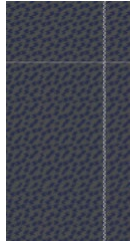


# AUTOMATE FĂRĂ PIERDERI

S.I. dr. Ing. Vlad-Cristian Miclea

Universitatea Tehnica din Cluj-Napoca  
Departamentul Calculatoare



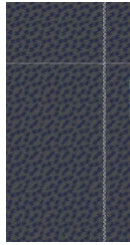
# CUPRINS

- 1) Introducere
- 2) Automate fara pierderi
- 3) Metode de identificare a pierderilor
- 4) Reconstituirea secventei de intrare
- 5) Concluzii



# PLAN CURS

- Partea 1 – VHDL
  1. Limbajul VHDL – 1
  2. Limbajul VHDL – 2
  3. Limbajul VHDL – 3
- Partea 2 – Implementarea sistemelor numerice
  4. Microprogramare
  5. Partea 1 - Unitate de comanda – exemplu cuptor
  5. Partea 2 - Unitate de executie – exemplu cuptor
- Partea 3 – Automate
  6. Automate finite
  7. Stari
  8. Automate sincrone
  9. Automate asincrone
  10. Identificarea automatelor
  - 11. Automate fara pierderi**
  12. Automate liniare
- Partea 4 – Probleme si discutii



# CONTEXT

## Cursurile trecute

- Automate finite
  - Abstractizarea circuitelor secventiale
  - Clasificarea automatelor (Moore, Mealy)
- Stari ale automatelor
  - Reducerea si codificarea eficienta a starilor
- Automate sincrone
  - Metode de eficientizare
- Automate asincrone
  - Curse Critice
- Identificarea automatelor
  - Aducerea automatelor in stari initiale/cunoscute



# AUTOMATE FĂRĂ PIERDERI

## Introducere

- Aplicație a automatelor finite: codificarea informației în vederea transmiterii ei
- Liniile de transmisie pot genera zgomote care alterează conținutul informației
- Apare necesitatea reconstituirii secvenței de intrare, cunoscând starea inițială, starea finală și secvența de ieșire a automatului
- **Definiție:** Un **automat** se numește **fără pierderi**, din punctul de vedere al informației, dacă pentru fiecare combinație a stării inițiale, stării finale și secvenței de ieșire **există cel mult o secvență de intrare** care ar putea produce secvența de ieșire dată
- La un automat se determină întâi dacă are pierderi, iar dacă nu are pierderi se determină secvența de intrare

# AUTOMATE FĂRĂ PIERDERI

## ■ Exemple: Automat 1

	0	1
A	A,0	B,0
B	C,0	D,1
C	C,1	B,1
D	D,0	C,0

- Din starea inițială C, pentru secvențele de intrare aplicate 0101 și 1111 avem:
  - Intrări 0101, stările CBCB și ieșirile 1101
  - Intrări 1111, stările BDCB și ieșirile 1101
- Automatul este cu pierderi, pentru că la secvențe diferite de intrări, plecând din aceeași stare (C) și ajungând în aceeași stare (B) generează aceleași secvențe de ieșiri

# AUTOMATE FĂRĂ PIERDERI

## ■ Exemple: Automat 2

	0	1
A	B,1	C,0
B	A,0	D,1
C	D,0	D,1
D	C,1	B,0

- Din starea inițială A, pentru secvențele de intrare aplicate 0101 și 0011 avem:
  - Intrări 0101, stările BDCD și ieșirile 1111
  - Intrări 0011, stările BACD și ieșirile 1001
- Automatul apare ca fiind fără pierderi, pentru că la secvențe diferite de intrări, plecând din aceeași stare (A) și ajungând în aceeași stare (D) generează secvențe de ieșiri diferite
- **Observație:** se va utiliza o metodă de determinare a pierderilor pentru a hotărî că automatul este sigur fără pierderi!!!



# METODE DE DETERMINARE A PIERDERILOR

## Determinarea pierderilor

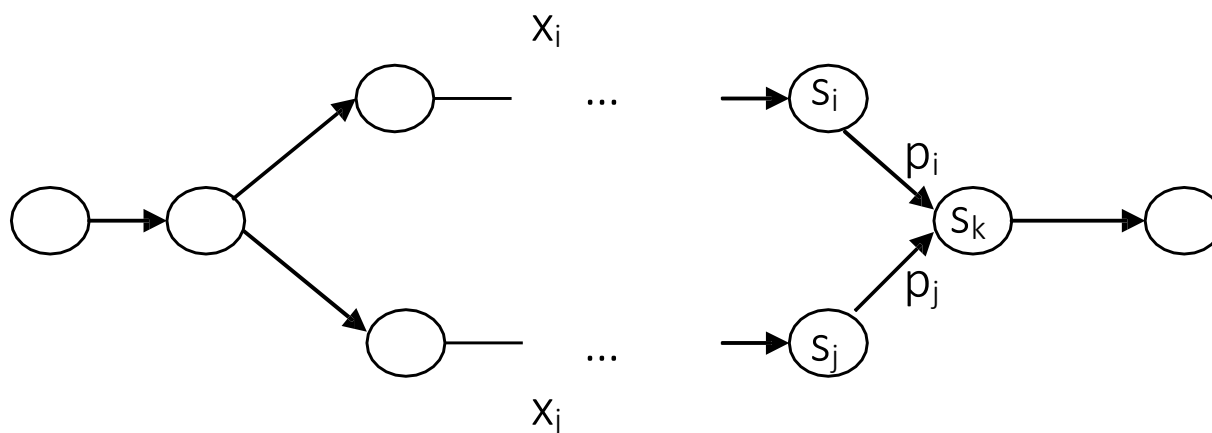
- Se testează automatul pentru cazurile care implică pierderi
- Trebuie să existe cel puțin 2 secvențe de intrare care să conducă automatul dintr-o stare inițială în una finală, generând aceeași secvență de ieșire
- Avem **2 situații** posibile:
  - Când secvența de intrare determină **aceeași secvență de stări**
    - Trebuie ca cel puțin o tranziție să poată fi determinată de 2 intrări diferite
    - O linie din tabelul de tranziție conține atunci elemente identice pe 2 coloane diferite
  - Când secvența de intrare determină **secvențe diferite de stări**
    - O pereche de astfel de secvențe se determină într-un punct de convergență a stărilor



# METODE DE DETERMINARE A PIERDERILOR

## Determinarea pierderilor

### ■ Convergența stărilor



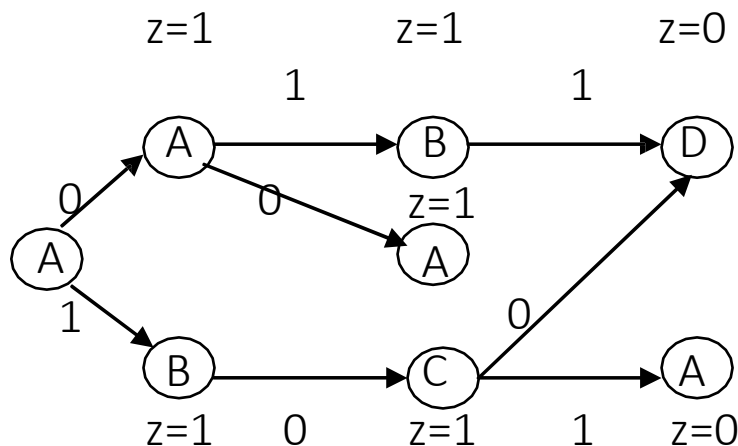
### ■ Se caută perechi de stări $s_i, s_j$ astfel:

- $s_i, s_j$  pot fi atinse dintr-o stare inițială comună, prin intervenția secvențelor de intrare  $x_i, x_j$ , care determină ieșiri identice
- Există valori de intrare  $p_i, p_j$  care determină tranziția automatului din stările  $s_i, s_j$  într-o stare finală comună  $s_k$ , generând aceeași ieșire

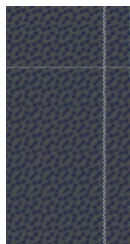
■ Exempu:

S \ x	0	1
A	A,1	B,1
B	C,1	D,0
C	D,0	A,0
D	C,0	D,1

- Se pleacă din starea A și secvența de ieșire obținută este 110



Plecând din A, pentru ieșire 110, avem secvențe diferite de stări, ABD și BCD, care au asociate 2 secvențe diferite de intrări, 011 și 100



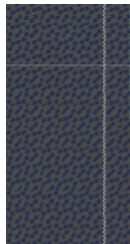
# METODE DE DETERMINARE A PIERDERILOR

## Determinarea pierderilor

- Tabelul stărilor următoare (**Tabel înainte**) – permite determinarea punctelor de convergență
  - Se consideră toate stările inițiale și succesorii lor și toate secvențele de ieșiri posibile

S \ x	0	1
A	A,1	B,1
B	C,1	D,0
C	D,0	A,0
D	C,0	D,1

	z=0	z=1
A	-	AB
B	D	C
C	AD	-
D	C	D
AB	D	ABC
AD	C	ABD
ABC	AD*	ABC
ABD	CD	ABCD
CD	ACD	D
ABCD	ACD*	ABCD
ACD	ACD	ABD



# METODE DE DETERMINARE A PIERDERILOR

## Metoda generală de testare pentru pierderi

### ■ Pașii:

- 1. Se verifică dacă în tabelul de tranziții apar 2 elemente identice pe aceeași linie => automatul este cu pierderi – caz simplu
- 2. Se alcătuieste tabelul stărilor următoare (tabel înainte), începând cu toate stările singulare și apoi cu succesorii lor
  - Dacă apar stări **duplicate** printre stările care se combină avem o convergență => automatul este **cu pierderi**
  - Dacă tabelul se completează fără să se găsească duplicate de stări => automatul este **fără pierderi**

# METODE DE DETERMINARE A PIERDERILOR

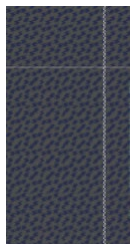
## Metoda generală de testare pentru pierderi

### ■ Exemple:

S \ x	0	1
A	B,1	C,1
B	C,1	D,0
C	B,0	A,1
D	A,0	C,0

	z=0	z=1
A	-	BC
B	D	C
C	B	A
D	AC	-
BC	BD	AC
AC	B	ABC
BD	ACD	C
ABC	BD	ABC*
ACD	ABC	ABC

- Deoarece în tabelul înainte există stări duplicate, automatul este cu pierderi



# METODE DE DETERMINARE A PIERDERILOR

## Metoda generală de testare pentru pierderi

### ■ Exemple:

$S \backslash x$	0	1
A	B,0	C,0
B	A,0	D,1
C	E,0	F,1
D	B,1	C,1
E	F,1	D,1
F	D,0	F,0

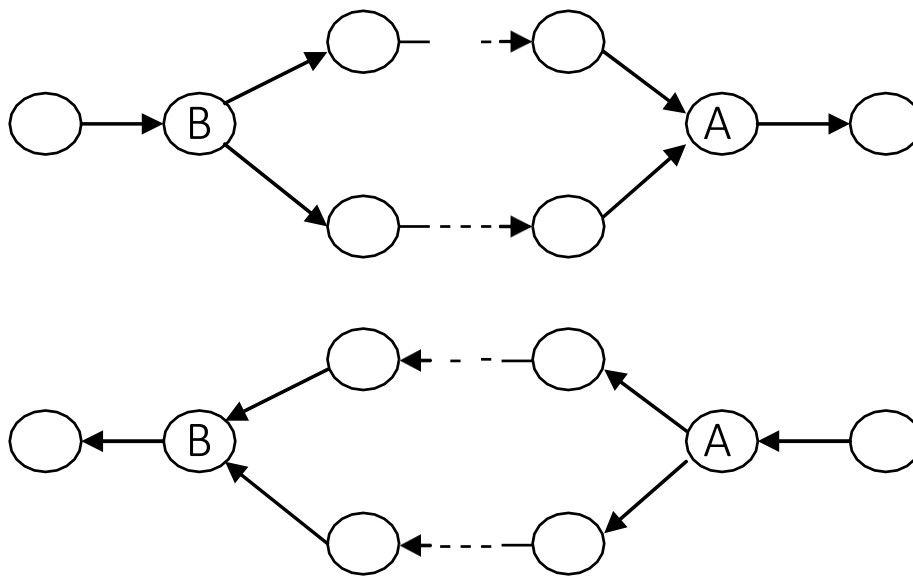
	$z=0$	$z=1$
A	BC	-
B	A	D
C	E	F
D	-	BC
E	-	DF
F	DF	-
BC	AE	DF
DF	DF	BC
AE	BC	DF

- Deoarece în tabelul înainte nu există stări duplicate, automatul este **fără pierderi**

# METODE DE DETERMINARE A PIERDERILOR

## Test alternativ de determinare a pierderilor

- Simetria figuri de la determinarea convergențelor (A) permite căutarea unor puncte (B) în care 2 secvențe de stări care au stare finală comună diverg pentru prima dată



- Se construiește un Tabel al stărilor anterioare (**Tabel înapoi**)

# METODE DE DETERMINARE A PIERDERILOR

## Test alternativ de determinare a pierderilor

### ■ Exemplu:

S \ x	0	1
A	B,1	C,1
B	C,1	D,0
C	B,0	A,1
D	A,0	C,0

z=0	z=1	
D	C	A
C	A	B
D	AB	C
B	-	D
CD	AC	AB
BD	AB	CD
D*	ABC	AC
BC	A	BD
CD*	A*BC	ABC
CD	A*B	BC

- Deoarece în tabelul înapoi există stări duplicate, automatul este cu pierderi

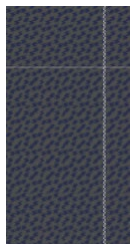




# RECONSTITUIREA SECVENȚEI DE INTRARE

## Metoda tabelelor individuale înainte / înapoi

- Reconstituirea secvenței de intrări este legată de reconstituirea secvenței de stări
- Metoda tabelelor individuale înainte / înapoi se poate aplica dacă tabelele au numai elemente singulare
- Dacă automatul este fără pierderi, există o singură succesiune de stări care duce automatul dintr-o stare inițială în una finală



# RECONSTITUIREA SECVENȚEI DE INTRARE

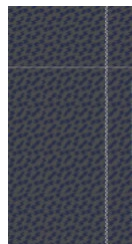
## Metoda tabelelor individuale înainte / înapoi

### ■ Exemplu:

<del>S</del> x	0	1
A	B,1	C,0
B	A,0	D,1
C	D,0	B,1
D	C,1	B,0

	z=0	z=1
A	C	B
B	A	D
C	D	B
D	B	C

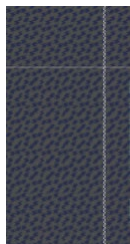
- În tabelul înainte nu există stări duplicate => automatul este fără pierderi
- Starea inițială este, de exemplu, B
- Secvența arbitrară de ieșiri este 0011110
- Reconstituirea secvenței de stări din tabelul înainte este BACBDCBA
- Reconstituirea secvenței de intrări pe baza tabelului de tranziții este 01110110



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Metoda de reconstrucție bilaterală

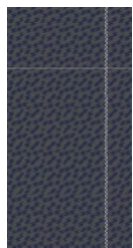
- Un automat fără pierderi evoluează dintr-o stare inițială cunoscută printr-o secvență de stări de lungime “m” și ajunge într-o stare finală cunoscută
- Cum se determină starea automatului după primele “k” ieșiri?
- Condiții care trebuie îndeplinite:
  - 1. Să existe o secvență de stări care să ducă automatul din starea inițială într-o stare “s”, generând la ieșire primele “k” valori
  - 2. Să existe o secvență de stări care să ducă automatul din starea “s” în starea finală, generând la ieșire “m-k” valori



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Metoda de reconstrucție bilaterală

- Setul de stări care satisface condiția 1 se determină prin aplicarea primelor “ $k$ ” valori de ieșire tabelului înainte, plecând de la elementul singular care rezultă din starea inițială cunoscută
- Setul de stări care satisface condiția 2 se determină prin aplicarea ultimelor “ $m-k$ ” valori de ieșire în ordine inversă tabelului înapoi, începând cu elementul singular care conține starea finală cunoscută
- Intersecția celor 2 seturi de stări generează o secvență de stări care satisface ambele condiții
- Pentru automate fără pierderi există o singură secvență de stări și de la ea se obține o singură secvență de intrări



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Metoda de reconstrucție bilaterală

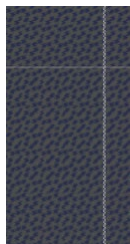
- **Exemplu:** Care este starea automatului între valorile 4 și 5 de intrare?

S \ x	0	1
A	B,0	C,0
B	A,0	D,1
C	E,0	F,1
D	B,1	C,1
E	F,1	D,1
F	D,0	F,0

	z=0	z=1
A	BC	-
B	A	D
C	E	F
D	-	BC
E	-	DF
F	DF	-
BC	AE	DF
DF	DF	BC
AE	BC	DF

z=0	z=1	
B	-	A
A	D	B
A	D	C
F	BE	D
C	-	E
F	CE	F
AC	D	BE
AC	D	CE
AB	D	AC
AB	D	AB

- Starea inițială este D
- Starea finală este F
- Secvența de ieșiri este 1101001



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Metoda de reconstrucție bilaterală

- Pentru secvența de ieșire dată, folosind tabelele înainte și înapoi, se obține prin intersectare succesiunea de stări

ieșiri		1	1	0	1	0	0	1
→ D	B	D	D	B	A	B	D	
	C	F	F	C	E	C	F	
	D	C	F	D	A	A	C	← F
	E			B	C	E		
	C	F	D	(B)	A	C		

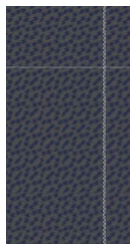
- Între valorile 4 și 5 de intrare automatul se află în starea B
- Secvența de stări rezultată plecând din D este CFDBACF
- Secvența de intrări corespunzătoare este 1100011 și este unică



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Test bilateral de determinare a pierderilor

- Se utilizează procedura anterioară de obținere a secvenței de stări
- La tabelele înainte și înapoi nu se verifică convergențele sau divergențele
- Dacă prin intersecția stărilor rezultă mai multe stări => există mai multe secvențe de stări care conduc automatul din starea inițială în starea finală => automatul are pierderi
- Dacă vreo intersecție de stări este vidă => nu există o secvență de stări care satisface cele 2 condiții



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Test bilateral de determinare a pierderilor

- **Exemplu:** Care este starea automatului între valorile 3 și 4 de intrare?

	0	1
A	A,0	B,0
B	C,0	D,1
C	C,1	B,1
D	D,0	C,0

	z=0	z=1
A	AB	-
B	C	D
C	-	BC
D	CD	-
AB	ABC	D
BC	C	BCD
CD	CD	BC
ABC	ABC	BCD
BCD	CD	BCD

z=0	z=1	
A	-	A
A	C	B
BD	C	C
D	B	D
AD	BC	BD
AD	B	AD
ABD	C	BC
AD	BC	ABD

- Starea inițială este A
- Starea finală este C
- Secvența de ieșiri este 0010101



# RECONSTITUIREA SECVENȚEI DE INTRARE

## Test bilateral de determinare a pierderilor

- Pentru secvența de ieșire dată, folosind tabelele înainte și înapoi, se obține prin intersectare succesiunea de stări

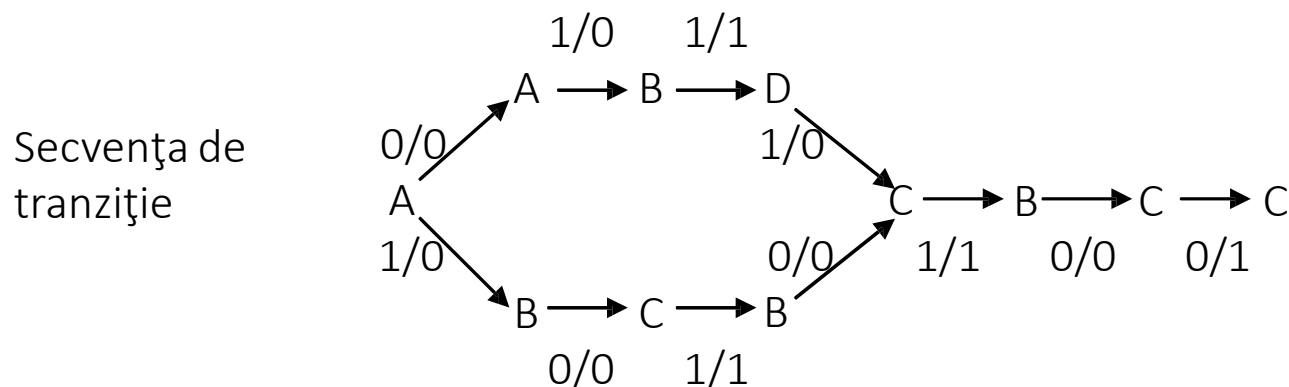
• ieșire

	0	0	1	0	1	0	1
A	→ AB	→ ABC	→ BCD	→ CD	→ BC	→ C	→ BC
AD	← ABD	← BC	← ABD	← BC	← BD	← C	← C

---

intersecția    A    AB    BC    (BD)    C    B    C    C

- Între intrarea 3 și 4 automatul se poate găsi în 2 stări diferite, B sau D  
=> automatul este cu pierderi



- Secvența de tranziții pune în evidență pierderile automatului



# CONCLUZII

## Automate fara pierderi

- Generalitati
- Determinarea pierderilor
  - Tabel inainte
  - Tabel inapoi
- Reconstructia secantei de intrare
  - Metoda de reconstructie bilaterala
- Test bilateral de determinare a pierderilor
- Data viitoare – automate liniare