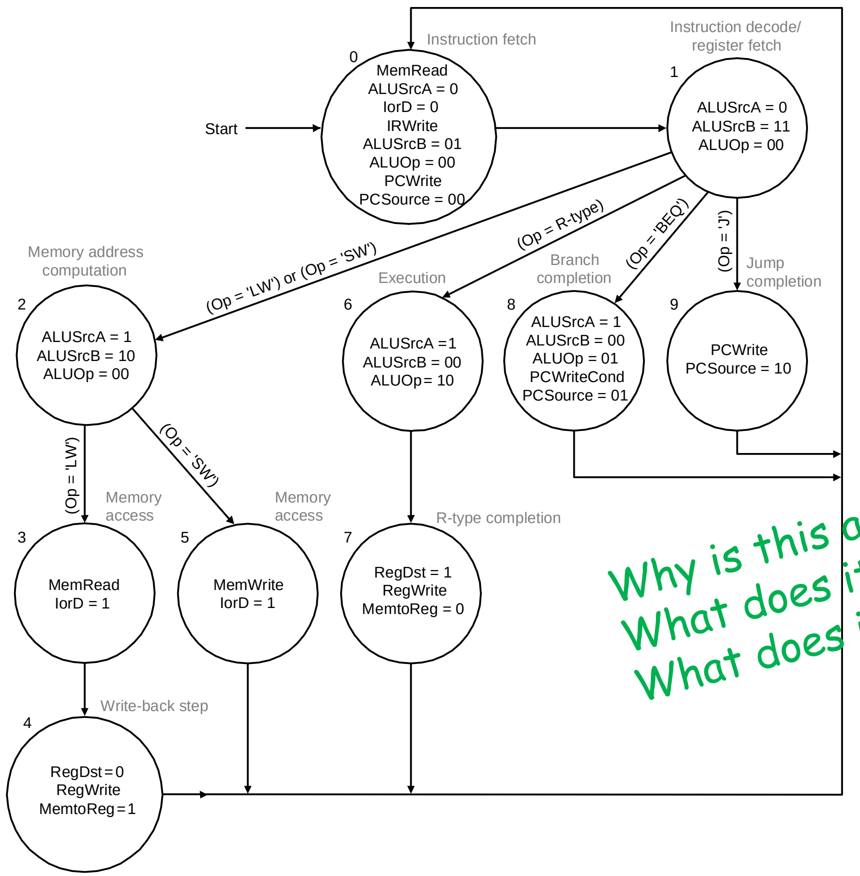
EE 044252: Digital Systems and Computer Structure

Spring 2018

Lecture 5: *Finite State Machines*



Why is this a FSM?
What does it do?
What does it mean?

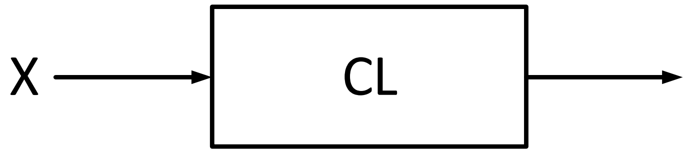
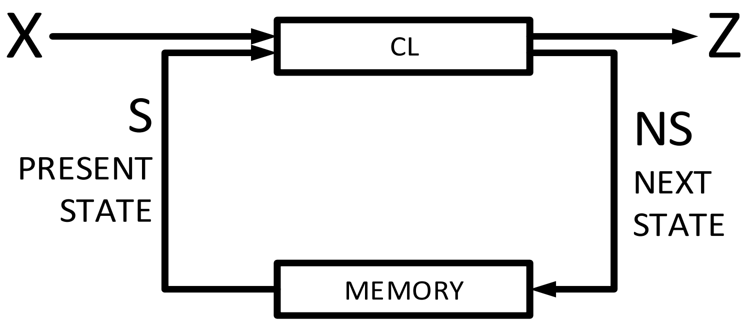
EE 044252: Digital Systems and Computer Structure

Topic	wk	Lectures	Tutorials	Workshop	Simulation
Arch	1	Intro. RISC-V architecture	Numbers. Codes		
Comb	2	Switching algebra & functions	Assembly programming		
	3	Combinational logic	Logic minimization	Combinational	
	4	Arithmetic. Memory	Gates		Combinational
Seq	5	Finite state machines	Logic		
	6	Sync FSM	Flip flops, FSM timing	Sequential	Sequential
	7	FSM equiv, scan, pipeline	FSM synthesis		
	8	Serial comm, memory instructions	Serial comm, pipeline		
μArch	9	Function call, single cycle RISC-V	Function call		
	10	Multi-cycle RISC-V	Single cycle RISC-V		Multi-cycle
	11	Interrupts, pipeline RISC-V	Multi-cycle RISC-V		
	12	Dependencies in pipeline RISC-V	Microcode, interrupts		
	13		Depend. in pipeline RISC-V		

Agenda

- Finite State Machine
  - Mealy and Moore FSM
  - Regular Expression
- FSM Synthesis

מערכת סדרתית / מערכת עקיבה / מכונת מצבים

מערכת צירופית (Combinational Circuit)	מערכת סדרתית (Sequential Circuit)
ערכי הפלט תלויים אך ורק בערכים הנוכחיים של משתני הקלט  Z=f(X)	• מערכת צירופית + זיכרון הזוכר "מצב" • המצב "זוכר היסטוריה" • ערכי הפלט תלויים בערכי הקלט ובמצב הקיים • ערכי המצב החדש גם הם תלויים בערכי הקלט ובמצב הקיים  Z=f(X,S), NS=g(X,S)

אם מספר המצבים סופי, זו מכונת מצבים סופית (FSM)

יש גם מכונות אינסופיות (לפחות בתיאוריה), למשל מחשב

מערכת סדרתית / מערכת עקיבה / מכונת מצבים

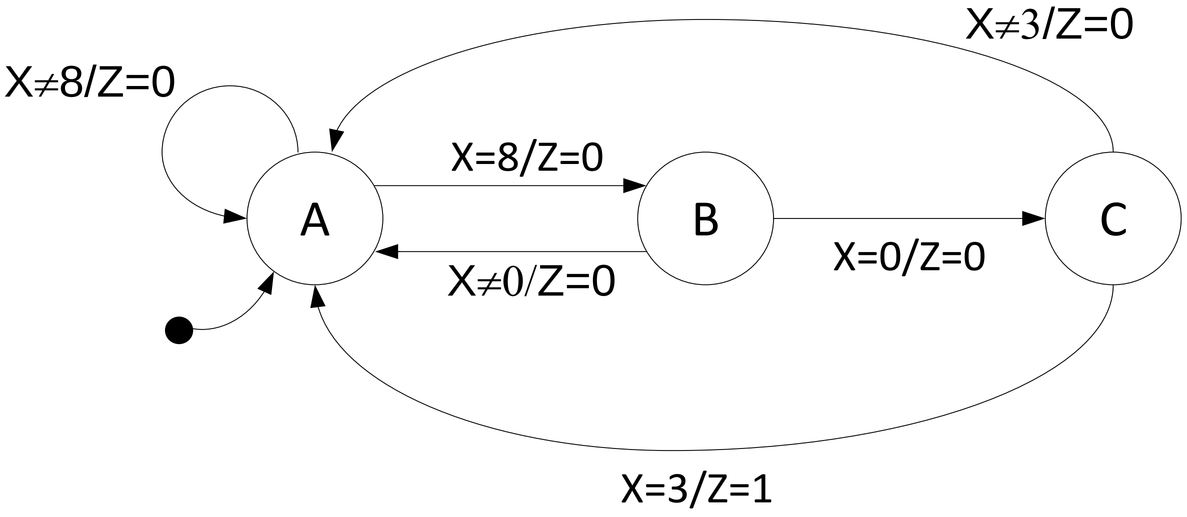
- מערכת עקיבה נמצאת בכל רגע נתון ב"מצב" מסוים
- המצב מיוצג ע"י ערכי הזיכרון של מערכת העקיבה
- המערכת יכולה לעבור ממצב אחד למצב אחר, בתלות במצב ובכניסות
  - ממצב אחד למצב העוקב, לכן "מערכת עקיבה"
  - סדרת מצבים וסדרת מעברים בין מצבים, לכן "מערכת סדרתית"
- מערכת עקיבה סינכרונית יכולה לעבור ממצב אחד למצב אחר רק בזמנים מסוימים
  - מערכת סינכרונית מקבלת אות שעון, והשעון קובע מתי יתבצעו מעברי המצב

מערכת סדרתית / מערכת עקיבה / מכונת מצבים

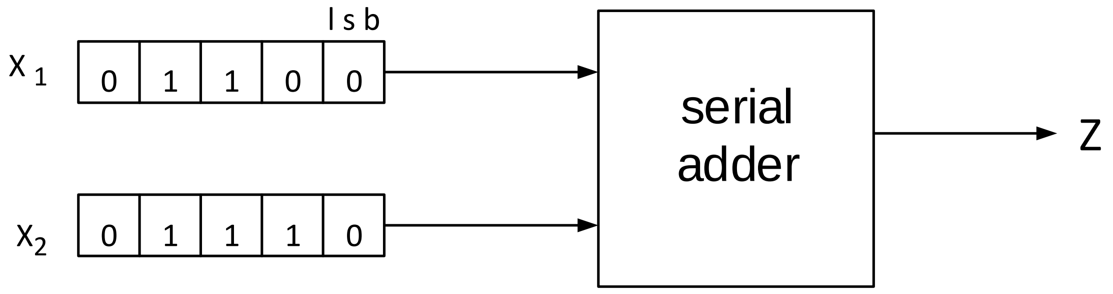
מערכת סדרתית (Sequential Circuit)	מערכת צירופית (Combinational Circuit)
	

דוגמה 1 —מנעול סדרתי

- סדרת הפתיחה של המנעול היא 803
  - קודם 8, אחר כך 0, לבסוף 3
- מצבים אפשריים במנעול:
  - A, מצב התחלתי
  - B, הוכנס המספר 8 (הכניסה מסומנת X)
  - C, הוכנס המספר 0 (לאחר שקודם לכן הוכנס 8)
- יציאה אפשרית Z ממכונת המצבים:
  - 0, נעול
  - 1, פתוח
- המכונה ניתנת לתיאור על ידי דיאגרמת מצבים (State Diagram)
  - על כל מעבר מסומנים הכניסה שגרמה לו והיציאה
- מכונת מילי (Mealy): תיאור סדרתי עקרוני
  - זמן מופשט

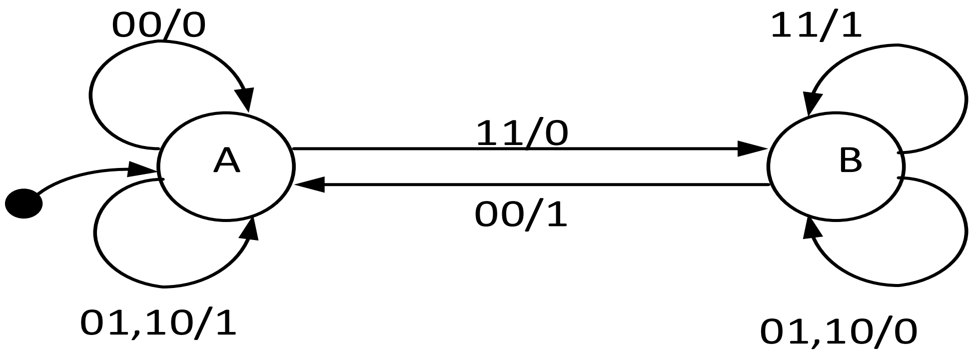


דוגמה 2 — מסכם בינארי טורי



- בתוך המסכם נשמר הנשא בין שלב לשלב
  - מצב A—נשא 0
  - מצב B—נשא 1

t_5	t_4	T_3	t_2	t_1	
0	1	1	0	0	X_1
0	1	1	1	0	X_2
1	1	0	1	0	Z

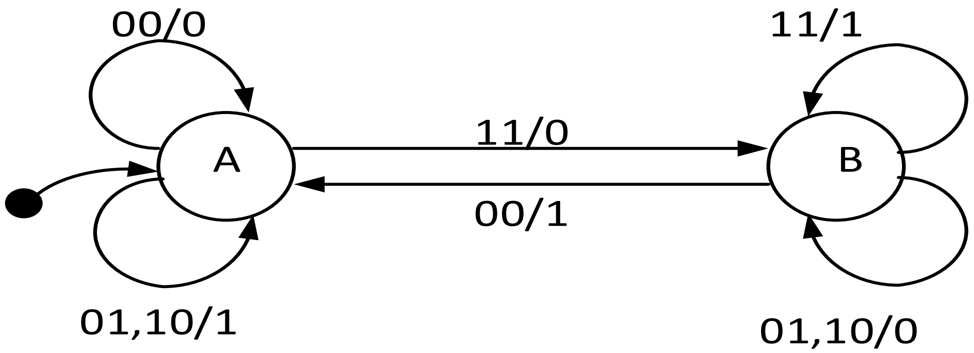


דוגמה 2 — מסכם בינארי טורי

PS (Present State)	NS (Next State), Z (Output) Inputs (x_1x_2) 00	01	11	10
A	A,0	A,1	B,0	A,1
B	A,1	B,0	B,1	B,0

PS (Present State)	NS (Next State) inputs (x_1x_2) 00	01	11	10	Z (Output) inputs (x_1x_2) 00	01	11	10
0	0	0	1	0	0	1	0	1
1	0	1	1	1	1	0	1	0

- נמיר את תיאור המכונה מדיאגרמת מצבים לטבלת מצבים
  - נסדר עמודות לפי קוד גריי
- נפריד את הטבלה ל-NS, Output
- הקצאת מצבים: נקצה ספרות לסימון המצבים
  - A=0, B=1

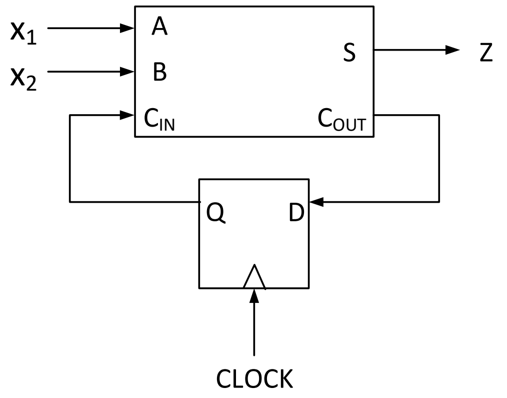


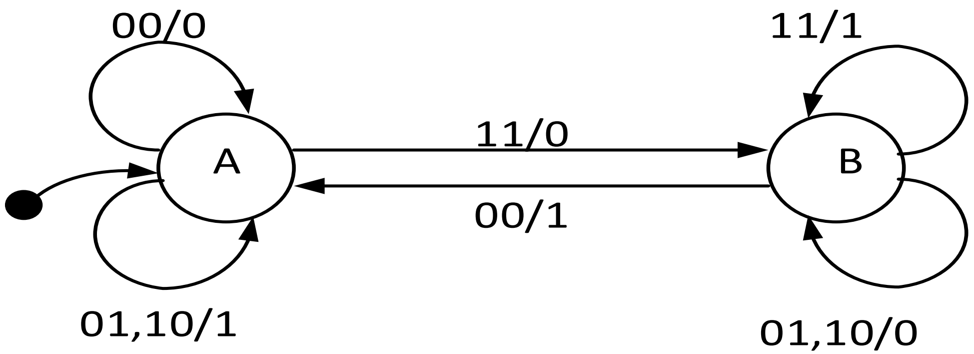
דוגמה 2 — מסכם בינארי טורי

PS (Present State)	NS (Next State) inputs (x_1x_2)				Z (Output) inputs (x_1x_2)			
	00	01	11	10	00	01	11	10
0	0	0	1	0	0	1	0	1
1	0	1	1	1	1	0	1	0

$NS = x_1x_2 + PS \cdot x_1 + PS \cdot x_2 = C_O$ $\quad Z = x_1 \oplus x_2 \oplus PS = Sum$

- נחשב את שתי הפונקציות NS, Z
  - למשל על ידי מפת קרנו (הטבלה כבר מסודרת בהתאם)
- הפונקציות (כמובן) זהות לאלו של FA
- במימוש, נשתמש ב-DFF לשמירת המצב
- השעון קובע מעבר בין שלבים
- מכונת מילי—זמן מופשט





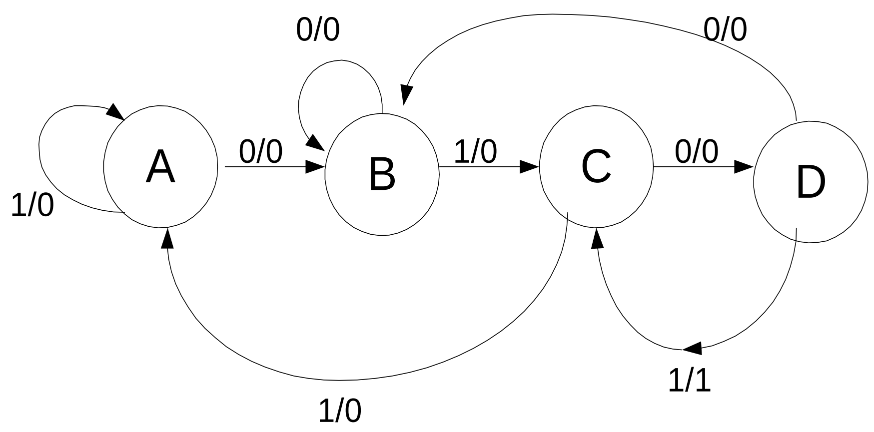
דוגמה 3—גלאי מחרוזת Sequence Detector

- נתכנן מערכת :
  - כניסה אחת
  - יציאה אחת
  - היציאה 1 אם״ם (אם ורק אם) מתגלה בכניסה המחרוזת 0101

here's another

here's one

x	0	0	0	1	0	1	0	1	…
z	0	0	0	0	0	1	0	1	…



דוגמה 3—גלאי מחרוזת Sequence Detector

PS	NS,z	
	X=0	X=1
A	B,0	A,0
B	B,0	C,0
C	D,0	A,0
D	B,0	C,1

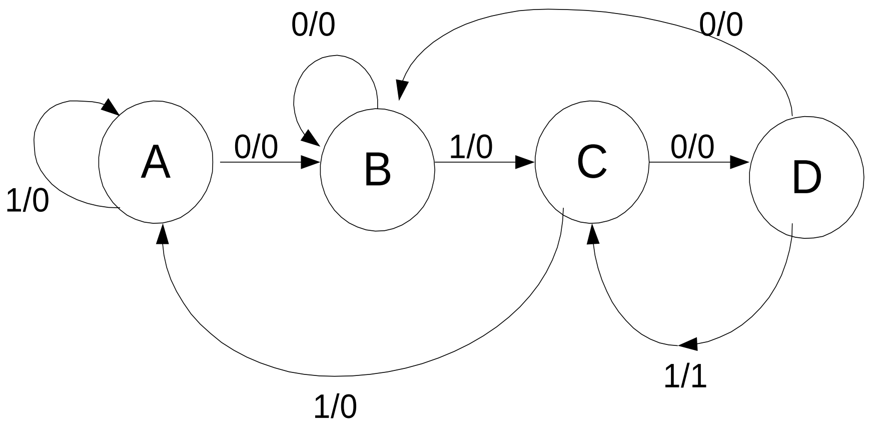
PS	NS(Y_1Y_2),Z	
y_1y_2	X=0	X=1
A=00	01,0	00,0
B=01	01,0	11,0
C=11	10,0	00,0
D=10	01,0	11,1

- תכן FSM מסוג מילי
  - טבלת מצבים
  - הקצאת מצבים (שני משתני מצב)
  - חישוב הפונקציות הצירופיות

$$Y_1 = xy_1'y_2 + x'y_1y_2 + xy_1y'_2$$

$$Y_2 = x'y'_1 + y'_1 y_2 + y_1y'_2$$

$$z = xy_1y'_2$$

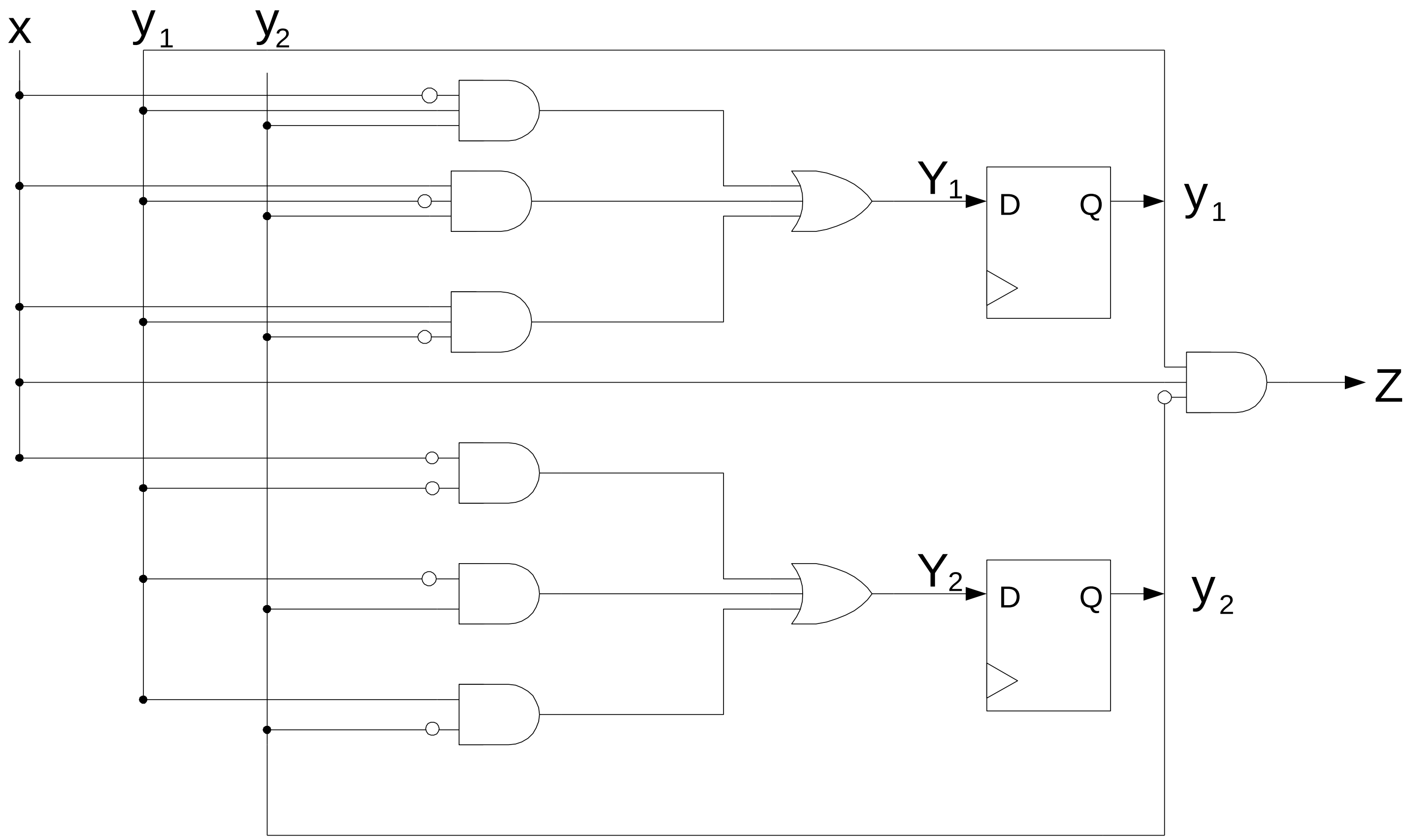


דוגמה 3—גלאי מחרוזות Sequence Detector

$$Y_1 = xy_1'y_2 + x'y_1y_2 + xy_1y'_2$$

$$Y_2 = x'y'_1 + y'_1\ y_2 + y_1y'_2$$

$$z = xy_1y'_2$$



סיבוכיות: 15 שערים של 2 כניסות

Sequence Detector דוגמה 3—גלאי מחרוזת

PS	NS,z	
	X=0	X=1
A	B,0	A,0
B	B,0	C,0
C	D,0	A,0
D	B,0	C,1

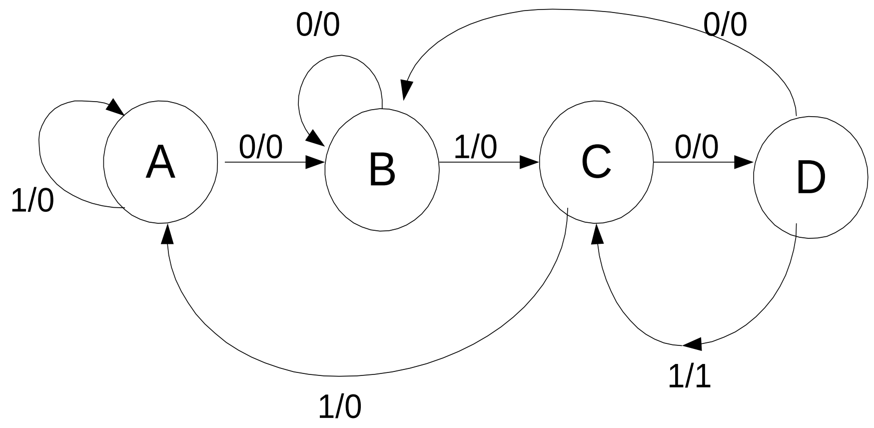
PS	NS(Y_1Y_2),Z	
y_1y_2	X=0	X=1
A=00	01,0	00,0
B=01	01,0	10,0
C=10	11,0	00,0
D=11	01,0	10,1

$$Y_1 = x' y_1 y'_2 + x y_2$$

$$Y_2 = x'$$

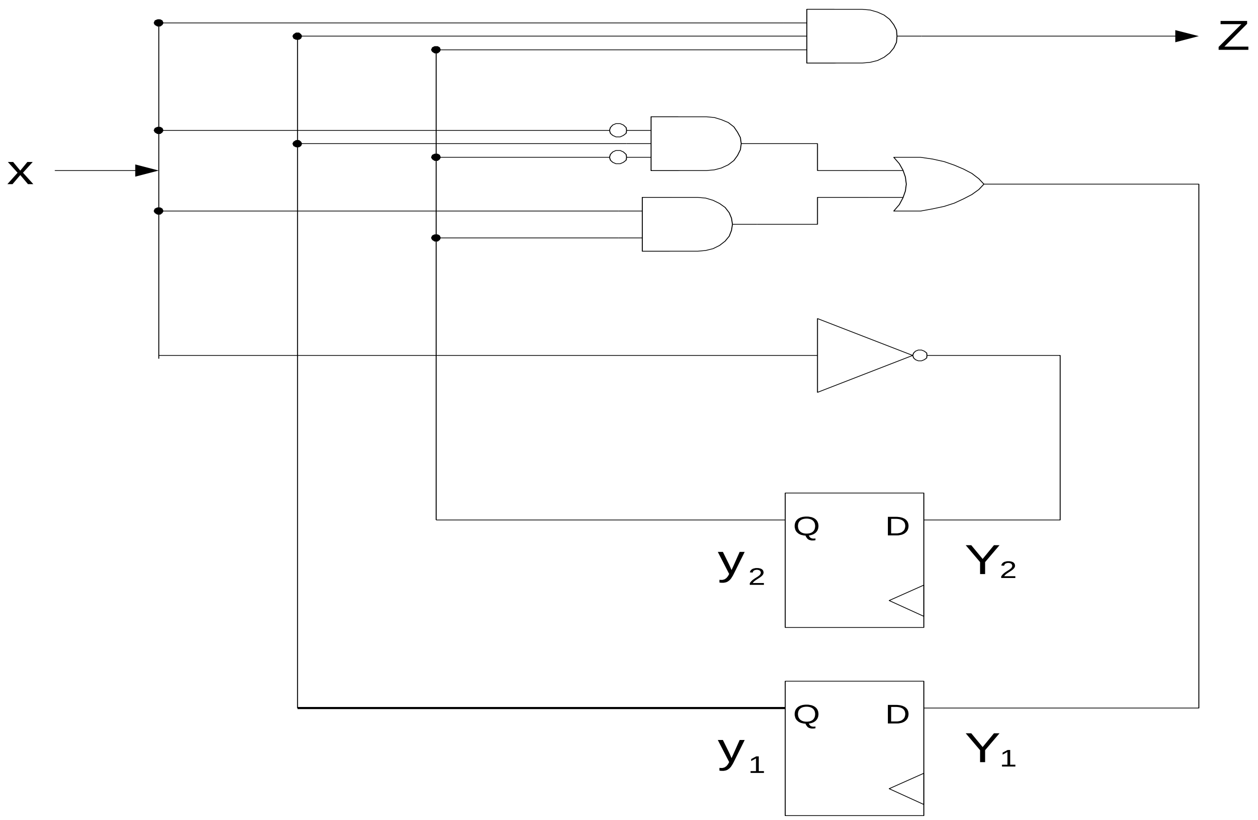
$$z = x y_1 y_2$$

- הקצאת מצבים שונה תביא למימוש שונה



דוגמה 3—גלאי מחרוזות Sequence Detector

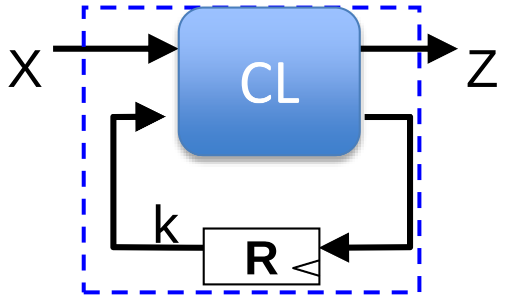
$$Y_1 = x' y_1 y'_2 + x y_2$$
$$Y_2 = x'$$
$$z = x y_1 y_2$$



סיבוכיות : 6 שערים של 2 כניסות (לעומת 15 במימוש קודם)

ביטוי רגולרי כתיאור מתמטי של מערכת סדרתית

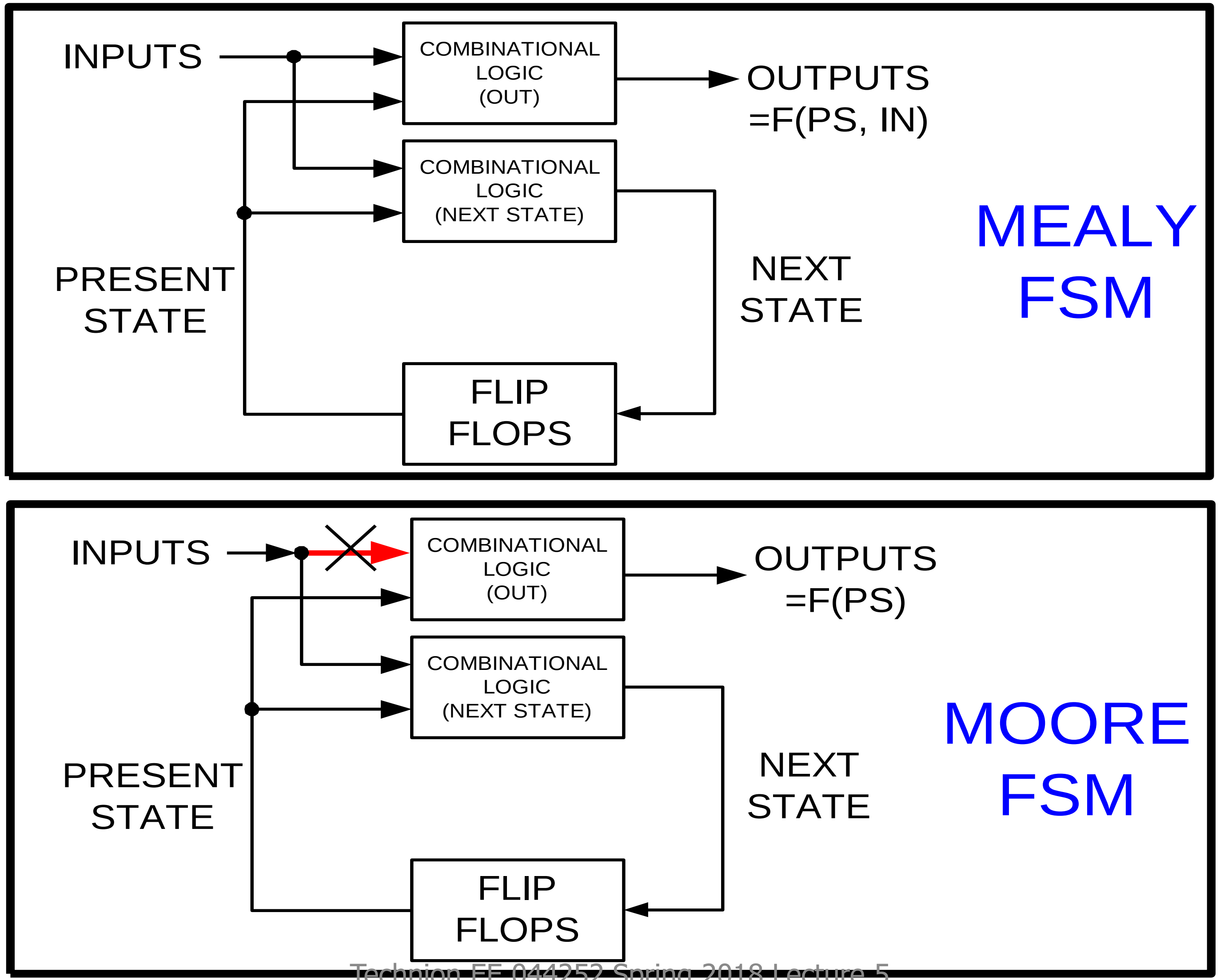
- מערכת סדרתית שמגלה מחרוזות ניתנת לתיאור מתמטי
- המערכת "מקבלת" מחרוזת ("מילה") השייכת לשפה
- השפה מתוארת על ידי ביטוי רגולרי
- נתון אלפבית סופי Σ , a (לכל $a\in\Sigma$) הוא ביטוי רגולרי שמתאר את השפה {a} או L[a]
- אם R,S ביטויים רגולרים אז $S \mid R$ הוא ביטוי רגולרי המתאר את השפה $L[R]\cup L[S]$
  - שפה בה מילה שייכת או לשפה האחת או לשנייה
- אם R,S ביטויים רגולרים אז $S \cdot R$ הוא ביטוי רגולרי המתאר את השפה $L[R]\cdot L[S]$
  - שפה בה מילה היא שרשור של שתי מלים, אחת שייכת לשפה האחת והשנייה שייכת לשפה השנייה
- R^* הוא ביטוי רגולרי המתאר את השפה $L[R^*]$
  - שפה בה מילה מורכבת ממספר כלשהו (כולל אפס) של מלים השייכות לשפה L[R]
- בדוגמה 3, המכונה מקבלת מלים השייכות לשפה $(0101)^*$
- בשימוש פרקטי מרחיבים את ההגדרה. מה מתארים הביטויים הרגולריים $05d^8$? 201[6-8] ?



הגדרת FSM

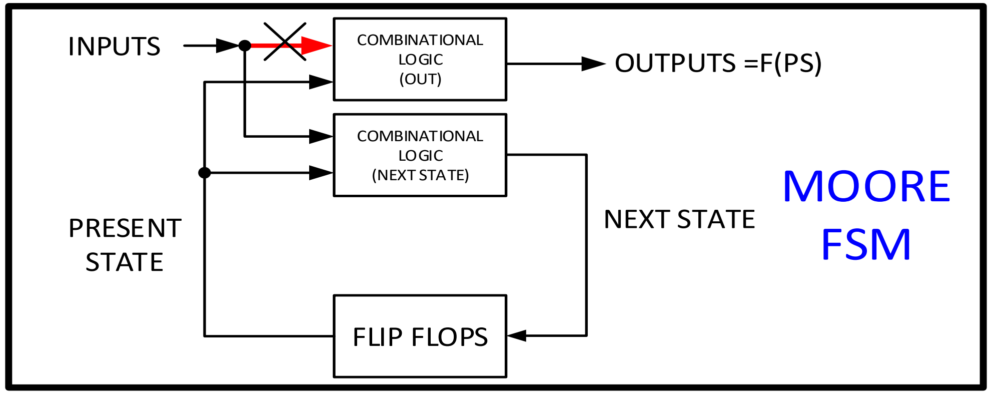
- מערכת עקיבה ממומשת ע"י מכונת מצבים סופית (Finite State-Machine, FSM) המוגדרת באמצעות מרכיביה
  - קבוצה סופית של מצבים $S=\{s_1,s_2,\ldots,s_K\}$. אחד מהם נקבע כמצב ההתחלתי. ניתן לייצג K מצבים ע"י $k=\lceil \log_2(K) \rceil$ משתני מצב בינאריים
  - קבוצה סופית של כניסות בינאריות $X=\{x_1,x_2,\ldots,x_L\}$
  - קבוצה סופית של יציאות בינאריות $Z=\{z_1,z_2,\ldots,z_M\}$
  - פונקצית מעבר $\lambda(S\times 2^X \rightarrow S)$ המגדירה לכל צירוף של מצב נוכחי s_i וערכי הכניסות $x_1,x_2,\ldots,x_L$ את המצב הבא s_i^*
  - פונקצית יציאה $\Omega_{MEALY}(S\times 2^X \rightarrow 2^Z)$ המגדירה לכל צירוף של מצב נוכחי s_i וערכי הכניסות $x_1,x_2,\ldots,x_L$ את ערכי היציאות $z_1,z_2,\ldots,z_M$
  - תזמוני כניסה t_S, t_H ותיזמוני יציאה $t_{cC\text{-}Q}$, $t_{pC\text{-}Q}$
- מכונת מצבים כזו קרויה ע"ש Mealy
- לעומתה, במכונת Moore שונה פונקצית היציאה והיא תלויה במצב הנוכחי בלבד: $\Omega_{MOORE}(S \rightarrow 2^Z)$

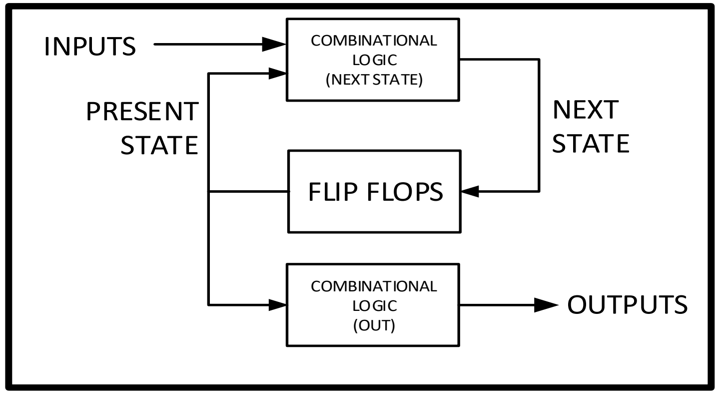
מכונת מצבים – Moore או Mealy?



מכונת מצבים – Moore או Mealy?

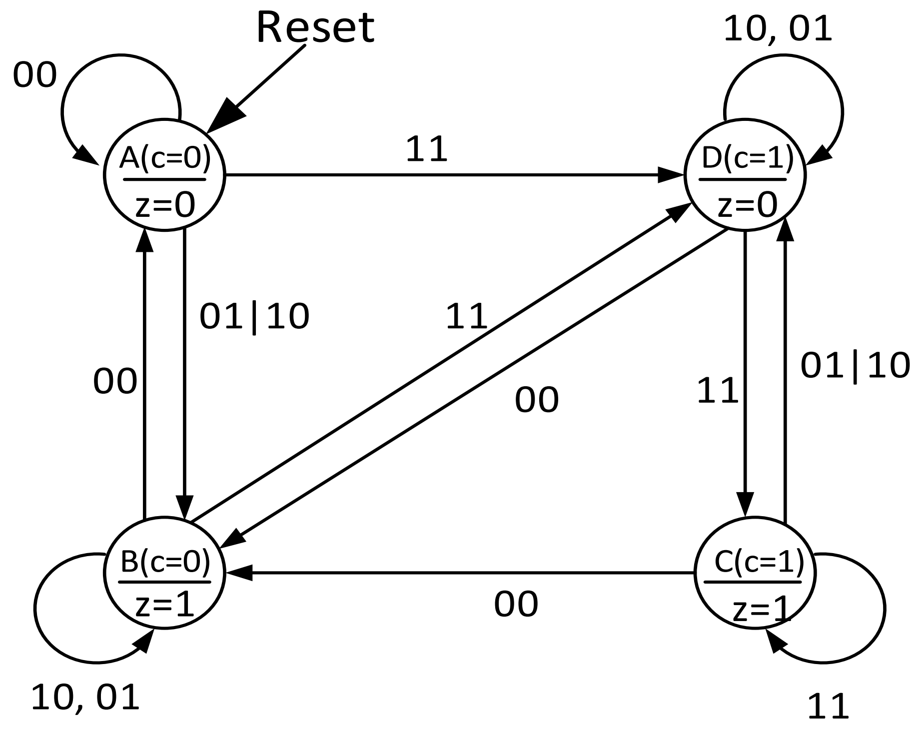
- Mealy לתיאור תיאורטי
  - המכונה מוגבלת לגילוי מלים השייכות לשפה
  - לא נשתמש לבניית מערכות ספרתיות המורכבות ממספר מכונות
  - הסיבה תובהר בהמשך (קושי בהגבלת האורך של מסלולים צירופיים)
- Moore לבניית מערכות מורכבות
  - נמנע מסלול צירופי מכניסה ליציאה

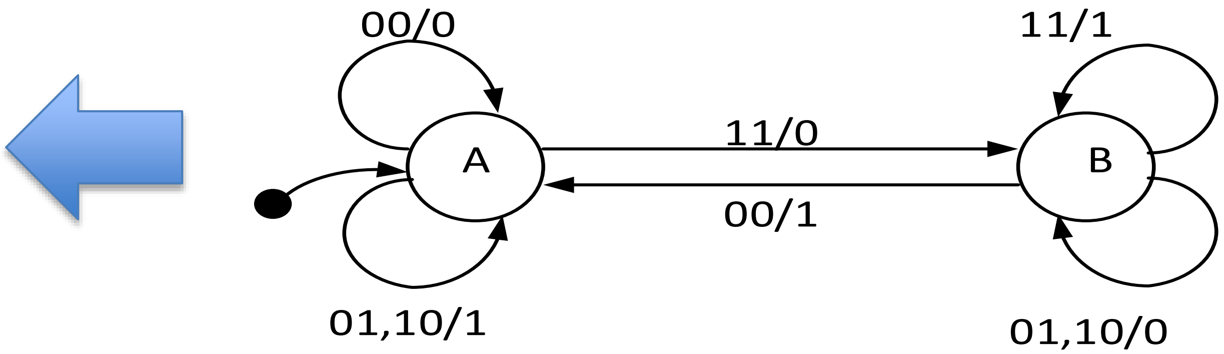




דוגמה 2 — מסכם בינארי טורי

- נתאר את המערכת כמכונת Moore במקום מכונת Mealy
  - דרושים יותר מצבים...



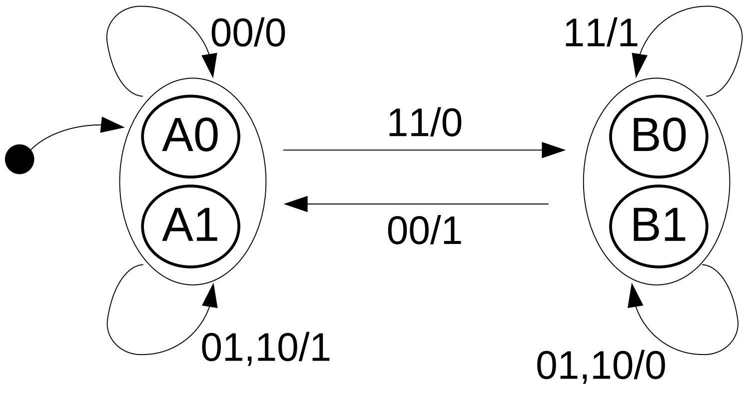


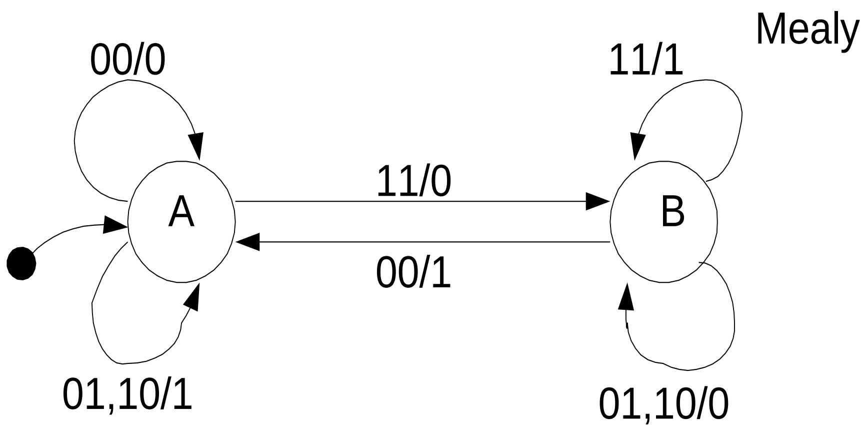
מעבר ממכונת Mealy למכונת Moore

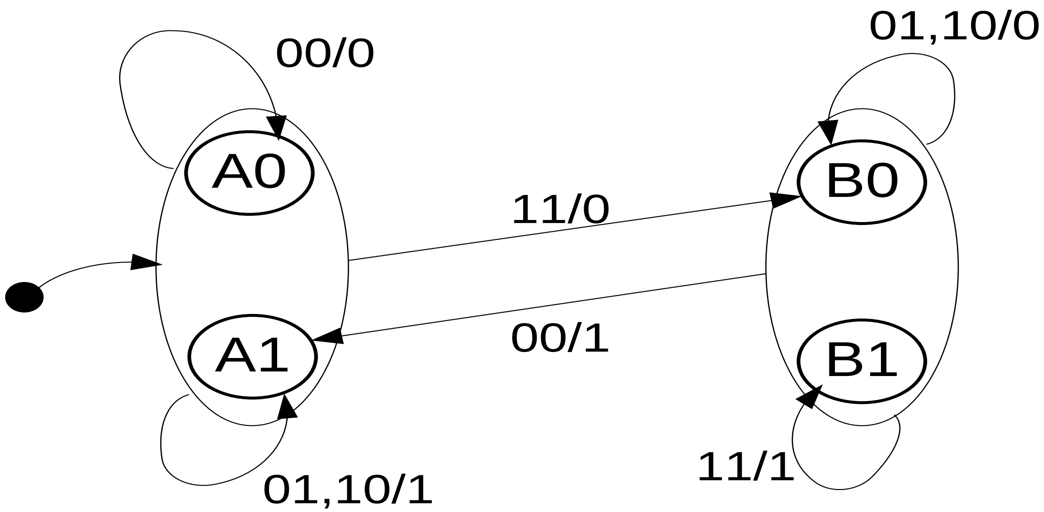
- ארבע דרכים להמיר מכונת Mealy למכונת Moore
  - תכנון מחדש כמכונת Moore
  - הוספת רגיסטרים בכניסה
  - הוספה רגיסטרים ביציאה
  - הוספת מצבים לטבלת המצבים היכן שהמוצא אינו זהה לכל הכניסות

  - דוגמאות להלן ללימוד עצמי

לימוד עצמי—מעבר ממכונת MEALY למכונת MOORE







הפרדת שני מצבים

לארבעה מצבים:

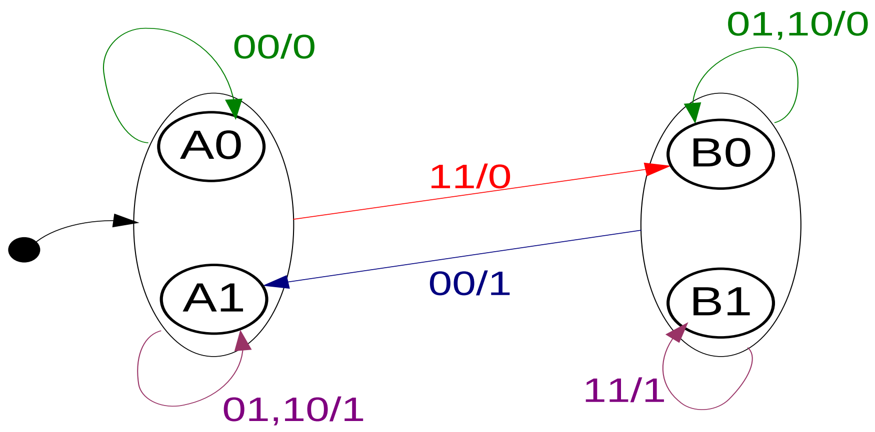
א. שכפול A ו-B

ב. התאמת מוצא לשמו

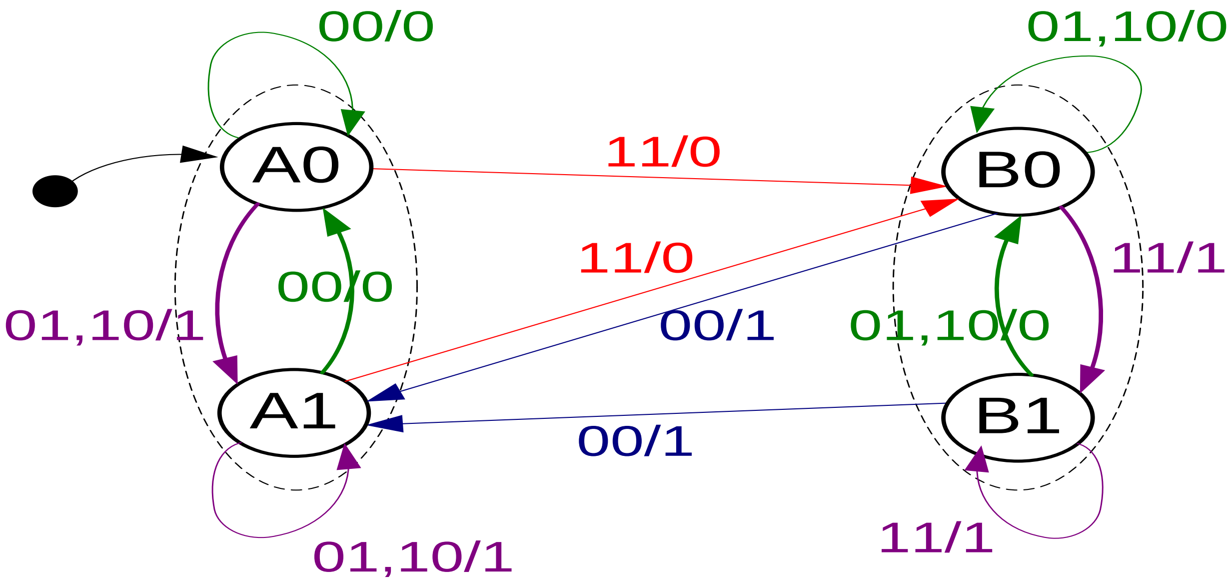
של המצב החדש אליו

החץמוביל

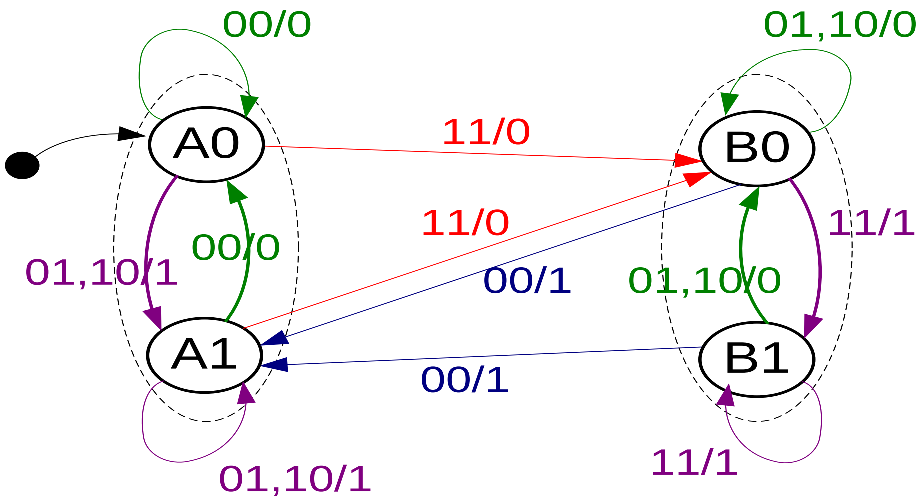
לימוד עצמי—מעבר ממכונת MEALY למכונת MOORE

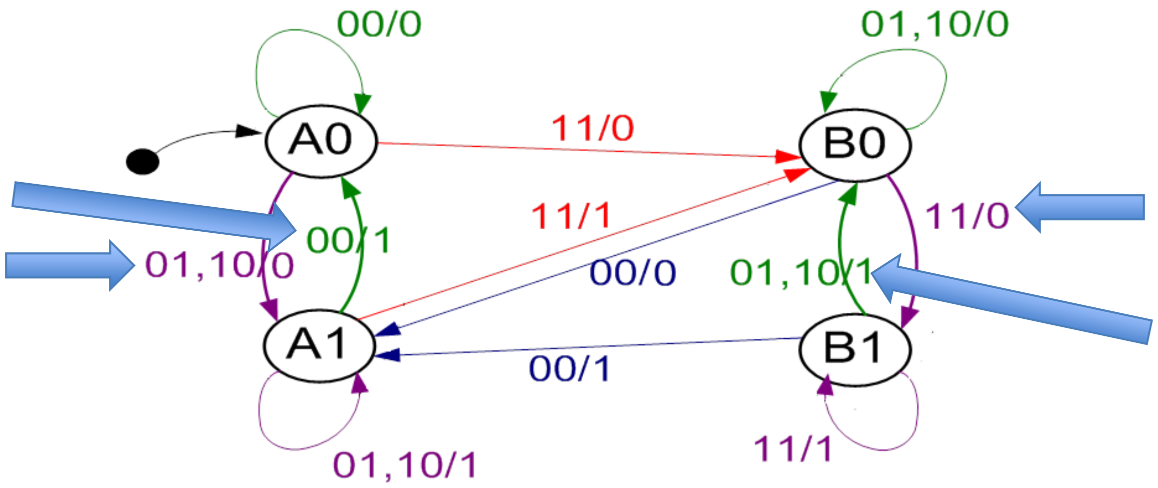


הפרדת שני מצבים
לארבעה מצבים:
ג. סיום ההפרדה



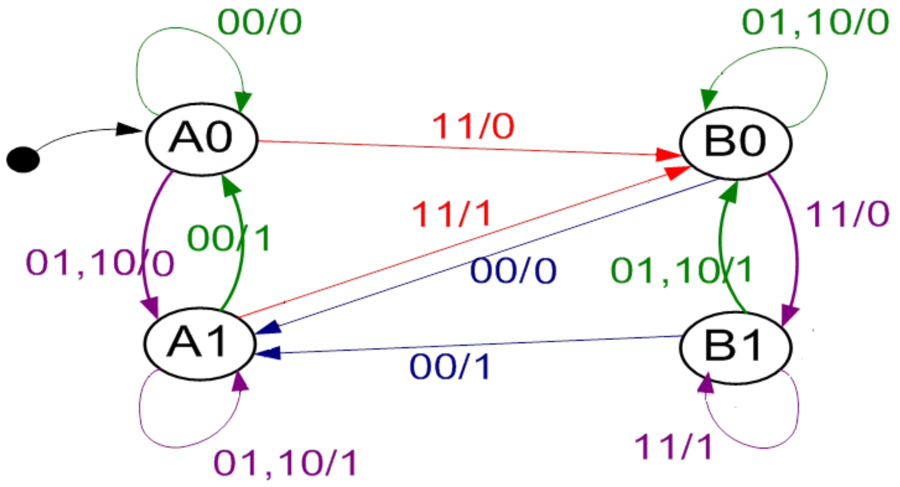
לימוד עצמי—מעבר ממכונת MEALY למכונת MOORE



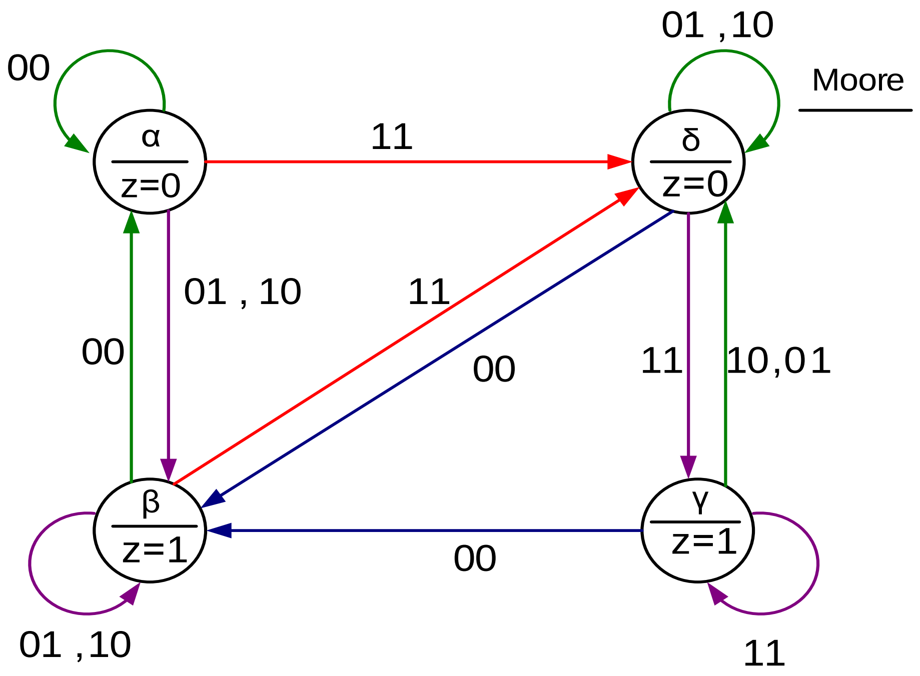


ד. הזזה בצעד זמן, כך שהמוצא במכונת Mealy החדשה קבוע בהתאם למצב (הנוכחי) ממנו יוצאים, ולא בהתאם למצב (הבא) אליו עוברים.

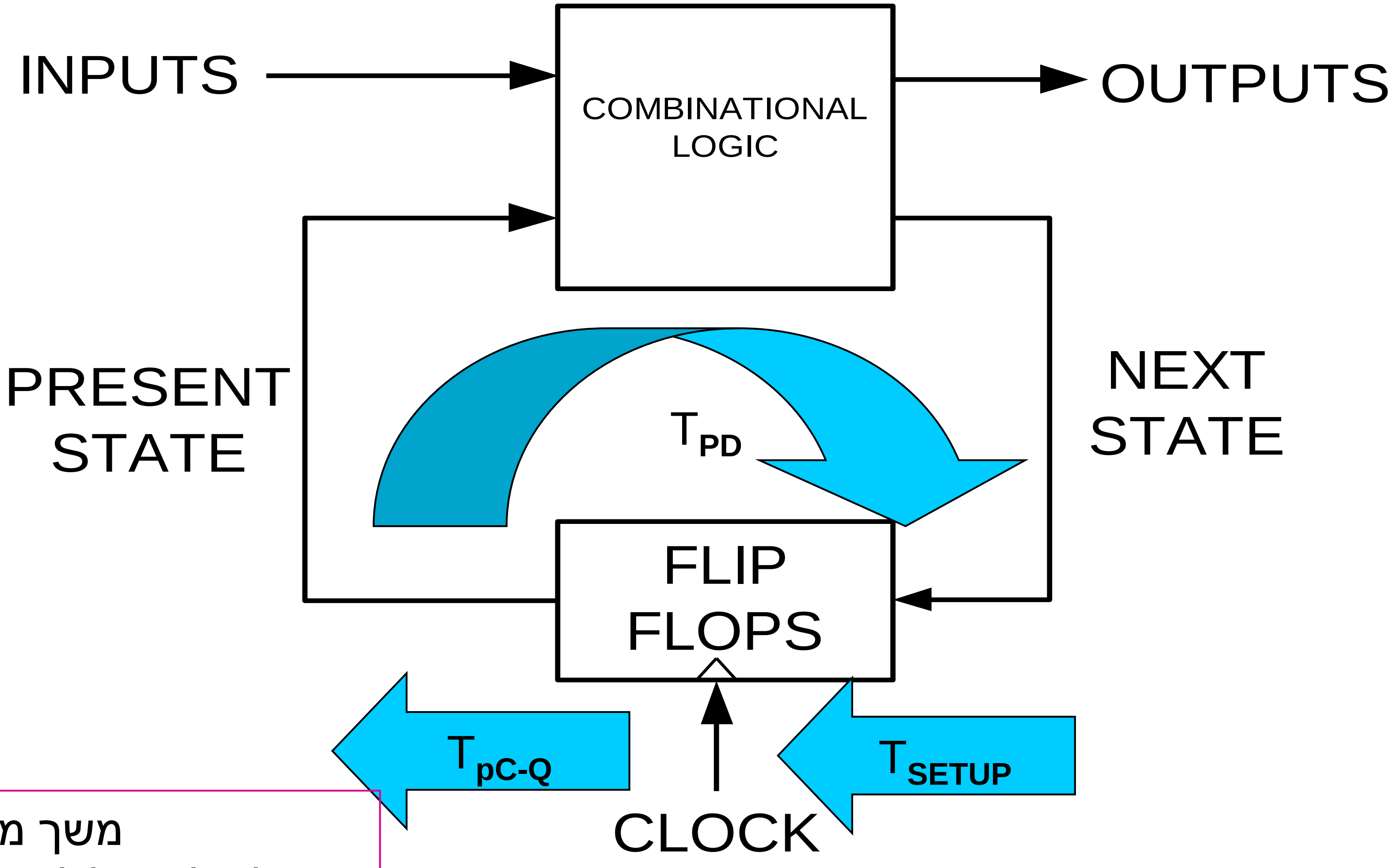
לימוד עצמי—מעבר ממכונת MEALY למכונת MOORE



ה. כעת המוצא תלוי במצב הנוכחי בלבד. קיבלנו מכונת Moore



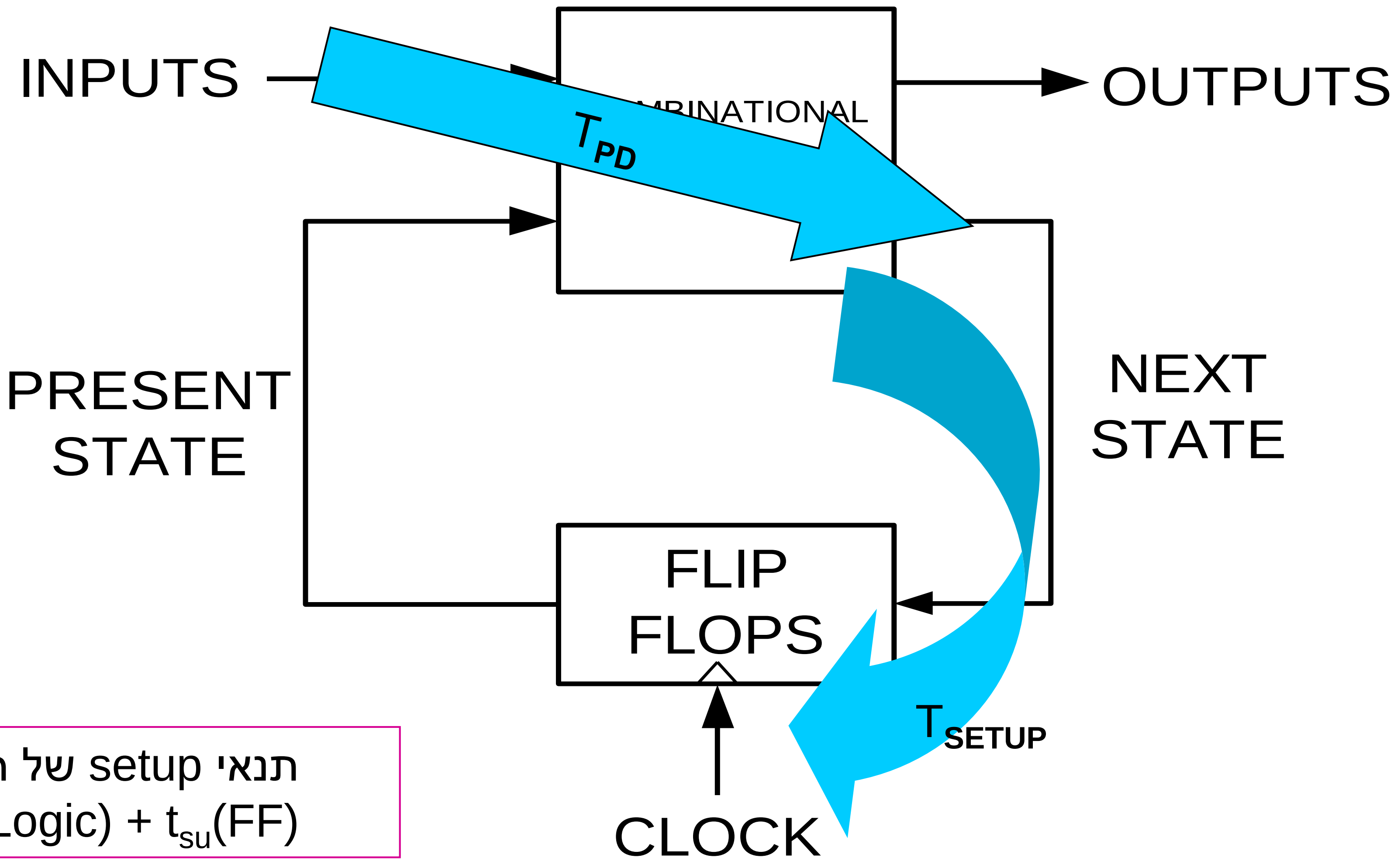
תזמון במכונת מצבים : זמן המחזור



משך מחזור השעון T:

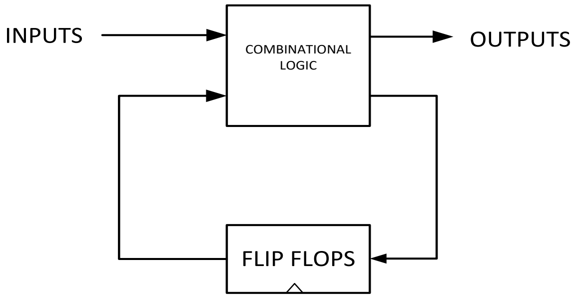
$T \geq t_{pCQ}(FF)+t_{pd}(C.Logic)+t_{su}(FF)$

תזמון במכונת מצבים : זמן SETUP



תנאי setup של הכניסה למכונה:

$t_{su}(IN) \geq t_{pd}(C.Logic) + t_{su}(FF)$



כללי התזמון

- ישנם ארבעה מסלולים, שניים שראינו: שעון → שעון, שעון → קלט, ושניים נוספים: פלט → קלט, פלט → שעון
- בכדי להבטיח פעולה תקינה של מערכת עקיבה יש להקפיד על שני כללים לכל מסלול. נרשום חלק מהם:
  - משך מחזור השעון T:

$$T \geq t_{pC-Q}+t_{pd}(C.Logic)+t_S$$

  - הכניסות למערכת הצירופית צריכות להיות תקפות בערכים הנכונים במשך $t_S(input)$:

$$t_S(input) \geq t_{pd}(C.Logic)+t_S$$

  - t_{cC-Q} + t_{cd} (C.Logic) במערכת הצירופית צריך להיות ארוך מאשר t_H של הזיכרונות
  - הכניסות למערכת הצירופית צריכות להיות תקפות בערכים הנכונים במשך $t_H(input)$ המקיים: $t_H(input) \geq t_H - t_{cd}(C.Logic)$

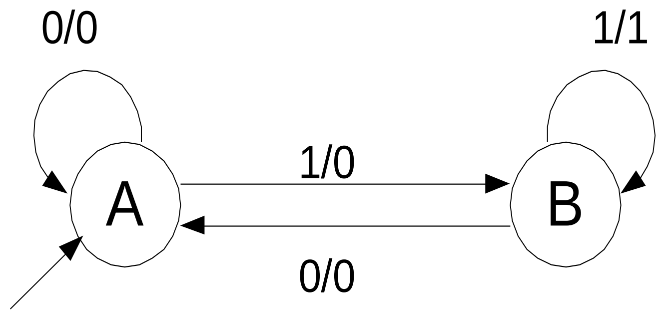
שלבי התכנון של מערכת עקיבה

1. למד את התיאור המילולי של המערכת הנדרשת
   - הבנת הדרישות: מספר כניסות, יציאות, מצבי זיכרון
2. בנה טבלת מצבים או דיאגרמת מצבים
3. (צמצם את טבלת המצבים — נלמד בהמשך)
4. בחר הקצאת מצבים ובחר רכיבי זיכרון
   - מסוג D-FF
5. רשום את טבלת המעברים ואת טבלת היציאה
6. מצא את פונקציות המעבר ופונקציות היציאה
7. שרטט את המעגל הממשש את המערכת

דוגמה 4—גלאי 11 בכניסה

- נתונה סדרת קלט $x(n)$ כאשר n הוא מונה זמן בדיד: $x(0), x(1), x(2), \ldots$
- יש לחשב סדרת פלט מתאימה $z(n)$ כך ש- $z(n)=x(n)\bullet x(n-1)$
- נגדיר "תנאי שפה" $z(0)=0$
- נצייר את דיאגרמת המצבים (מסוג Mealy). צריך לזכור רק את הערך האחרון של x

n	0	1	2	3	4	5	6	7	8	...
x	1	0	0	1	0	1	1	1	0	...
z	0	0	0	0	0	0	1	1	0	...



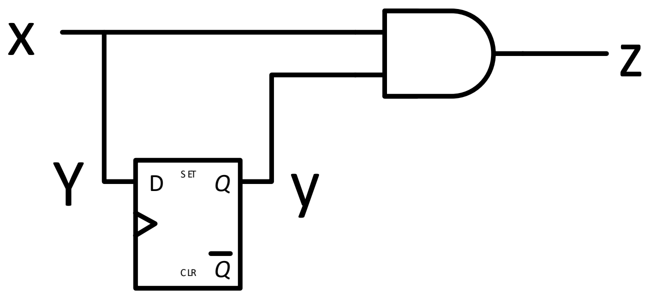
דוגמה 4—גלאי 11 בכניסה

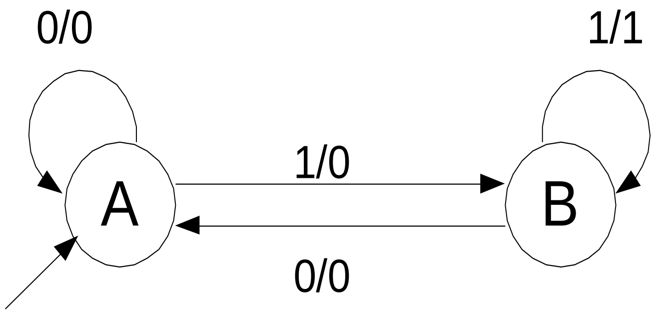
PS(y)	NS(Y),z x=0	x=1
A=0	0,0	1,0
B=1	0,0	1,1

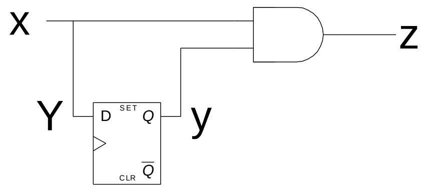
$$Y = x$$

$$z = x \cdot y$$

- נתכנן מימוש מכונת Mealy

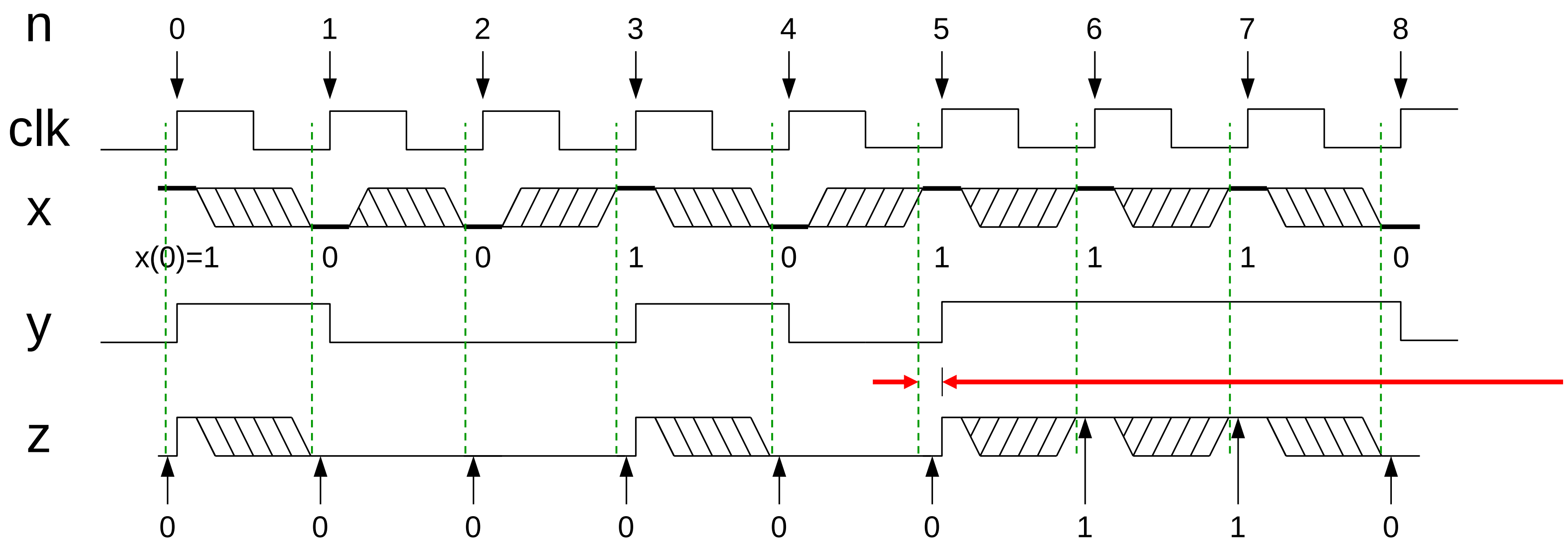






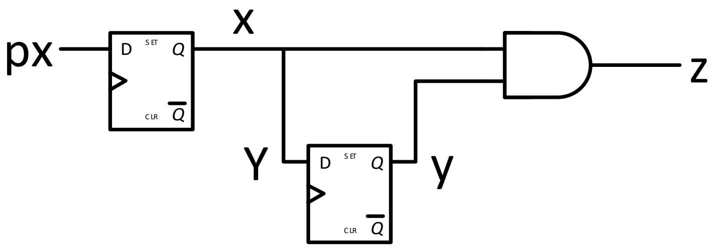
תזמון במכונת Mealy

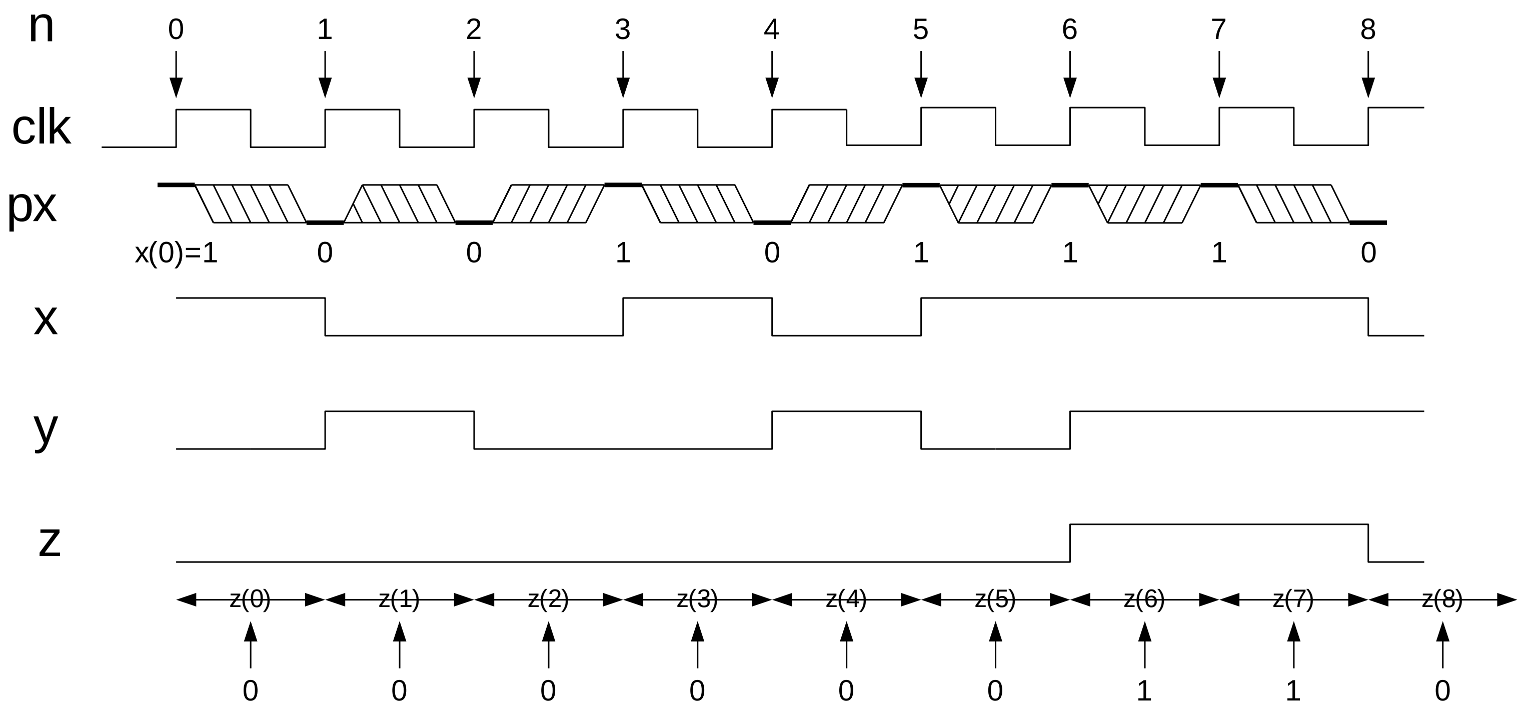
- נבחן מה קורה בכל רגע



היציאה
תקפה
רק לזמן
קצר

המרה למכונת Moore על ידי הוספת רגיסטר בכניסה

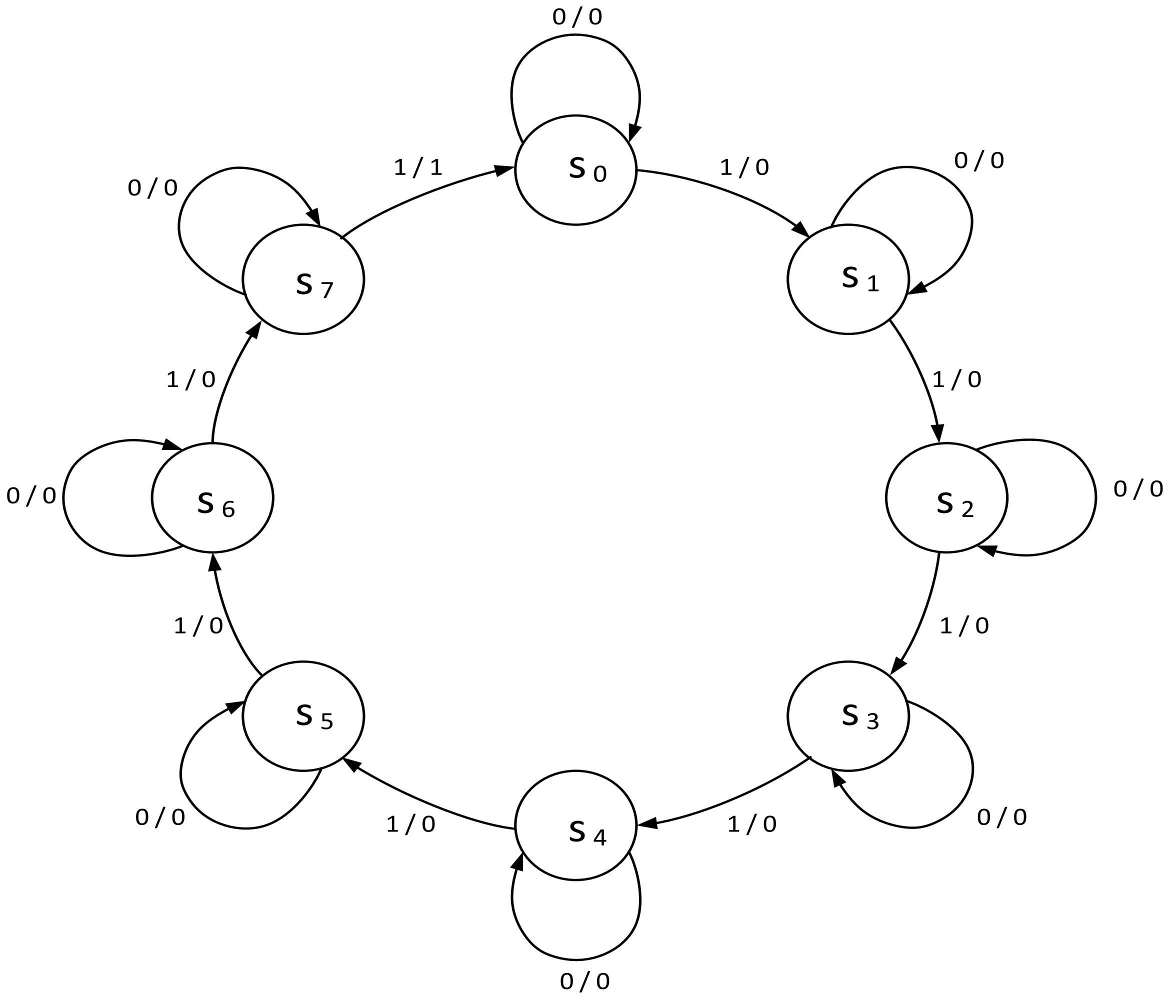




דוגמה: מונה בינרי מודולו 8 **(לימוד עצמי)**

יציאת המונה 1 עם קבלת ה- '1' השמיני, הששה עשר, העשרים ורביעי, ... בקלט x

	NS,z	
PS	x=0	x=1
S_0	S_0,0	S_1,0
S_1	S_1,0	S_2,0
S_2	S_2,0	S_3,0
S_3	S_3,0	S_4,0
S_4	S_4,0	S_5,0
S_5	S_5,0	S_6,0
S_6	S_6,0	S_7,0
S_7	S_7,0	S_0,1



דוגמה: מונה בינרי מודולו 8 (לימוד עצמי)

המימוש מצויר בהפשטה—החיבורים המפורשים מסתתרים בתוך BUS

$Y_3 = x'y_3 + y_3y_2'$
$+ y_3y_1' + xy_3'y_2y_1$

$Y_2 = x'y_2 + y_2y_1'$
$+ xy_2'y_1$

$Y_1 = x'y_1 + xy_1'$
$= x \oplus y_1$

$z = xy_3y_2y_1$