# דפי עזר לבחינה במערכות ספרתיות

21 בפברואר 2005

## 1. מעבר מבסיס 10 לבסיס b

<u>שלם</u>:

$$\left(d_{n-1}d_{n-2}\ldots d_1d_0\right)_b = d_{n-1}b^{n-1} + d_{n-2}b^{n-2} + \ldots + d_1b + d_0$$
$$= \left(d_{n-1}b^{n-2} + d_{n-2}b^{n-3} + \ldots + d_1\right)b + d_0$$

כלומר, לאחר חלוקה ראשונה ב-$b$, $d_0$ היא השארית המתקבלת.

באופן דומה ממשיכים ומקבלים את $\ldots\ d_1, d_2,$ .

<u>שבר</u>:

$$\left(0.d_{-1}d_{-2}\ldots d_{-m}\right)_b = d_{-1}b^{-1} + d_{-2}b^{-2} + \ldots + d_{-m}b^{-m} =$$
$$= \frac{d_{-1} + d_{-2}b^{-1} + \ldots + d_{-m}b^{-m+1}}{b}$$

כלומר, לאחר הכפלה ראשונה ב-$b$, $d_{-1}$ הוא השלם המתקבל.

באופן דומה ממשיכים ומקבלים את $d_{-2},\ d_{-3},\ \ldots$

## 2. ייצוג מספרים בעלי סימן

ייצוג בעזרת $n$ סיביות $b_{n-1}b_{n-2}\ldots b_1b_0$ :

$b_{n-1}$ היא סיבית הסימן:

$$b_{n-1}\begin{cases} 0 & \text{אם המספר חיובי} \\ 1 & \text{אם המספר שלילי} \end{cases}$$

$b_{n-2}\ldots b_1b_0$ הן סיביות הערך:

**בשיטת גודל וסימן sign and magnitude**

*הערך המוחלט של המספר בייצוג בינארי* $= b_{n-2}\ldots b_1b_0$

**בשיטת 1's complement**

$$b_{n-2}\ldots b_1b_0 = \begin{cases} \text{אם חיובי – הערך הבינארי של המספר} \\ \text{אם שלילי – המשלים של הערך הבינארי של המספר} \end{cases}$$

**בשיטת 2's Complement**

$$b_{n-2}\ldots b_1b_0 = \begin{cases} \text{אם חיובי – הערך הבינארי של המספר} \\ \text{אם שלילי – המשלים של הערך הבינארי של המספר פלוס 1} \end{cases}$$

טווח ייצוג בגודל וסימן וב-1's complement : $-\left(2^{n-1}-1\right) \le A \le 2^{n-1}-1$

טווח ייצוג ב-2's complement : $-2^{n-1} \le A \le 2^{n-1}-1$

### 3. ייצוג מספרים ממשיים בשיטת Floating Point

ייצוג בעזרת $n = m + e + 1$ סיביות $b_{m+e}b_{m+e-1}\ldots b_{m+1}b_m b_{m-1}\ldots b_1 b_0$ :

<u>המנטיסה</u> M מיוצגת ע"י $m+1$ סיביות $b_{m-1}\ldots b_i b_0$ (ערך) ו- $b_{m+e}$ (סימן) :

$-2^m \le M \le 2^m - 1$.

<u>האקספוננט</u> מיוצג ע"י $e$ סיביות $b_{m+e-1}\ldots b_{m+1}b_m$ ($b_{m+e-1}$ עבור סימן) :

$-2^{e-1} \le E \le 2^{e-1} - 1$.

חישוב הערך העשרוני של המספר : $N = M \cdot 2^E$

טווח יצוג : $-2^m \cdot 2^{2e-1} \le N \le \left(2^m - 1\right) \cdot 2^{2e-1} - 1$.

### 4. קודים, הגדרות

<u>קוד משוקלל</u>: כל ספרה מיוצגת ע"י 4 סיביות $x_4x_3x_2x_1$ ולכל סיבית משקל $\omega_i$, ערך המספר הינו $N = \sum_{i=1}^{4} \omega_i x_i$.

| ספרה / משקלים | 8 4 2 1 (קוד BCD) | 2 4 2 1 | 6 4 2 -3 |
|---|---|---|---|
| 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 |
| 1 | 0 0 0 1 | 0 0 0 1 | 0 1 0 1 |
| 2 | 0 0 1 0 | 0 0 1 0 | 0 0 1 0 |
| 3 | 0 0 1 1 | 0 0 1 1 | 1 0 0 1 |
| 4 | 0 1 0 0 | 0 1 0 0 | 0 1 0 0 |
| 5 | 0 1 0 1 | 1 0 1 1 | 1 0 1 1 |
| 6 | 0 1 1 0 | 1 1 0 0 | 0 1 1 0 |
| 7 | 0 1 1 1 | 1 1 0 1 | 1 1 0 1 |
| 8 | 1 0 0 0 | 1 1 1 0 | 1 0 1 0 |
| 9 | 1 0 0 1 | 1 1 1 1 | 1 1 1 1 |

<u>קוד המשלים את עצמו</u>: קוד שבו המשלים ל- 9 של כל ספרה מתקבל ע"י הפיכת אחדים לאפסים ולהיפך במילת הקוד של הספרה.

<u>קוד ציקלי</u>: קוד שבו מילת קוד של ספרה כלשהי שונה ממילת הקוד של הספרה שלפניה ואחריה בסיבית אחת בלבד.

<u>קוד משוקף</u>: קוד עם $n$ סיביות מתקבל ע"י שיקוף הקוד עם $n$-1 סיביות והוספת $0$ משמאל ל- $2^{n-1}$ מילות הקוד הראשונות, ו- 1 משמאל ל- $2^{n-1}$ מילות הקוד הבאות.

קוד Gray : (המרה מבינארי ל- Gray ולהיפך)

אם $g_n g_{n-1} \cdots g_1 g_0$ היא מילת קוד Gray המייצגת את המספר הבינארי $b_n b_{n-1} \ldots b_1 b_0$, אזי

- המרה מבינארי ל- Gray :

$$g_n = b_n$$
$$g_i = b_i \oplus b_{i+1} \qquad \forall 0 \le i < n$$

- המרה מ- Gray לבינארי :

$$b_i = \begin{cases} g_i & \text{אם מספר ה-1-ים משמאל ל-} g_i \text{ זוגי} \\ \overline{g_i} & \text{אחרת} \end{cases}$$

**5. קודים לגילוי ותיקון שגיאות**

יכולת גלוי או תקון שגיאות : אם המרחק המינימלי בקוד הוא $k$, אזי ניתן לגלות עד $k\text{-}1$ שגיאות. אם מניחים שנפלו לכל היותר $\left\lfloor \frac{k-1}{2} \right\rfloor$ שגיאות, אזי ניתן לתקן אותן.

Parity code : מוסיפים סיבית אחת לכל מלת קוד, כך שמספר האחדים במלה יהיה זוגי.

Repetition code : כל סיבית משודרת r פעמים.

X out of Y code : כל מילות הקוד באורך Y מכילות בדיוק X 1-ים.

**6. כמה מזהויות המיתוג**

| | |
|---|---|
| עצמיות : | $x + x = x \qquad x \cdot x = x$ |
| אדיש : | $x \cdot 1 = x \qquad x + 0 = x$ |
| שולט : | $x \cdot 0 = 0 \qquad x + 1 = 1$ |
| חילוף : | $x + y = y + x \qquad xy = yx$ |
| קיבוץ : | $(x+y)+z = x+(y+z) \qquad (xy)z = x(yz)$ |
| השלמה : | $x + x' = 1 \qquad x \cdot x' = 0$ |
| פילוג : | $x(y+z) = xy + xz \qquad x + yz = (x+y)(x+z)$ |
| בליעה : | $x + xy = x \qquad x(x+y) = x$ |
| בליעה שני : | $x + x'y = x + y \qquad x(x'+y) = xy$ |
| הסכמה : | $xy + x'z + yz = xy + x'z$ |
| דה-מורגן : | $(x+y)' = x' \cdot y'$<br>$(xy)' = x' + y'$ |

**עקרון הדואליות** :

אם מתקיים : P(x,y,…,0,1,AND,OR,NOT)= Q(x,y,…,0,1,AND,OR,NOT)

אז מתקיים גם : P(x,y,…,1,0,OR,AND,NOT)= Q(x,y,…,1,0,OR,AND,NOT)

הערה : בזהות מיתוג ניתן להחליף כל משתנה בביטוי מיתוג.

## 7. ייצוג פונקציות

minterm: צרוף כל המשתנים ע"י פעולת AND כך ש- $x_i$ מופיע כ- $x_i$ אם ערכו בצרוף הוא 1, וכ- $\bar{x}_i$ אם ערכו הוא 0.

Maxterm: צרוף כל המשתנים ע"י פעולת OR כך ש- $x_i$ מופיע כ- $x_i$ אם ערכו בצרוף הוא 0, וכ- $\bar{x}_i$ אם ערכו הוא 1.

סכום מכפלות קנוני: סכום כל ה- minterms בהם הפונקציה מקבלת ערך 1.

מכפלת סכומים קנונית: מכפלת כל ה- maxterms בהם הפונקציה מקבלת ערך 0.

## 8. מפות קרנו

גורר: ליטרל או מכפלת ליטרלים המכוסה ע"י הפונקציה.

גורר ראשוני (PI): גורר שכל השמטה של ליטרל ממנו יוצרת מכפלה שאינה מכוסה ע"י הפונקציה.

גורר ראשוני חיוני (EPI): גורר ראשוני המכסה מינטרם שאינו מכוסה ע"י אף גורר ראשוני אחר.

### מפות קרנו הכוללות משתני ברירה (don't care)

א. הנח כי כל משתני הברירה (Φ) הם '1'.

ב. סמן את כל ה-PI במפה המתקבלת משלב א' (התעלם מ-PI המכסים אך ורק משתני ברירה).

ג. כל PI המכסה מינטרם ('1') מקורי (שאינו Φ במקור) באופן בלעדי יחשב EPI.

ד. 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.

ה. הפוך ל-'1' את משתני הברירה המכוסים ע"י הפונקציה המינימלית.

ו. הפוך ל-'0' את שאר משתני הברירה.

### מפת קרנו ל- 5 משתנים:

| yz \ vwx | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
|---|---|---|---|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 | 24 | 28 | 20 | 16 |
| 01 | 1 | 5 | 13 | 9 | 25 | 29 | 21 | 17 |
| 11 | 3 | 7 | 15 | 11 | 27 | 31 | 23 | 19 |
| 10 | 2 | 6 | 14 | 10 | 26 | 30 | 22 | 18 |

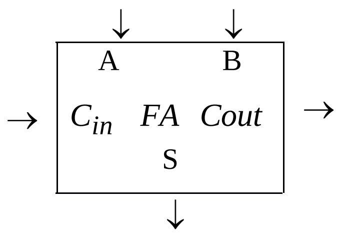
**משתנים בתוך המפה -- תהליך הפישוט:**

1. נגדיר את המשתנים במפה כ- 0 ונמצא כיסוי מינימלי למפה שנוצרה.
2. למשתנה מסויים במפה:
   א. מגדירים אותו כ- 1, את יתר המשתנים כ- 0 ואת האחדים המקוריים כ- $\phi$.
   ב. מוצאים כיסוי מינימלי למפה הנוצרת וכופלים במשתנה.
3. חוזרים על הסעיף הקודם עבור כל המשתנים, כאשר לכל השלבים בתהליך הפישוט משתנה והמשלים שלו נחשבים כשונים.
4. הכיסוי הסופי (והפשוט יותר) יהיה סכום כל הביטויים שהתקבלו.

### 9. Full Adder

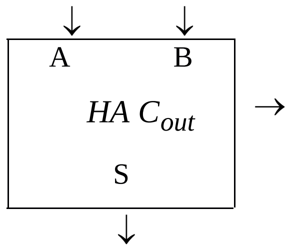
- כניסות: $A, B, C_{in}$. יציאות: $S,\ C_{out}$.
- מבצע: $A+B+C_{in}$. (חיבור, לא OR) נותן את הסכום ב-S ואת הנשא ב- $C_{out}$:

$$S = A \oplus B \oplus C_{in} \qquad C_{out} = AB + AC_{in} + BC_{in}$$



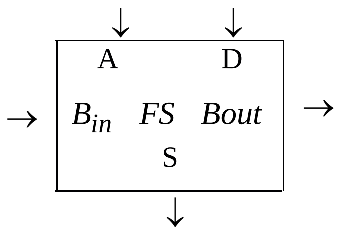

### 10. Half Adder

כמו Full Adder, אך ללא כניסת $C_{in}$.




### 11. Full Subtractor

- כניסות: $A, D, B_{in}$. יציאות: $S,\ B_{out}$
- מבצע: $A-D-B_{in}$. נותן את התוצאה ב- S ואת ה- Borrow ב- $B_{out}$.



## 12. Selector (בורר)

- שמות נוספים: Multiplexor, MUX. סימון: $2^n \to 1$.
- כניסות: $D_o, \ D_1, \ \ldots, D_{2^n-1}$ - כניסות מידע (data inputs),

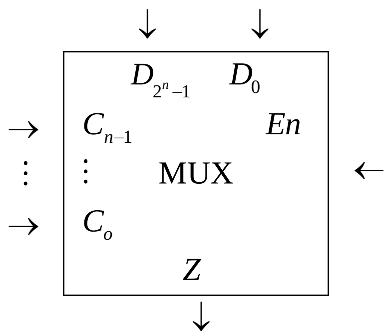
  $C_o, C_1, \ldots, C_{n-1}$ - כניסות בקרה (control inputs),

  $En$ – כניסת איפשור (Enable) (לא הכרחית).
- יציאה: $Z$.
- מבצע (אם קיימת כניסת איפשור):

  - אם $En = 0$, אזי $Z=0$,

  - אם $En = 1$, אזי $Z=D_i$, כש- $i$ הוא המספר המופיע בכניסות הבקרה.

  - אם <u>אין</u> כניסת איפשור,מניחים כי $En=1$.




## 13. Decoder (מפענח)

- סימון: $n \to 2^n$.
- כניסות: $C_o, \ C_1, \ldots, C_{n-1}$ - כניסות בקרה (control inputs),

  $En$ – כניסת איפשור (Enable) (לא הכרחית).
- יציאות: $Z_o, \ Z_1, \ldots, Z_{2^n-1}$.
- מבצע (אם קיימת כניסת איפשור):

  - אם $En = 0$, אזי $Z_i = 0$

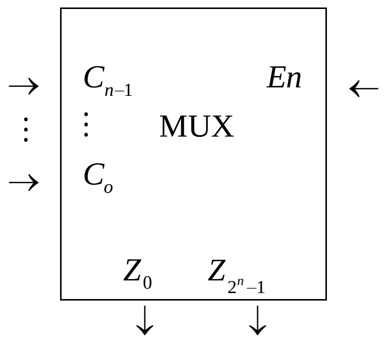
  $\forall \ i \in \{0,1,\ldots,2^n-1\}$.

  - אם $En=1$ אזי $Z_i = 1$ כש- $i$ הוא המספר המופיע בכניסות הבקרה,

  ו- $Z_j = 0$ בכל שאר האינדקסים:

  $\forall \quad j \in \{0,1,\ldots,2^n-1\}, \quad j \neq i$.

  - אם אין כניסת איפשור, מניחים כי $En=1$.



## 14. Flip – Flops – טבלאות עירור

*SR-FF*

| S | R | y(t) |
|---|---|---|
| 0 | 0 | y(t-1) |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | מצב אסור |

*D-FF*

| D | y(t) |
|---|---|
| 0 | 0 |
| 1 | 1 |

T-FF

| T | y(t) |
|---|---|
| 0 | y(t-1) |
| 1 | y'(t-1) |

## 15. זמנים ברכיבים לוגיים

$T_{pd}$ זמן השהייה (propagation delay) – משך הזמן מרגע שינוי בכניסת הרכיב ועד לעדכון היציאה.

$T_{cd}$ זמן זיהום (contamination delay) – משך הזמן מרגע שינוי בכניסת הרכיב, שבו נשארת היציאה בערכה היציב <u>הקודם</u>.

## 16. זמנים ברכיבי זכרון (Latches או Flip-Flops)

$T_{setup}$ - משך הזמן <u>לפני</u> נעילת השעון, שעל המידע בכניסות להיות יציב.

$T_{hold}$ - משך הזמן <u>אחרי</u> נעילת השעון, שעל המידע בכניסות להשאר יציב.

$T_{pdC\to Q}$ - ההשהייה מרגע הנעילה ועד להתעדכנות היציאות.

$T_{cdC\to Q}$ - משך הזמן מרגע הנעילה, שבו נשארת היציאה בערכה היציב <u>הקודם</u>.

אם הכניסה לרכיב זכרון איננה מקיימת את הדרישות $T_{setup}$, $T_{hold}$ עלולה היציאה להיות במצב על-יציב (meta-stable) במשך זמן מה. במצב זה היציאה בלתי יציבה ובלתי מוגדרת. לאחר מכן עלולה היציאה להתייצב באפס או באחד באופן אקראי, ללא קשר למצב הקודם ולערך הכניסה.

## 17. זמנים במערכות עקיבה סינכרוניות

א) מחזור השעון ארוך מסכום $t_{pC\text{-}Q}$ של הזכרונות, $t_{pd}$ של הלוגיקה הצירופית, ו-$t_s$ של הזכרונות:

$$T_{Clk} > t_{pdC\to Q}(FF) + t_{pd}(\text{logic}) + t_{setup}(FF)$$

ב) $t_{hold}$ של הזכרונות צריך להיות קטן מסכום $t_{cd}$ של הזכרונות ושל המערכת הצירופית:

$$t_{hold}(FF) < t_{cdC\to Q}(FF) + t_{cd}(\text{logic}).$$

ג) הכניסות למערכת הצירופית צריכות להיות מוחזקות בערכים החוקיים הנכונים במשך לפחות $t_{pd}$ של הלוגיקה הצירופית ועוד $t_s$ של הזכרונות לפני עליית השעון.

ד) הכניסות למערכת הצירופית צריכות להיות מוחזקות בערכים החוקיים במשך לפחות $t_{hold}$ של רכיבי הזיכרון פחות $t_{cd}$ של הלוגיקה הצירופית אחרי עליית השעון.

מערכת שפונקציות המוצא שלה תלויות במצב הנוכחי ובכניסות הנוכחיות נקראת מכונת Mealy.

מערכת שפונקציות המוצא שלה תלויות אך ורק במצב הנוכחי נקראת מכונת Moore.

**.18 אנליזה של מערכת עקיבה סינכרונית**

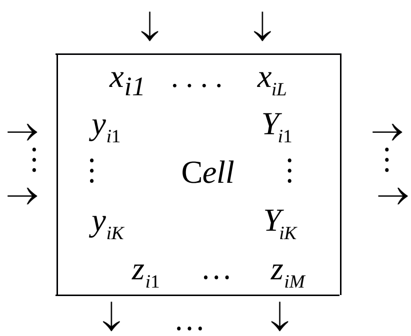
א. מציאת פונקציות הכניסות לרכיבי הזכרון, ופונקציות המוצא.
ב. טבלת עירור.
ג. טבלת מצבים סימבולית.
ד. דיאגרמת מצבים.
ה. מה המערכת עושה?

**.19 סינתזה של מערכת עקיבה סינכרונית**

א דיאגרמת מצבים.
ב. טבלת מצבים.
ג. צמצום מצבים.
ד. בחירת אלמנטי זכרון (סוג ה- FF).
ה. הקצאת מצבים (קוד לכל מצב).
ו. טבלת עירור.
ז. מציאת פונקציות הכניסות לרכיבי הזכרון, ופונקציות המוצא.
ח. שרטוט המעגל.

**.20 מערכת איטרטיבית**

המערכת בנויה מ-$N$ תאים זהים. במערכת איטרטיבית המקום מחליף את הזמן. כל סדרת הכניסה (באורך של עד $N\times L$ סיביות) מופיעה במקביל.
תא אופייני:




- $x_{i1},\ldots,x_{iL}$ - כניסות "מבחוץ". במערכות סינכרוניות אלה היו כניסות של מחזור $i$–י.
- $y_{i1},\ldots,y_{iK}$ - כניסות מהתא הקודם. במערכות סינכרוניות אלה היו ערכים ברכיבי זכרון בתחילת מחזור $i$–י.
- $Y_{i1},\ldots,Y_{iK}$ - יציאות לתא הבא. במערכות סינכרונית אלה היו ערכים ברכיבי זכרון בסוף מחזור $i$–י.
- $z_{i1},\ldots,z_{iM}$ - יציאות "החוצה". במערכות סינכרוניות אלה היו יציאות של מחזור $i$–י.

ניתן לחבר $N$ תאים בחוג ע"י $K$ רכיבי זכרון, לצורך עיבוד קלט באורך "אינסופי" המחולק לקטעים באורך $N\times L$ כל אחד.

### 21. צמצום מכונות מצבים

א. החלוקה ההתחלתית היא קבוצת כל המצבים (שהם 0-שקולים): $P_0=\{S_i\}$.

ב. חלוקה ראשונה ($P_1$) לפי היציאות: כל המצבים באותה מחלקת שקילות הם 1-שקולים זה לזה, כלומר יש להם אותן יציאות בדיוק.

ג. המשך החלוקה לא לפי היציאות, אלא לפי משפט Moore: שני מצבים הם k-שקולים אס"ם הם:

(1) (k-1)-שקולים, וגם -

(2) המצבים העוקבים שלהם (בהתאמה לפי הכניסות) גם הם (k-1)-שקולים.

ד. בכל שלב, החלוקה הינה עידון של החלוקה הקודמת.

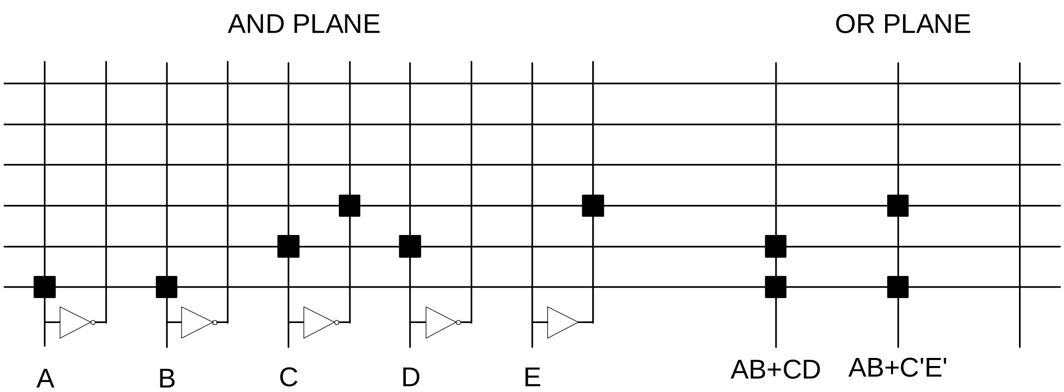
האלגוריתם עוצר כאשר $P_i=P_{i-1}$.

### 22. מבנים רגולריים:

**Prgramable Logic Array (PLA)**

- יש רק לבחור אילו שערים משתתפים במימוש = "תכנות" של ה- PLA.
- לפשטות הסימון מסמנים רק את המקומות של השערים המשתתפים.

<u>דוגמה</u>:




**אפיון ה- PLA שבדוגמה**:

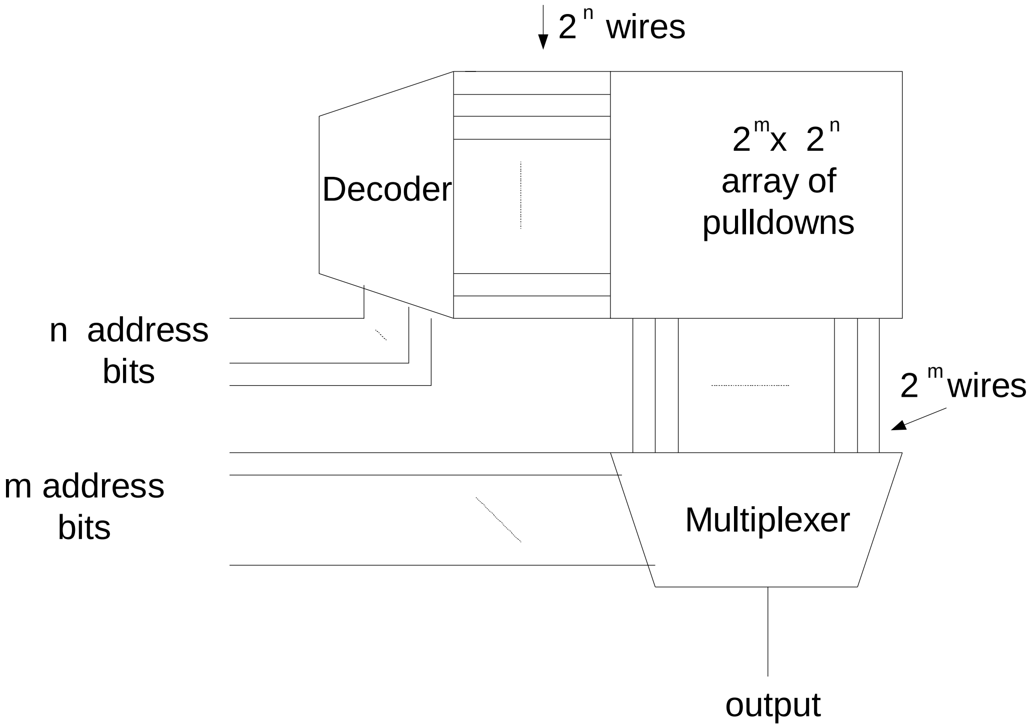
מספר משתני הכניסה – 5.

מספר המכפלות האפשריות – 6.

מספר היציאות האפשריות – 3.

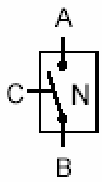
## Read Only Memory (ROM)

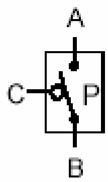
המבנה העקרוני של ROM דו-מימדי בעל $m+n$ קווי כתובת ( $2^{m+n}$ תאי זיכרון) הוא:




**23. מימוש שערים ע"י מתגים:**

טבלאות האמת של שני סוגי המתגים:




**מתג $P$**

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

**מתג $N$**

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