
שכבה פיזית, מסגרות, גילוי שגיאות

5 מאי 2026

הרצאה 3

Slides Credits: Steve Zdancewic (UPenn)

נושאים להיום

- שכבה הפיזית
- שכבת העורק
- מסגרות
- שגיאות
- גילוי שגיאות

אתגרים : חיבור פיזי

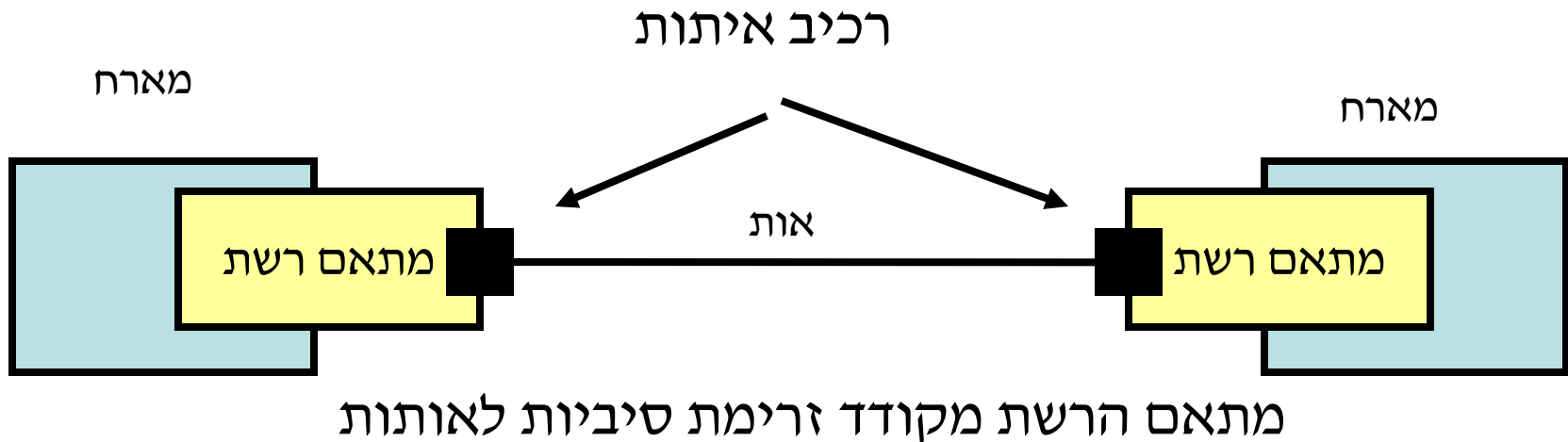
קידוד
ופיענוח
סיביות

שידור אותות

שידור אמין

גילוי ותיקון
שגיאות

רכיבי איתות



הפשטה: נניח שיש 2 סוגי אותות – גבוה ונמוך
בפועל: הפרשי מתח על חוטי נחושת
(מוביל לאתגרי ובעיות קידוד מעניינות)

כרטיסי מתאם רשת (Network Interface Cards – NIC)

Edimax EN-9260TX-E PCI Express 10/100/1000Mbps כרטיס רשת

מק"ט: 26394

מחיר: **₪59**

או 4.92 ₪ לחודש ב- 12 תשלומים (סה"כ 59 ₪)

זמין כעת במלאי בטבריה [בדיקת מלאי בסניף אחר]



לחץ כאן להגדלה ולתמונות נוספות

התמונה להמחשה בלבד

EDIMAX
NETWORKING PEOPLE TOGETHER

TP-Link TL-WN781ND nLITE N PCI Express 150Mbps כרטיס רשת אלחוטי

מק"ט: 24824

מחיר: **₪45**

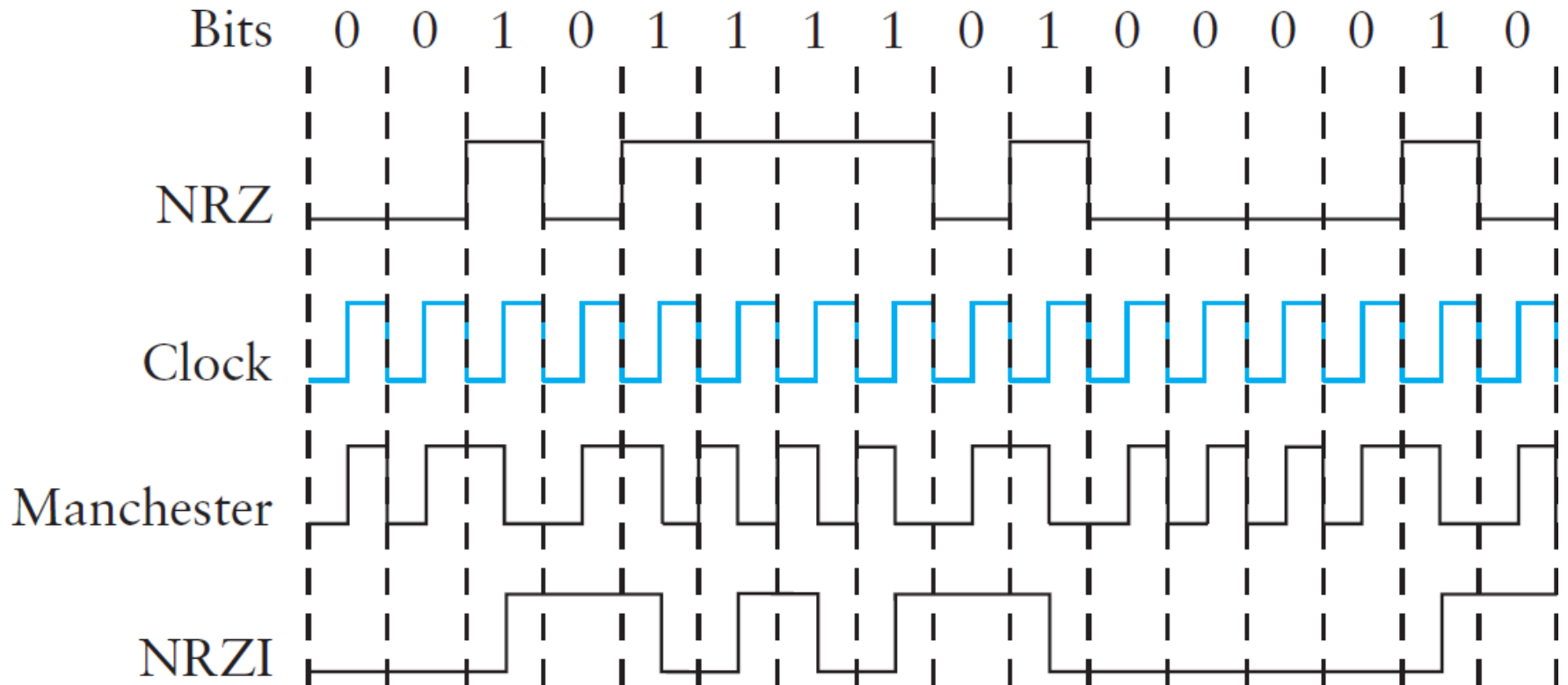
או 3.75 ₪ לחודש ב- 12 תשלומים (סה"כ 45 ₪)

זמין כעת במלאי בטבריה [בדיקת מלאי בסניף אחר]

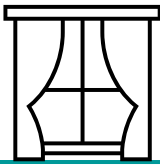


Source: KSP.co.il

שיטות קידוד



- שכבה הפיזית
- שכבת העורק
 - מסגרות
 - שגיאות
- גילוי שגיאות



מסגרות

- צריכים דרך לשלוח בלוקים של נתונים
- איך מתאם הרשת יודע היכן הרצף מתחיל והיכן הוא מסתיים?
- מסגרות הינן יחידות השליחה של נתונים בשכבת העורק
- מבוסס בתים מול מבוסס סיביות
- נקודה לנקודה
(Point-to-point (e.g., **PPP**)
לעומת
גישה מרובה
Multiple access (e.g., **Ethernet**))



- מתייחסים לכל מסגרות כרצף של **בתים**

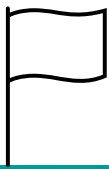
DDCMP

- *Digital Data Communication Message Protocol*
- היה בשימוש במערכת Digital Equipment Company של חברת DEC

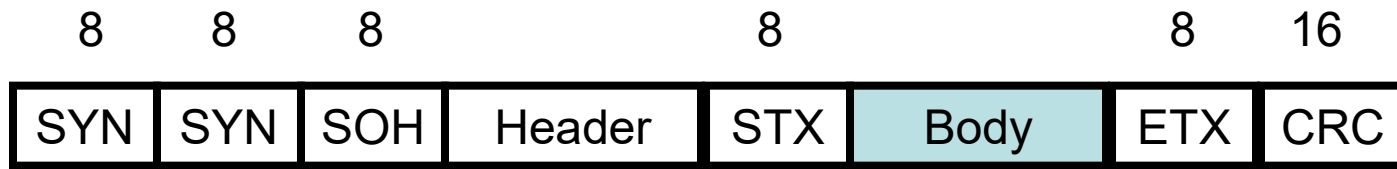
BISYNC

- *Binary Synchronous Communication protocol*
- פותח על ידי IBM בשנות ה-1960

- **שאלה מרכזית:** איזה בתים שייכים למסגרת?



גישת הזקיף : BISYNC



מבנה המסגרת של BISYNC

- זקיפים {
- SYN - סנכרון
 - SOH - תחילת הכותרת (start of header)
 - STX - תחילת הטקסט (start of text)
 - ETX - סוף הטקסט (end of text)
 - CRC - בדיקת תקינות (cyclic redundancy check)

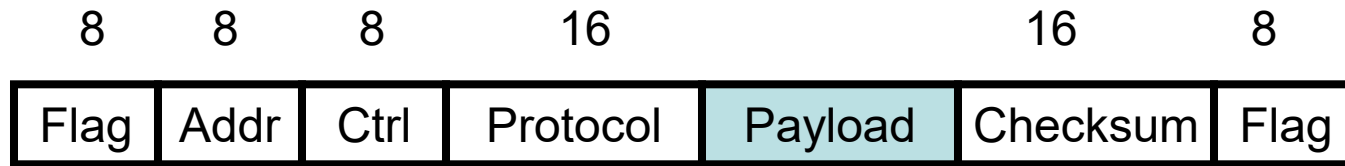


דחיפת אותיות

- מה יקרה אם הקוד ETX (0x03) מופיע בתוך שדה ה-BODY?
עולים לחשוב בטעות שזה סוף ההודעה
- משתמשים ב-"אות חילוץ"
- **DLE** – "חילוץ נתוני עורק" (Data Link Escape) (0x10)
משתמשים בו כמו שכותבים האות \ בשפת C, פייתון, או Java
- "quotes in \"quotes\""
- "slash is \\"

```
>>> print ("Hello World")
Hello World
>>> print ("\\"Hello World\\")
"Hello World"
>>> print ("\'"Hello World\'")
'"Hello World"'
>>> print ("\\\'"Hello World\'")
\'"Hello World"'
```

(PPP) Point-to-Point Protocol



מבנה המסגרת של PPP

דגל – זקוף 01111110

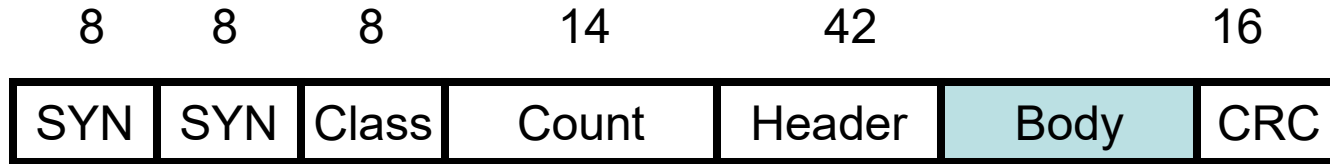
בשימוש על ידי מודמים
(חייגן)

גודל ה-Payload נקבע על ידי משא ומתן

- 1500 בתים ברירת מחדל
- **Link Control Protocol (LCP)**

פרוטוקול – מפתח פענוח ריבוב מזהה את הפרוטוקול ברמה היותר גבוה כמו IP או LCP

ספירת בתים: DDCMP



מבנה המסגרת DDCMP

- במקום זקיפים, כותבים מספר הבתים במסגרת
- מה יקרה אם השדה Count מגיע שגוי?

- שכבה הפיזית
- שכבת העורק
- מסגרות
- שגיאות
- גילוי שגיאות

אתגר : גילוי ותיקון שגיאות

- סיביות שגויות מתרחשות מפעם לפעם
 - הפרעה אלקטרונית
 - רעש בעקבות חום
- יכול להחליף סיביות בודדות (Bit errors)
- יכול לאפס או להחליף רצף של סיביות (Burst errors)

מה עושים
כאשר מגלים
שגיאה?

איך מגלים
שגיאה?

גילוי שגיאות

- עקרון: הכנסת מידע מיותר

שיטות פשוטות לגילוי שגיאות

סיבית זוגיות

- 7 סיביות נתונים
- סיבית 8 הינה סכום 7 סיביות הראשונות $mod 2$

שיטות פשוטות לגילוי שגיאות

אלגוריתם האינטרנט ל-Checksum

- מחברים את כל מילות ההודעה, שולחים את הסכום
- חיבור של 16 סיביות במשלים במתמטיקה משלים ל-1
- <https://www.youtube.com/watch?v=EmUuFRMJbss>

Cyclic Redundancy Check



פרוטוקולי עורק משתמשים
בו

- CRC-32 באתרנט, 802.5, PKzip, ...
- CRC-CCITT ב-HDLC
- CRC-8, CRC-10, CRC-32 ב-ATM

קל לממש

יותר טוב מסיבית זוגיות ו-
checksum

- (למשל, 32 סיביות לשלוח 12000)

ליותר מידע

- ערך בויקיפדיה
- https://en.wikipedia.org/wiki/Cyclic_redundancy_check

Cyclic Redundancy Check (CRC)

- נתייחס להודעה עם $(n + 1)$ סיביות כאל פולינום במעלת n
- מתמטיקה פולינומים מודולו 2
- ערכי הסיביות בהודעה הינן המקדמים
- הודעה = 10011010
- פולינום:

$$\begin{aligned}M(z) &= (1 \times z^7) + (0 \times z^6) + (0 \times z^5) + (1 \times z^4) \\ &+ (1 \times z^3) + (0 \times z^2) + (1 \times z^1) + (0 \times z^0) \\ &= z^7 + z^4 + z^3 + z^1\end{aligned}$$

אלגוריתם CRC

1. השולח והמקבל מחליטים על פולינום מחלק $C(z)$ במעלה k

- דוגמה: $k = 3$

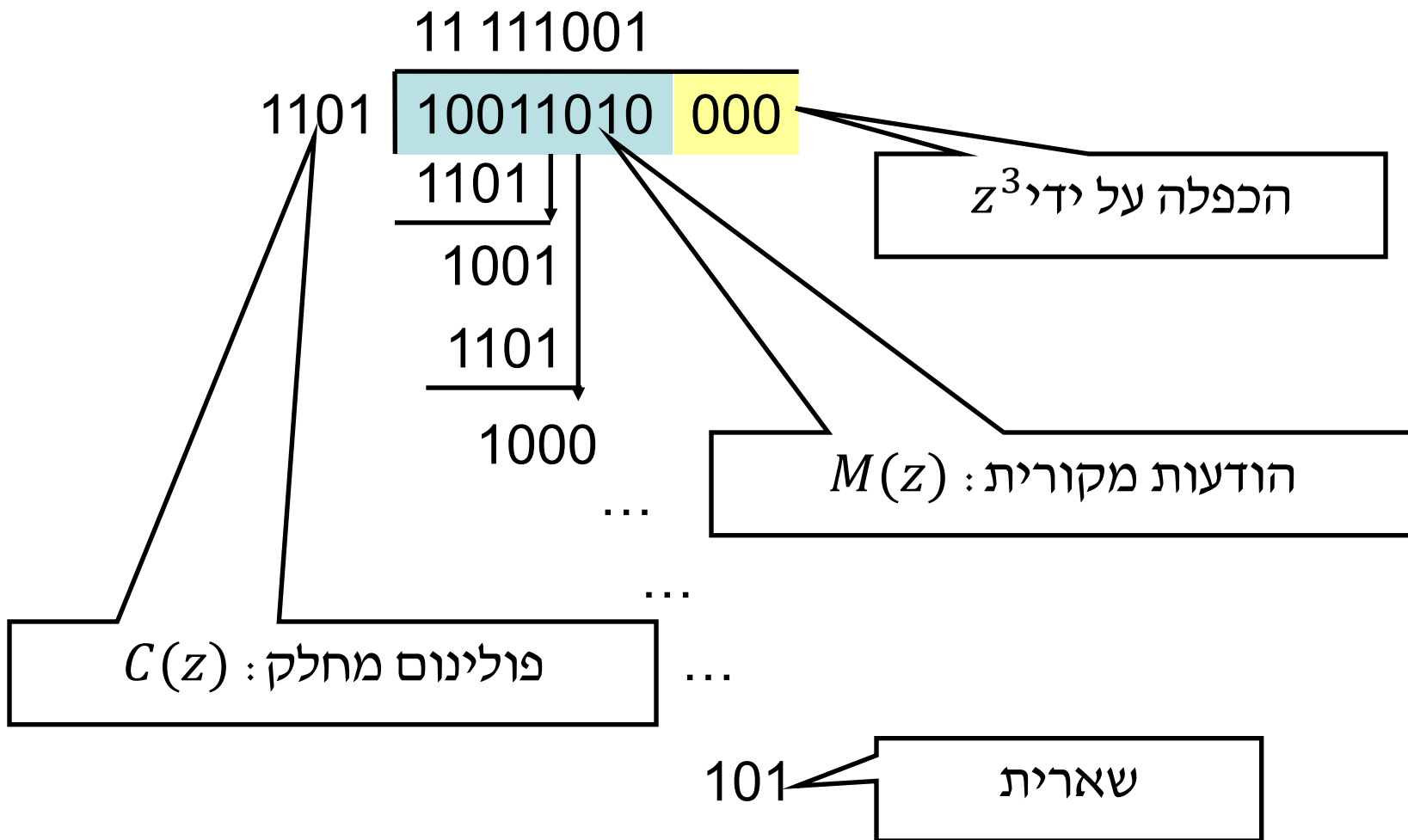
- $C(z) = z^3 + z^2 + 1$

- מקדמים: 1101

2. סיביות גילוי השגיאות הינן השארית של $(M(z) \times z^k)$ מחולק על ידי $C(z)$

3. התוצאה הינה פולינום שליחה בגודל $n + k$ סיביות $P(z)$ שמתחלק על ידי $C(z)$

חישוב CRC לדוגמה



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1					1	1	1	1	1	0	0	1			
2	1	1	0	1	1	0	0	1	1	0	1	0	0	0	0
3					1	1	0	1							
4					0	1	0	0	1						
5						1	1	0	1						
6						0	1	0	0	0					
7							1	1	0	1					
8							0	1	0	1	1				
9								1	1	0	1				
10								0	1	1	0	0			
11									1	1	0	1			
12									0	0	0	1	0	0	0
13												1	1	0	1
14												0	1	0	1

חישוב CRC לדוגמה

$$\begin{array}{r} z^3 \times M(z) \quad 10011010 \quad 000 \\ + \quad \text{שארית} \quad 101 \\ \hline P(z) \text{ הודעה לשלוח} \quad 10011010 \quad 101 \end{array}$$

- המקבל בודק ש- $C(z)$ מחלק את ההודעה שהתקבלה

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1					1	1	1	1	1	0	0	1			
2	1	1	0	1	1	0	0	1	1	0	1	0	1	0	1
3					1	1	0	1							
4					0	1	0	0	1						
5						1	1	0	1						
6						0	1	0	0	0					
7							1	1	0	1					
8							0	1	0	1	1				
9								1	1	0	1				
10								0	1	1	0	0			
11									1	1	0	1			
12									0	0	0	1	1	0	1
13												1	1	0	1
14												0	0	0	0

גילוי שגיאות על ידי CRC

- צריכים לבחור מחלק $C(z)$ טוב
- יש הרבה אפשרויות תקניות:

– CRC-8, CRC-10, CRC-12, CRC-16, CRC-32, –

– CRC-32: $0x04C11DB7$ (1 0000 0100 1100 0001 0001 1101 1011 0111 or $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$)

מגלה:

כל שגיאת
רצף בגודל
 $\leq k$

כל מספר אי
זוגי של
שגיאות אם
 $(z + 1)$
מחלק $C(z)$

כל שגיאה
של 2 סיביות
כל עוד ו-
 $C(z)$ בנוי
מ-3 רכיבים

כל שגיאה
של 1 סיבית
כל עוד
ומקדמי z^k
ו- z^0 שוות 1

מימוש CRC

- קל למימוש בחומרה
- חיסור בבסיס 2 הוא פשוט XOR
- רגיסטר הזזה עם k סיביות עם שערי XOR לפני כל מקדם 1 ב- $C(z)$ (חוץ מהראשון)
- מזיזים את ההודעה לשבב, השארית מתמלאת ברגיסטרים

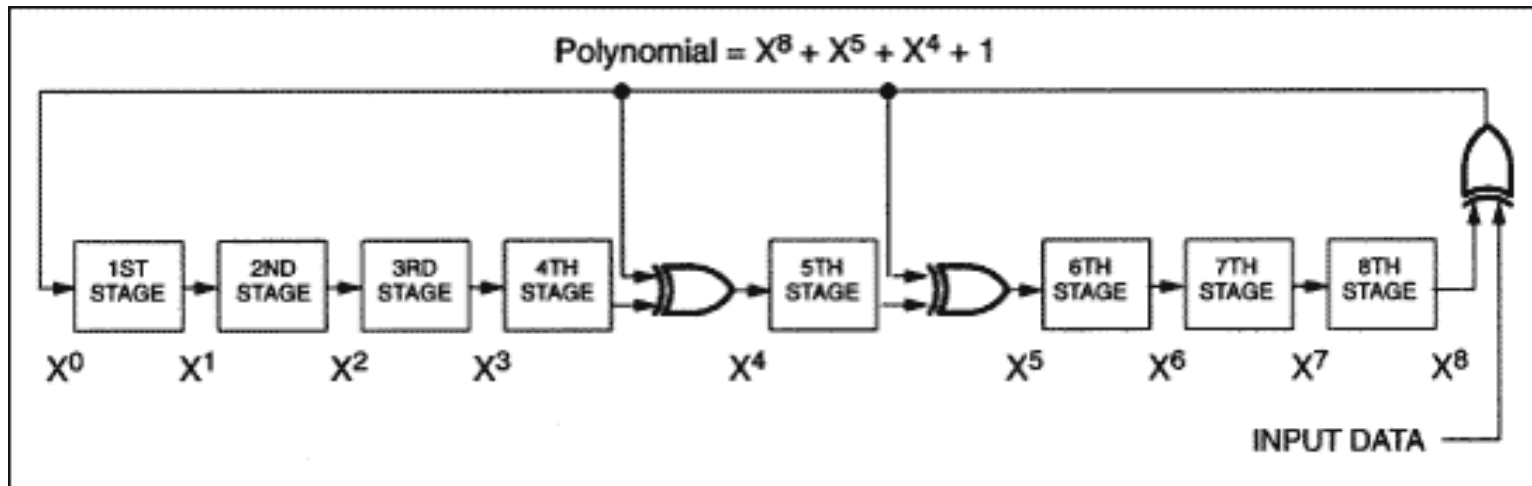
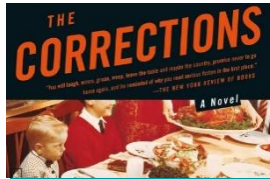


Image source: <https://www.analog.com/en/resources/technical-articles/understanding-and-using-cyclic-redundancy-checks-with-maxim-1-wire-and-ibutton-products.html>



קודים לתיקון שגיאות

שיקול דעת הנדסי

- גילוי שגיאות מחייב שליחה מחדש
- תיקון שגיאות מחייב שליחת יותר סיביות תמיד

ניתן להשתמש במידע מיותר לתקן שגיאות מסוימות

- בדרך כלל דורש יותר מידע מיותר

Forward Error Correction שימושי כאשר :

- הסיכוי לשגיאות גבוה (למשל, חיבור אל-חוטי)
- השהייה ארוך מדי לשליחה מחדש יעילה (למשל, שידורי לוויין)

- שכבה הפיזית
- שכבת העורק
- מסגרות
- שגיאות
- גילוי שגיאות