



Universitatea  
Transilvania  
din Brașov

UNIVERSITATEA TRANSILVANIA BRAȘOV

Facultatea de Inginerie Electrică și Știința Calculatoarelor

Automatică și Tehnologia Informației

Rodina Valentin

Tudor Georgian Iulian

# Proiect ASCN II

Îndrumator

Prof. dr. Ing. MOLDOVEANU Florin

2021

## FIȘA PROIECTULUI DE DISCIPLINĂ

### Disciplina:

Analiza și sinteza circuitelor numerice II, an III/sem. 5.

### Tema de proiect nr. 28:

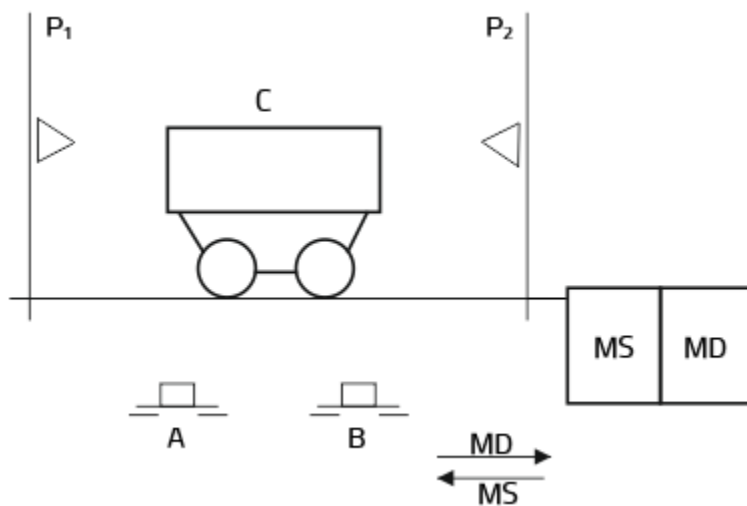
Să se proiecteze un automat secvențial asincron al cărui caiet de sarcini este descris mai jos.

În proiectarea automatului secvențial asincron se va folosi metoda organigramei.

### Caiet de sarcini:

Se consideră instalația tehnologică din figura de mai jos, care conține următoarele elemente componente:

- MS și MD sunt două motoare electrice alimentate prin intermediul unor relee notate în același mod. Atunci când motorul MS este alimentat, căruciorul C se deplasează de la dreapta spre stânga, iar când motorul MD este alimentat, căruciorul C se deplasează de la stânga spre dreapta;
- $P_1$  și  $P_2$  sunt două limitatoare de cap de cursă;
- A și B sunt două butoane.



Se va avea în vedere proiectarea unei structuri de comandă care să opereze după următorul protocol:

- dacă se apasă pe butonul A de pornire, atunci când căruciorul este oprit în poziția  $P_1$ , acesta părăsește poziția  $P_1$ , ajunge în  $P_2$  și revine în  $P_1$ , unde se oprește;
- dacă se apasă pe butonul B, atunci când căruciorul se deplasează între  $P_1$  și  $P_2$  (de la  $P_1$  către  $P_2$ ), căruciorul C va efectua, la sfârșitul ciclului  $P_1 \rightarrow P_2$

->  $P_1$ , un nou ciclu identic (fără a se acționa asupra lui A);

- în absența acțiunii asupra butonului B sau dacă aceasta are loc în timpul parcurgerii  $P_2 \rightarrow P_1$ , ciclul inițial nu este repetat (căruciorul se oprește în poziția  $P_1$ , la sfârșitul ciclului);
- orice acțiune asupra butonului A în timpul parcurgerii traseului nu are nicio influență asupra desfășurării acestuia;

- butonul A permite deci pornirea ciclului, iar acționarea asupra lui B (în intervalul  $P_1 \rightarrow P_2$ ) permite să fie repetat ciclul (de fiecare dată când se dorește).

### Cerințe de proiectare:

În rezolvarea temei proiectului se vor trata următoarele probleme:

- Transpunerea caietului de sarcini al automatului într-o diagramă a stărilor.
- Întocmirea organigramei funcționale a automatului secvențial descris mai sus.
- Stabilirea diagramei stărilor la momentul "t".
- Întocmirea diagramelor stărilor următoare (de la momentul "t+1").
- Obținerea expresiilor funcțiilor de excitație pentru automatele elementare ce compun registrul de stare.
- Obținerea expresiilor funcțiilor de ieșire ale automatului.
- Implementarea funcțiilor logice obținute cu circuite integrate (se vor utiliza bistabili de tip D pentru implementarea registrului de stare (secțiunii de memorie) și porți logice pentru structura logică combinațională).
- Analiza schemei logice obținute.

Pe schema logică obținută se vor specifica tipul și gradul de utilizare al fiecărui circuit integrat.

### Bibliografia recomandată:

- [1] Ștefan, Gh. – *Circuite și sisteme digitale*, Ed. Tehnică, București, 2000.
- [2] Wakerly, J.F. – *Circuite digitale*, Ed. Teora, București, 2002.
- [3] Wilkinson, B. – *Electronică digitală. Bazele proiectării*, Ed. Teora, București, 2002.
- [4] Mano, M.M. – *Digital Design*, Prentice Hall International, London, 2002.
- [5] Moldoveanu, F., Floroian, D. – *Circuite logice și comenzi secvențiale. Circuite logice combinaționale*, Ed. Universității Transilvania din Brașov, 2003.
- [6] Toacșe, Gh., Nicula, D. – *Electronică digitală*, vol. I, Ed. Tehnică, București, 2005.

### Condiții de redactare:

Pentru redactare se va folosi template-ul recomandat de cadrul didactic îndrumător.

### Evaluări pe parcurs:

S-a stabilit o viză pentru evaluarea pe parcurs a proiectului, în data de ???.2020.

**Termenul de predare și susținere:**

Proiectul se va preda și susține în ultima săptămână a semestrului.

**Notarea proiectului:**

Forma finală a proiectului trebuie să conțină rezolvări pentru toate punctele cerute prin temă iar, din punct de vedere al redactării, să aibă forma solicitată. În cursul susținerii, studentul trebuie să dovedească cunoașterea metodelor specifice de rezolvare pentru problemele date, utilizarea corectă și fluentă a termenilor specifici și interpretarea corectă a rezultatelor. Notarea va porni de la nota 10, dacă studentul a primit viza de evaluare și de la nota 7 în cazul în care studentul nu are această viză.

Octombrie 2020

Titular activități de proiect,  
Prof. dr. ing. Florin Dumitru MOLDOVEANU

## Cuprins

1. Transpunerea caietului de sarcini într-o diagrama a stărilor .....	6
1.1 Noțiuni introductive .....	6
1.2 Diagrama stărilor .....	7
1.3 Codificarea stărilor .....	8
2. Intocmirea organigramei funcționale a automatului secvențial descris mai sus .....	9
3. Stabilirea diagramei stărilor la momentul "t" .....	11
4. Intocmirea diagramei stărilor următoare .....	13
5. Obținerea funcțiilor de excitație pentru automatele elementare ce compun registrul de stare .....	15
6. Obținerea funcțiilor de ieșire ale automatului .....	18
7. Implementarea funcțiilor logice obținute cu circuite integrate .....	19
8. Analiza schemei logice obținute .....	21
9. Bibliografie .....	23

# 1. Transpunerea caietului de sarcini într-o diagrama a stărilor

## 1.1 Noțiuni introductive

Un circuit logic secvențial (CLS) este un circuit de comutare în cazul căruia starea externă (ieșirea) la un moment dat depinde nu numai de starea intrărilor la momentul de timp considerat, ci și de stările anterioare ale acestora. Din acest motiv, CLS trebuie să aibă memorie în care trebuie păstrată informația referitoare la evoluția lui anterioară.

Definirea CLS se bazează pe introducerea conceptului de stare internă. Informația păstrată în memorie și pe baza căreia poate fi descrisă complet evoluția anterioară a CLS se numește stare internă a acestuia.

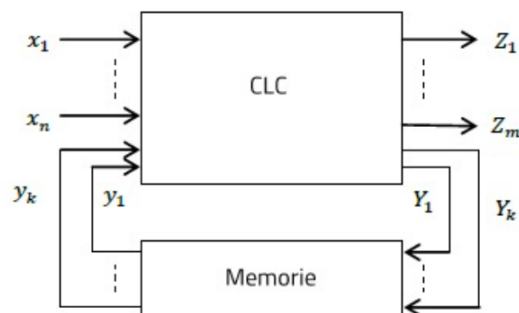


Fig 1.1.1 Schema bloc generală al unui circuit logic secvențial

Un CLS este un circuit de prelucrare a informației discrete în care se pun în evidență următoarele seturi de mărimi: un set sau o mulțime a variabilelor de intrare (primare/principale)  $X = (x_1, x_2, \dots, x_n)$ , un set al variabilelor de ieșire  $Z = (z_1, z_2, \dots, z_m)$  și un set al variabilelor de stare  $Y = (y_1, y_2, \dots, y_k)$ .

## 1.2 Diagrama stărilor

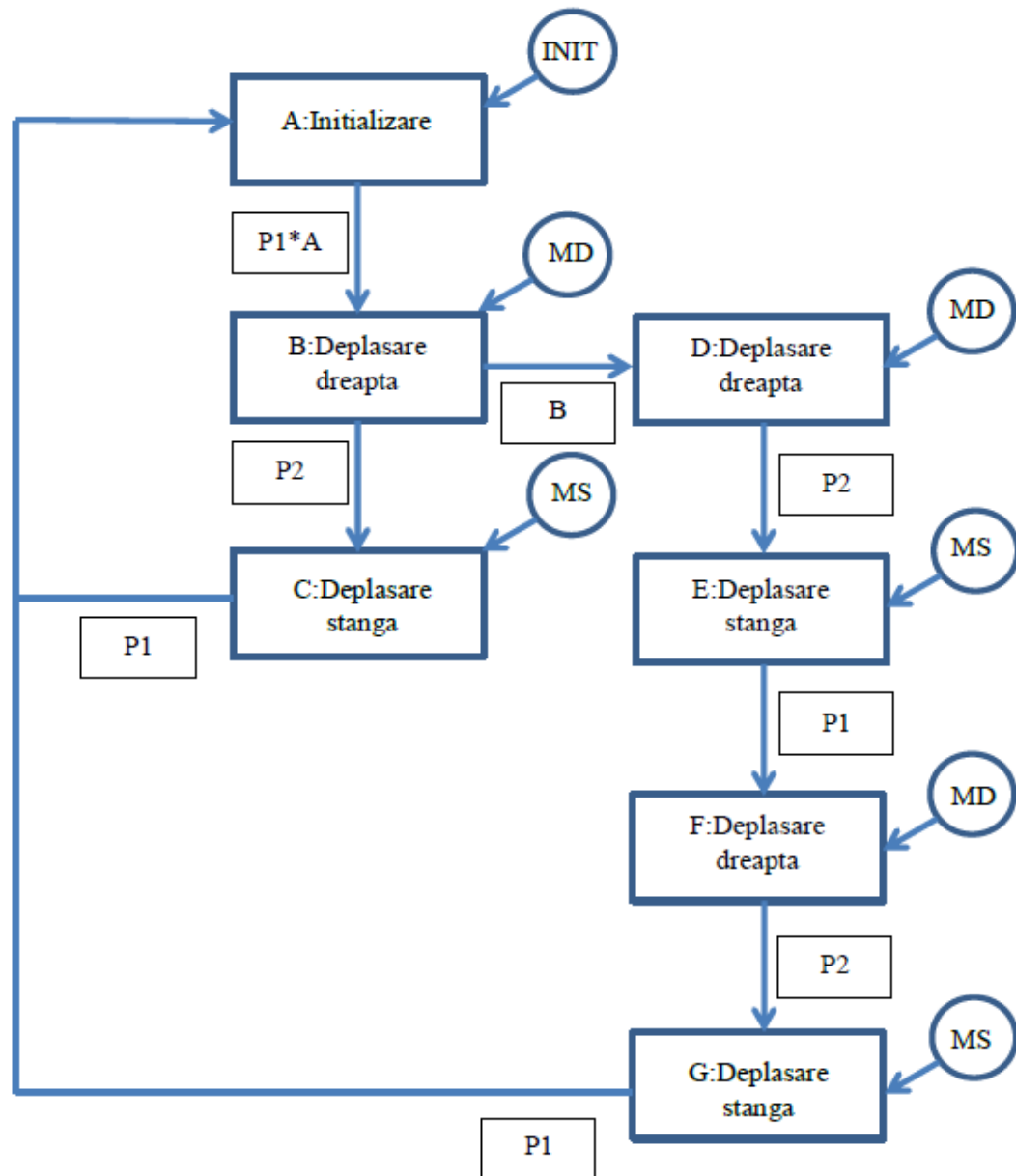


Fig. 1.2.1 Diagrama stărilor

### 1.3 Codificarea stărilor

Având în vedere faptul că problema codificării stărilor circuitelor secvențiale nu are soluție unică și nu există o metodă care să garanteze o soluție optimă, am ales să utilizăm 4 variabile de stare și să introducem stări suplimentare pentru a putea codifica adiacente stările.

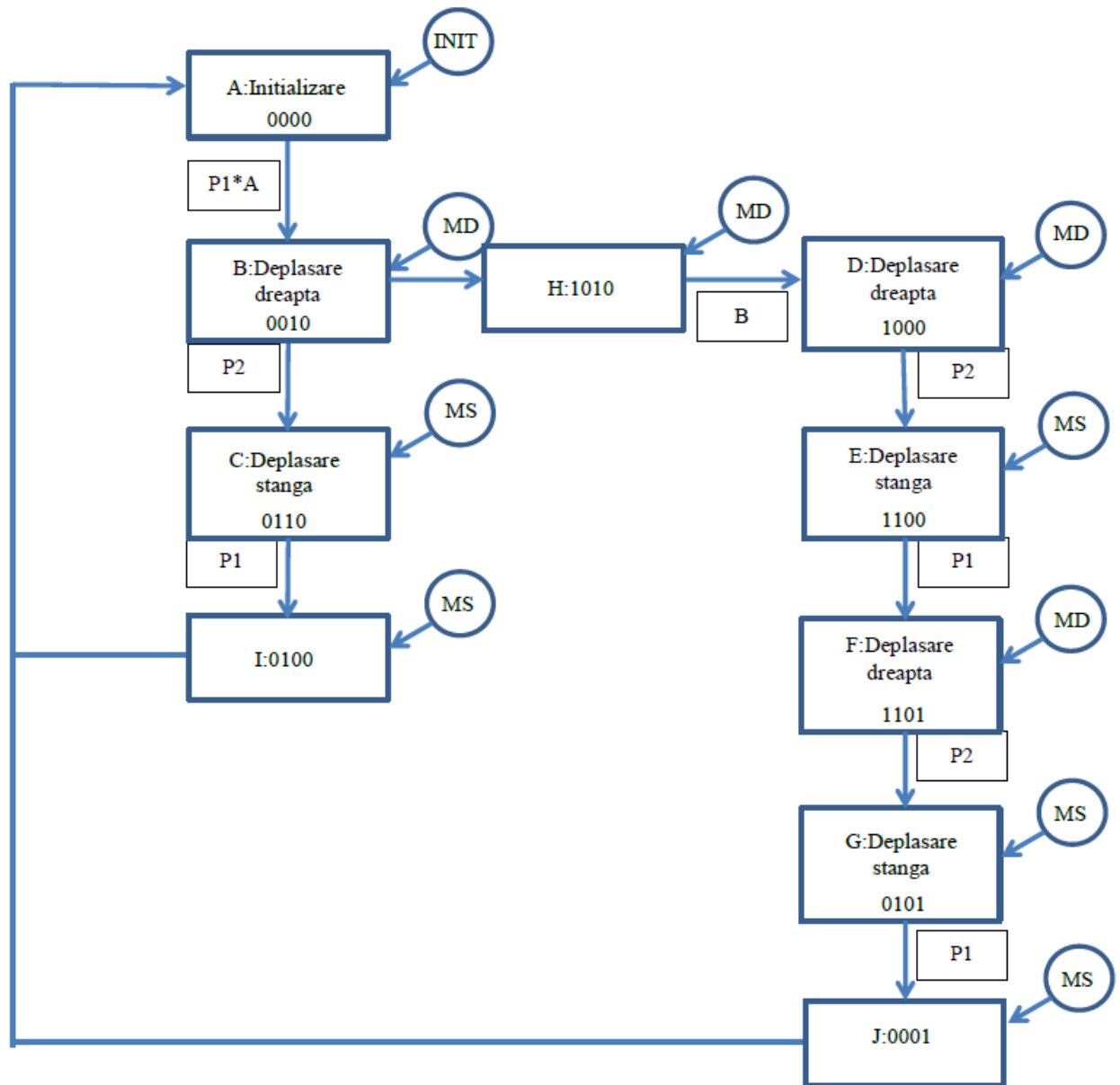


Fig. 1.3.1 Codificarea stărilor



## 2. Intocmirea organigramei funcționale a automatului secvențial descris mai sus

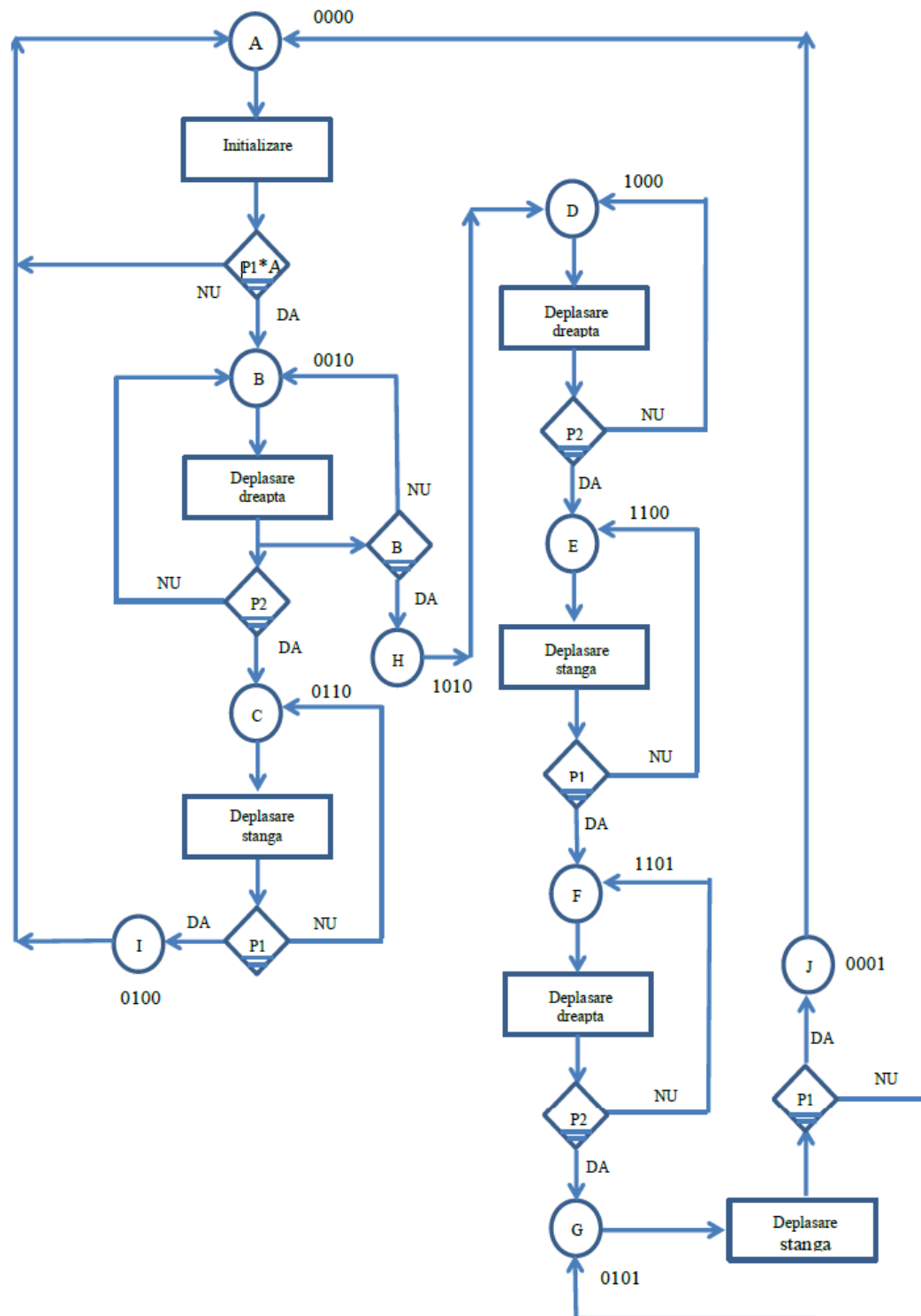
Conform caietului de sarcini, se întocmește organigrama în care se pun în evidență stările, tranzițiile între stări, semnalele de intrare sau deciziile care determină aceste tranziții și evoluția semnalelor de ieșire.

Pentru întocmirea organigramei din caietul de sarcini a fost necesar să se utilizeze  $n = 4$  variabile de stare ( $y_1, y_2, y_3, y_4$ ), prin urmare numărul maxim de stări este  $2^n = 16$ , din care au fost folosite doar **10** stări.

Utilizarea organigramei constituie o metodă grafică care permite transpunerea direct, rapidă și intuitivă a tuturor condițiilor de operare pe care trebuie să le îndeplinească un CLS sub forma unui program sau schema logică cu instrucțiuni sau decizii, stări și ieșiri.

Elementele de bază ale unei organigrame sunt: elemente de instrucțiune, control sau decizie, elemente de stare și elemente de ieșire.

Organigrama se construiește pornindu-se de la tabelul de tranziții sau de la graful de tranziții al circuitului și dintr-o stare considerată stare initial.



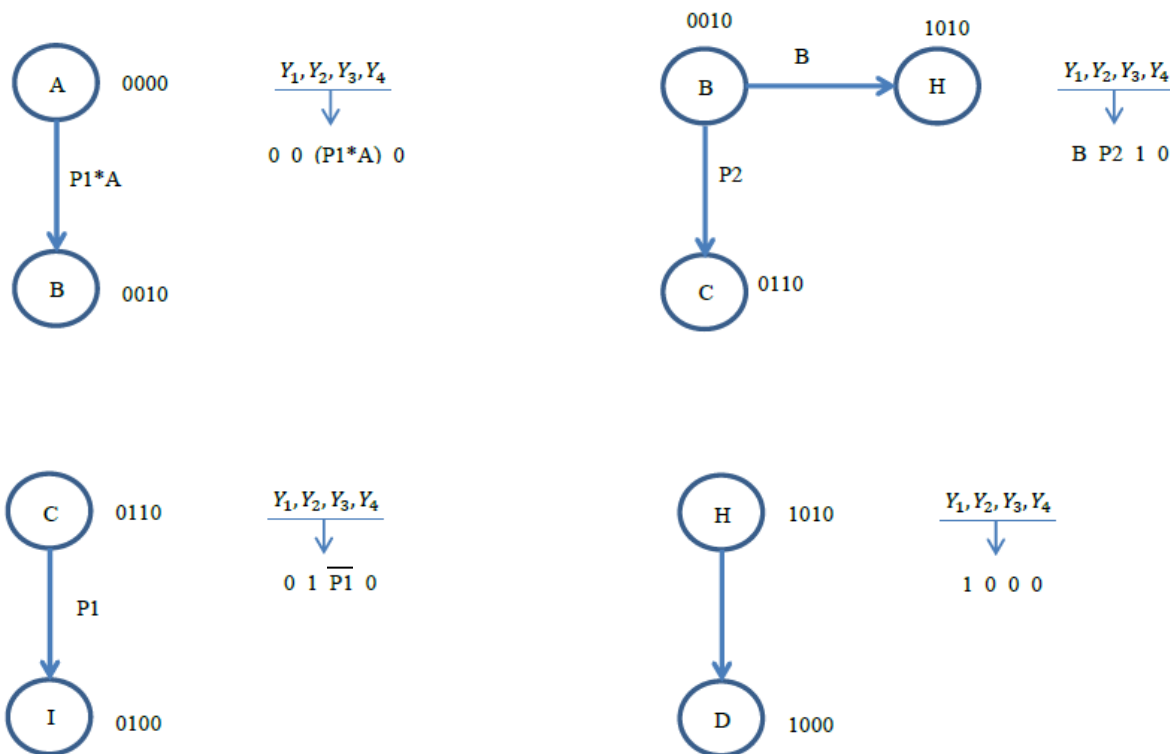
10

### 3. Stabilirea diagramei starilor la momentul "t"

Pentru completarea diagramei stărilor la momentul "t" se vor introduce în locațiile diagramei stările în funcție de codurile lor.

$y_1, y_2$ $y_3, y_4$	00	01	11	10
00	A	I	E	D
01	J	G	F	—
11	—	—	—	—
10	B	C	—	H

Fig. 3.1 Diagrama stărilor la momentul "t"



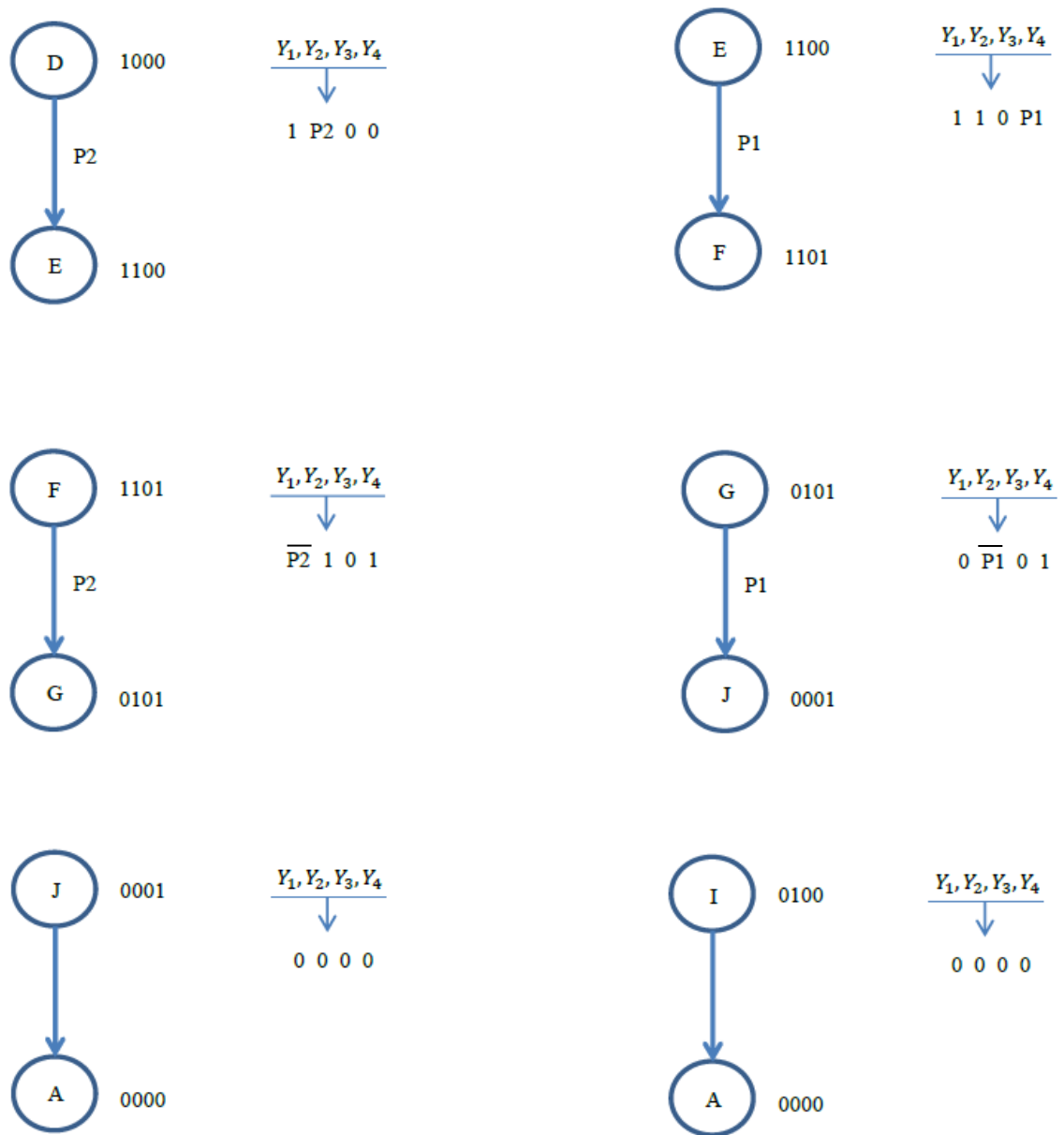


Fig 3.1 Tranzitiile starilor urmatoare

## 4. Intocmirea diagramei stărilor următoare

Pentru întocmirea diagramelor la momentul "t+1" trebuie introduse variabilele de intrare în diagramele VID ale stărilor următoare. Pentru sistemul din caietul de sarcini avem de introdus în diagramele VID 8 variabile, ( $y_1, y_2, y_3, y_4, A, B, P_1, P_2$ ) (4 variabile de stare și 4 variabile de intrare).

Pentru completarea matricelor de tranziții se vor analiza toate tranzițiile dintre stările automatului urmărindu-se modul în care se modifică sau nu variabilele de stare. Când în cursul tranziției o variabilă de stare se modifică din 0 în 1, atunci în locația corespunzătoare din matricea tranzițiilor se va trece semnalul de intrare care a determinat această tranziție. Se modifică din 1 în 0, iar atunci se trece semnalul de intrare negat. Pentru o tranziție dintr-o stare suplimentară fără semnal de intrare se va trece în locațiile matricelor de tranziție codul stărilor următoare (fig 3.1).

Având în vedere tranzițiile de mai sus se vor scrie tabelele de excitație.

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	1	1
01	0	0	$\overline{P_2}$	—
11	—	—	—	—
10	B	0	—	1

Tab. 4.1 Tabel de excitație pentru Y1

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	1	$P_2$
01	0	$\overline{P_1}$	1	—
11	—	—	—	—
10	$P_2$	1	—	0

Tab. 4.2 Tabel de excitație pentru Y2

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	$P1 * A$	0	0	0
01	0	0	0	—
11	—	—	—	—
10	1	$\overline{P1}$	—	0

Tab. 4.3 Tabel de excitație pentru Y3

$y_3, y_4 \backslash y_1, y_2$	0	01	11	10
00	0	0	$P1$	0
01	0	1	1	—
11	—	—	—	—
10	0	0	—	0

Tab. 4.4 Tabel de excitație pentru Y4

## 5. Obținerea funcțiilor de excitație pentru automatele elementare ce compun registrul de stare

Tehnica determinării funcțiilor de excitație pentru diagramele VID, cuprinde următoarele etape:

- I. Se consideră toate variabilele înglobate egale cu 0, și se formează subcuburi cu 1-urile din diagramă
- II. Se consideră toate 1-urile indiferente (\*) și se formează subcuburi cu variabilele înglobate
- III. Se consideră conjunctiva variabilelor înglobate cu implicanții primi obținuți la etapa II.
- IV. Se face disjuncția implicanților primi obținuți în etapele I. și III.

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	1	1
01	0	0	0	—
11	—	—	—	—
10	0	0	—	1

Etapa I. pentru Y1:  $y_1, \bar{y}_4$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	*	*
01	0	0	$\bar{P}2$	—
11	—	—	—	—
10	B	0	—	*

Etapa II. pentru Y1:  $y_1, y_3 \bar{y}_4$

Etapa III. pentru Y1:

$$\overline{P2} * y_1; B * y_3 \overline{y_4}$$

Etapa IV. pentru Y1:

$$Y1 = y_1 \overline{y_4} + \overline{P2} * y_1 + B * y_3 \overline{y_4}$$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	1	0
01	0	0	1	—
11	—	—	—	—
10	0	1	—	0

Etapa I. pentru Y2:  $y_1 y_2; y_2 y_3$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	*	P2
01	0	P1	*	—
11	—	—	—	—
10	P2	*	—	0

Etapa II. pentru Y2:  $y_2 y_4; y_1 \overline{y_3}; \overline{y_1} y_3$

Etapa III. pentru Y2:

$$P2 * \overline{y_1} y_3; \overline{P1} * y_2 y_4; P2 * y_1 \overline{y_3}$$

Etapa IV. pentru Y2:

$$Y2 = y_1 y_2 + y_2 y_3 + P2 * \overline{y_1} y_3 + \overline{P1} * y_2 y_4 + P2 * y_1 \overline{y_3}$$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	0	0
01	0	0	0	—
11	—	—	—	—
10	1	0	—	0

Etapa I. pentru Y3:  $\overline{y_1} y_2 y_3$



$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	$P1 * A$	0	0	0
01	0	0	0	—
11	—	—	—	—
10	*	$\overline{P1}$	—	0

Etapa II. pentru Y3:  $y_1 \overline{y_3}$ ;  $\overline{y_1} \overline{y_2} \overline{y_4}$

Etapa III. pentru Y3:

$$P1 * A * \overline{y_1} \overline{y_2} \overline{y_4}; \overline{P1} * y_1 \overline{y_3}$$

Etapa IV. pentru Y3:

$$Y3 = \overline{y_1} \overline{y_2} y_3 + P1 * A * \overline{y_1} \overline{y_2} \overline{y_4} + \overline{P1} * y_1 \overline{y_3}$$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	0	0
01	0	1	1	—
11	—	—	—	—
10	0	0	—	0

Etapa I. pentru Y4:  $y_2 y_4$

$y_3, y_4 \backslash y_1, y_2$	0	01	11	10
0	0	0	$P1$	0
00	0	0	$P1$	0
01	0	*	*	—
11	—	—	—	—
10	0	0	—	0

Etapa II. pentru Y4:  $y_1 y_2$

Etapa III. pentru Y4:

$$P1 * y_1 y_2$$

Etapa IV. pentru Y4:

$$Y4 = y_2 y_4 + P1 * y_1 y_2$$

## 6. Obținerea funcțiilor de ieșire ale automatului

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	1	0	0	0
01	0	0	0	—
11	—	—	—	—
10	0	0	—	0

Tab. 6.1 Ieșirea INIT

$$INIT = \overline{y_1} \overline{y_2} \overline{y_3} \overline{y_4}$$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	1	1	0
01	1	1	0	—
11	—	—	—	—
10	0	1	—	0

Tab. 6.2 Ieșirea MS

$$MS = \overline{y_1} y_4 + y_2 \overline{y_4}$$

$y_3, y_4 \backslash y_1, y_2$	00	01	11	10
00	0	0	0	1
01	0	0	1	—
11	—	—	—	—
10	1	0	—	1

Tab. 6.1 Ieșirea MD

$$MD = \overline{y_2} y_3 + y_1 \overline{y_2} + y_1 y_4$$

## 7. Implementarea funcțiilor logice obținute cu circuite integrate

Pentru implementarea funcțiilor logice cu circuite basculante bistabile de tip D și porți logice am folosit următoarele notații:

$$y_1 = Q_1^t; y_2 = Q_2^t; y_3 = Q_3^t; y_4 = Q_4^t$$

$$Y_1 = Q_1^{t+1}; Y_2 = Q_2^{t+1}; Y_3 = Q_3^{t+1}; Y_4 = Q_4^{t+1}$$

Ecuatiile intrărilor bistabililor:

$$D_1^t = Q_1^{t+1} = Y_1 = Q_1^t \overline{Q_4^t} + \overline{P2} Q_1^t + B Q_3^t \overline{Q_4^t}$$

$$D_2^t = Q_2^{t+1} = Y_2 = Q_1^t Q_2^t + Q_2^t Q_3^t + P2 \overline{Q_1^t} Q_3^t + \overline{P1} Q_2^t Q_4^t + P2 Q_1^t \overline{Q_3^t}$$

$$D_3^t = Q_3^{t+1} = Y_3 = \overline{Q_1^t} \overline{Q_2^t} Q_3^t + P1 * A \overline{Q_1^t} \overline{Q_2^t} \overline{Q_3^t} + P1 Q_1^t \overline{Q_3^t}$$

$$D_4^t = Q_4^{t+1} = Y_4 = Q_2^t Q_4^t + P1 Q_1^t Q_2^t$$

Ecuatiile ieșirilor:

$$Z_1 = \overline{Q_1^t} Q_4^t + Q_2^t \overline{Q_4^t}$$

$$Z_1 = \overline{Q_2^t} Q_3^t + Q_1^t \overline{Q_2^t} + Q_1^t Q_4^t$$

$$INIT = \overline{Q_1^t} \overline{Q_2^t} \overline{Q_3^t} \overline{Q_4^t}$$

Implementarea cu porți logice ȘI-NU:

Pentru intrări:

$$D_1^t = \overline{\overline{D_1^t}} = \overline{\overline{Q_1^t Q_4^t} * \overline{\overline{P2} Q_1^t} * \overline{\overline{B Q_3^t Q_4^t}}}$$

$$D_2^t = \overline{\overline{D_2^t}} = \overline{\overline{Q_1^t Q_2^t} * \overline{\overline{Q_2^t Q_3^t} * \overline{\overline{P2} Q_1^t Q_3^t} * \overline{\overline{P1} Q_2^t Q_4^t} * \overline{\overline{P2} Q_1^t Q_3^t}}}$$

$$D_3^t = \overline{\overline{D_3^t}} = \overline{\overline{Q_1^t \overline{Q_2^t} Q_3^t} * \overline{\overline{P1} * A \overline{Q_1^t} \overline{Q_2^t} \overline{Q_3^t} * \overline{\overline{P1} Q_1^t Q_3^t}}}$$

$$D_4^t = \overline{\overline{D_4^t}} = \overline{\overline{Q_2^t Q_4^t} * \overline{\overline{P1} Q_1^t Q_2^t}}}$$

Pentru ieșiri:

$$Z_1 = \overline{\overline{Z_1}} = \overline{\overline{\overline{Q_1^t} Q_4^t} * \overline{\overline{Q_2^t Q_4^t}}}$$

$$Z_1 = \overline{\overline{Z_2}} = \overline{\overline{\overline{Q_2^t} Q_3^t} * \overline{\overline{Q_1^t Q_2^t} * \overline{\overline{Q_1^t} Q_4^t}}}$$

$$INIT = \overline{\overline{INIT}} = \overline{\overline{\overline{Q_1^t} \overline{Q_2^t} \overline{Q_3^t} \overline{Q_4^t}}}$$

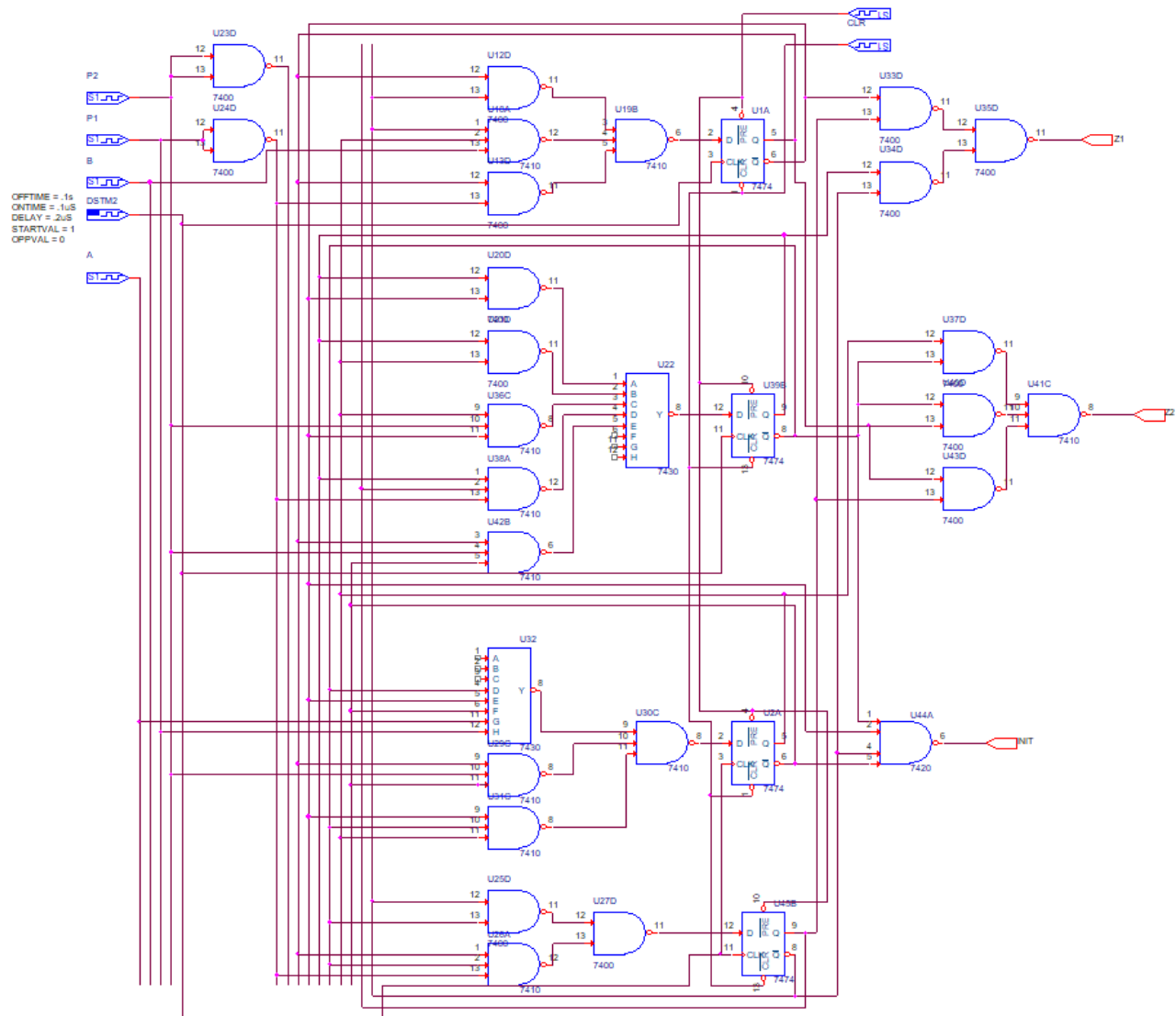


Fig. 7.1 Implementarea funcțiilor logice cu bistabili de tip D și porți ȘI-NU

## 8. Analiza schemei logice obținute

Schema logică obținută a fost realizată cu ajutorul mediului de lucru OrCAD. Pentru implementarea funcțiilor am folosit următoarele porți logice: 7 x 7410, 13 x 7400, 2 x 7430, 1 x 7420, 4 x 7474.

Pentru implementarea cu circuite integrate se pot folosi următoarele: 2x 7474(-), 3x 74LS10 (-2), 4x 74LS00(-3), 1x 70LS20 (-1), 2x 74LS30(-).

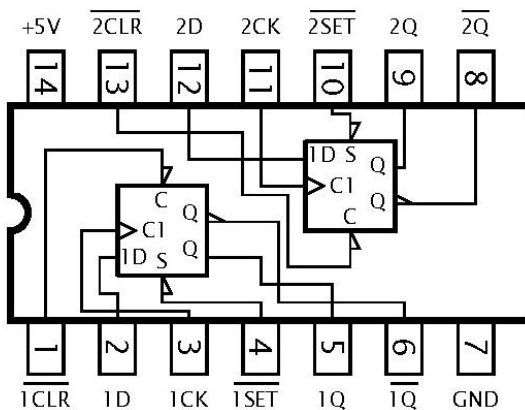


Fig. 8.1 CI 7474

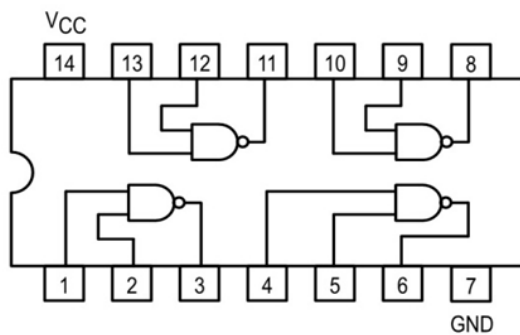


Fig. 8.2 CI 74LS00

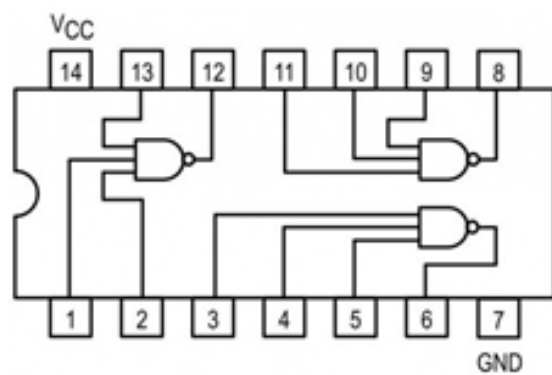


Fig. 8.3 CI 74LS10

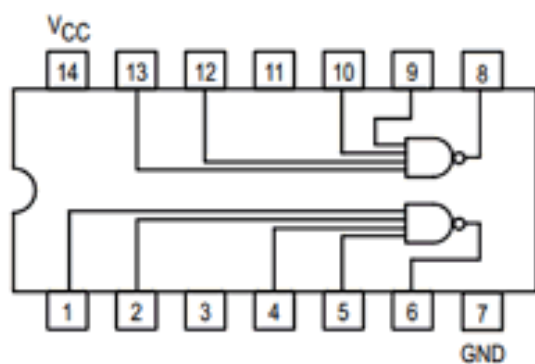


Fig. 8.4 CI 74LS20

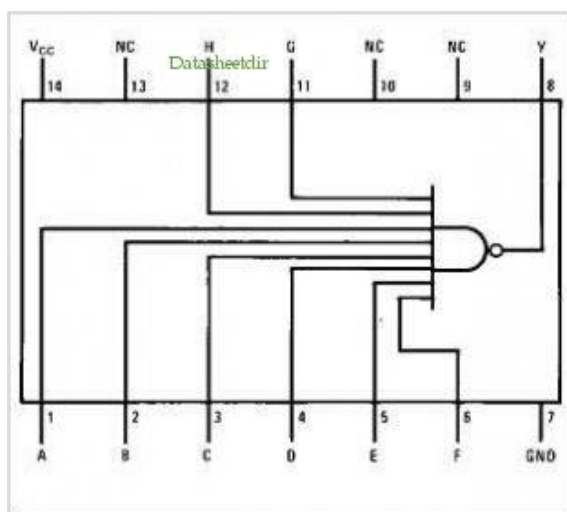


Fig. 8.5 CI 74LS30

## 9. Bibliografie

1. Moldoveanu F., Floroian D., Circuite logice secvențiale. Circuite logice combinaționale, Ed. Universității Transilvania Brașov, 2003.
2. <http://www.ti.com/> – pagina oficială Texas Instruments
3. [http://www.datasheetcatalog.com/datasheets\\_pdf](http://www.datasheetcatalog.com/datasheets_pdf) – pagina de căutare a diferitelor date de catalog.