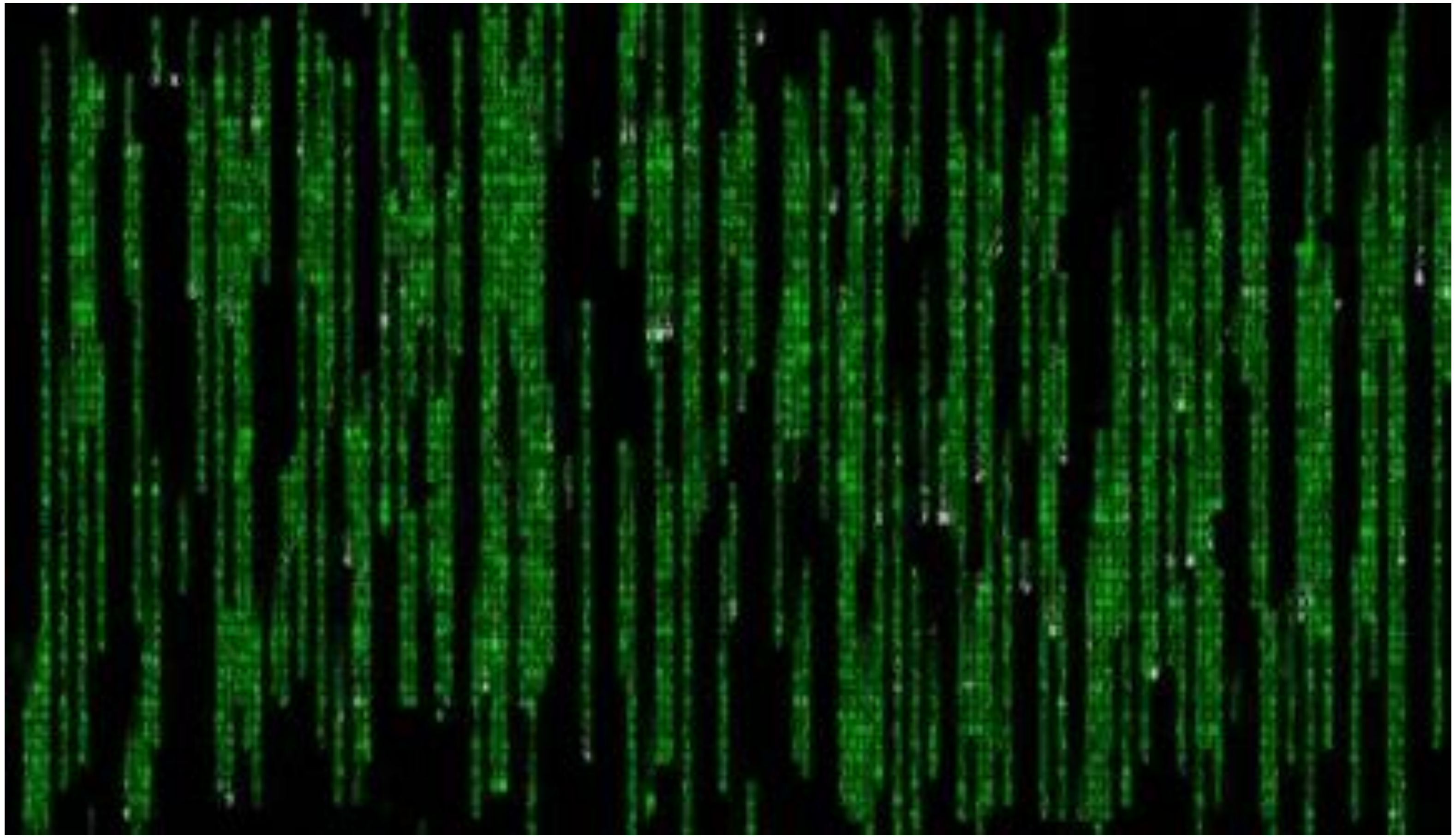
# EE 044252: Digital Systems and Computer Structure

Spring 2018

## Lecture 3: *Combinational Logic*



# 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

- Switching Function Minimization
  - Karnaugh, don’t-care
- Gates
- Switching components
  - Mux, dec, enc, sel, switch, bus
- Electrical Digital Abstraction
- Four Propagation Delay Models
  - zero, unit, physical, practical
- Switching Energy
- Gates as Switches

# ביטוי מינימלי

- המטרה: בהינתן פונקציית מיתוג $f(x_1, \dots, x_n)$, מצא ביטוי שקול ל- $f$ (מה פירוש שקול?) בצורת סכום מכפלות כך ש:
  1. מספר המכפלות יהיה מינימלי
  2. מבין הביטויים המקיימים את 1, מצא את הביטוי בעל מספר ליטרלים מינימלי
- אנו שואפים למינימום של מכפלות ולמינימום של איברים בכל מכפלה
- מדוע? האם זו האופטימיזציה הנכונה? כיצד בוחרים מטרת אופטימיזציה?
  - בהמשך הלימוד נשקול גם מטרות אחרות: מזעור זמן חישוב ומזעור האנרגיה הנדרשת לחישוב

**הערה: אין "מרשם" המבטיח הגעה לביטוי מינימלי. השיטות שתילמדנה הינן אמצעי עזר בלבד! (כמו בצמצום פולינומים...)**

# דוגמא למציאת ביטוי מינימלי

- נתונה פונקציה בצורתה הקנונית:

$$f(x,y,z)=\underset{a}{x'yz'}+\underset{b}{x'y'z'}+\underset{c}{xy'z'}+\underset{d}{x'yz}+\underset{e}{xyz}+\underset{f}{xy'z}$$

- נמזג את a,b, את b,c, את d,e, ואת e,f ונקבל (8 ליטרלים):

$$f(x,y,z)=x'z'+y'z'+yz+xz$$

  – כל מחיקת מכפלה או ליטרל בביטוי זה תיתן ביטוי שאינו שקול ל- f. לכן לא ניתן יותר לצמצם את f

- ניתן לבצע גם מיזוגים אחרים, b-a, f-c, e-d (6 ליטרלים):

$$f(x,y,z)=x'z'+xy'+yz$$

- וכן מיזוג d-a, c-b, f-e (6 ליטרלים):

$$f(x,y,z)=x'y+y'z'+xz$$

- <u>ביטוי בלתי ניתן לצמצום</u> ≡ סכום מכפלות, שמחיקת מכפלה או ליטרל ממנו יוצרת ביטוי שאיננו שקול לביטוי המקורי
- ביטוי בלתי-ניתן לצמצום **איננו** בהכרח ביטוי מינימלי
- ביטוי מינימלי **אינו** בהכרח יחיד

# צמצום פונקציות מיתוג קטנות בעזרת מפת קרנו (Karnaugh)

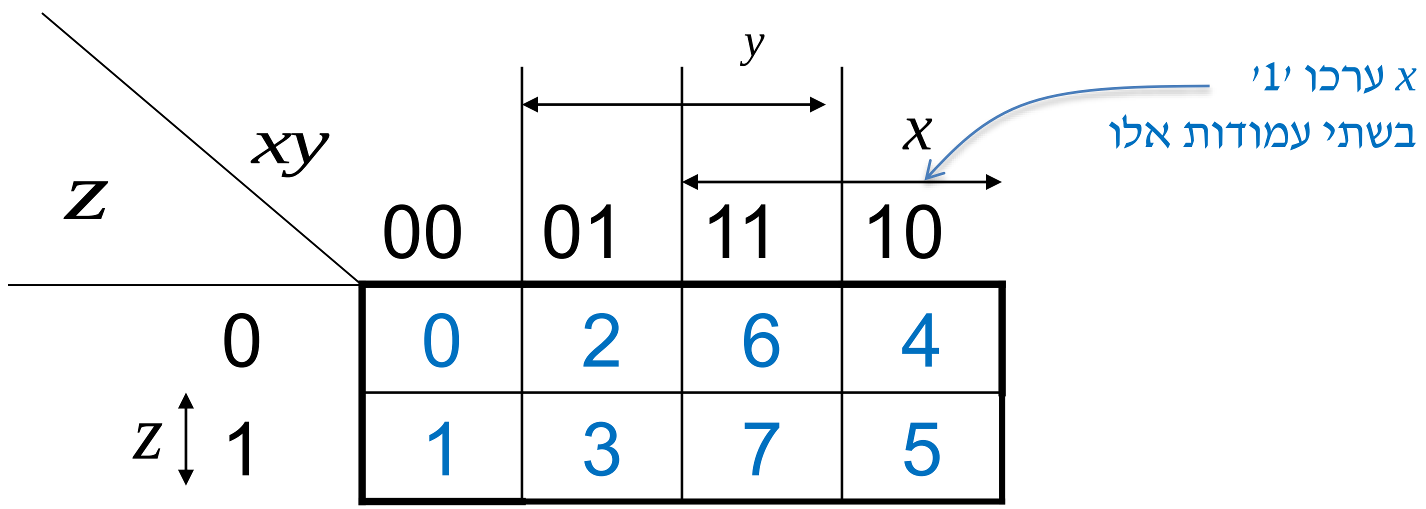
- מפת קרנו
  - שיטה לרישום טבלת האמת של פונקציית מיתוג
  - מקילה על מציאת ביטוי מינימלי של הפונקציה
- שיטה זו טובה למספר קטן של משתנים: 3 עד 5

# סדר העמודות במפת קרנו מאורגן לפי קוד Gray

- רק סיבית אחד משתנה בין קודים עוקבים (מרחק האמינג = 1)
- הקוד ציקלי—רק סיבית אחת משתנה בין הקוד האחרון לראשון

| N=1 | N=2 | N=3 |
|---|---|---|
| 0 | 00 | 000 |
| 1 | 01 | 001 |
| | 11 | 011 |
| | 10 | 010 |
| | | 110 |
| | | 111 |
| | | 101 |
| | | 100 |

# מפת קרנו של שלושה משתנים




- העמודות והשורות מסודרות לפי קוד Gray
- בשרטוט מסומנים תחומי האמת של כל משתנה
- המספרים במפה (0 עד 7) מציינים את מספרי השורות בטבלת האמת של הפונקציה, שהם גם מספרי המינטרמים
- הערה: המפה שלעיל "ריקה", שכן במקום כל מספר סידורי (בכחול) יש לרשום את ערך הפונקציה בשורה המתאימה בטבלת האמת

# מפת קרנו וטבלת אמת

| $z$ \ $xy$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 |

| | | | |
|---|---|---|---|
| 0 | 2 | 6 | 4 |
| 1 | 3 | 7 | 5 |

| Decimal code | x,y,z | f(x,y,z) |
|---|---|---|
| 0 | 0 0 0 | 0 |
| 1 | 0 0 1 | 0 |
| 2 | 0 1 0 | 1 |
| 3 | 0 1 1 | 0 |
| 4 | 1 0 0 | 1 |
| 5 | 1 0 1 | 0 |
| 6 | 1 1 0 | 1 |
| 7 | 1 1 1 | 1 |

# מפת קרנו של ארבעה משתנים

| $yz \backslash wx$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

# דוגמא לרישום מפת קרנו של 4 משתנים

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | x | | |
| | | | | | w | |
| | | yz \ wx | 00 | 01 | 11 | 10 |
| | | 00 | | 1 | 1 | 1 |
| | z | 01 | | 1 | 1 | |
| y | | 11 | | | 1 | |
| | | 10 | | | 1 | |

$$f=\sum(4,5,8,12,13,14,15)$$

| | | | |
|---|---|---|---|
| 0 | **4** | **12** | **8** |
| 1 | **5** | **13** | 9 |
| 3 | 7 | **15** | 11 |
| 2 | 6 | **14** | 10 |

# משבצות שכנות

הגדרה

- שתי משבצות במפת קרנו תיקראנה שכנות, אם ייצוגן (הבינרי) נבדל בסיבית אחת בלבד
  - כאשר המפה מייצגת טבלת אמת של n משתנים,
    לכל משבצת $n$ שכנים

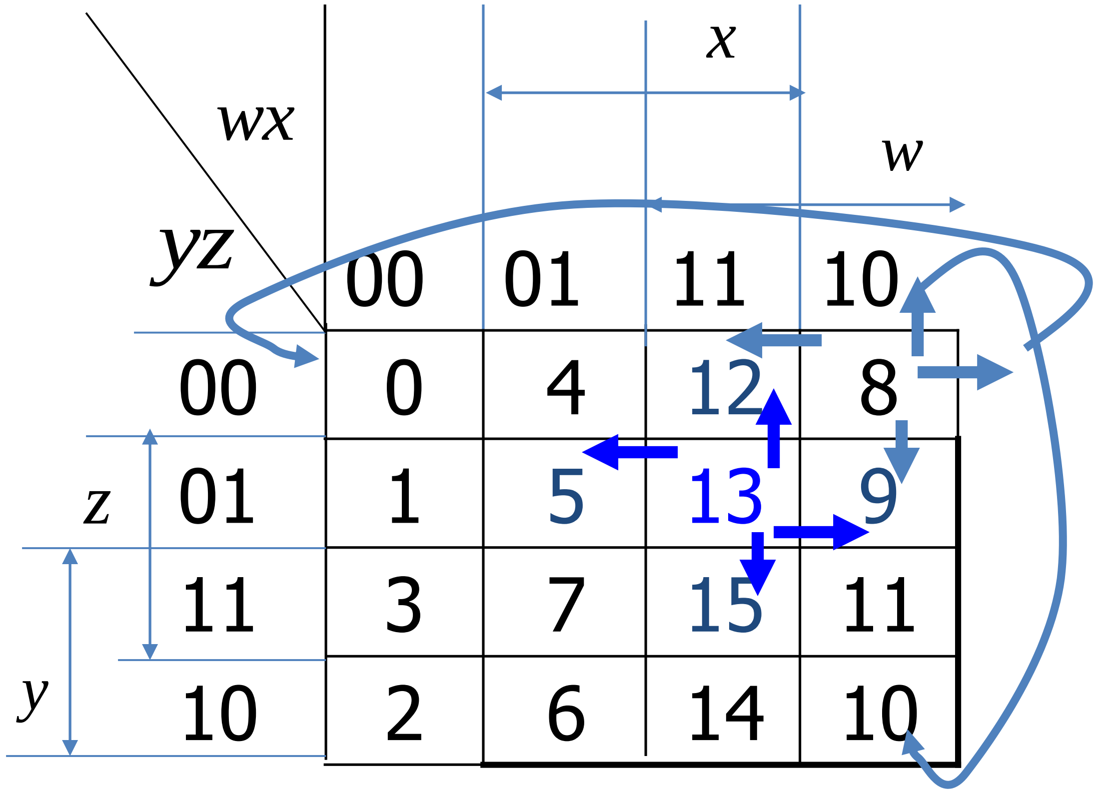
# דוגמה-משבצות שכנות במפת שלושה משתנים

- במפת קרנו של שלושה משתנים, לכל משבצת שלושה שכנים
  - השכנים של 5 הם 4,7,1
  - **השכנים** של 4?
  - הם 0,5,6

| $z \backslash xy$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | 2 | 6 | 4 |
| 1 | 1 | 3 | 7 | 5 |

# דוגמה-משבצות שכנות במפת ארבעה משתנים

- במפת קרנו של ארבעה משתנים, לכל משבצת ארבעה שכנים
  - השכנים של 13 הם 5,12,9,15
  - השכנים של 8 הם?
    - 9,12,10,0



# צמצום פונקציות בעזרת מפת קרנו

- צמצום פונקציות מבוסס על הזהות:

Ax’+Ax=A

- שתי משבצות שכנות המסומנות ב-1, המתאימות לשני מינטרמים הנבדלים בליטרל אחד, ניתנות לליכוד למכפלה של n-1 הליטרלים המשותפים

| $z \backslash xy$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | | | |
| 1 | | 1 | 1 | |

$$x'yz + xyz = yz$$

# קובייה $m$-מימדית

קובייה $m$-מימדית (במפת קרנו של $n$ משתנים) ≡ אוסף מכסימלי של $2^m$ משבצות שכולן זהות בייצוגן הבינרי ב-$n-m$ מקומות

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | | 1 |
| 01 | | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 1 | | | 1 |

m=2, 2 same bits (wx)

m=1, 3 same bits (wxy)

# משפט הקובייה

משפט : קובייה $m$-מימדית (במפת-קרנו של $n$ משתנים) מתאימה למכפלה של $m-n$ ליטרלים

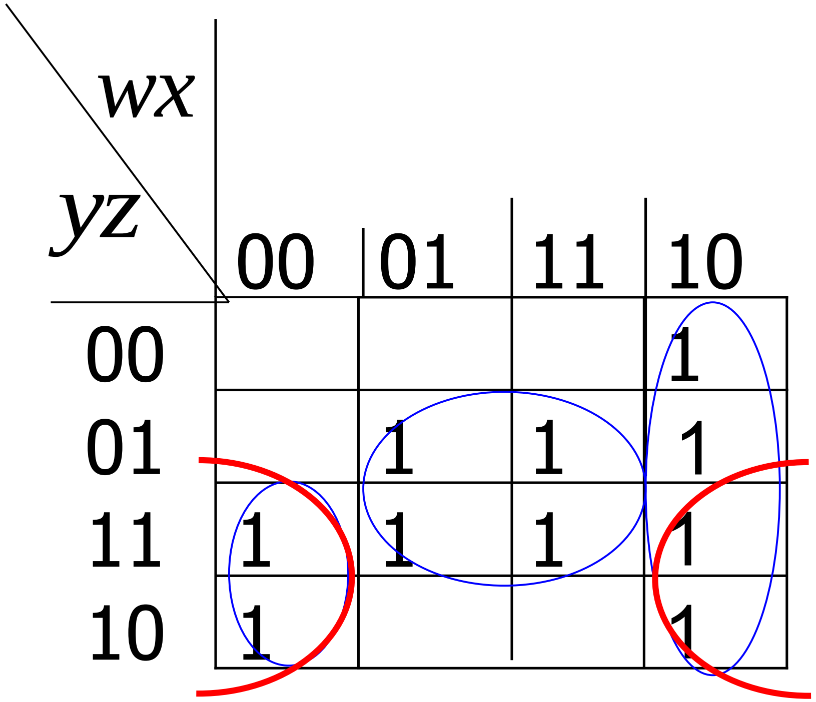
דוגמה :

| $yz$ \ $wx$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | | 1 |
| 01 | | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 1 | | | 1 |

$$f(w,x,y,z)=\sum(2,3,5,7,9,11,12,13,14,15)=w'x'y+xz+wx'$$

# מינימיזציה באמצעות מפת קרנו

- מטרת המינימיזציה (במובן של מספר ליטרלים מינימלי) : כיסוי ה-1-ים במפה על ידי מינימום קוביות <u>גדולות ככל האפשר</u>



# דוגמא של מינימיזציה באמצעות מפת קרנו

$$f(w,x,y,z)=\sum(0,4,5,7,8,9,13,15)$$

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | | 1 |
| 01 | | 1 | 1 | 1 |
| 11 | | 1 | 1 | |
| 10 | | | | |

$$f(w,x,y,z)=x'y'z'+w'xy'+wy'z+xz$$

או

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | | 1 |
| 01 | | 1 | 1 | 1 |
| 11 | | 1 | 1 | |
| 10 | | | | |

$$f(w,x,y,z)=w'y'z'+wx'y'+xz$$

מי משני הביטויים קטן יותר?

# פונקציות מינימליות ותכונותיהן

- תהיינה $f$ ו-$g$ שתי פונקציות מיתוג ב-$n$ משתנים. נאמר ש-$f$ <u>מכסה</u> (*covers*) את $g$ (נרשום $g \subseteq f$) אם

$$g(x_1, \dots, x_n) = 1 \quad \Rightarrow f(x_1, \dots, x_n) = 1$$

- כלומר, בטבלת האמת של $f$ יש 1 בכל שורה בה בטבלת האמת של $g$ יש 1

- הגדרה: אם $g \subseteq f$ ו-$g$ היא **מכפלת ליטרלים**, אזי $g$ נקרא גורר (Implicant) של $f$ ונסמן $f \longrightarrow g$

- הגדרה: גורר **ראשוני** (**Prime** Implicant, PI): גורר שכל השמטת ליטרל ממנו יוצרת מכפלה שאיננה מכוסה ע"י הפונקציה

- הגדרה: גורר ראשוני **חיוני** (**Essential** Prime Implicant, EPI): גורר ראשוני המכסה מינטרם שאיננו מכוסה ע"י אף גורר ראשוני אחר

# פונקציות מינימליות ותכונותיהן—הדגמה

<table>
<tr><th>פונקציה</th><th>מפת קרנו</th><th>ביטוי<br>כסכום מכפלות</th><th>כיסוי</th><th>גורם<br>implicant</th><th>גורם ראשוני<br>(PI)</th></tr>
<tr>
<td>$f=\sum(1,2,3,5,7)$</td>
<td><table><tr><th>z \ xy</th><th>00</th><th>01</th><th>11</th><th>10</th></tr><tr><td>0</td><td></td><td>1</td><td></td><td></td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table></td>
<td>$f(x,y,z)=x'y+xz+y'z$<br>(לא ביטוי מינימלי)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$h=\sum(2,3)$</td>
<td><table><tr><th>z \ xy</th><th>00</th><th>01</th><th>11</th><th>10</th></tr><tr><td>0</td><td></td><td>1</td><td></td><td></td></tr><tr><td>1</td><td></td><td>1</td><td></td><td></td></tr></table></td>
<td>$h=x'y$</td>
<td>$f \supseteq h$<br>$f$ מכסה את $h$</td>
<td>$h \rightarrow f$<br>$h$ is implicant of $f$<br>$h$ implies $f$</td>
<td>$h$ PI of $f$</td>
</tr>
<tr>
<td>$k=\sum(3)$</td>
<td><table><tr><th>z \ xy</th><th>00</th><th>01</th><th>11</th><th>10</th></tr><tr><td>0</td><td></td><td></td><td></td><td></td></tr><tr><td>1</td><td></td><td>1</td><td></td><td></td></tr></table></td>
<td>$k=x'yz$</td>
<td>$f \supseteq k$</td>
<td>$k \rightarrow f$</td>
<td>$k$ NOT<br>PI of $f$</td>
</tr>
<tr>
<td>$l=\sum(3,5)$</td>
<td><table><tr><th>z \ xy</th><th>00</th><th>01</th><th>11</th><th>10</th></tr><tr><td>0</td><td></td><td></td><td></td><td></td></tr><tr><td>1</td><td></td><td>1</td><td></td><td>1</td></tr></table></td>
<td>$l=x'yz+xy'z$</td>
<td>$f \supseteq l$</td>
<td>$l \not\rightarrow f$<br>כי $l$ לא מכפלה</td>
<td>—</td>
</tr>
</table>

# משפט

- כל סכום מכפלות בלתי ניתן לצמצום השקול לפונקציית מיתוג $f$ הינו סכום של גורדים ראשונים של $f$

- הוכחה:
  - על דרך השלילה:
  - נניח שקיים סכום מכפלות בלתי-ניתן לצמצום השקול ל- $f$ והמכיל מכפלה $k$ כך ש- $k \rightarrow f$ ושאיננה גורר ראשוני של $f$.
  - כלומר, כלומר, ניתן להשמיט ליטרל אחד או יותר מ- $k$ ולקבל מכפלה $k_0$ כך ש- $k_0 \rightarrow f$.
  - נראה ש-$f$ ניתנת לצמצום מ- $f=g+k$ ל- $f=g+k_0$.
  - זו סתירה לנתון שסכום המכפלות $f=g+k$ איננו ניתן לצמצום.

- למען השלמות הפורמלית יש לבסס את הטיעון על משפטי הגרירה להלן.

# משפטי גרירה ***

טענה 1 :

$$g \supseteq g\cdot h \quad ; \quad g+h \supseteq g$$

הוכחה:

$$g(x_1,\dots,x_n)+h(x,\dots,x_n)=1 \quad \Leftarrow \quad g(x_1,\dots x_n)=1$$

$$g(x_1,\dots x_n)=1 \quad \Leftarrow \quad g(x_1,\dots,x_n)\cdot h(x,\dots,x_n)=1$$

טענה 2 :

אם $f \supseteq g$ וגם $g \supseteq f$ אז $f = g$

הוכחה:

$$g(x_1,\dots,x_n)=1 \quad \Leftrightarrow \quad f(x_1,\dots,x_n)=1$$

# משפטי גרירה ***

טענה 3: $f \supseteq g+h$ אמ"ם $f \supseteq g$ וגם $f \supseteq h$

הוכחה:

כוון אחד: נניח שנתון $f \supseteq g$ וגם $f \supseteq h$ , צ"ל $f \supseteq g+h$

נניח ש- $g(x_1,\dots,x_n)+h(x_1,\dots,x_n)=1$

אזי אם $g(x_1,\dots,x_n)=1$ נובע מתוך $f \supseteq g$ ש- $f(x_1,\dots,x_n)=1$

ואם $h(x_1,\dots,x_n)=1$ אזי מתוך $f \supseteq h$ נובע ש- $f(x_1,\dots,x_n)=1$

מכאן ש- $f \supseteq g+h$

כוון שני:

נניח כי- $f \supseteq g+h$ (*) ונראה כי $f \supseteq g$ (ולכן גם $f \supseteq h$ ).

מתוך טענה 1 $g+h \supseteq g$ , ולכן אם $g(x_1,\dots x_n)=1$ אזי גם $g(x_1,\dots,x_n)+h(x_1,\dots,x_n)=1$

מהנחה (*) נובע $f(x_1,\dots,x_n)=1$ , כלומר $f \supseteq g$ .

# *** משפט

כל סכום מכפלות בלתי ניתן לצמצום השקול לפונקציית מיתוג $f$ הינו סכום של גוררים ראשונים של $f$.

<u>הוכחה מורחבת, המבוססת על משפט הגרירה:</u>

נניח בשלילה שקיים סכום מכפלות בלתי-ניתן לצמצום השקול ל- $f$ והמכיל מכפלה $k$ כך ש- $k \to f$ ושאיננה גורר ראשוני של $f$.

כלומר, ניתן להשמיט ליטרל אחד או יותר מ- $k$ ולקבל מכפלה $k_0$ כך ש- $k_0 \to f$.

נראה ש-$f$ ניתנת לצמצום מ- $f=g+k$ ל- $f=g+k_0$ , סתירה.

1. $k_o \supseteq k$    <u>טענה 1:</u> $g \supseteq g \cdot w$
2. $g+k_o \supseteq g$ ; $g+k_o \supseteq k$
3. $g+k_0 \supseteq g+k=f$    <u>טענה 3</u>
4. $g \subseteq f$    $(f=g+k)$    <u>טענה 3</u>
5. $k_0 \subseteq f$    (נתון $k_0 \to f$)
6. $g+k_0 \subseteq f$    (נתון 4+5)    <u>טענה 3</u>
7. $f=g+k_0$    (נתון 6+3)    <u>טענה 2</u>

כלומר $f$ ניתנת לצמצום וזו סתירה. מש"ל.

ראינו כי:

- גוררים ראשוניים מתאימים לקוביות במפת-קרנו שאינן מוכלות בקוביות גדולות יותר
- ביטוי מינימלי של פונקציה הינו סכום של גוררים ראשוניים

השאלה היא: אילו גוררים ראשוניים נמצאים בביטוי המינימלי?

הגדרה:

מכפלת ליטרלים P נקראת גורר ראשוני חיוני (EPI, Essential Prime Implicant)

של פונקציית מיתוג $f$ אם

א) P הוא גורר ראשוני של $f$

ב) P מכסה לפחות מינטרם אחד של $f$ שאיננו מכוסה על ידי אף גורר ראשוני אחר

ביטוי מינימלי חייב להכיל את כל ה- EPI

# קבלת ביטוי מינימלי לפונקציה

1) מצא את כל הגורמים הראשוניים (PI) של $f$

2) מצא מתוכם את כל הגורמים הראשוניים החיוניים (EPI) והכנס אותם לביטוי

3) הוצא מרשימת ה- PI את כל ה- EPI וכן את כל ה-PI שכבר כוסו על ידי ה-EPI

4) אם קבוצת ה-EPI מכסה את $f$ סיימנו. אחרת:

5) בחר PI נוספים כך ש- $f$ תכוסה כולה וכן שמספרם יהיה מינימלי, ומבין כן הסכומים המקיימים תנאי זה בחר את בעל מספר הליטרלים הקטן ביותר

# דוגמה 1 : גוררים ראשוניים הכרחיים

$$f=\sum(4,5,8,12,13,14,15)$$

| $yz \backslash wx$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | 1 | 1 | 1 |
| 01 | | 1 | 1 | |
| 11 | | | 1 | |
| 10 | | | 1 | |

גוררים ראשוניים: $xy'\ ,\ wx\ ,\ wy'z'$

כולם EPI

# דוגמא 2

$$f = \sum(0,2,3,4,5,7)$$

$xy$

| $z \backslash xy$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 1 | 1 | | 1 |
| 1 | | 1 | 1 | 1 |

גוררים ראשוניים: $x'z',\ x'y,\ yz,\ xz,\ xy',\ y'z'$

אף גורר ראשוני אינו EPI, וכולם באותו גודל. זוהי <u>מפה ציקלית</u>

$$f = x'y + xz + y'z'$$

$$f = x'z' + yz + xy'$$

# צירופי ברירה (Don’t Care)

לעיתים ערך הפונקציה איננו מוגדר עבור ערכים מסוימים של משתנים. למשל,

- צירופי כניסה מסוימים לא ייתכנו, או
- ערך היציאה עבור צירופי כניסה כאלה אינו מעניין

צירופי כניסות כאלה ייקרא **צירופי ברירה** (Don’t-care combinations)

היציאה עבורם תסומן ב-Ø

- ניתן להחליף את Ø ב-0 או ב-1 לפי הנוחות (לצורך קבלת ביטוי מינימלי)
- כיוון שכל Ø יכול לציין 0 או 1, פונקציה עם K צירופי Don’t-care מתארת למעשה $2^K$ פונקציות שונות
- מקרב פונקציות אלה נבחר את זו הנותנת את הביטוי הפשוט ביותר (מינימלי)

# דוגמה לצירופי ברירה—קוד BCD

צמצם את הפונקציה $f$ שהקלט שלה הוא ספרה עשרונית בקוד BCD והפלט הוא $f=1$ אם הספרה מתחלקת ב-3 ללא שארית

<u>פתרון</u>

נקצה ארבעה משתני כניסה $w,x,y,z$ לציון הספרה. צירופי הכניסה 10 עד 15 אינם חוקיים. ערך הפלט יהיה 1 עבור הקלטים 0,3,6,9 :

$$f=\sum(0,3,6,9)+\sum_{\emptyset}(10,11,12,13,14,15)$$

| $yz \backslash wx$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | | $\emptyset$ | |
| 01 | | | $\emptyset$ | 1 |
| 11 | 1 | | $\emptyset$ | $\emptyset$ |
| 10 | | 1 | $\emptyset$ | $\emptyset$ |

$$f_{\min}=w'x'y'z'+x'yz+xyz'+wz$$
$$=\sum(0,3,6,9,11,13,14,15)$$

מדוע לא נכללה גם המכפלה wx בביטוי המינימלי? מהי התוצאה ללא צירופי ברירה?

# דוגמה שנייה לצירופי ברירה

צמצם את הפונקציה (מרובת היציאות) המתרגמת ספרה BCD לקוד Excess-3, $f(A)=A+3$

<u>פתרון</u> : נסמן את ארבע הכניסות w,x,y,z ואת ארבע הפונקציות $f_1 f_2 f_3 f_4$ ונקבל

| ספרה | wxyz | $f_1 f_2 f_3 f_4$ |
|---|---|---|
| 0 | 0000 | 0011 |
| 1 | 0001 | 0100 |
| 2 | 0010 | 0101 |
| 3 | 0011 | 0110 |
| 4 | 0100 | 0111 |
| 5 | 0101 | 1000 |
| 6 | 0110 | 1001 |
| 7 | 0111 | 1010 |
| 8 | 1000 | 1011 |
| 9 | 1001 | 1100 |

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | ∅ | 1 |
| 01 | | 1 | ∅ | 1 |
| 11 | | 1 | ∅ | ∅ |
| 10 | | 1 | ∅ | ∅ |

$f_1$

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | 1 | ∅ | |
| 01 | 1 | | ∅ | 1 |
| 11 | 1 | | ∅ | ∅ |
| 10 | 1 | | ∅ | ∅ |

$f_2$

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | ∅ | 1 |
| 01 | | | ∅ | |
| 11 | 1 | 1 | ∅ | ∅ |
| 10 | | | ∅ | ∅ |

$f_3$

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | ∅ | 1 |
| 01 | | | ∅ | |
| 11 | | | ∅ | ∅ |
| 10 | 1 | 1 | ∅ | ∅ |

$f_4$

$$f_1 = w + xz + xy$$

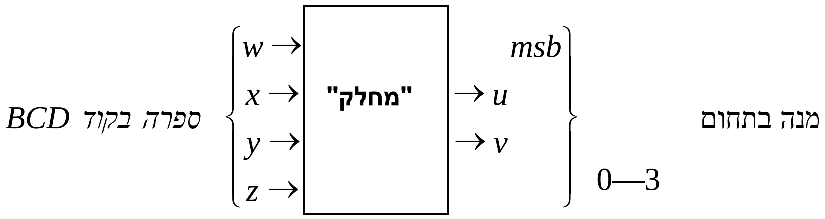
$$f_2 = x'z + x'y + xy'z'$$

$$f_3 = y'z' + yz$$

$$f_4 = z'$$

# דוגמה שלישית לצירופי ברירה

צמצם את הפונקציה שהקלט שלה הוא ספרה עשרונית בקוד BCD והפלט הוא מנת החלוקה של הספרה ב-3




צרופי הכניסה 10 עד 15 **אינם חוקיים**

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | ∅ | 1 |
| 01 | | | ∅ | 1 |
| 11 | | 1 | ∅ | ∅ |
| 10 | | 1 | ∅ | ∅ |

$u$

| yz \ wx | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | 1 | ∅ | |
| 01 | | 1 | ∅ | 1 |
| 11 | 1 | | ∅ | ∅ |
| 10 | | | ∅ | ∅ |

$v$

$$u = \sum(6,7,8,9) + \sum_{\phi}(10,11,12,13,14,15)$$

$$v = \sum(3,4,5,9) + \sum_{\phi}(10,11,12,13,14,15)$$

במימוש המינימלי:

$$u_{min} = w + xy$$

$$v_{min} = wy' + wz + x'yz$$

# סיכום: מינימיזציה באמצעות מפות קרנו הכוללות משתני ברירה (don't care)

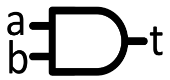
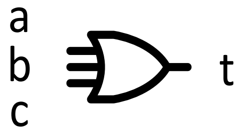
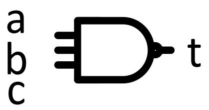
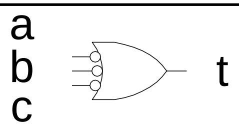
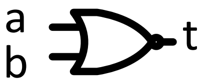
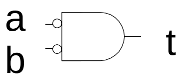
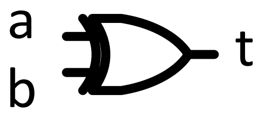
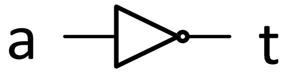
- הנח כי כל משתני הברירה (Φ) הם '1'
- במפה המתקבלת, סמן את כל ה-PI
  - התעלם מ-PI המכסים אך ורק משתני ברירה
- כל PI המכסה מינטרם ('1') מקורי (לא Φ מקורי) באופן בלעדי יחשב EPI
- בחר את כל ה-EPI, וכן אוסף מינימלי של PI, כך שכל המינטרמים המקוריים יכוסו
  - קבוצת ה-EPI וה-PI שבחרת תהווה את הפונקציה המינימלית
- הפוך ל-'1' את משתני הברירה המכוסים ע"י הפונקציה המינימלית
- הפוך ל-'0' את שאר משתני הברירה

# Agenda

- Switching Function Minimization
  - Karnaugh, don't-care
- **Gates**
- **Switching components**
  - **Mux, dec, enc, sel, switch, bus**
- **Electrical Digital Abstraction**
- **Four Propagation Delay Models**
  - **zero, unit, physical, practical**
- **Switching Energy**
- Gates as Switches

# שערים לוגיים

- שער לוגי (Logic Gate) הוא התקן הממש פונקציית מיתוג
- אלקטרוני (ביולוגי, אופטי, ...)
- ניתן לתיאור באמצעות
  - ציור / סמל
  - טבלת אמת
  - ביטוי מיתוג
  - Verilog – שפה לתיאור חומרה
    HDL, Hardware Description Language

| | | | |
|---|---|---|---|
| שער AND: | $t = a \cdot b$ |  | |
| שער OR: | $t = a + b + c$ |  | |
| שער NAND: | $t = \overline{a \cdot b \cdot c}$ |  |  |
| שער NOR: | $t = \overline{a + b}$ |  |  |
| שער XOR: | $t = a \oplus b$ |  | |
| שער NOT (מהפך): | $t = \bar{a}$ |  | |

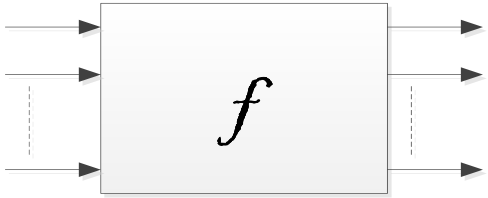
# תיאור שער בשפת Verilog

שער AND

```
module and_gate(
   input logic a,
   input logic b,
   output logic out
);
  assign out = a & b ;
endmodule
```

תיאור היררכי: שער NAND מתואר באמצעות AND

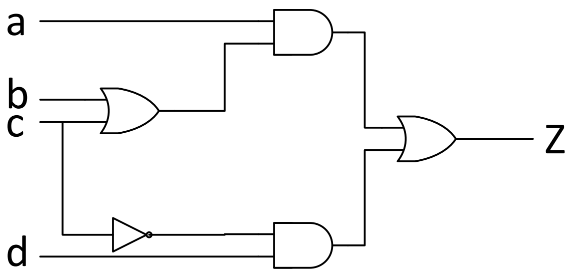
```
module nand_gate(
   input logic x, input logic y,
   output logic out
);
  logic x_and_y;
  assign out = ~x_and_y ;
  and_gate foo(.a(x), .b(y), .out(x_and_y));
endmodule
```




# רכיב צירופי combinational component

- הגדרה: רכיב צירופי הוא רכיב הממש פונקציית מיתוג בעלת התכונות הבאות:
  1. כניסה אחת או יותר של משתני מיתוג
  2. יציאה אחת או יותר של משתני מיתוג
  3. התאמה של ערך מיתוג (0 או 1) לכל יציאה בעבור כל צירוף אפשרי של ערכי הכניסות
     - טבלת אמת
  4. מגבלות תזמון (פיזיקליות)—על כך בהמשך...

# מעגל צירופי combinational circuit



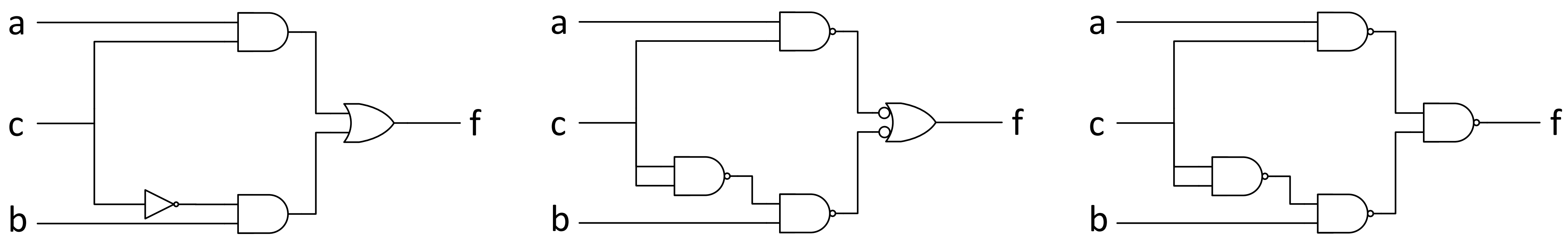

$$Z = a(b + c) + c'd$$

- הגדרה: מעגל צירופי הוא מעגל הכולל רכיבים המחוברים ביניהם, המקיים:
  1. כל רכיב במעגל הוא רכיב צירופי
  2. כל כניסה לרכיב במעגל היא או כניסה למעגל או יציאה של רכיב אחר במעגל
  3. יציאות של רכיבים במעגל יכולות להתחבר רק לכניסות של רכיבים אחרים במעגל **(ולא ליציאות אחרות)**
  4. כל מסלול במעגל העובר דרך רכיביו בכיוון מן הכניסה אל היציאה עובר דרך כל רכיב לכל היותר פעם אחת (כלומר אין "חוגים" - cycles)
     - בהמשך הקורס נלמד גם על מעגלים עם חוגים. אלו **לא יהיו** מעגלים צירופיים
  5. כיציאות המעגל ניתן לבחור כל כניסה למעגל וכל יציאה של רכיב במעגל
- המעגל קרוי צירופי כי יציאותיו תלויות אך ורק בצירוף הערכים בכניסותיו
- מסקנה: מעגל בו יש רכיב שיציאתו מחוברת לאחת מכניסותיו **אינו** מעגל צירופי

- מה משמעות השם "מעגל" אם אין פה מעגלים / חוגים סגורים?

# תכן לוגי (שלב ראשון): מימוש פונקציה כמעגל צירופי

- ממש את הפונקציה $f(a,b,c)=\sum(2,5,6,7)=ac+bc'$




- שלושת המעגלים מממשים את אותה פונקציה
- דה מורגן מסייע לנו לתרגם לשער אוניברסלי
  - כאן NAND בעל שתי כניסות בלבד

# תכן לוגי - יעדים

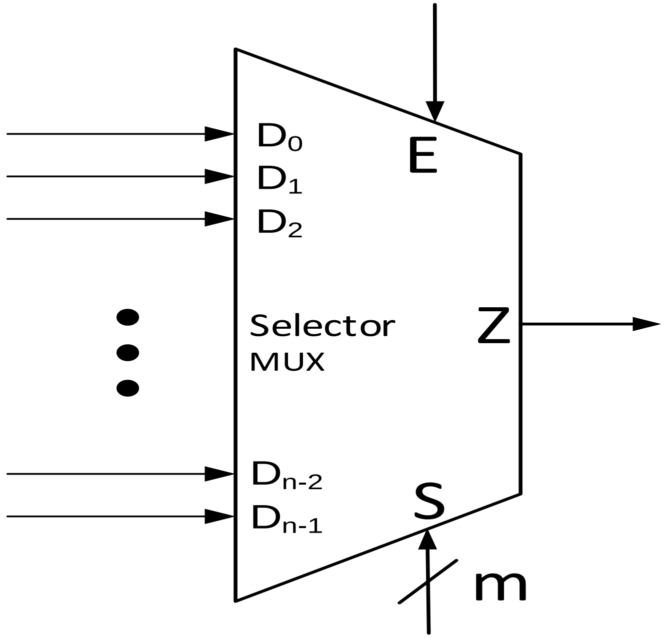
יעדי התכן הלוגי **משתנים** עם השתנות הטכנולוגיה

- בשנים הראשונות של התכן הלוגי (עד שנות הששים)
  - המימוש נעשה באמצעות שערים בודדים
  - יעד התכן: מימוש פונקציות מיתוג בעזרת מספר מינימאלי של שערים
- מאוחר יותר (עד שנות השמונים)
  - המימוש נעשה באמצעות רכיבים מורכבים יותר הכוללים מספר רב של שערים (נלמד עליהם בהמשך)
  - יעד התכן: מימוש פונקציית מיתוג באמצעות מספר מינימאלי של רכיבים מורכבים ולאו דווקא של שערים
- שבבים מתקדמים
  - כאשר המימוש הלוגי הינו חלק מתכנון גדול, מצטרפות לתכן מטרות נוספות
  - מינימיזציה של השטח על השבב (הנקבע ע"י מספר השערים וסוגיהם)
  - מינימיזציה של ההספק החשמלי
  - מינימיזציה של זמן החישוב
  - פשטות התכנון (חסכון בכוח אדם וזמן פיתוח)

# Agenda

- Switching Function Minimization
  - Karnaugh, don’t-care
- Gates
- **Switching components**
  - **Mux, dec, enc, sel, switch, bus**
- Electrical Digital Abstraction
- Four Propagation Delay Models
  - zero, unit, physical, practical
- Gates as Switches
- Switching Energy

# Selector / multiplexer / mux – בורר




<u>קלט:</u> m כניסות **בקרה** $S_o, S_1, ..., S_{m-1}$

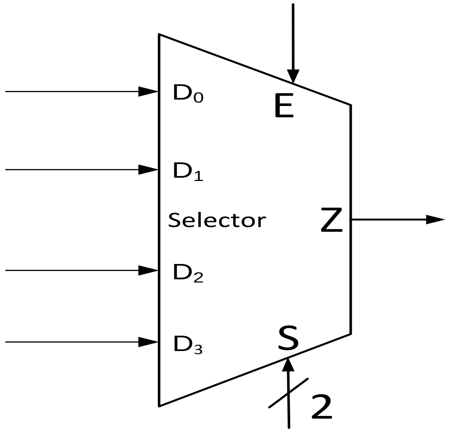
$n = 2^m$ כניסות **נתונים** $D_o, D_1, ..., D_{n-1}$

$E$ כניסת $enable$

<u>פלט:</u> אם $E=0$ אז '0'

אחרת הסיבית $D_i$

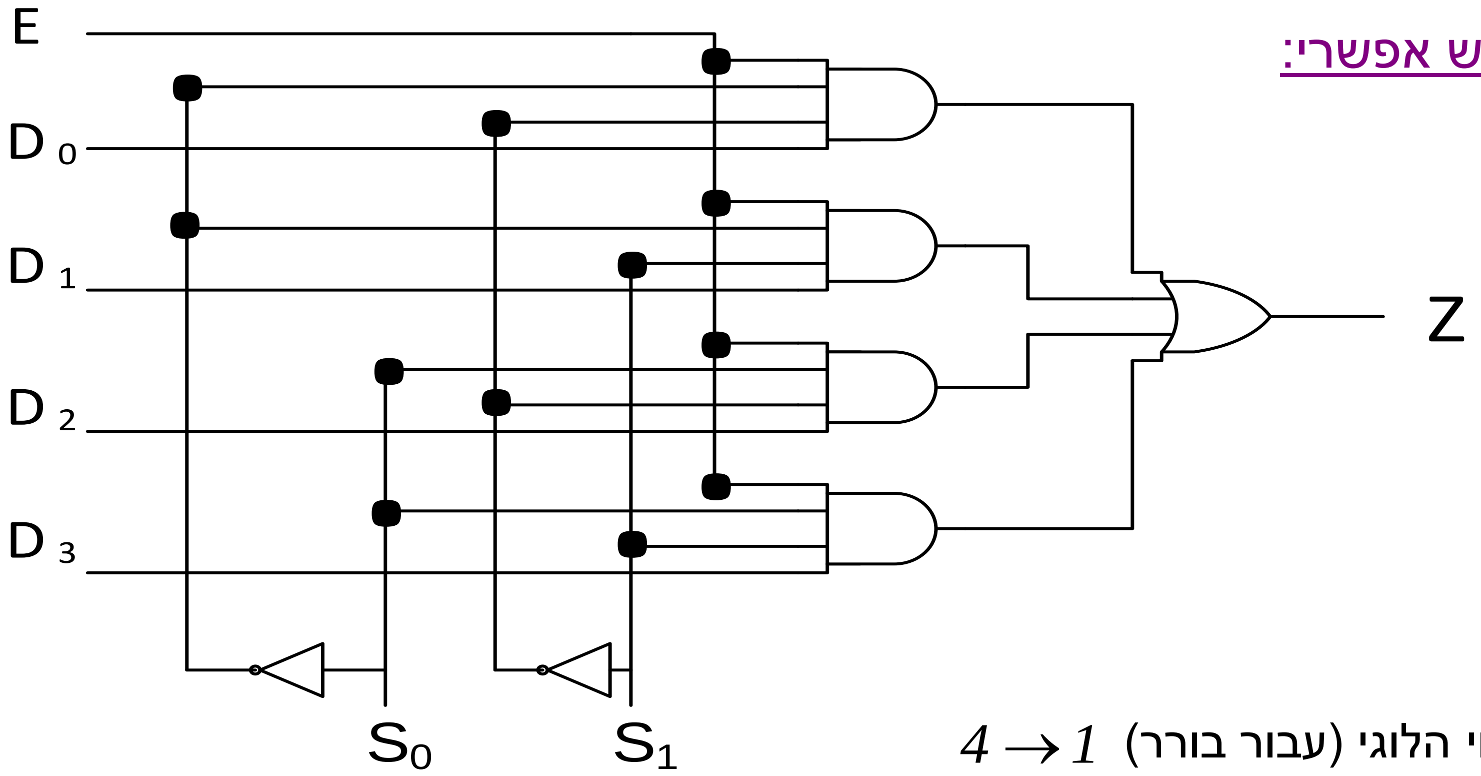
$i$ הינו המספר שיצוגו הבינרי מופיע ב-m סיביות הבקרה




# בורר

- מהו הגודל של טבלת אמת של הבורר 4 ל-1?

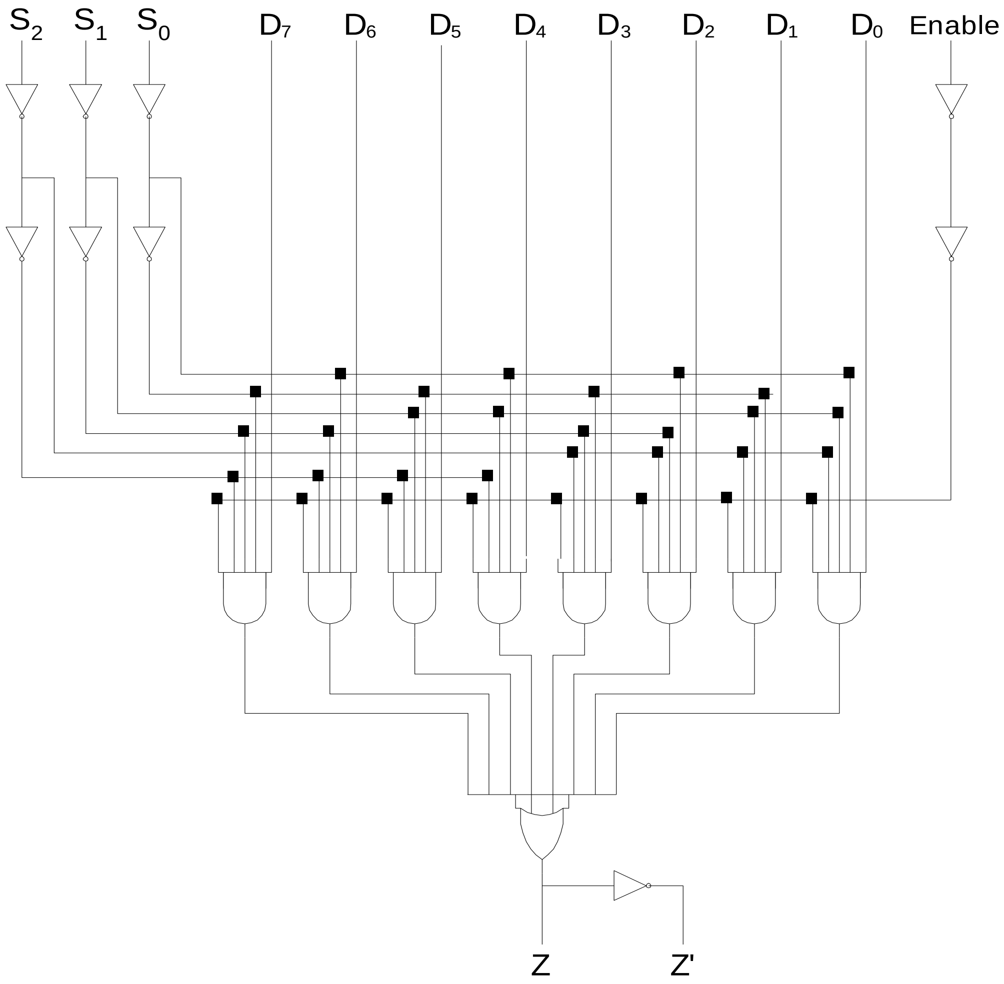
מימוש אפשרי:

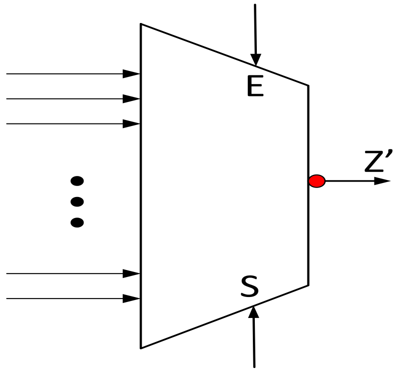



והביטוי הלוגי (עבור בורר $4 \rightarrow 1$)

$$z = E\overline{S}_o\overline{S}_1D_o + E\overline{S}_oS_1D_1 + ES_o\overline{S}_1D_2 + ES_oS_1D_3$$

# 8-input MUX and Input Decoupling



# *** ממוש פונקציות באמצעות בוררים (1)

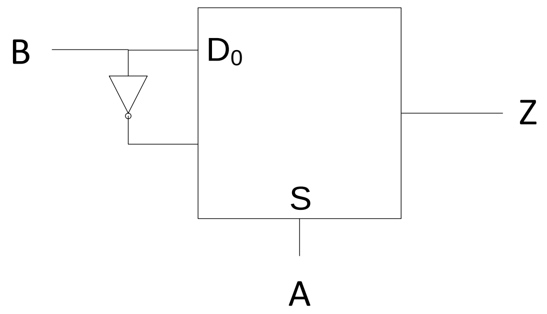
לדוגמא, עבור בורר $2 \rightarrow 1$ ללא enable, הפונקציה הממומשת היא

$$z = SD_1 + \overline{S}\ D_o$$

אם נציב $S = A$ , $D_o = B$ , $D_1 = \overline{B}$ נקבל

$$z = A\overline{B} + \overline{A}B = A \oplus B$$

קיבלנו ממוש $XOR$:

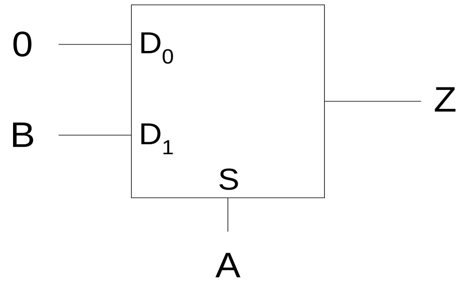


# *** מימוש פונקציות באמצעות בוררים (2)

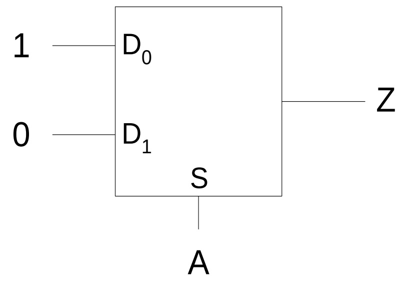
נראה שיחד עם הקבועים $0, 1$ נקבל מערכת פעולות שלמה:

אם נציב $S=A$ , $D_o=0$ , $D_1=B$ נקבל $z=AB+\overline{A}\cdot 0=A\cdot B$

קיבלנו מימוש $AND$:




וכן $\overline{A}$




בדקו בבית: האם בורר בלבד מהווה מערכת פעולות שלמה? בורר עם '0'? בורר עם '1'?

# *** ממוש פונקציות באמצעות בוררים (3)

נראה כיצד ניתן לממש פונקציה של $n$ משתנים באמצעות בוררים

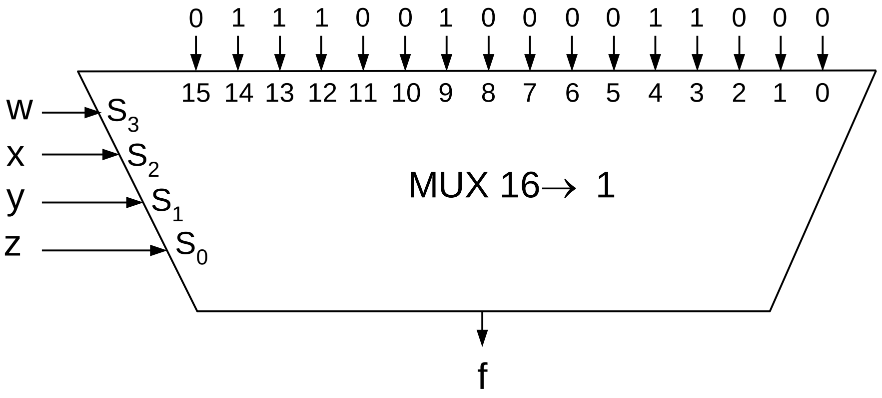
נממש לדוגמא את הפונקציה:

$$f(w,x,t,z)=\sum(3,4,9,12,13,14)$$

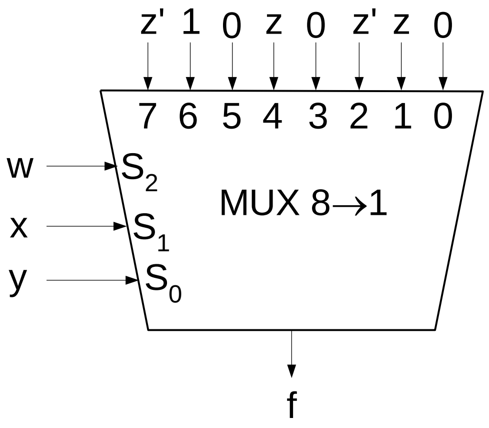
| w | x | y | z | | f | $16 \to 1$ בורר<br>$S_3 = w \quad S_1 = y$<br>$S_2 = x \quad S_o = z$ | $8 \to 1$ בורר<br>$S_2 = w \quad S_0 = y$<br>$S_1 = x$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | | 0 | $D_o = 0$ | $D_o = 0$ |
| 0 | 0 | 0 | 1 | | 0 | $D_1 = 0$ | |
| 0 | 0 | 1 | 0 | | 0 | $D_2 = 0$ | $D_1 = z$ |
| 0 | 0 | 1 | 1 | | 1 | $D_3 = 1$ | |
| 0 | 1 | 0 | 0 | | 1 | $D_4 = 1$ | $D_2 = \overline{z}$ |
| 0 | 1 | 0 | 1 | | 0 | $D_5 = 0$ | |
| 0 | 1 | 1 | 0 | | 0 | $\ldots$ | $D_3 = 0$ |
| 0 | 1 | 1 | 1 | | 0 | | |
| 1 | 0 | 0 | 0 | | 0 | | $D_4 = z$ |
| 1 | 0 | 0 | 1 | | 1 | | |
| 1 | 0 | 1 | 0 | | 0 | | $D_5 = 0$ |
| 1 | 0 | 1 | 1 | | 0 | | |
| 1 | 1 | 0 | 0 | | 1 | | $D_6 = 1$ |
| 1 | 1 | 0 | 1 | | 1 | | |
| 1 | 1 | 1 | 0 | | 1 | | $D_7 = \overline{z}$ |
| 1 | 1 | 1 | 1 | | 0 | | |

# ממוש פונקציות באמצעות בוררים (4) ***

מימוש ישיר , בורר בעל $n$ כניסות :




מימוש עם בורר בעל $n$-1 כניסות :

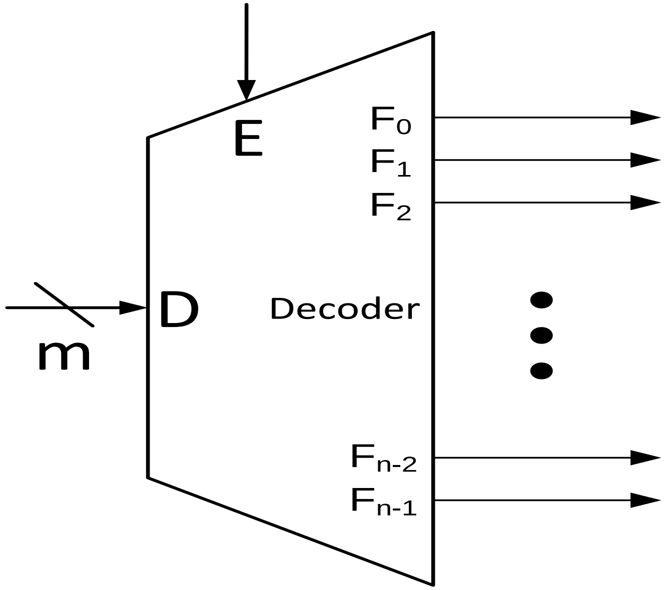


# Decoder – מפענח

קלט: m כניסות **בקרה** $\mathrm{d}_o, \mathrm{d}_1, \ldots, \mathrm{d}_{m-1}$

כניסת *Enable*

פלט: $n \le 2^m$ יציאות $\mathrm{f}_o, \mathrm{f}_1, \ldots, \mathrm{f}_{n-1}$

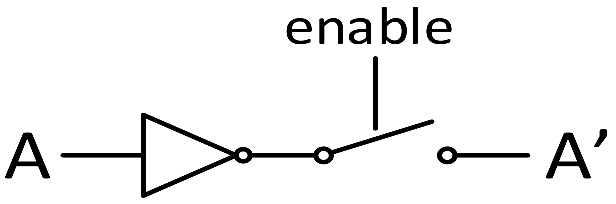


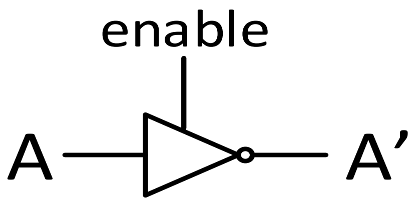

- אם *Enable=0* $f_i = 0$ לכל *i* .
- אם *Enable=1* , $f_i = 1$ כאשר *i* הוא המספר הבינרי המיוצג ע"י כניסות הבקרה, וכל שאר היציאות הן 0

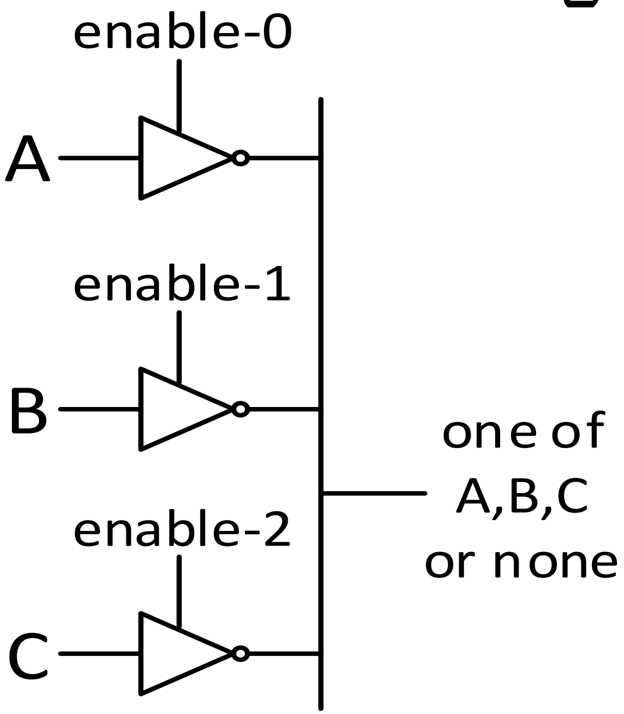
זהו מפענח $m \to n$

(אפשר להתייחס למפענח כאל *Demultiplexer* , כאשר *E* הוא בעצם כניסת המידע *D*)

# מיתוג יציאת השער באמצעות מתגים Three-State Output

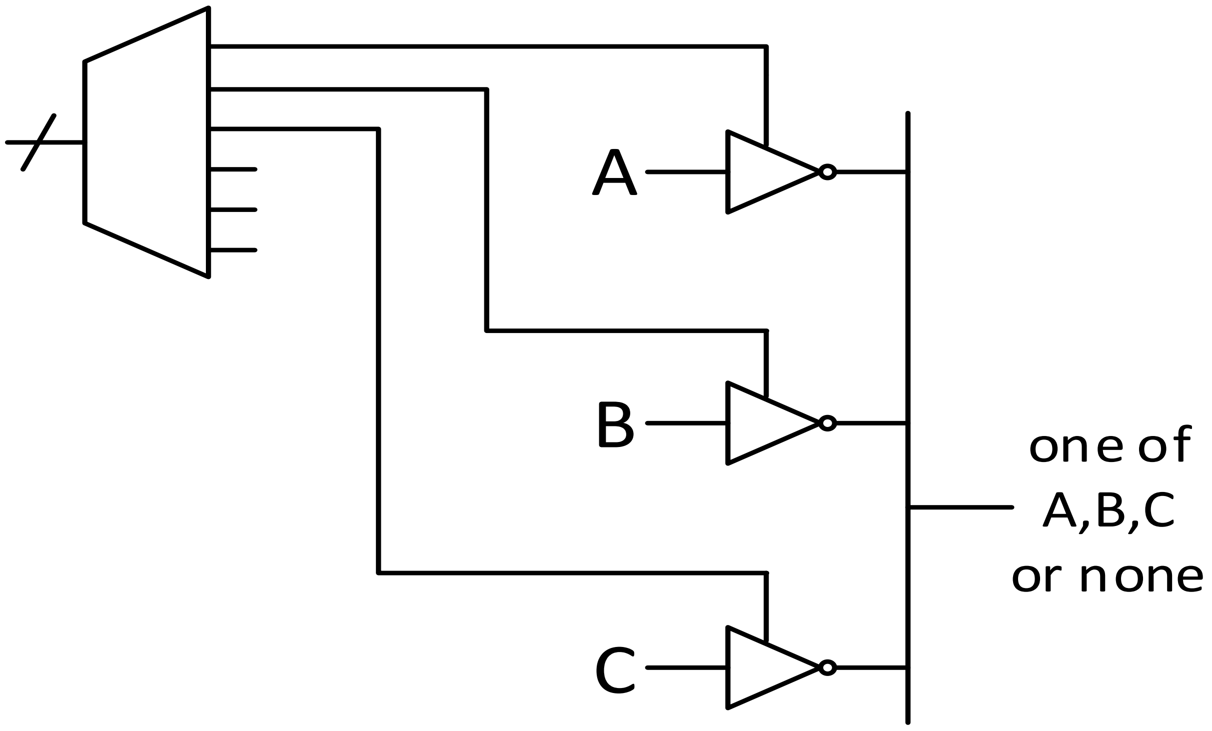


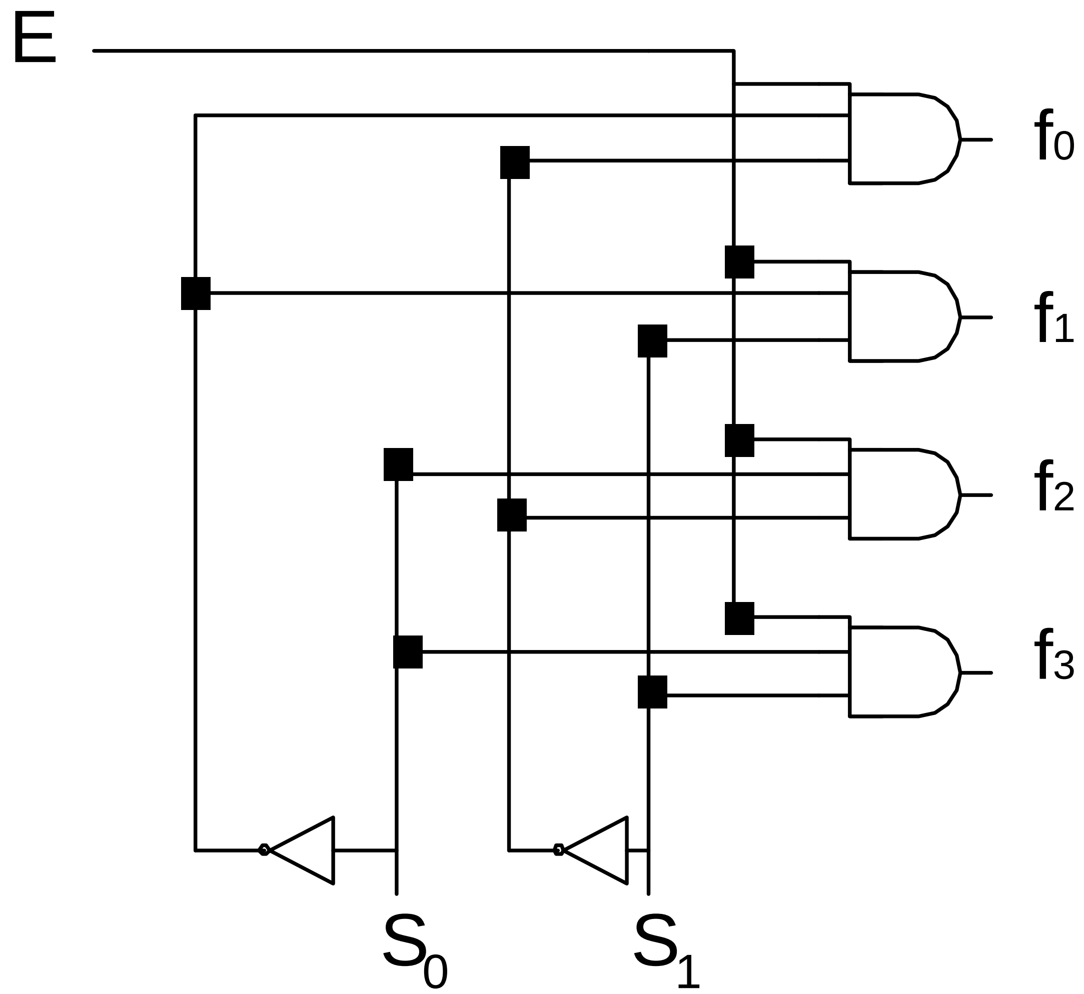

- מתג מנתק/מחבר את היציאה
- ליציאה שלושה ערכים אפשריים
  - '1'
  - '0'
  - "צפה", 3-state, Z
- מאפשר לחבר מספר יציאות לחוט יחיד—BUS
- חובה לדאוג שלכל היותר רק אחד מקווי enable יקבל את הערך '1'

# המפענח (decoder) חיוני למימוש BUS



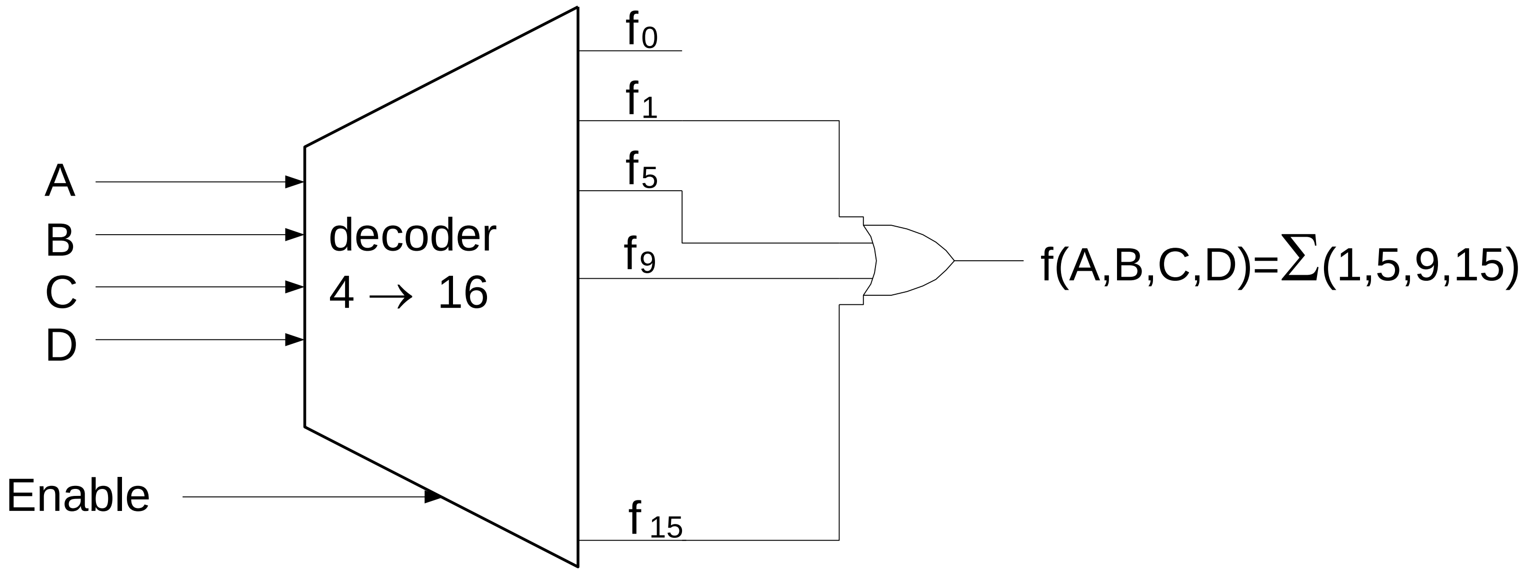

- כבר קבענו כי חובה לדאוג שלכל היותר רק אחד מקווי enable יקבל את הערך '1'
- לכך בדיוק דואג המפענח

# מימוש Decoder



# *** מימוש פונקציה באמצעות מפענח

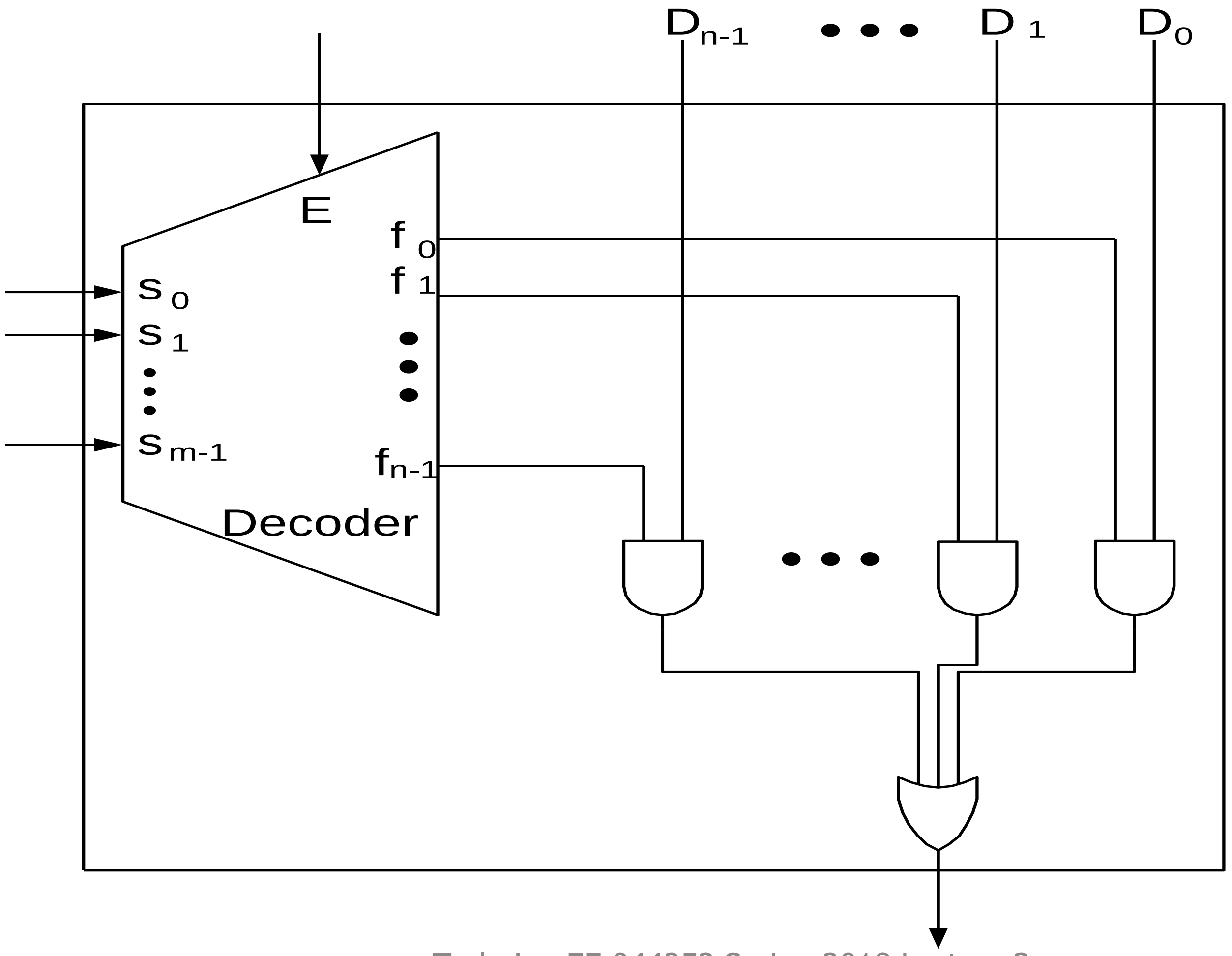
$$f(A,B,C,D)=\sum(1,5,9,15)$$



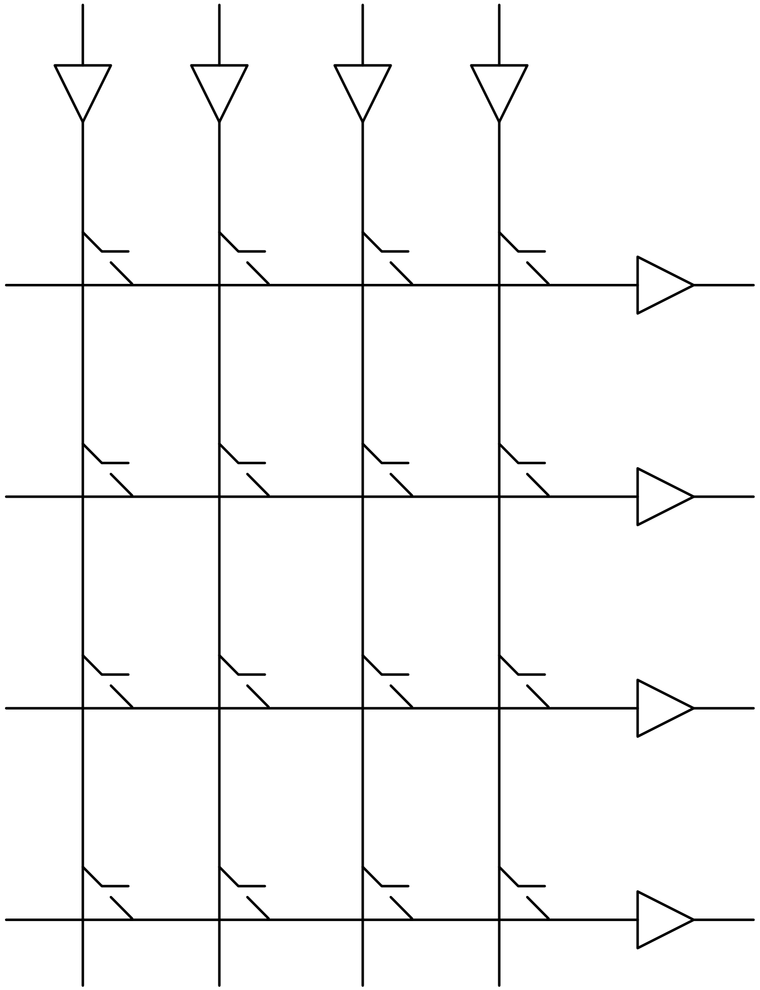

איזה ערכים יש לשים בכניסת ENABLE במעגל זה?

# מימוש בורר ע"י מפענח

הדמיון בממוש: ניתן לממש *Selector* תוך שמוש ב- *Decoder*:

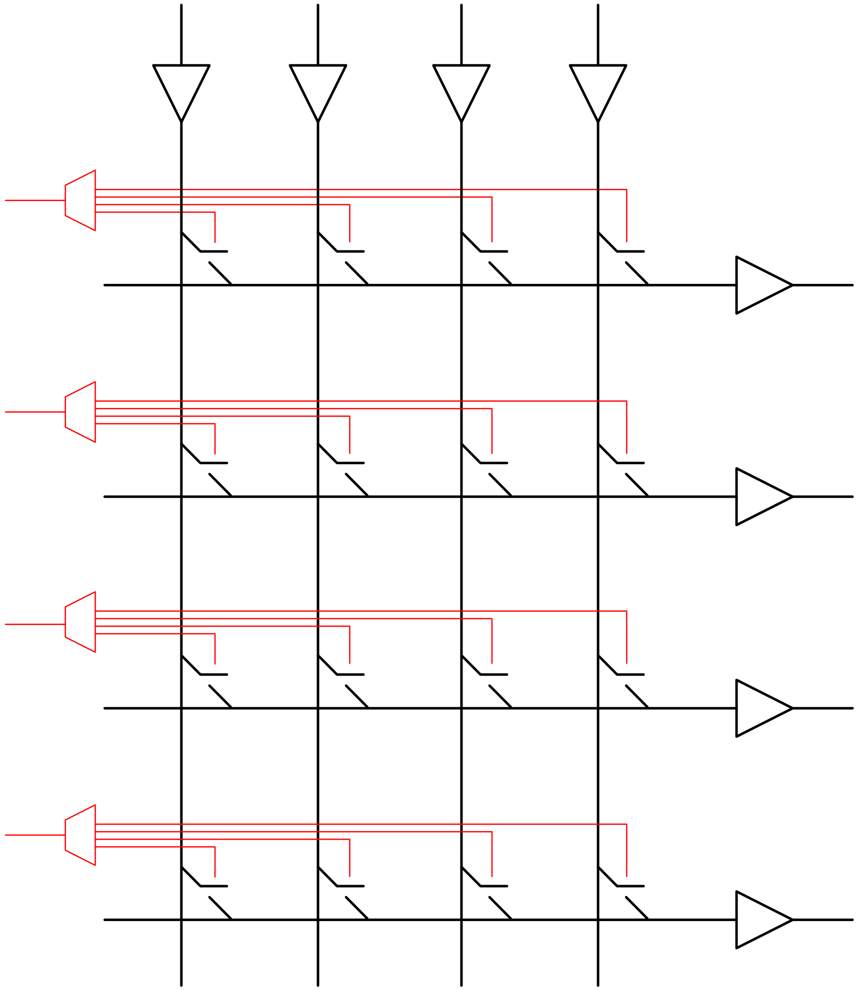


# Crossbar מתג



From telephones to the internet:
Search the web for crossbar images and videos

# מתג Crossbar מבוקר מפענחים



כזכור, רק שער אחד רשאי לכתוב על BUS

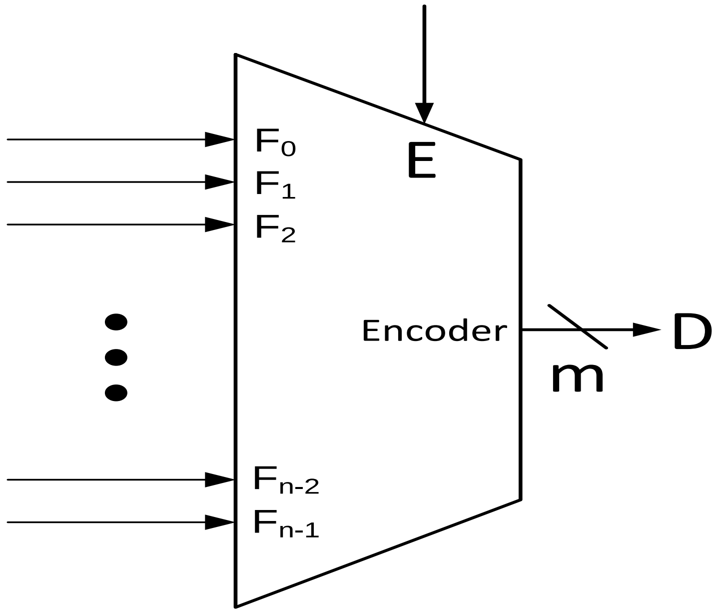
# Encoder – מקודד

<u>קלט:</u> n כניסות $F_o, \quad F_1, \ldots, F_{n-1}$

כניסת *Enable*

<u>פלט:</u> $m = \log n$ יציאות

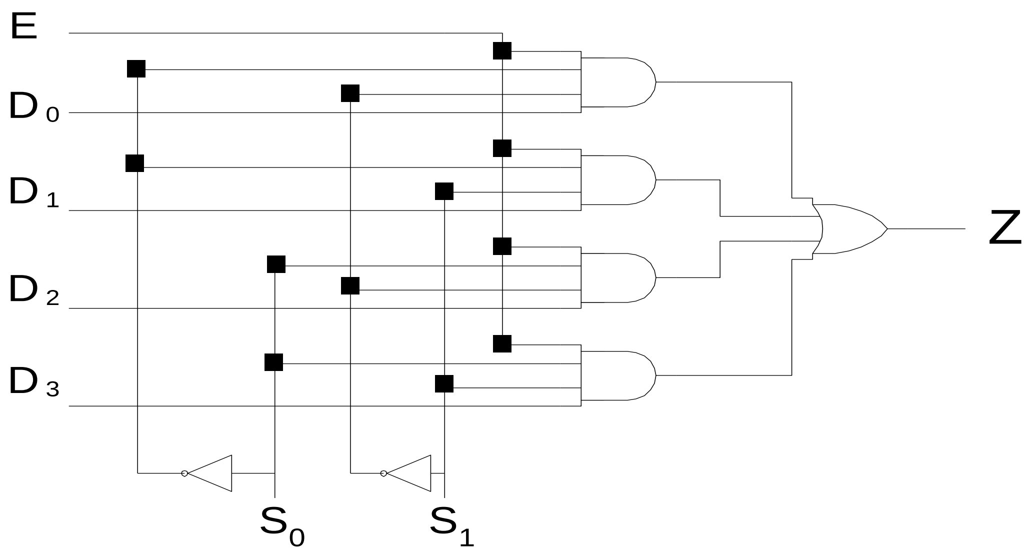
.




היציאה היא הקוד הבינרי של
הכניסה (היחידה) שהיא '1'

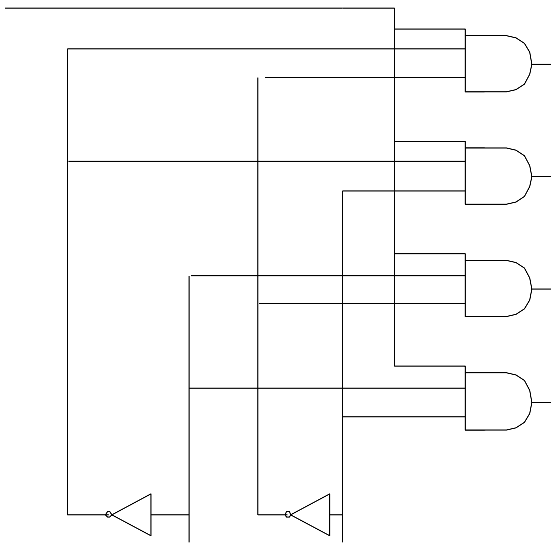
# סיבוכיות הבורר

- בורר בעל n+1 כניסות בקרה ו- $K=2^n$ כניסות נתונים מכיל :
  - $2(n+1)$ מהפכים [כדי לאפשר FO=1 מבחוץ]
  - $2^n$ שערי AND בעלי n+2 כניסות כ"א (השקולים ל- n+1 שערי AND בעלי שתי כניסות כ"א) וביחד $(n+1)2^n$ שערים
  - שער OR בעל $2^n$ כניסות השקול ל – $2^n-1$ שערים בעלי שתי כניסות
- סה"כ, מחירו של הבורר הוא $(n+2)2^n+2n+1=O(n2^n)=O(K \log_2 K)$ שערים



# סיבוכיות המפענח

- מפענח בעל $n+1$ כניסות ו- $K=2^n$ יציאות מכיל:
  - $2(n+1)$ מהפכים
  - $2^n$ שערי AND בעלי $n+1$ כניסות כ"א (השקולים ל- $n$ שערי AND בעלי שתי כניסות כ"א) וביחד $n2^n$ שערים
- סה"כ, מחירו של הבורר הוא
  - $n2^n+2(n+1)=O(n2^n)=O(K \log_2 K)$ שערים

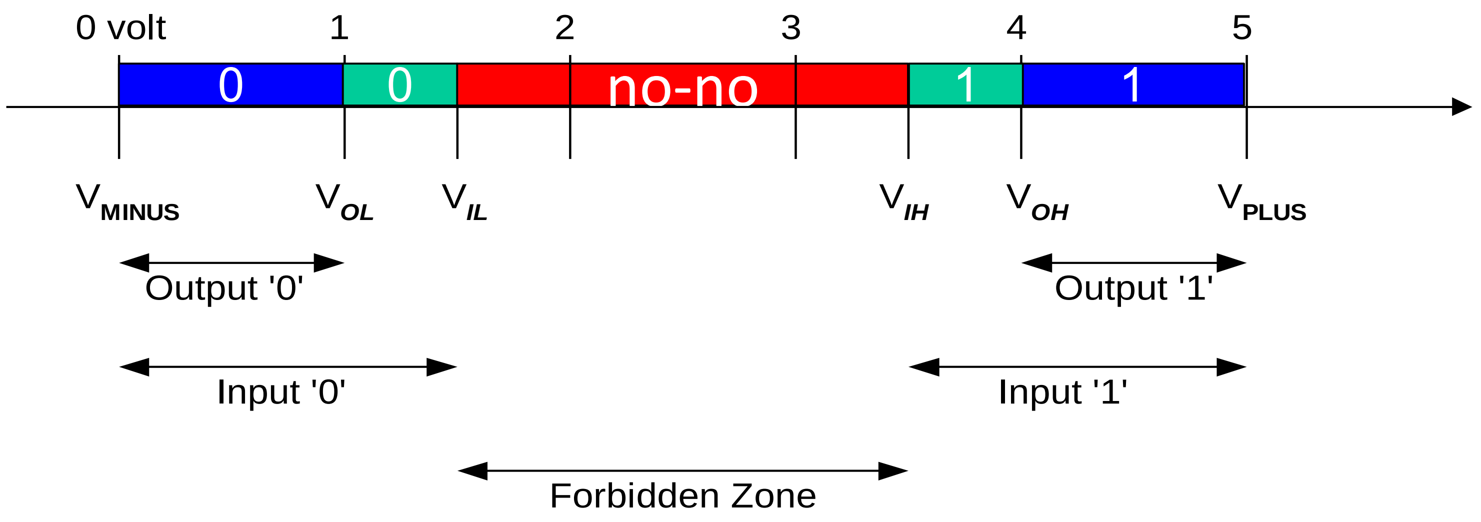


# Agenda

- Switching Function Minimization
  - Karnaugh, don't-care
- Gates
- Switching components
  - Mux, dec, enc, sel, switch, bus
- **Electrical Digital Abstraction**
- Four Propagation Delay Models
  - zero, unit, physical, practical
- Switching Energy
- Gates as Switches

# ההפשטה הספרתית (1) : ממתח רציף לערכים בדידים

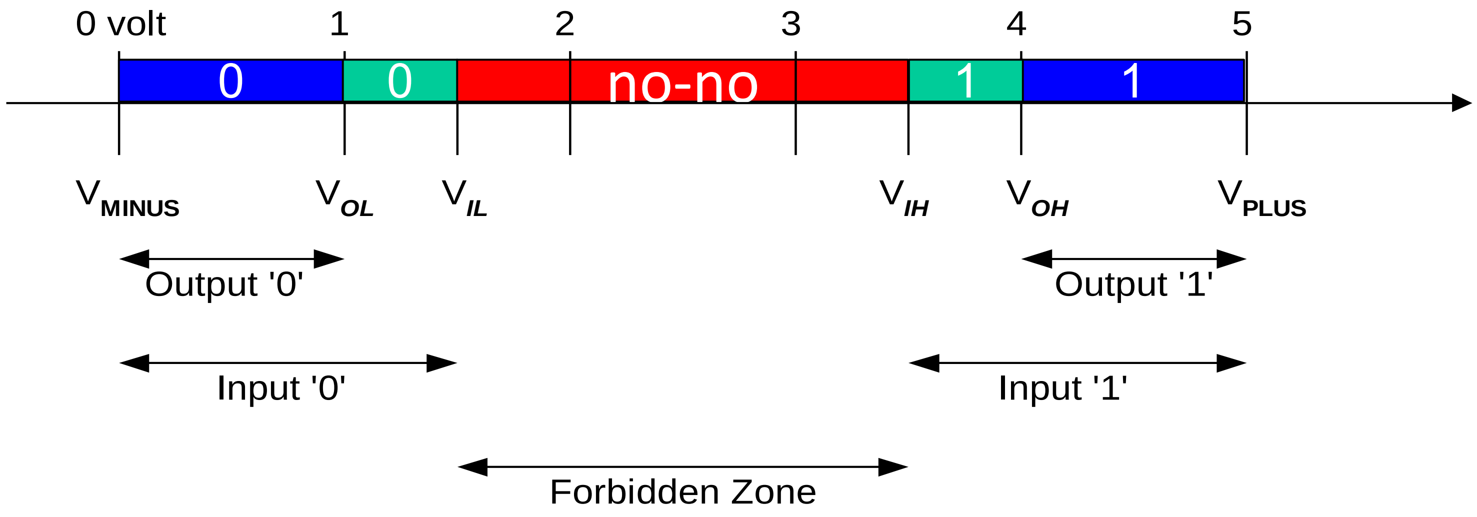
- שערים לוגיים (אלקטרוניים) מקבלים ערכי מתח חשמלי רציפים בכניסות, מייצרים מתח חשמלי ביציאות
- ערכי מיתוג בדידים (דיסקרטיים--0,1) מיוצגים על ידי <u>תחומים</u> של ערכי מתח חשמלי



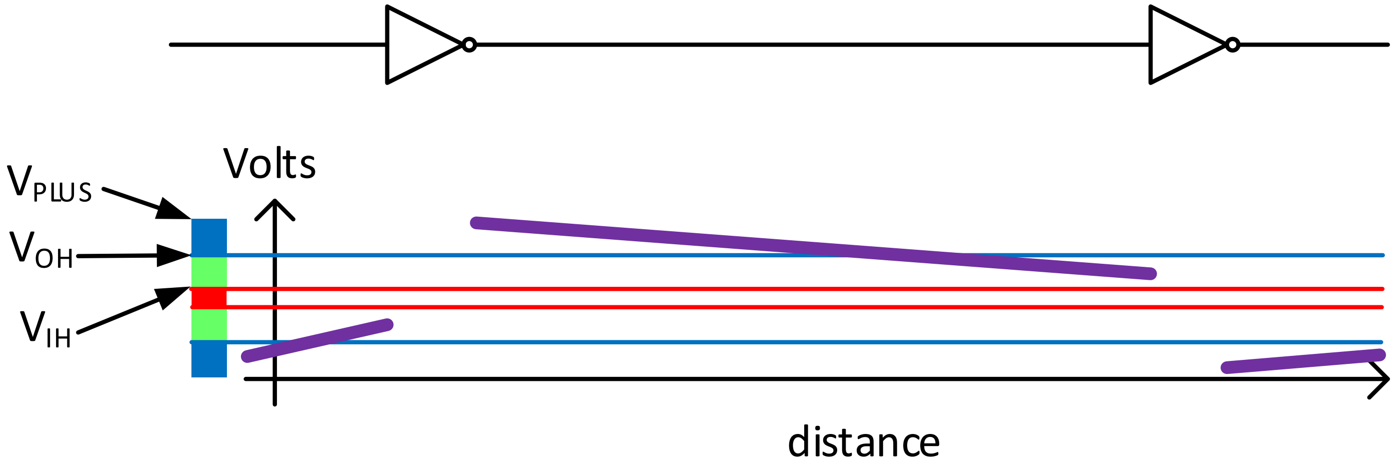

- השער "משפר" את איכות האות החשמלי—גם אם האות בכניסתו מתקרב לתחום האסור (ירוק), ביציאתו מובטח שהאות יהיה בתחום הרצוי (כחול)

# ההפשטה הספרתית (2): "רעש" ושולי רעש

- אות המופק על ידי שער אחד מועבר על גבי חוטי חשמל לשער שני
- בדרך יתכן שהמתח ישתנה בגלל הפרעות חשמליות ובגלל איבוד אנרגיה
  – רמז: חוט החשמל מתנגד למעבר החשמל. נדרשת אנרגיה כדי להתגבר על ההתנגדות. מפל המתח לפחות IR
- לכן $V_{IH}<V_{OH}$ ולכן $V_{OL}<V_{IL}$

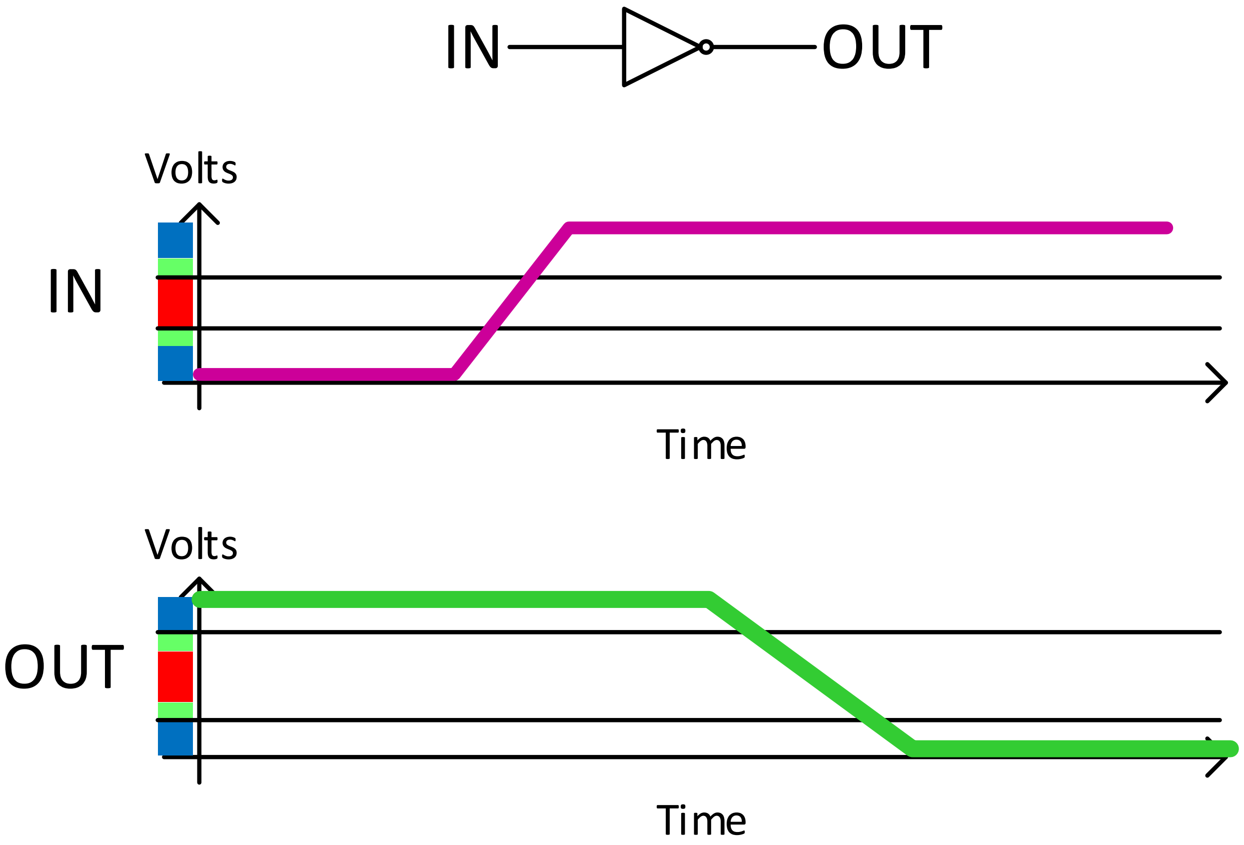


# ההפשטה הספרתית (3) : רגנרציה




- השער ה"דוחף" משמאל מייצר אות ברמת מתח בין $V_{PLUS}$ לבין $V_{OH}$
- המתח על האות בדרך כלל "נחלש" (מתקרב לאזור האסור) לאורך החוט
- השער מימין מפרש את האות המגיע אליו בתור 1 לוגי כל עוד המתח מעל $V_{IH}$
- ביציאה שלו, האות עומד בדרישה המחמירה יותר, וכך יש יצירה מחדש, **regeneration**, של האות, והוא יכול לעבור בנתיב הכולל אלפי ואף מיליוני שערים באמינות גבוהה מאוד. (Great idea #1)

# ההפשטה הספרתית (4): מזמן רציף לזמן בדיד




- בכניסה וגם ביציאה האותות עוברים בתחום האסור במשך זמן סופי > 0
- קיים משך זמן בו צריך "לעצום את העיניים" ולהתעלם מהיציאה
- במשך כמה זמן? יש לבחור במודל התזמון המתאים

# הגדרת זמני ההשהיה – פרוט המדידה (1)

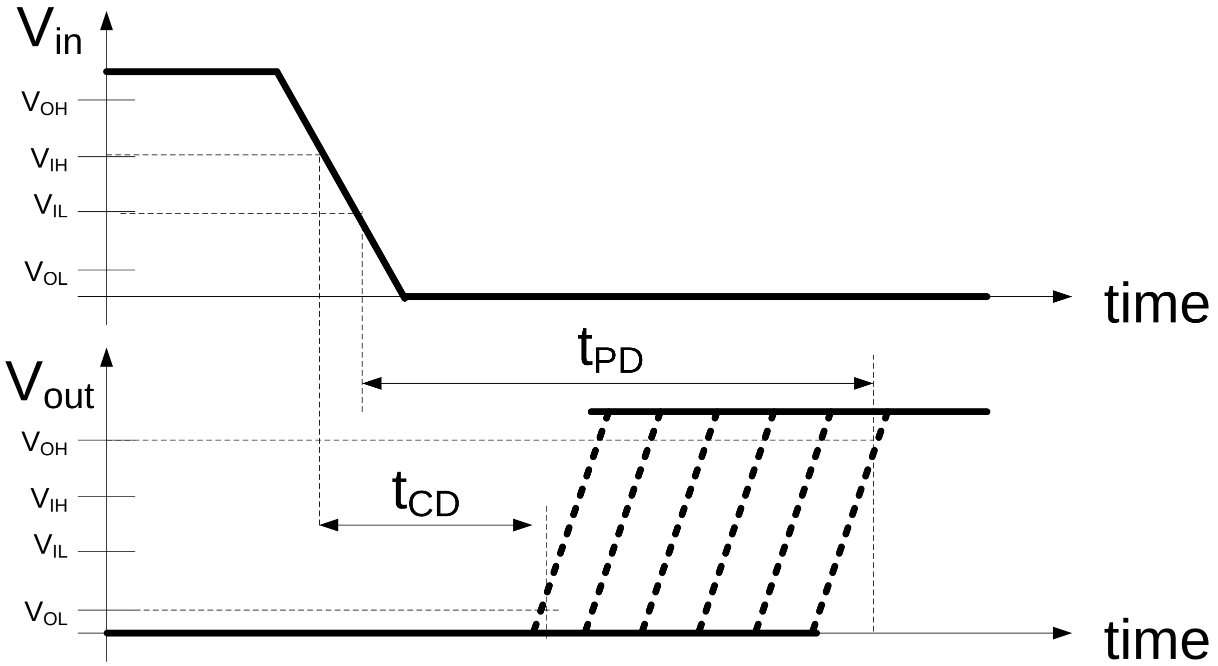
- $t_{CD}$ – משך הזמן בו מובטח שיציאת רכיב לא תספיק להגיב לשינוי בכניסתו. (Contamination Delay)
- $t_{PD}$ – משך הזמן שאחריו מובטח שהיציאה תשקף את השינוי בערך הכניסה. (Propagation Delay)

- בעיה: ממתי (ערך המתח **בכניסה**) עד מתי (ערך המתח **ביציאה**) נמדוד את ההשהיה?

- $t_{CD}$: יש למדוד ***מהמועד*** בו הכניסה "עזבה" את ערכה הלוגי **הקודם**,
  שכן עד אז אין כל עילה להשתנות היציאה,
  **עד** הרגע בו המתח ביציאה חורג מהתחום החוקי עבור ערך היציאה **הישן**.
  – התרחיש הקובע (כלומר זה בו נמדד משך הזמן המזערי) יהיה לרוב כאשר הכניסה משתנה באופן חד מאוד, דבר המחיש את שינוי היציאה.

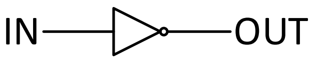
- $t_{PD}$: יש למדוד ***מהמועד*** בו הכניסה "הגיעה" לערכה הלוגי **החדש**,
  שכן עד אז אין כל מחויבות לשינוי היציאה,
  **עד** הרגע בו המתח ביציאה נכנס לתחום החוקי עבור ערך היציאה **החדש**.
  – התרחיש הקובע (כלומר זה בו נמדד משך הזמן המירבי) הוא לרוב כאשר הכניסה משתנה באופן איטי מאוד, שכן אז השתנות היציאה עשויה להשתהות.

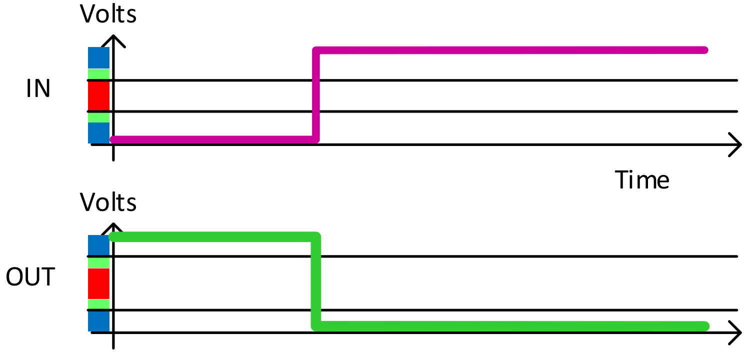
# הגדרת זמני השהייה – פרוט המדידה (מהפך כדוגמה) (2)



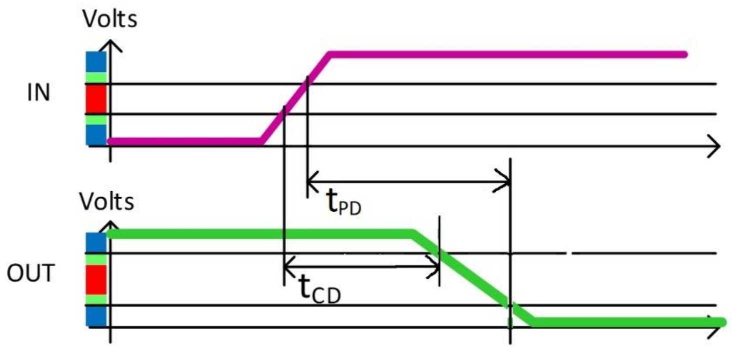

- כניסת המהפך 1 בתחילה, ואי"כ יורדת ל-0
- היציאה אינה משתנה מיד : במשך $\mathbf{t_{CD}}$ מרגע שינוי הכניסה מובטח שהיציאה תישאר עדיין בערכה הקודם (לא "תזדהם")
  - בדרך כלל היציאה תישאר עדיין בערכה הקודם גם קצת מאוחר יותר. החסם מאפיין את המקרה הגרוע ביותר
- במקרים שונים (מהפכים שונים ותנאי מתח וטמפרטורה שונים) תשתנה היציאה בזמנים שונים
  - מיוצג על ידי קווים לא רצופים בשרטוט
- מובטח שהיציאה תגיע לערכה החדש (ותישאר בו) לכל המאוחר לאחר $\mathbf{t_{PD}}$
- החסמים מהווים "אחריות יצרן"

# ההפשטה הספרתית (5): מודלים חלופיים "נוחים" לתזמון



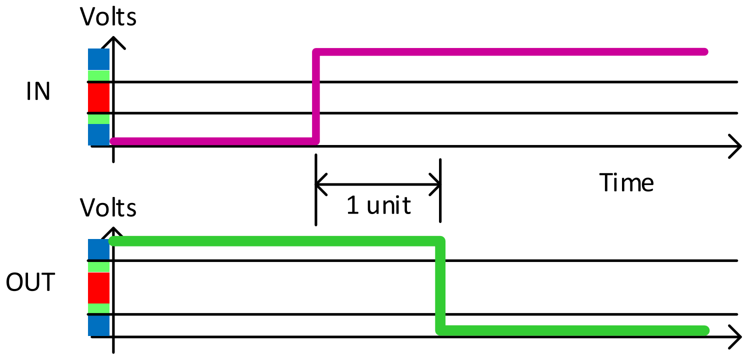




ZERO DELAY MODEL

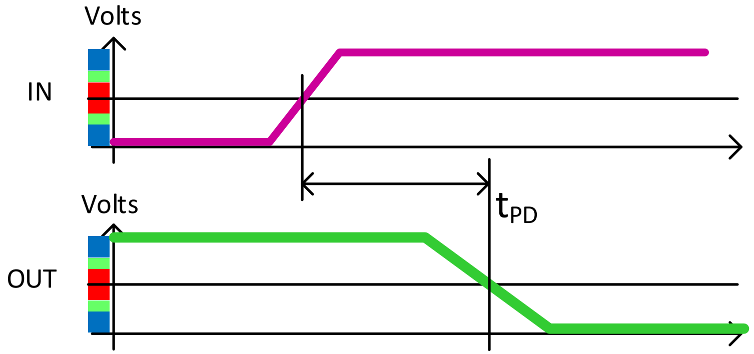



PHYSICAL DELAY MODEL

$t_{PD}$, $t_{CD}$ מייצגים חסמים ולא זמני מיתוג אקטואליים

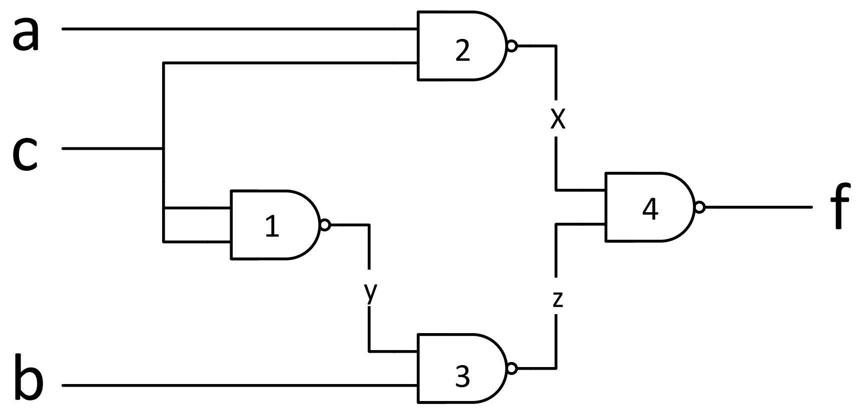



UNIT DELAY MODEL



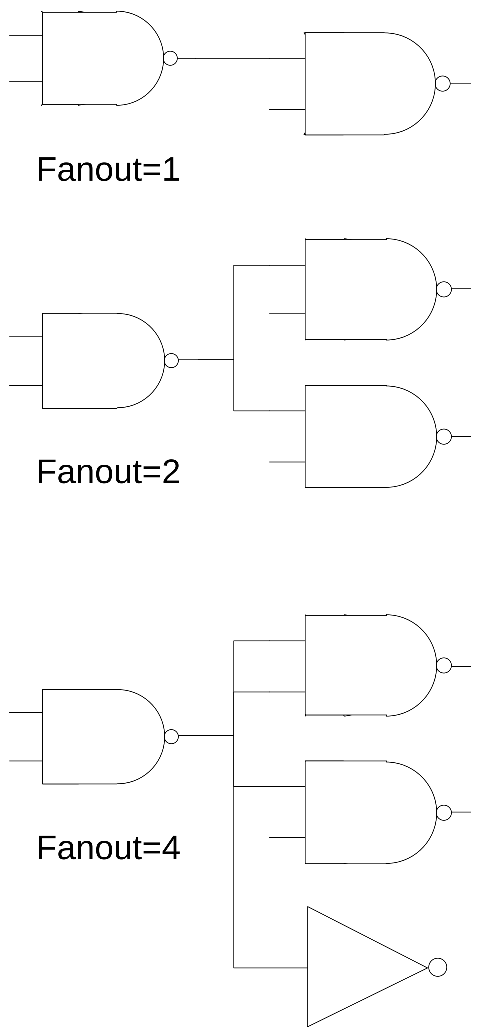

PRACTICAL (50%) DELAY MODEL

# השהייה במעגל צירופי



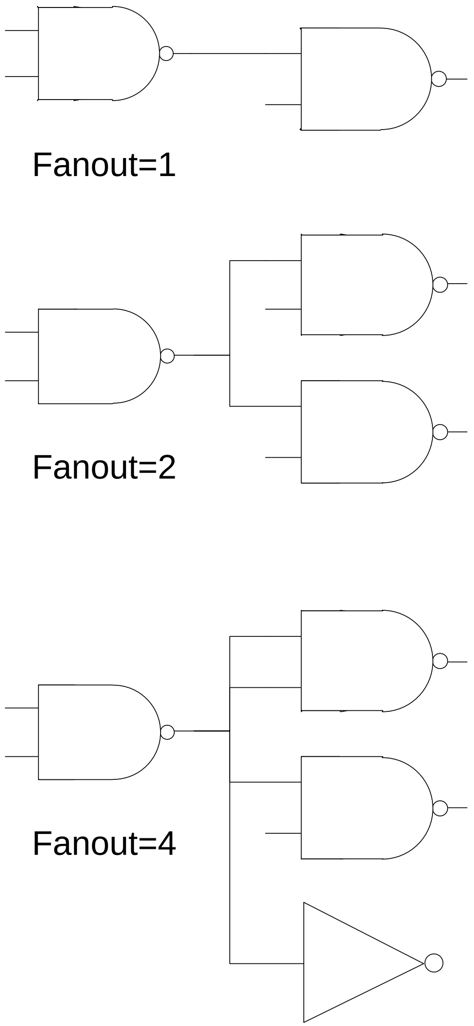

| DELAY MODEL | a,c → x | c → y | y,b → z | x,z → f | Inputs → f |
|---|---|---|---|---|---|
| Zero Delay | 0 | 0 | 0 | 0 | 0 |
| Unit Delay | 1 | 1 | 1 | 1 | 3 |
| Physical Delay | $t_{PD}(2)$ | $t_{PD}(1)$ | $t_{PD}(3)$ | $t_{PD}(4)$ | Max[ $t_{PD}(1)+t_{PD}(3)+t_{PD}(4)$, $t_{PD}(2)+t_{PD}(4)$ ] |
| Practical Delay | " | " | " | " | " |

# זמן המיתוג האמיתי תלוי גם בעומס



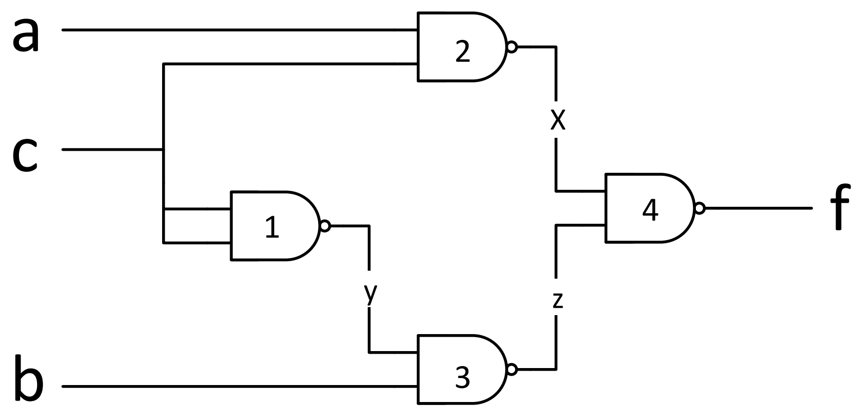

- $t_{PD} = t_0 + FO \cdot t_1$
- קצת פיזיקה:
  - מוצא השער הדוחף מתנהג כמקור זרם קבוע
  - כניסת השער המקבל מתנהגת כקבל
  - ככל שקיבול העומס **גדל**, זמן טעינתו / פריקתו בזרם קבוע **גדל**
- בעצם, גם מספר הכניסות לשער משפיע על השהייתו
  - באופן מעשי, אפשר להסתפק בשערים בעלי שתי כניסות
  - זה מנטרל את השפעת מספר הכניסות על ההשהייה

# ההפשטה הספרתית (6) : האנרגיה הנצרכת בחישוב ספרתי



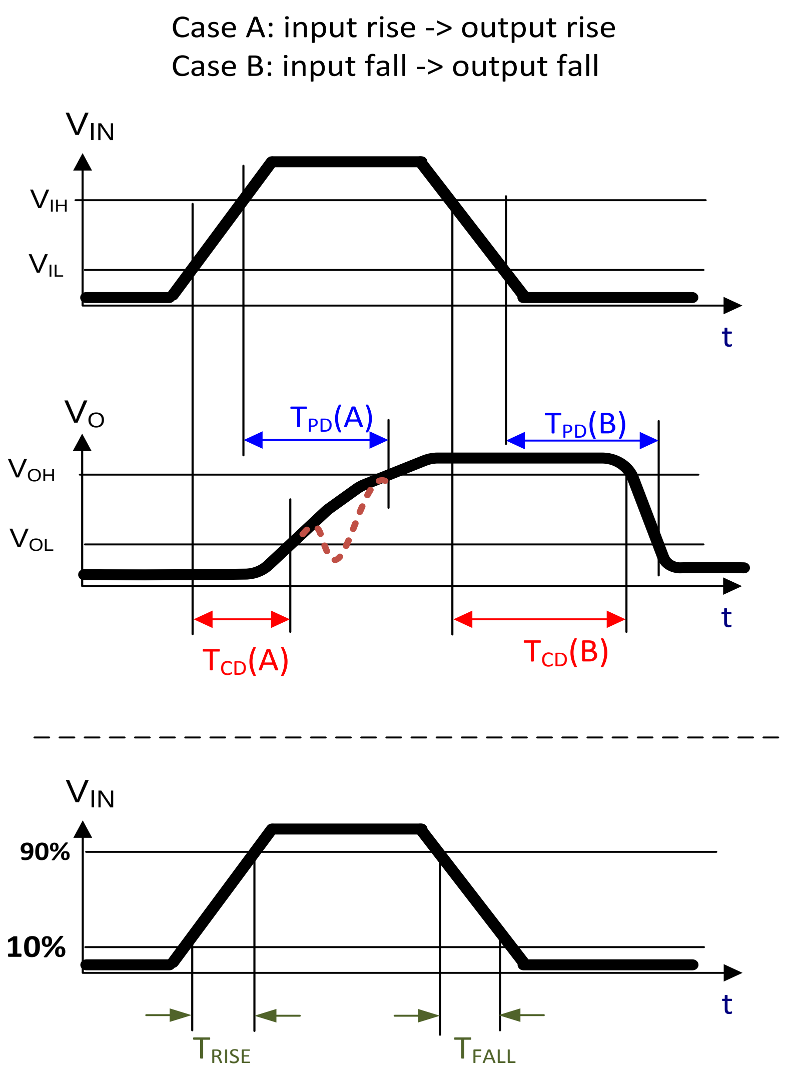

- עוד קצת פיזיקה :
  - טעינת קבל (קיבול C) במתח V צורכת אנרגיה $\frac{1}{2}CV^2$
  - לכן כל מיתוג (החלפת 0/1) בכל כניסת שער צורכת יחידת אנרגיה "דינמית"
  - טכנולוגיות שונות (בתנאי עבודה שונים) צורכות גם אנרגיה "סטטית" ללא קשר לחישוב
- הפשטה דיגיטלית :
  - בכל חישוב, נספור את מספר המיתוגים בכניסות שערים
  - יש להתחשב במעבר מצירוף כניסות אחד לאחר

# האנרגיה במעגל צירופי—דוגמאות



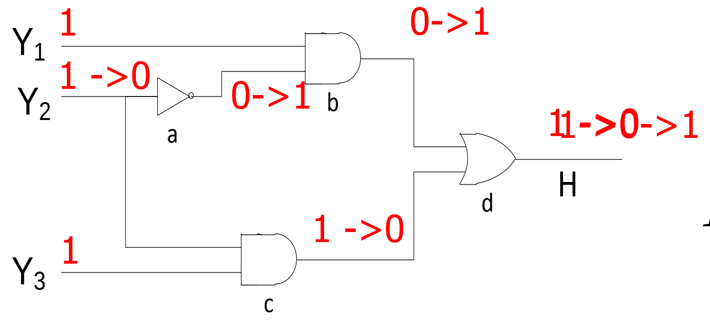

| Old abc | Old xyz | Old f | New abc | New xyz | New f | Energy | Switching |
|---|---|---|---|---|---|---|---|
| 000 | 111 | 0 | 001 | 101 | 0 | 4 | c (three inputs), y |
| 000 | 111 | 0 | 111 | 001 | 1 | 8 | a, b, c (three inputs), x, y, f |

# סיכום : מדדי זמן




- זמן זיהום המוצא, $t_{CD}$
- זמן התפשטות או שיהוי, $t_{PD}$
- זמן עליית האות, $t_{RISE}$,
  וזמן ירידתו, $t_{FALL}$
- בזמן שבו האות משתנה יתכן ויעבור ממצבו ההתחלתי לסופי דרך ערכי ביניים
  – האזארד סטטי (מתיצבים על ערך ישן) , למשל 0-1-0
  – האזארד דינמי (מתיצבים על ערך חדש), למשל 0-1-0-1

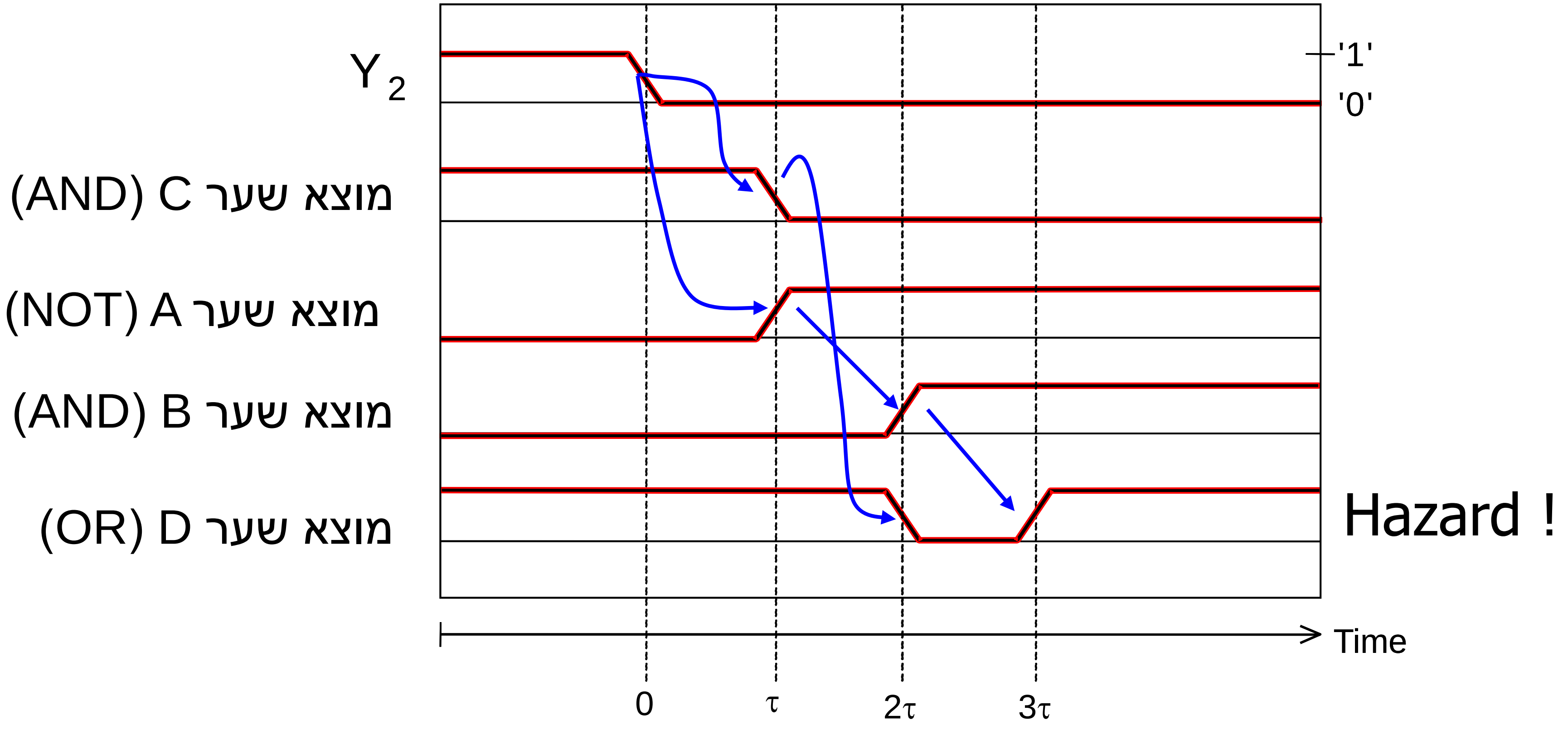
# הזרד (הבהוב) סטטי

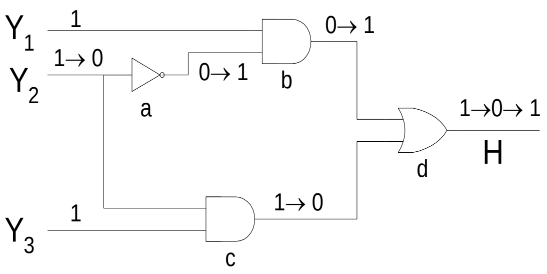



$$H = y_1\overline{y_2} + y_3y_2$$

- הסכמנו לא להסתכל על יציאת המעגל הצירופי לפני תום זמן ההשהיה
- לעיתים, במהלך זמן ההשהיה, עלולה היציאה לקבל ערך ביניים לא נכון

נניח מודל Unit Delay (או השהיה פיזיקלית עם זמני השהייה זהים לכל השערים: $t_{PD}= t_{CD} =t$).






# הבהובים סטטיים

- הבהוב סטטי (Static Hazard) : היציאה אמורה להיות סטטית, אבל היא עלולה להבהב
- אופייני למעבר (במפת קרנו) מגורם אחד לגורם אחר
- נגרם ע״י הבדלים בזמני ההתפשטות ברכיבים שונים

| $Y_3$ \ $Y_1Y_2$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | | | 1 |
| 1 | | 1 | 1 | 1 |

$$H = y_1\overline{y_2} + y_3y_2$$

# מניעת הבהובים סטטיים

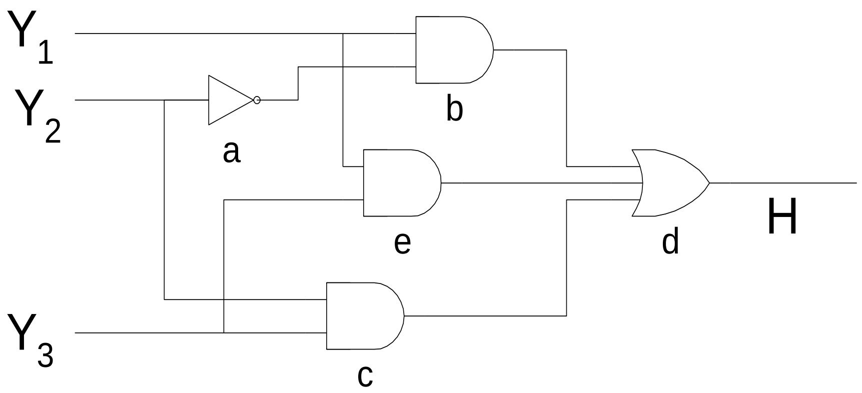
ניתן למנוע הבהובים סטטיים ע"י שינוי המעגל. ראשית, יש להניח כי :

א. <u>בו זמנית לא משתנה יותר מכניסה אחת למעגל,</u>

ב. שינויים נוספים בכניסות לא יקרו עד אשר יסתיימו כל השינויים בתוך המעגל הנובעים משנוי הכניסה האחרון

מוסיפים למעגל **גורם** נוסף $Y_1 \cdot Y_3$ , המכסה את החץ שהופיע במפת קרנו. ערכו של גורם זה אינו משתנה כאשר $Y_2$ משתנה מ-1 ל-0 :

| $Y_3$ \ $Y_1Y_2$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | | | 1 |
| 1 | | 1 | 1 | 1 |

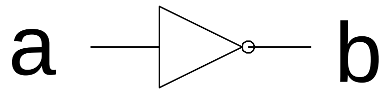


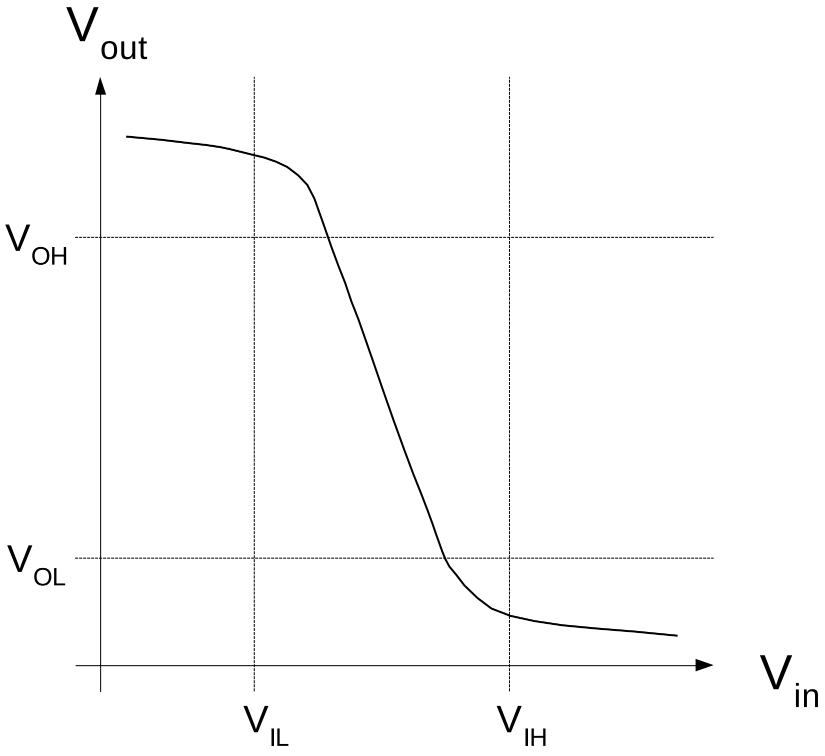

מעגל נקרא **Hazard-Free** אם הוא מממש ביטוי בצורת סכום מכפלות, כך שכל זוג משבצות שכנות במפת קרנו המכילות '1' מכוסה על ידי אחת המכפלות (לפחות)

# הבהובים דינמיים (Dynamic Hazards)

- קורים כאשר יציאת המעגל אמורה להשתנות (למשל $0 \rightarrow 1$) אבל השינוי נעשה תוך שלושה מעברים לפחות (למשל $0 \rightarrow 1 \rightarrow 0 \rightarrow 1$)
- בעייה זו אינה כל כך מעניינת מבחינה מעשית (מדוע?)
- אם כן רוצים לפתור אותה, יש לזכור שהיא מסובכת יותר מבעיית ההבהובים הסטטיים
  - הפתרונות דומים, אך אין פתרון כללי ויש מקרים שאינם ניתנים לפתרון
- כמובן שלפני זמן הזיהום ולאחר זמן ההשהייה היציאה תהיה תקינה

# המהפך – אופיין המתח (פונקציית תמסורת)



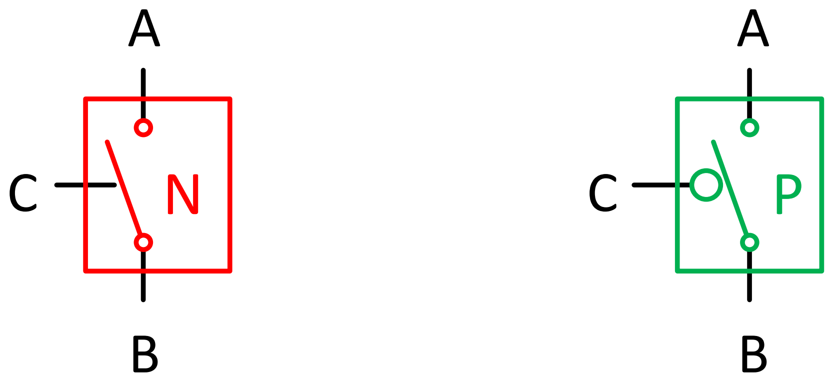

- שיפוע הגרף גדול ביותר באזור האסור
- כל שינוי קל במתח הכניסה יביא לשינוי חזק במתח היציאה
- המטרה:
  - המעבר באזור האסור יתרחש בזמן קצר ככל האפשר
  - תקטין רגישות המעגל לרעש

# Agenda

- Switching Function Minimization
  - Karnaugh, don’t-care
- Gates
- Switching components
  - Mux, dec, enc, sel, switch, bus
- Electrical Digital Abstraction
- Four Propagation Delay Models
  - zero, unit, physical, practical
- Switching Energy
- **Gates as Switches**

# בנית שערים לוגיים באמצעות מתגים

- המימוש הטכנולוגי של שערים לוגיים נעשה באמצעות טרנזיסטורים המשמשים כמתגים
- לכל מתג שלוש קצוות : כניסת בקרה (C) ושני קצוות (B,A) שהמתג יכול לחבר ביניהם
- נגדיר שני סוגי מתגים, P ו-N , באמצעות טבלאות אמת :




### מתג $N$

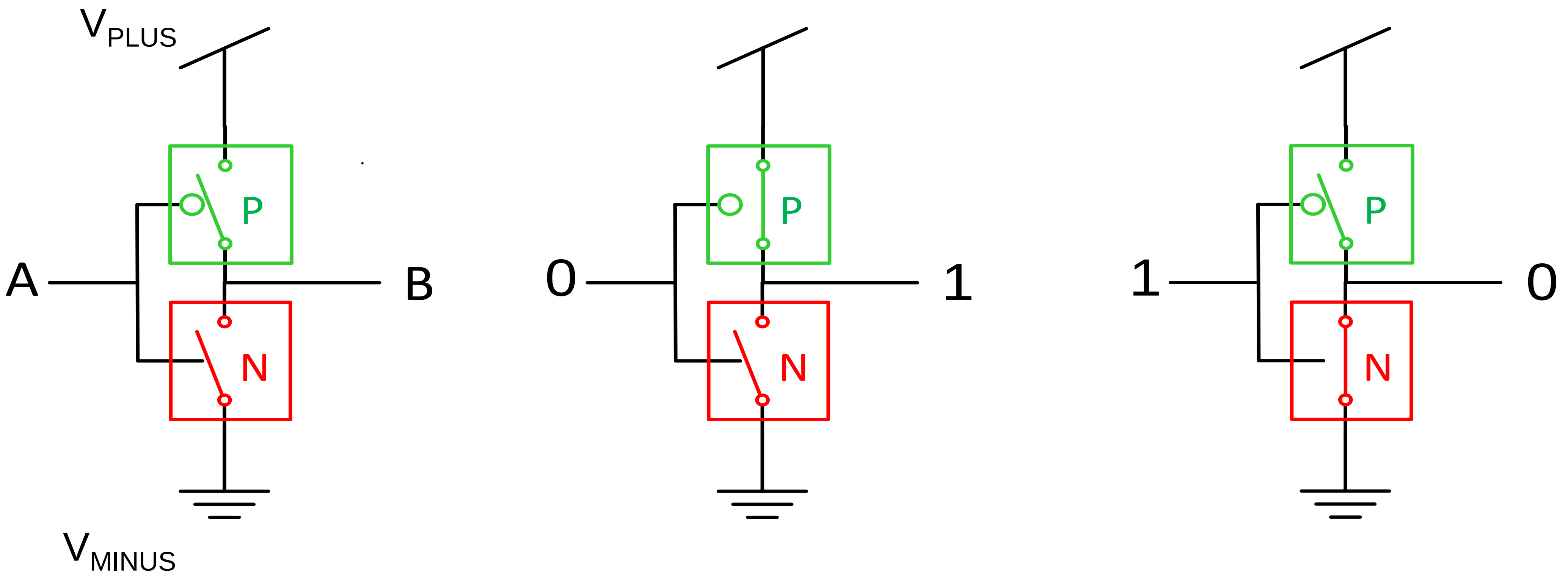
| כניסת הבקרה $C$ | מצב המתג |
|---|---|
| 0 | מנותק |
| 1 | מחובר |

### מתג $P$

| כניסת הבקרה $C$ | מצב המתג |
|---|---|
| 0 | מחובר |
| 1 | מנותק |

# בנית מהפך באמצעות מתגים

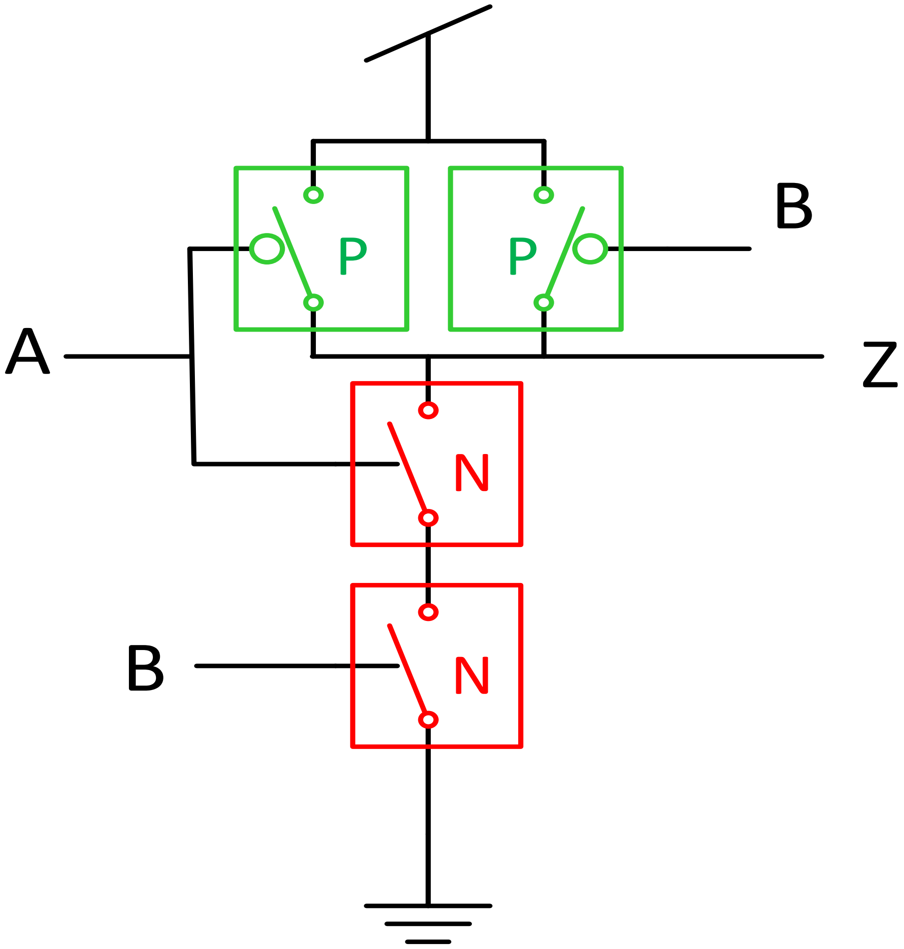
- זוג מתגים המחוברים בטור בין הקבועים '1' ו-'0' ($V_{PLUS}$, $V_{MINUS}$)
- כאשר A=0, מתג N מנותק ומתג P מחובר, וכך עובר הקבוע '1' ליציאה B
- כאשר A=1, מתג N מחובר ומתג P מנותק, וכך עובר הקבוע '0' ליציאה B



# בנית שער NAND באמצעות מתגים

- שער NAND מורכב מארבעה מתגים כלהלן

נסה להבין את פעולתו :



# סיכום

- Switching Function Minimization
  - Karnaugh, don’t-care
- Gates
- Switching components
  - Mux, dec, enc, sel, switch, bus
- Electrical Digital Abstraction
- Four Propagation Delay Models
  - zero, unit, physical, practical
- Switching Energy
- Gates as Switches