

תרגיל בית – 4

שאלה מספר 1 - פענוח

נתון קוד ליניארי מעל $GF(3)$ הנפרש על ידי המטריצה היוצרת $G = \begin{pmatrix} 0 & 2 & 2 & 1 \\ 1 & 2 & 0 & 2 \end{pmatrix}$.

- בנו Standard Array למטרת פענוח.
- ציינו מיהם ה Cosset Leaders ואת הסינדרומים שלהם.
- בהנחה שכל סימבול מיוצג בעזרת שלושה ביטים, מהו גודל הזיכרון הנדרש למימוש מפענח על בסיס SA? מה גודל הזיכרון הנדרש למימוש מפענח על בסיס טבלת סינדרום-CL?
- נתקבלה המילה (1022) , מצאו מה שודר:

- לפי Standard Array
- לפי טבלת סינדרום-CL

פתרון:

א.

| | 00 | 20 | 10 | 02 | 22 | 12 | 01 | 21 | 11 |
|---|------|------|------|------|------|------|------|------|------|
| 1 | 0000 | 0112 | 0221 | 2101 | 2210 | 2022 | 1202 | 1011 | 1120 |
| 2 | 1000 | 1112 | 1221 | 0101 | 0210 | 0022 | 2202 | 2011 | 2120 |
| 3 | 2000 | 2112 | 2221 | 1101 | 1210 | 1022 | 0202 | 0011 | 0120 |
| 4 | 0100 | 0212 | 0021 | 2201 | 2010 | 2122 | 1002 | 1111 | 1220 |
| 5 | 0200 | 0012 | 0121 | 2001 | 2110 | 2222 | 1102 | 1211 | 1020 |
| 6 | 0010 | 0122 | 0201 | 2111 | 2220 | 2002 | 1212 | 1021 | 1100 |
| 7 | 0020 | 0102 | 0211 | 2121 | 2200 | 2012 | 1222 | 1001 | 1110 |
| 8 | 0001 | 0110 | 0222 | 2102 | 2211 | 2020 | 1200 | 1012 | 1121 |
| 9 | 0002 | 0111 | 0220 | 2100 | 2212 | 2021 | 1201 | 1010 | 1122 |

השורה הראשונה בטבלה מכילה את כל מילות הקוד. העמודה השמאלית מכילה את אוסף כל ווקטורי השגיאה הניתנים לתיקון (הקוד הנ"ל בעל יכולת תיקון שגיאה אחת בלבד). כל שאר התאים הם סכום השורה והעמודה המתאימות להם.

הערה: הסכום מתבצע לכל בית בנפרד ובמודול 3.

ב. ה- Cosset Leader הוא הווקטור בעל משקל Hamming הנמוך ביותר בכל שורה בטבלה. לחישוב הסינדרום יש למצוא את מטריצת הבדיקה H :

$$G = \begin{pmatrix} 0 & 2 & 2 & 1 \\ 1 & 2 & 0 & 2 \end{pmatrix} \xrightarrow{II \leftrightarrow I} \begin{pmatrix} 1 & 2 & 0 & 2 \\ 0 & 2 & 2 & 1 \end{pmatrix} \xrightarrow{2 \cdot II \rightarrow I} \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 2 & 2 & 1 \end{pmatrix} \xrightarrow{2 \cdot II \rightarrow II} \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{pmatrix} \begin{matrix} \\ I_k \\ A \end{matrix}$$

$$\Rightarrow$$

$$H = \begin{pmatrix} -1 & -1 & 1 & 0 \\ -1 & -2 & 0 & 1 \end{pmatrix} \xrightarrow{\text{mod } 3} \begin{pmatrix} 2 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \begin{matrix} \\ -A^T \\ I_{n-k} \end{matrix}$$

הסינדרום מתקבל על ידי הכפלת כל Cosset Leader במטריצת הבדיקה.
לסיכום, קיבלנו טבלת סינדרום-CL:

| Cosset Leader | Syndrome $s_1 s_0$ |
|---------------|--------------------|
| 0000 | 00 |
| 1000 | 22 |
| 2000 | 11 |
| 0100 | 21 |
| 0200 | 12 |
| 0010 | 10 |
| 0020 | 20 |
| 0001 | 01 |
| 0002 | 02 |

1. גודל הזיכרון הנדרש במימוש מפענח על בסיס SA:

ישנם $q^n = 3^4 = 81$ תאים בטבלה. כל תא בטבלה מכיל ארבעה סימבולים וכן מיוצג בעזרת שלושה ביטים לכן נצטרך סה"כ $81 \cdot 4 \cdot 3 = 972$ יחידות זיכרון.

גודל הזיכרון הנדרש במימוש מפענח על בסיס טבלת סינדרום-CL:

ישנם $2 \cdot q^r = 2 \cdot 3^2 = 18$ תאים בטבלה כאשר חצי מהם מכילים CL וחצי מכילים סינדרום. כל תא בטבלה המייצג CL מכיל ארבעה סימבולים לעומת זאת, תא המייצג סינדרום מכיל שני סימבולים בלבד. כל סימבול מיוצג על ידי שלושה ביטים ולכן נצטרך סה"כ $9 \cdot 4 \cdot 3 + 9 \cdot 2 \cdot 3 = 162$ יחידות זיכרון.

רואים אם כן מימוש מפענח על בסיס טבלת סינדרום-CL עדיף על SA מבחינת החיסכון בחומרה (מספר יחידות הזיכרון הנדרשות לפענוח).

ד. לפי Standard Array :

כאשר מתקבלת המילה (1022) יש להתבונן ב-SA ולראות תחת איזה מילת קוד (מתוך השורה הראשונה) נמצאת המילה המתקבלת. במקרה שלנו המילה המתקבלת (1022) נמצאת בעמודה השייכת למילה (2022) ולכן נחליט כי שודר (2022) .

לפי טבלת סינדרום-CL:

כאשר מתקבלת המילה (1022) יש לחשב את הסינדרום שלה:

$$S = Hy^T = \begin{pmatrix} 2 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

לאחר מכן, יש למצוא מתוך טבלת סינדרום-CL את ה-Cosset Leader המתאים

לסינדרום $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. במקרה שלנו ה-Cosset Leader הוא (2000) .

כעת נחסר את ה-Cosset Leader (ווקטור השגיאה) מתוך המילה המתקבלת ונקבל:

$$\begin{array}{r} 1022 \\ 2000 \\ \hline 2022 \end{array}$$

ולכן נחליט כי שודר (2022) .

הערה: פעולת החיסור מתבצעת לכל סימבול בנפרד ובמודול 3.

שאלה מספר 2 - מימוש בחומרה:

א. נתון קוד ליניארי מעל $GF(3)$ הנפרש על ידי המטריצה היוצרת $G = \begin{pmatrix} 1 & 2 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$. ממשו מעגל

מקודד (צירופי) על פי מטריצה G . (ניתן להשתמש ברכיבים שתכננו בכיתה כקופסא שחורה)

ב. ממשו מפענח על פי מטריצה H עבור הקוד שראינו בתרגול כיתה.

רמז: השתמשו במעגל חישוב הסינדרום כקופסא שחורה וכן נצלו את העובדה כי קיים Cosset

Leader יחיד עבור כל Cosset ב-Standard Array.

פתרון:

א. תכנון מקודד לפי $G = \begin{pmatrix} 1 & 2 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$ מעל $GF(3)$:

משתני כניסה: M_1, M_0 כאשר כל אחד מהם בגודל שני ביטים (לפי המיפוי להלן).

משתני יציאה: C_3, C_2, C_1, C_0 כאשר כל אחד מהם בגודל שני ביטים (לפי המיפוי להלן).

כניסות ויציאות המערכת נקבעות על פי המיפוי הבא:

| סימבול | מיפוי |
|--------|-------|
| 0 | 00 |
| 1 | 01 |
| 2 | 10 |

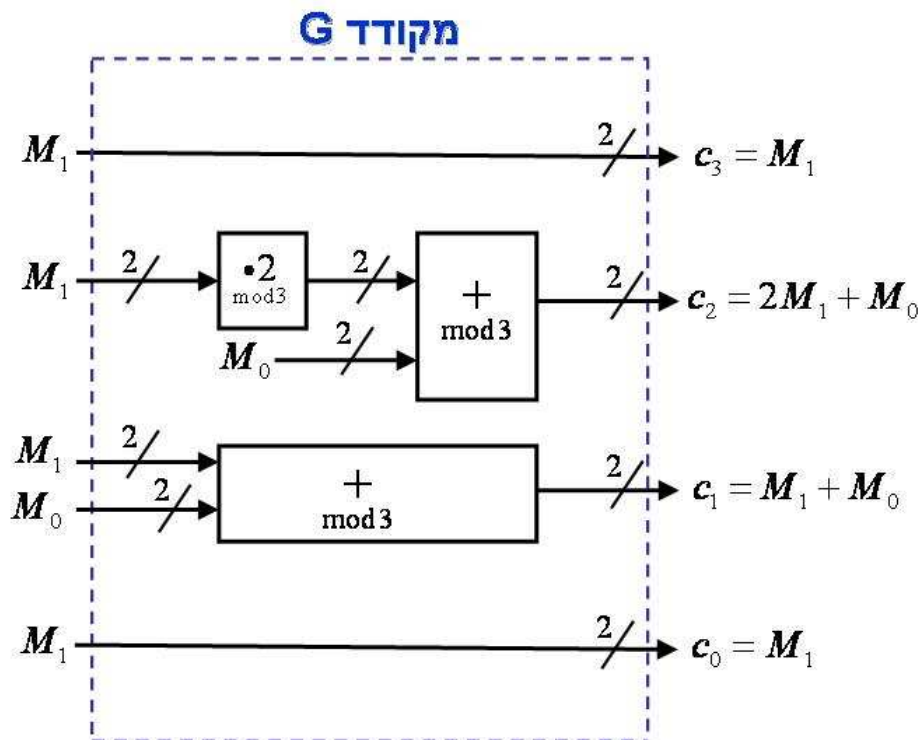
תיאור המערכת:



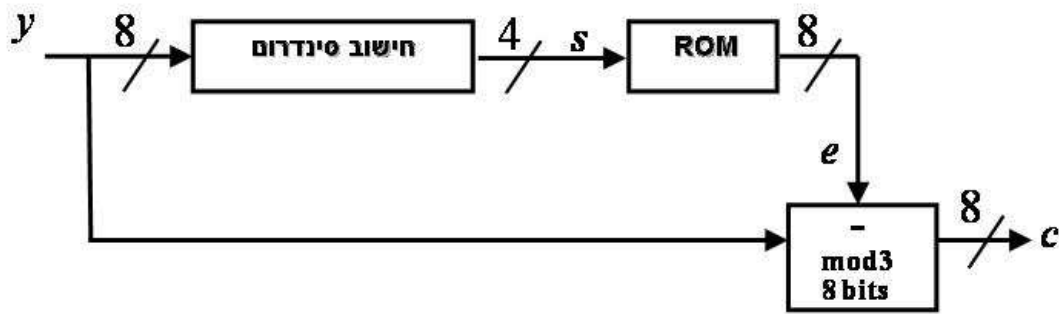
מימוש:

פעולת הקידוד היא מכפלת מילת הכניסה $M_1 M_0$ במטריצה היוצרת G כלומר:

$$(M_1 M_0) \begin{pmatrix} 1 & 2 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} = (M_1, 2M_1 + M_0, M_1 + M_0, M_1)$$



ב. המפענח שנבחר לממש יהיה על בסיס טבלת סינדרום-CL (משיקולי חיסכון בחומרה שכבר הזכרנו)



ייצוג כל סימבול יתבצע לפי אותו מיפוי שראינו בסעיף קודם.

גודל המילה בכניסה (y) הוא 8 ולא 4 כיוון שצריך שני ביטים לייצוג כל סימבול (מדובר ב $GF(3)$).

ה-ROM מכיל את טבלת סינדרום-CL. גודל ה-ROM (חישובו בכיתה) 108 יחידות זיכרון.

ניתן לייעל את המפענח על ידי צמצום ה-ROM בעזרת מפת קרנו עבור כל אחת מיציאותיו.

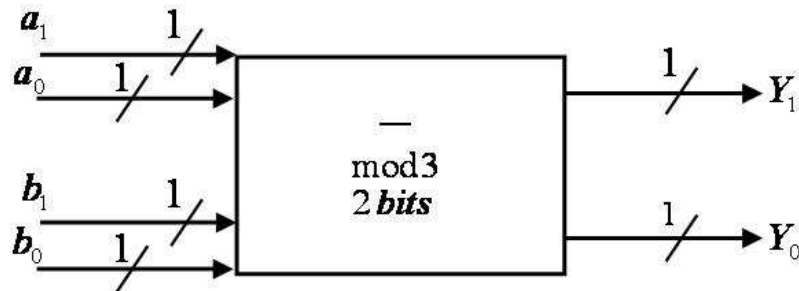
את המחסר שמונה ביטים במודול 3 יש לממש בנפרד.

ראשית נממש מחסר עבור שני ביטים:

משתני כניסה: a_1, a_0, b_1, b_0 כאשר כל אחד מהם בגודל ביט אחד.

משתני יציאה: Y_1, Y_0 כאשר כל אחד מהם בגודל ביט אחד.

תיאור המערכת:



טבלת אמת

| a_1, a_0 | b_1, b_0 | Y_1, Y_0 |
|------------|------------|------------|
| 00 | 00 | 00 |
| | 01 | 10 |
| | 11 | ** |
| | 10 | 01 |
| 01 | 00 | 01 |
| | 01 | 00 |
| | 11 | ** |
| | 10 | 10 |
| 10 | 00 | 10 |
| | 01 | 01 |
| | 11 | ** |
| | 10 | 00 |
| 11 | 00 | ** |
| | 01 | ** |
| | 11 | ** |
| | 10 | ** |

נבצע מינימיזציה בעזרת מפת קרנו:

עבור Y_1 מתקבל:

| a_1, a_0 b_1, b_0 | 00 | 01 | 11 | 10 |
|--------------------------|----|----|----|----|
| 00 | 0 | 0 | * | 1 |
| 01 | 1 | 0 | * | 0 |
| 11 | * | * | * | * |
| 10 | 0 | 1 | * | 0 |

$$Y_1 = a_1 \bar{b}_1 \bar{b}_0 + \bar{a}_1 \bar{a}_0 b_0 + a_0 b_1$$

עבור Y_0 מתקבל:

| a_1, a_0 b_1, b_0 | 00 | 01 | 11 | 10 |
|--------------------------|----|----|----|----|
| 00 | 0 | 1 | * | 0 |
| 01 | 0 | 0 | * | 1 |
| 11 | * | * | * | * |
| 10 | 1 | 0 | * | 0 |

$$Y_0 = a_1 b_0 + a_0 \bar{b}_1 \bar{b}_0 + \bar{a}_1 \bar{a}_0 b_1$$

בכדי לממש מחסר 8 ביטים יש להשתמש בארבעה רכיבים מהסוג שתכננו ולהתייחס ליציאות כל רכיב כעל תוצאה נפרדת. ניתן לעשות זאת כיוון שפעולת החיסור נעשית לכל סימבול בנפרד.

שאלה מספר 3

נתון קוד ליניארי מעל $GF(2)$ הנפרש על ידי המטריצה היוצרת $G = \begin{pmatrix} 1010101010 \\ 0101010101 \end{pmatrix}$

ממשו (בצורה מלאה ללא קופסאות שחורות) מפענח המתקן עד שגיאה אחת ומוציא הודעת שגיאה אם חלו יותר שגיאות.

פתרון:

המפענח שנבחר לממש יהיה על בסיס טבלת סינדרום-CL. תהליך הפענוח מתבסס על חישוב הסינדרום של המילה המתקבלת, מציאת ווקטור השגיאה (Cosset Leader) המתאים לסינדרום זה ולבסוף חיסור (במודול 2) ווקטור השגיאה מתוך המילה המתקבלת.

משתני כניסה: y_9, \dots, y_0 כאשר כל אחד מהם בגודל ביט אחד. (המילה המתקבלת)

משתני יציאה: c_9, \dots, c_0 כאשר כל אחד מהם בגודל ביט אחד. (המילה המתוקנת)

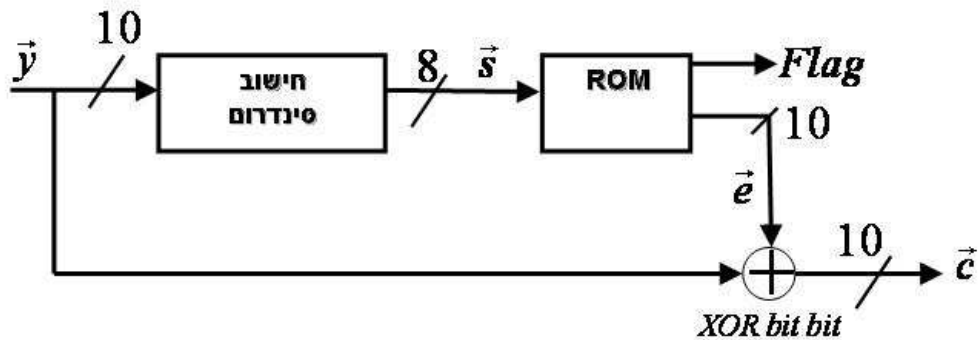
$Flag$ - ביט אשר תפקידו לציין האם חלו לפחות שתי שגיאות במילה המתקבלת.

$Flag = 1 \leftarrow$ חלו לפחות שתי שגיאות במילה המתקבלת.

תיאור המערכת הכללית:



תיאור המערכת ברמת components:



מימוש חישוב סינדרום:

משתני כניסה: y_9, \dots, y_0 כאשר כל אחד מהם בגודל ביט אחד. (המילה המתקבלת)

משתני יציאה: S_7, \dots, S_0 כאשר כל אחד מהם בגודל ביט אחד. (סינדרום המילה המתקבלת)

מימוש:

ראשית נמצא את H למטרת חישוב הסינדרום.

$$G = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \Rightarrow H = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -A^T & I_8 \end{pmatrix}$$

חישוב הסינדרום מתבצע באופן הבא:

$$S = Hy^T = \begin{pmatrix} 1010000000 \\ 0101000000 \\ 1000100000 \\ 0100010000 \\ 1000001000 \\ 0100000100 \\ 1000000010 \\ 0100000001 \end{pmatrix} \begin{pmatrix} y_9 \\ y_8 \\ y_7 \\ y_6 \\ y_5 \\ y_4 \\ y_3 \\ y_2 \\ y_1 \\ y_0 \end{pmatrix} = \begin{pmatrix} S_7 \\ S_6 \\ S_5 \\ S_4 \\ S_3 \\ S_2 \\ S_1 \\ S_0 \end{pmatrix}$$

לאחר פתרון כפל המטריצות מתקבל את משוואות המערכת:

$$S_7 = y_9 + y_7$$

$$S_6 = y_8 + y_6$$

$$S_5 = y_9 + y_5$$

$$S_4 = y_8 + y_4$$

$$S_3 = y_9 + y_3$$

$$S_2 = y_8 + y_2$$

$$S_1 = y_9 + y_1$$

$$S_0 = y_8 + y_0$$

משוואות אלו ניתנות למימוש בעזרת שערים לוגיים (שערי OR).

מימוש ROM:

משתני כניסה: S_7, \dots, S_0 כאשר כל אחד מהם בגודל ביט אחד. (סינדרום המילה המתקבלת)

משתני יציאה: c_9, \dots, c_0 כאשר כל אחד מהם בגודל ביט אחד. (המילה המתוקנת)

$Flag$ - ביט אשר תפקידו לציין האם חלו לפחות שתי שגיאות במילה המתקבלת.

$$Flag = 1 \leftarrow \text{חלו לפחות שתי שגיאות במילה המתקבלת.}$$

מימוש:

רכיב זה ניתן לממש במספר דרכים. הדרך הפשוטה ביותר היא להשתמש בזיכרון המכיל את טבלת סינדרום-CL עם שינויים קלים. השינוי הראשון הוא שבמקרה וחלה יותר משגיאה אחת נוציא את מילת האפס כווקטור שגיאה (וכך לא נתקנה). בכדי שנוכל להבדיל האם המילה הייתה תקינה מלכתחילה או שחלה יותר משגיאה אחת נשתמש ב-Flag. Flag יתווסף לטבלה וערכו יקבע לפי ערך ווקטור השגיאה המקורי (אם משקל Hamming של ווקטור השגיאה המקורי גדול מאחד נוציא $1 = Flag$, אחרת $0 = Flag$).

ברור כי דרך זו היא בזבזנית כיוון שיש לנו טבלה שיכולה לתקן שתי שגיאות ואנו מתקנים שגיאה אחת בלבד.

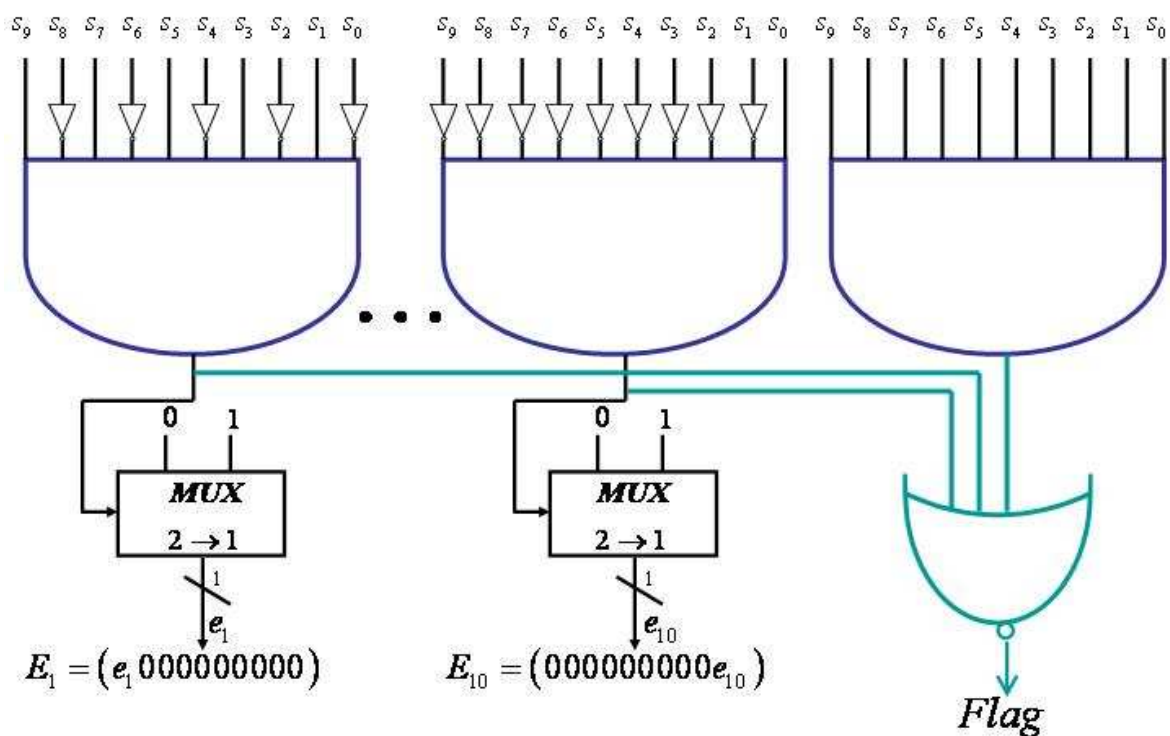
דרך נוספת היא לחשב רק את המינטרמים של הסינדרומים שאמורים להוציא ווקטור שגיאה עם משקל Hamming קטן מ-2. לאחר מכן, ניעזר בעובדה שתבנית ווקטור השגיאה מאופיינת בכך שרק ביט אחד דלוק והשאר אפסים (במקרה שחלה שגיאה אחת) או לחילופין כולם כבויים (במקרה שלא חלה אף שגיאה).

שגיאה אחת יכולה לקרות בכל אחד מעשרת הביטים במילה ולכן ישנם עשרה מינטרמים שונים של סינדרומים שנרצה לממש (אם לא חלה אף שגיאה הסינדרום הוא המינטרם אפס).

מינטרם הסינדרום המתאים לשגיאה במיקום i הוא עמודה h_i במטריצת הבדיקה (חשבו את

הסינדרום וודאו זאת).

נממש בשערים לוגיים:



$$\vec{e} = E_1 \otimes E_2 \otimes E_3 \otimes E_4 \otimes E_5 \otimes E_6 \otimes E_7 \otimes E_8 \otimes E_9 \otimes E_{10}$$

שער AND הימני ביותר מייצג את המינטרם עבור סינדרום אפס. עשרת שערי AND הנותרים מייצגים את כל ה-CL האפשריים של שגיאה אחת. בכל פענוח רק יציאה אחת (מתוך אחד עשר שערי ה-AND) שווה 1 או שכולן שוות לאפס. האחת ששווה 1 תקבע את ערך ווקטור השגיאה (E_i). אם כל היציאות הן אפס הרי שלפנינו מילה עם לפחות שתי שגיאות ונדליק Flag (שער NOR).