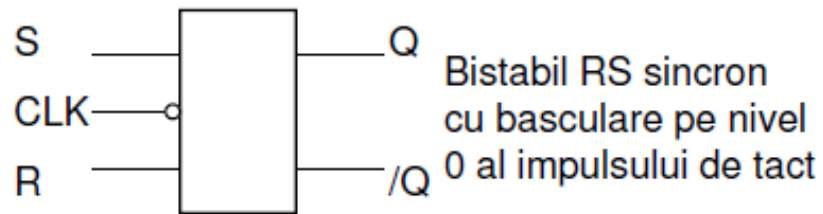
Bistabil T sincron
cu basculare pe nivel
1 al impulsului de tact



Bistabil JK sincron
cu basculare pe nivel
1 al impulsului de tact

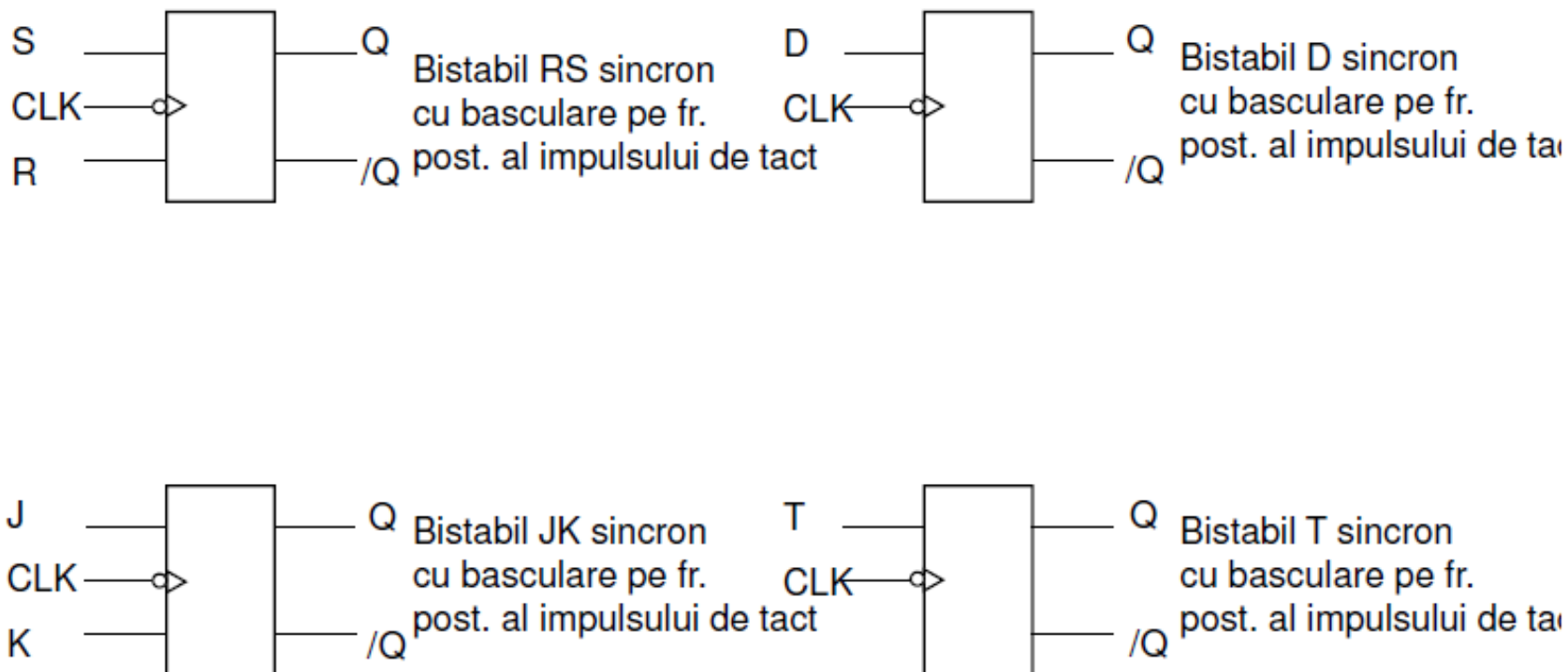
Proiectarea cu Microprocesoare

□ Simboluri:



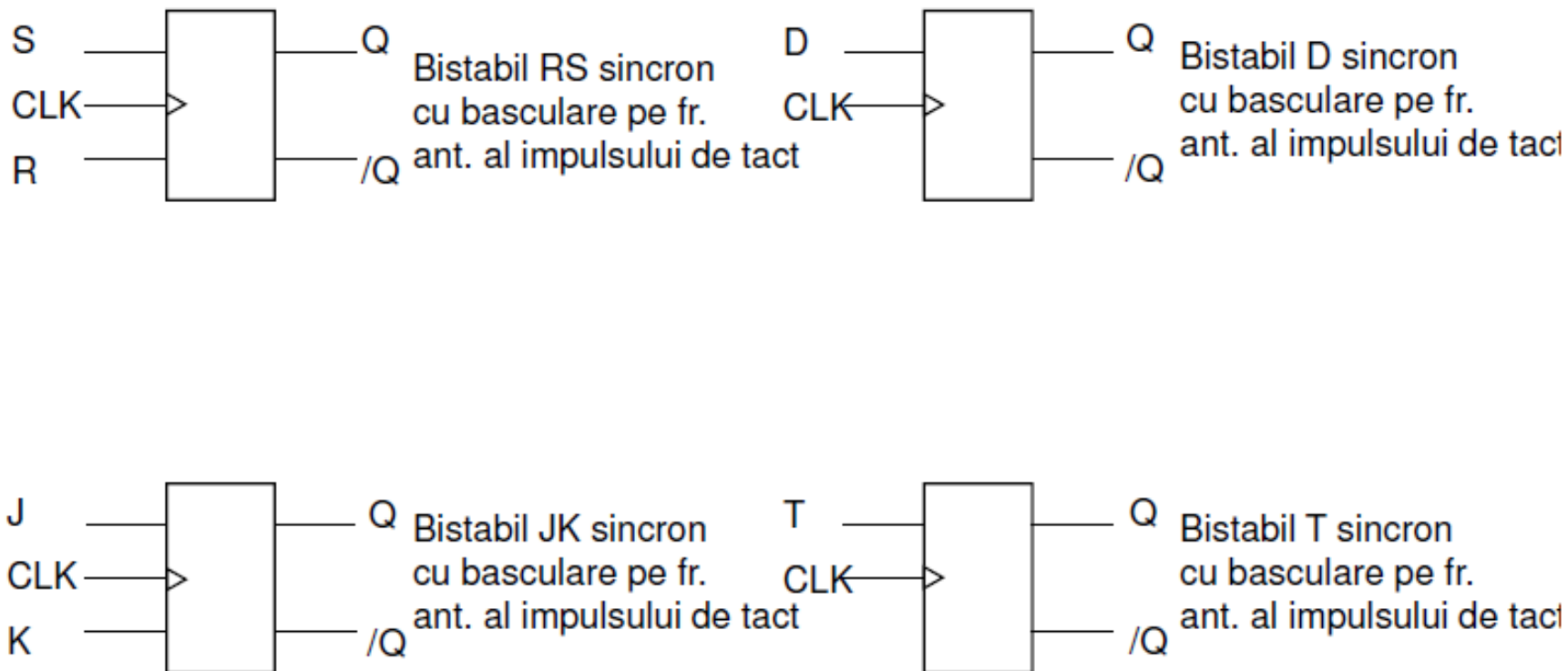
Proiectarea cu Microprocesoare

□ Simboluri:



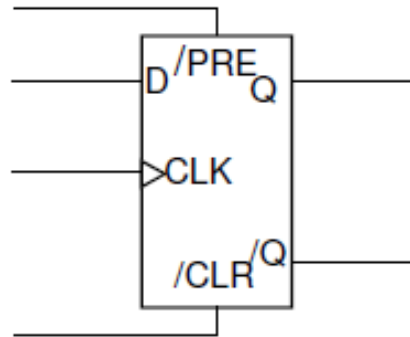
Proiectarea cu Microprocesoare

□ Simboluri:

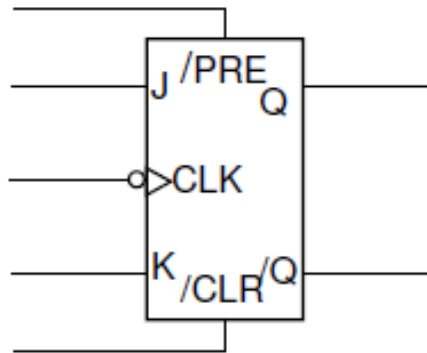


Proiectarea cu Microprocesoare

- Aplicații ale bistabilelor: divizarea frecvenței unui semnal, sincronizarea;
- Circuite integrate care conțin bistabile:
 - 74x74: conține 2 bistabile D, active pe frontul anterior al impulsului de tact; simbolul unui bistabil este:



- 74x76: conține 2 bistabile JK, active pe frontul posterior al impulsului de tact; simbolul unui bistabil este:

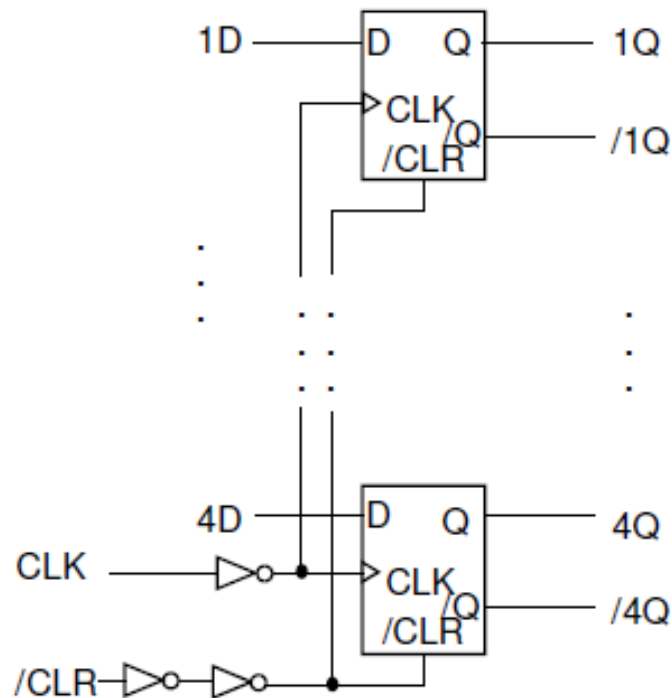


Proiectarea cu Microprocesoare

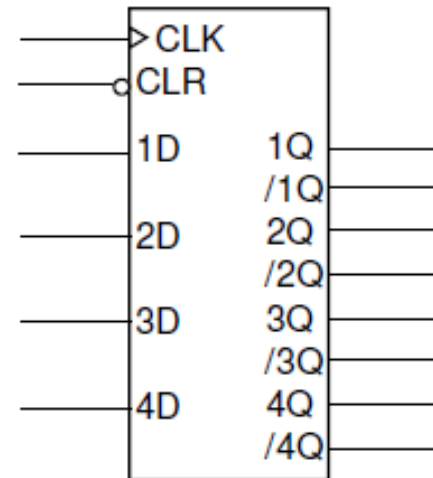
■ Registrul

- Ansamblu ordonat de bistabile și circuite de comandă;
- Circuite integrate care conțin bistabile:
 - 74x175: conține 4 bistabile de tip D, cu tact comun, activ pe frontul anterior și cu intrare de ștergere:

Schema:



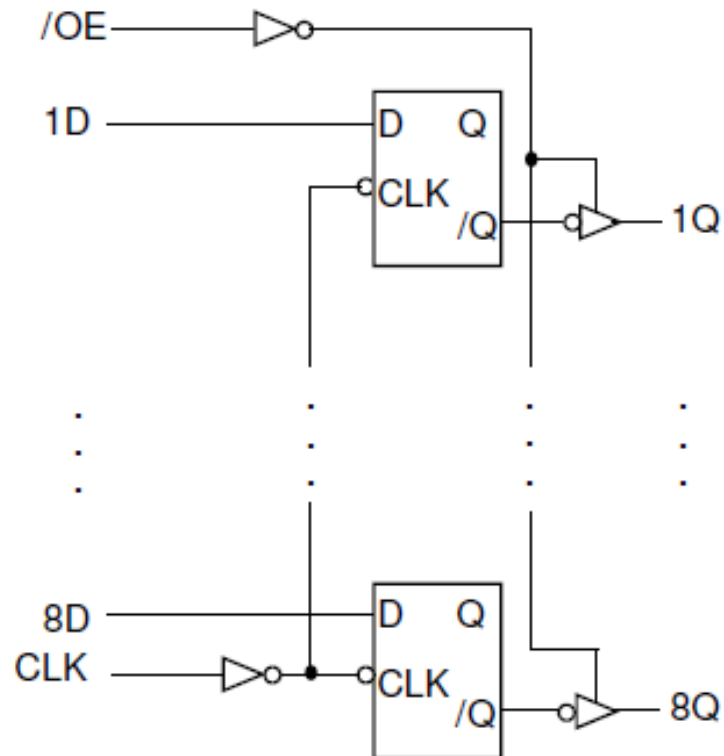
Simbolul:



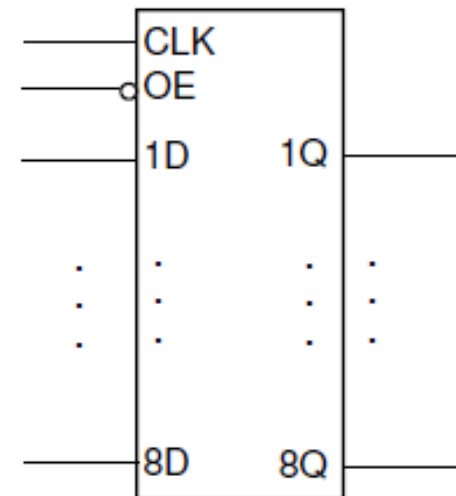
Proiectarea cu Microprocesoare

- 74x373: conține 8 bistabile de tip D, cu tact comun, activ pe nivel 1 și cu ieșiri cu 3 stări:

Schema

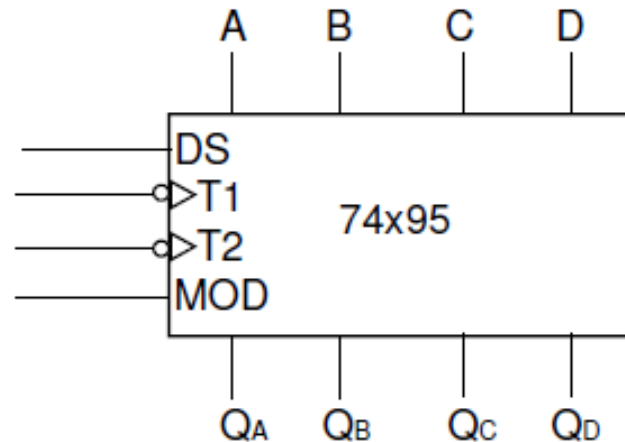


Simbolul:



Proiectarea cu Microprocesoare

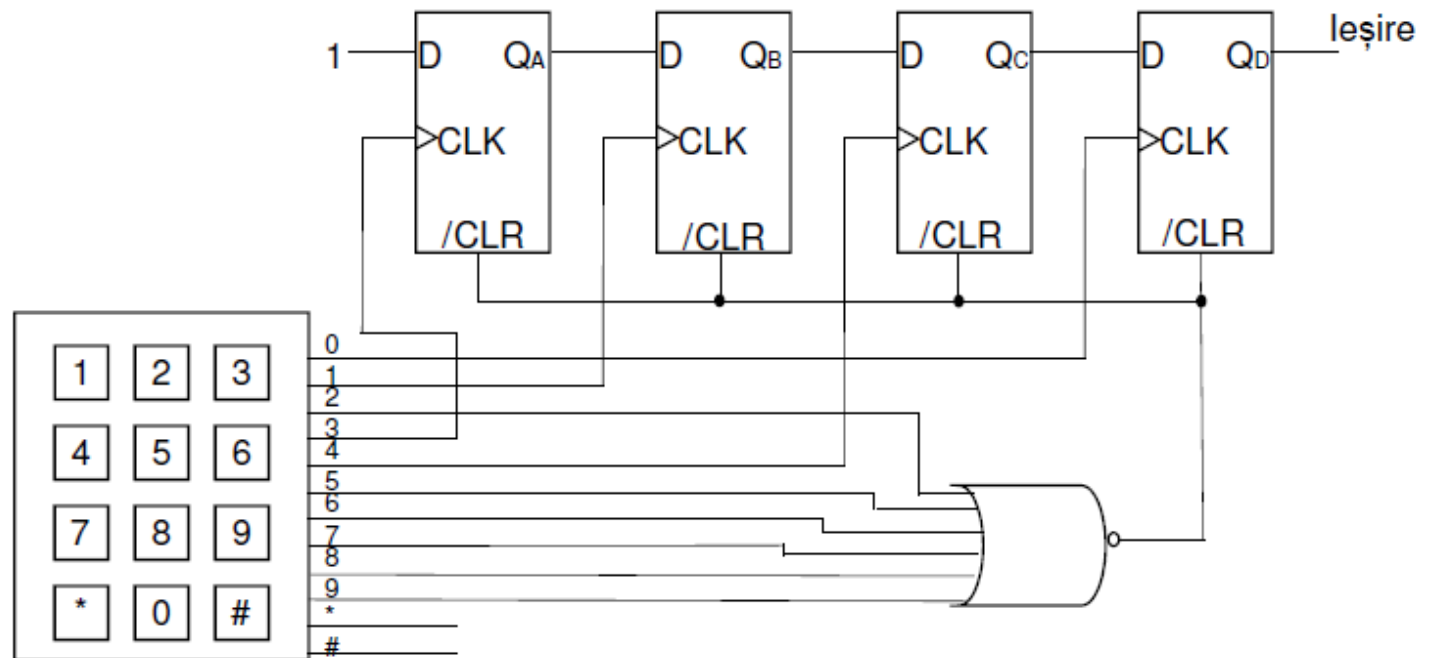
- ❑ Registre de deplasare
- ❑ Operații: deplasare la stânga, la dreapta, rotire la stânga, la dreapta, încărcare paralelă; registrul universal asigură toate operațiile fără conexiuni externe;
- ❑ Circuitul integrat registru 74x95:
 - Este un registru pe 4 ranguri, de tip paralel in – paralel out, și cu posibilitatea de a deplasa informația la dreapta;
 - Simbolul:



- A – D sunt intrările paralele, $Q_A - Q_D$ sunt ieșirile paralele, DS este intrarea serială, intrarea MOD alege între deplasare la dreapta, MOD = 0, și încărcare paralelă, MOD = 1, iar T1 și T2 sunt intrări pentru impulsuri de tact: impulsurile pe T1 provoacă deplasarea la dreapta iar cele de pe T2 provoacă încărcarea paralelă a informației; impulsurile de pe intrările T1 și T2 sunt active pe frontul posterior;

Proiectarea cu Microprocesoare

- Aplicații ale registrelor: deplasarea și rotirea informației, implementarea înmulțirii și împărțirii;
- Detector de cod
 - Este o aplicație a unui registru de tip serial in – serial out;
 - Circuitul detectează un anumit cod, format secvențial, acționând asupra unei minitastaturi și, la apariția codului, va genera un impuls care va putea fi folosit, de exemplu, la deschiderea unei uși; schema:



Proiectarea cu Microprocesoare

- Pentru a se activa ieșirea este necesară introducerea codului în succesiunea indicată de conexiuni; codul corespunzător schemei este 3140;
- La fiecare acționare a unei taste, se generează un impuls la ieșirea corespunzătoare a tastaturii;
- Se observă că dacă se acționează una din tastele 2, 5, 6, 7, 8 sau 9, în orice ordine, registrul va fi resetat (anulat);
- Dacă se acționează tasta 3, va apare 1 logic la ieșirea QA ;
- Dacă urmează acționarea tastei 1, 1 logic de la intrarea bistabilului B se va transmite la ieșirea sa, QB ;
- Dacă urmează acționarea tastei 4, 1 logic de la intrarea bistabilului C se va transmite la ieșirea sa, QC ;
- La final, dacă urmează acționarea tastei 1, 1 logic de la intrarea bistabilului D se va transmite la ieșirea sa, QD care este și ieșirea întregului circuit; acest impuls va putea comanda o yală electromagnetică sau/și va putea fi monitorizat de sistemul de securitate;
- Acționarea oricărei alte taste numerice, oricând, va anula registrul ca urmare este necesară reintroducerea întregului cod corect;
- Dezactivarea ieșirii poate fi făcută oricând, prin acționarea unei taste care nu corespunde codului;
- Schemă simplă dar nu permite modificarea codului.

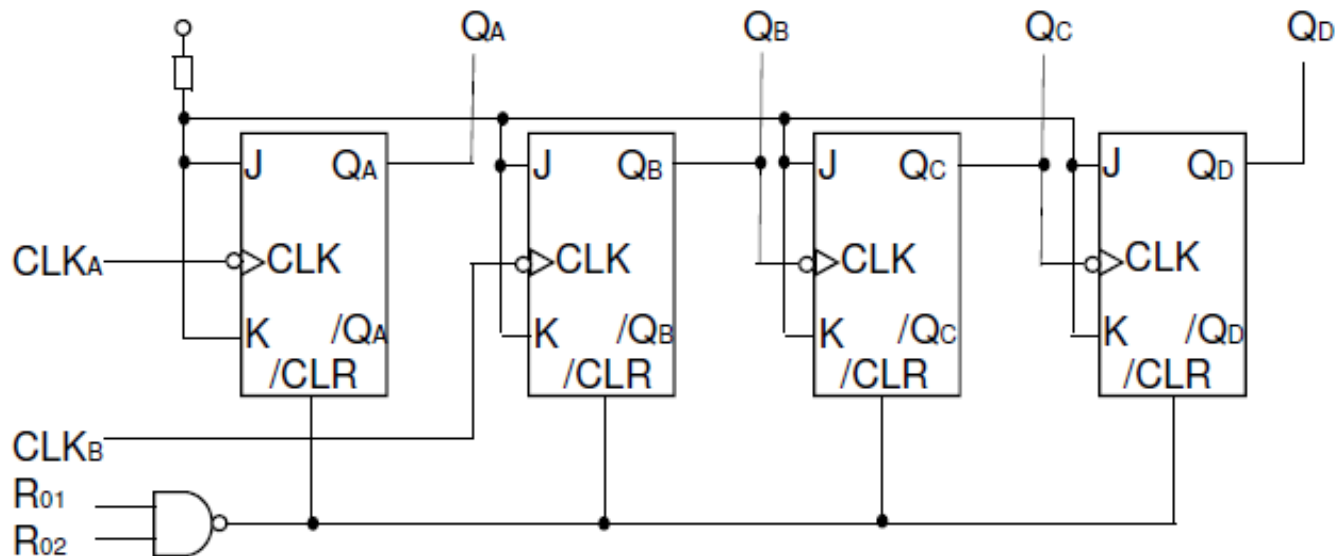
Proiectarea cu Microprocesoare

■ Numărătorul

- ❑ Constituie o clasă particulară a circuitelor secvențiale fără intrări, ale căror modele matematice se numesc automate finite autonome;
- ❑ Tranzițiile acestor circuite se fac la comanda impulsului de tact după o anumită lege, numai pe baza stării prezente a acestora;
- ❑ Modul de funcționare al numărătorului este complet specificat prin secvența de numărare care reprezintă succesiunea de stări binare a acestuia;
- ❑ 2 roluri: numărarea unei succesiuni de impulsuri, divizarea frecvenței unui semnal;
- ❑ Numărătoarele se clasifică după următoarele criterii:
 - Felul de codificare a informației: decadic, octal, binar, modulo n etc.; prin acest criteriu distincția se face în funcție de numărul de stări al numărătorului;
 - Sincronismul bistabilelor: sincrone și asincrone;
 - Sensul numărării: directe, inverse și reversibile;
- ❑ Orice numărător pornește ciclul de numărare dintr-o stare inițială; aceasta poate fi 0 pe toate bistabilele, 1 pe toate bistabilele sau orice altă valoare; pentru încărcarea stării inițiale există circuite combinaționale dedicate.

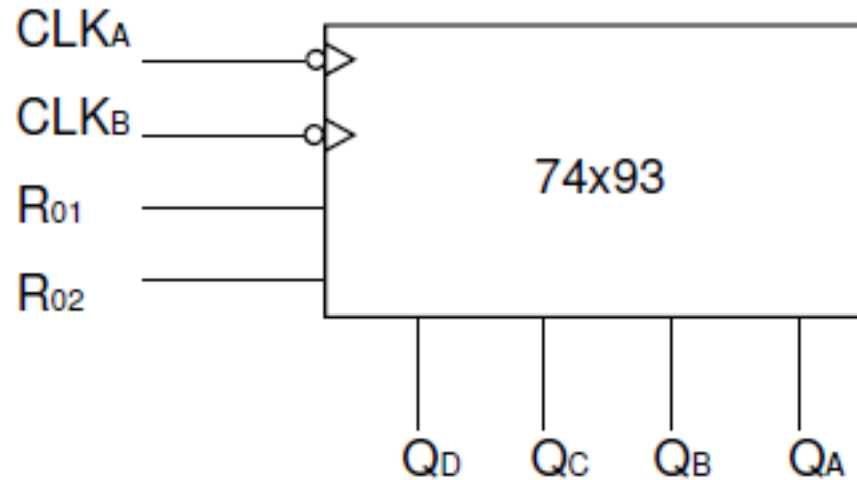
Proiectarea cu Microprocesoare

- ❑ Numărătoare asincrone: se obțin înlănțuind mai multe bistabile, ieșirea unuia fiind conectată la intrarea de tact a următorului;
 - Avantaje: schema simplă, ușor scalabil;
 - Dezavantaje: întârziere variabilă cu numărul bistabilelor, întârziere mare care duce la limitarea frecvenței tactului;
- ❑ Circuitul integrat numărător asincron 74x93
 - Este un numărător asincron care poate lucra ca numărător mod 16 sau ca numărător mod 8 și numărător mod 2; schema internă este:



Proiectarea cu Microprocesoare

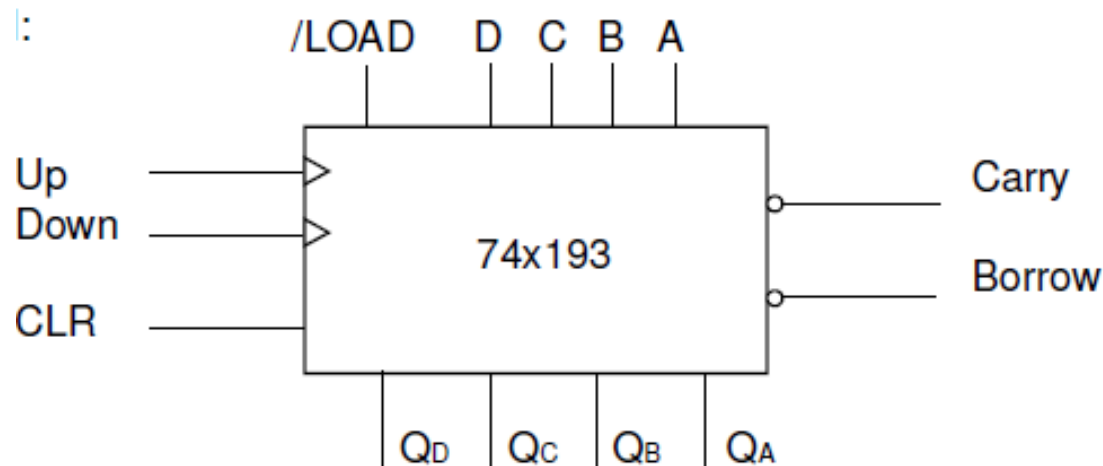
- Simbolul circuitului este:



- Se observă că numărătorul trece din o stare în alta pe frontul posterior al impulsului (sau impulsurilor) de tact;
- Dacă la intrările R01 și R02 există simultan 1 logic, numărătorul va fi forțat în starea inițială 0000; dacă pe cel puțin una dintre ele este 0 logic, numărătorul va parcurge secvența proprie de stări;
- Dacă impulsul de tact se leagă la intrarea CLKA, la ieșirea QA se obține un semnal cu frecvența pe jumătate față de cea a semnalului de intrare (numărător mod 2);
- Dacă impulsul de tact se leagă la intrarea CLKB, bistabilele B, C și D vor funcționa ca un numărător mod 8;

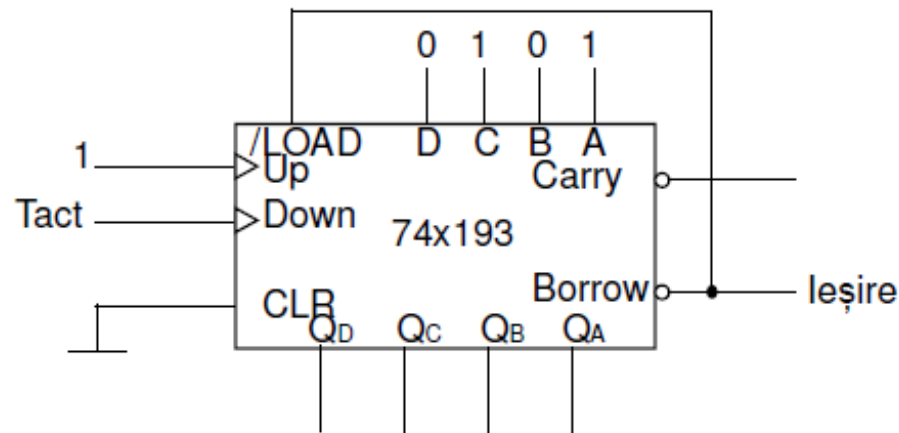
Proiectarea cu Microprocesoare

- ❑ Numărătoare sincrone: sunt numărătoare la care impulsul de tact este aplicat simultan la toate bistabilele, ca urmare acestea se modifică simultan, în conformitate cu tabelul stărilor;
 - Avantaje: întârziere fixă dată de întârzierea pe un rang;
 - Dezavantaje: schemă mai complicată;
- ❑ Circuitul integrat 74x193
 - Este un numărător binar, sincron, mod 16, reversibil și cu preîncărcare;
 - Simbolul:



Proiectarea cu Microprocesoare

- Dacă se aduce 1 la intrarea CLR, circuitul va fi șters, adică va avea 0 în toate bistabilele;
- Dacă se aduce 0 la intrarea /LOAD, circuitul va fi încărcat cu combinația de la intrările A, B, C, D;
- Dacă se dorește ca circuitul să numere, trebuie adus impuls de tact la intrarea Up sau Down, funcție de direcția de numărare dorită; intrarea de tact nefolosită trebuie legată la 1 logic;
- Circuitul generează:
 - Un impuls la ieșirea Carry, activ la 0 logic, atunci când apare depășirea capacității sale adică la trecerea din starea 1111 în starea 0000 când numără în sus;
 - Un impuls la ieșirea Borrow, activ la 0 logic, atunci când apare trecerea din starea 0000 în starea 1111 când numărătorul numără în jos;
- Datorită existenței ieșirilor Carry și Borrow și a intrării /LOAD, se pot obține scheme mai simple pentru divizoare cu diferite constante; ex.: divizor la 5:



Proiectarea cu Microprocesoare

1.1.3 Proiectarea automatelor finite

- Majoritatea aplicațiilor cer o parte de execuție și una de comandă;
- Partea de comandă este aceea care asigură execuția secvențială a aplicației, ia decizii în funcție de intrări și generează semnale de comandă către blocurile de execuție sau, eventual, către alte părți de comandă;
- Partea de comandă poate fi implementată cu un automat finit;
- Pentru proiectarea unui automat finit se parcurg pașii:
 - Se specifică, în cuvinte, comportarea circuitului; se consideră printr-o succesiune de stări; o stare este definită prin operațiile pe care le execută automatul la un moment dat; operațiile simultane se execută în aceeași stare; într-o stare automatul poate să ia o decizie, pe baza unei intrări, să genereze un semnal de ieșire sau să treacă într-o altă stare; trecerea dintr-o stare în alta se face la momentele stabilite de un semnal de tact;
 - Se caută ca numărul de stări să fie minim;

Proiectarea cu Microprocesoare

- ❑ Se stabilește modul de reprezentare al înlănțuirii stărilor: cu tabele sau cu diagrame (în continuare se va folosi varianta cu diagrame);
- ❑ Se alege un set de variabile de stare și se asignează variabilele stărilor automatului; numărul variabilelor depinde de numărul stărilor conform relației:
$$2^{\text{numărul variabilelor}-1} \leq \text{numărul stărilor} \leq 2^{\text{numărul variabilelor}}$$
- ❑ Se alege tipul de bistabil; bistabilele vor fi suportul fizic pentru stările automatului; numărul de bistabile este egal cu numărul de variabile;
- ❑ Se codifică variabilele de stare;
- ❑ Se verifică dacă nu apare hazard (se reamintește că hazardul apare dacă la tranziția din o stare în alta se modifică mai mult de o variabilă de stare); dacă apare hazard și dacă acesta deranjează trebuie eliminat prin introducerea de stări suplimentare; ca urmare automatul nu va mai avea număr minim de stări și este posibil ca și numărul de bistabile să crească dar este eliminat hazardul (atunci când este necesar);
- ❑ Se construiesc tabelele de excitație pentru secvența de stări obținută;
- ❑ Se obțin ecuațiile intrărilor bistabilelor pornind de la tabelele de excitație și minimizându-le;
- ❑ Similar se obțin și ecuațiile ieșirilor;
- ❑ Se desenează schema

Proiectarea cu Microprocesoare

- Avantaje și dezavantaje ale proiectării cu automate finite
 - Avantaje: este o metodă generală;
 - Dezavantaje:
 - Soluția poate fi lentă: tranzițiile automatului se fac sub controlul unui tact extern și frecvența acestuia va limita viteza automatului;
 - Dacă numărul de stări este mare, schema va fi complexă;
 - Alternativa: utilizarea microcontrolerelor.