Chapter 10
Error Detection and Correction

Jirasak Sittigorn

Department of Computer Engineering
Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang

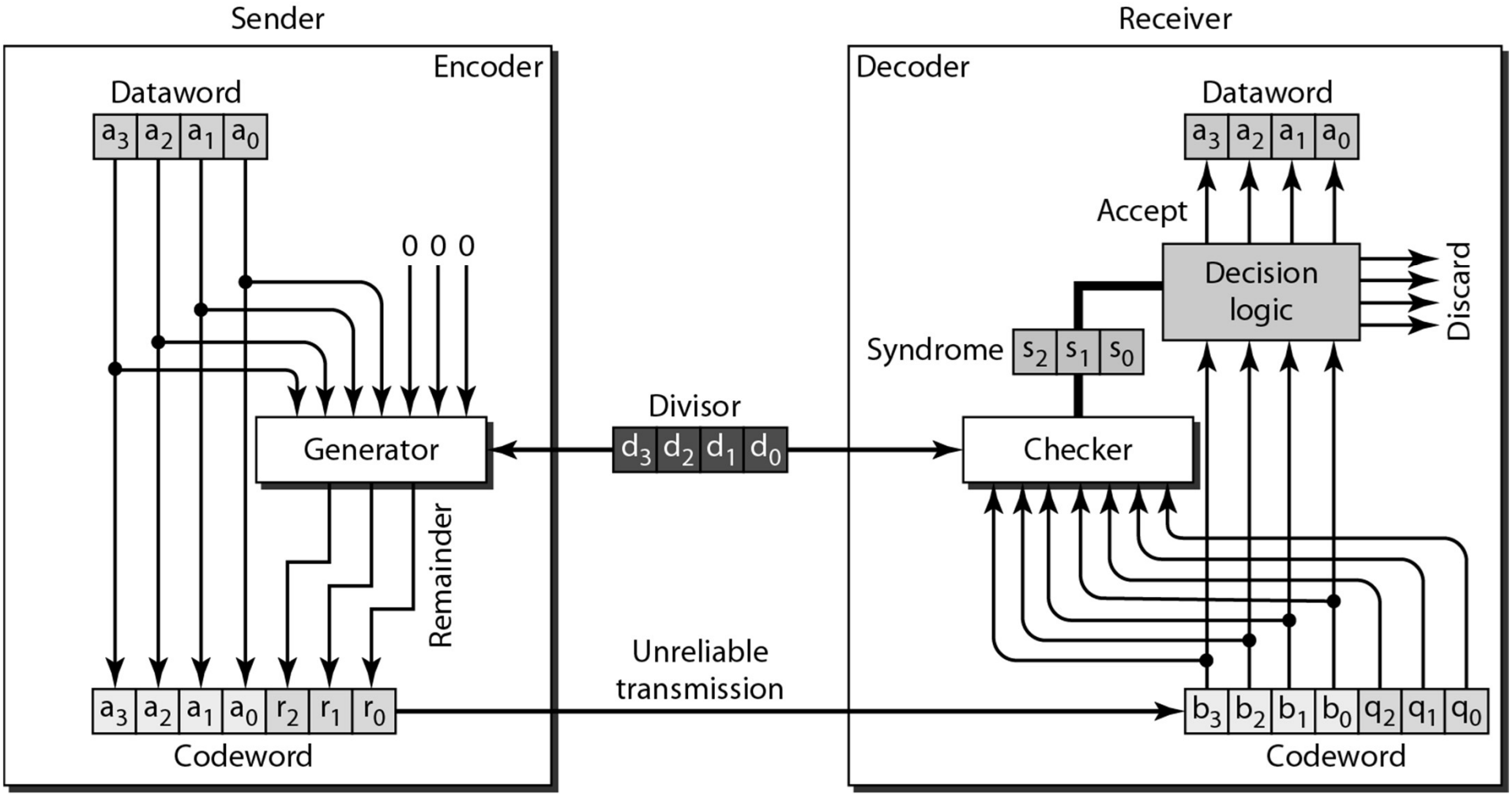
CYCLIC CODES

- Cyclic Redundancy Check CRC
- Hardware Implementation
- Polynomials
- Cyclic Code Analysis

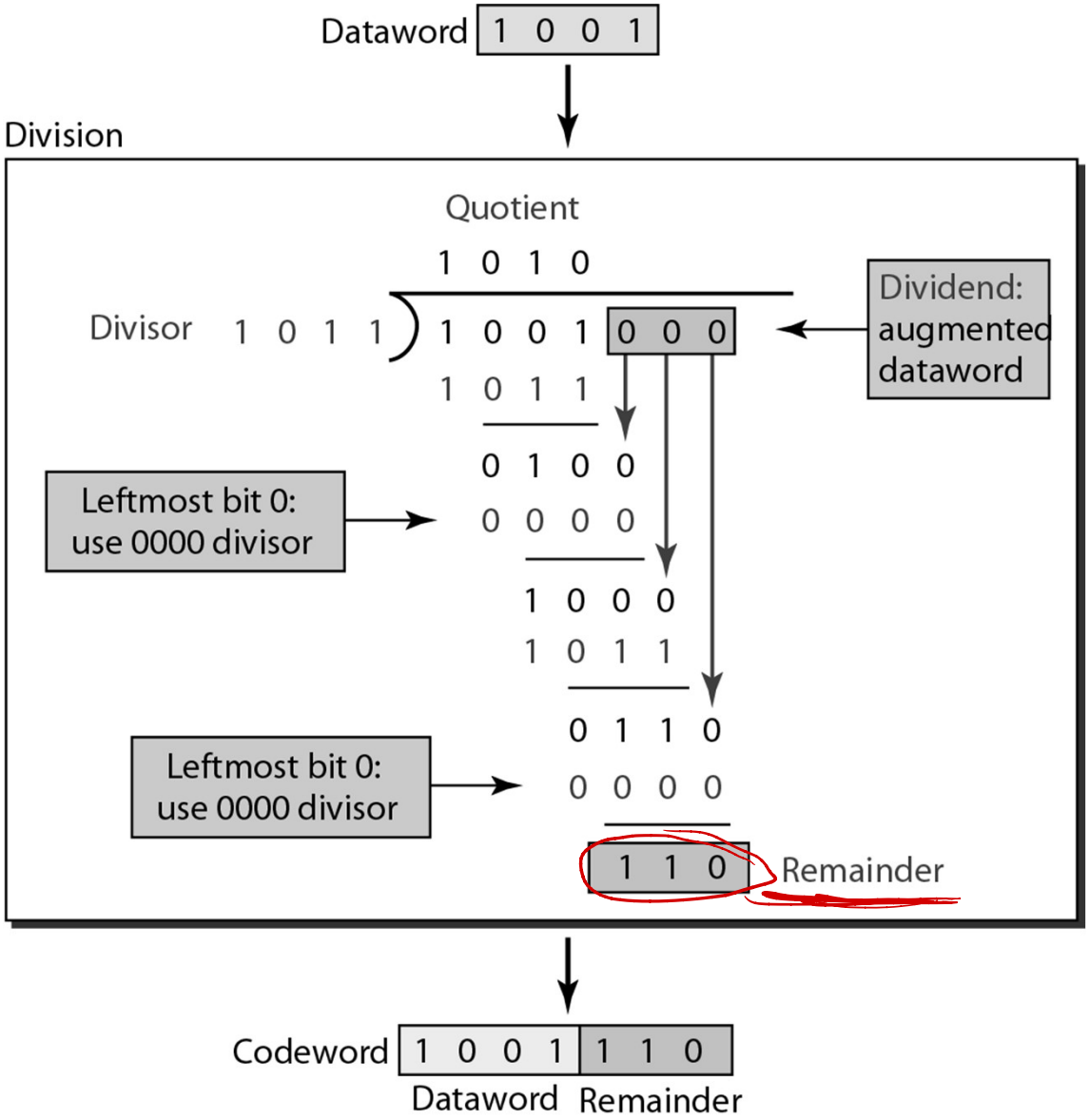
A CRC code with C(7, 4)

Dataword	*Codeword*	*Dataword*	*Codeword*
0000	0000000	1000	1000101
0001	0001011	1001	1001110
0010	0010110	1010	1010011
0011	0011101	1011	1011000
0100	0100111	1100	1100010
0101	0101100	1101	1101001
0110	0110001	1110	1110100
0111	0111010	1111	1111111

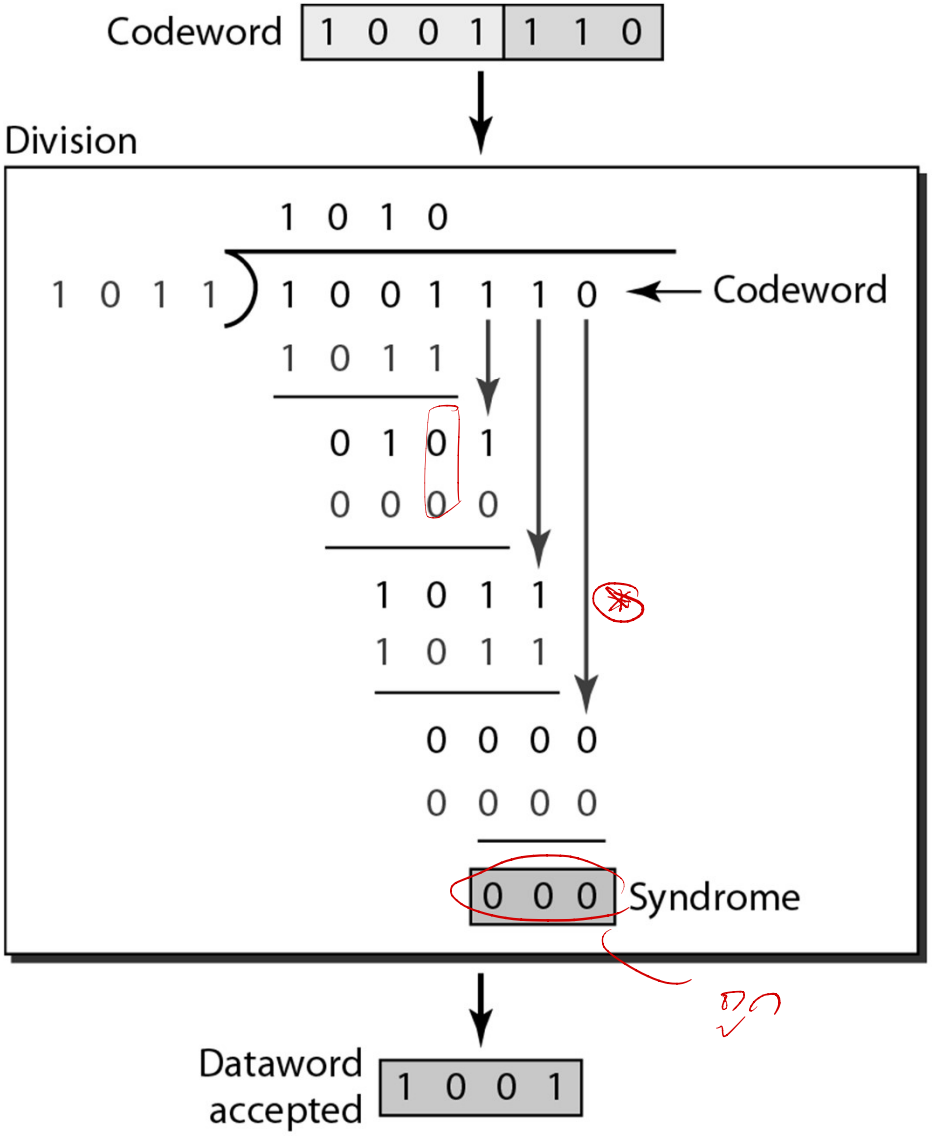
Cyclic Redundancy Check

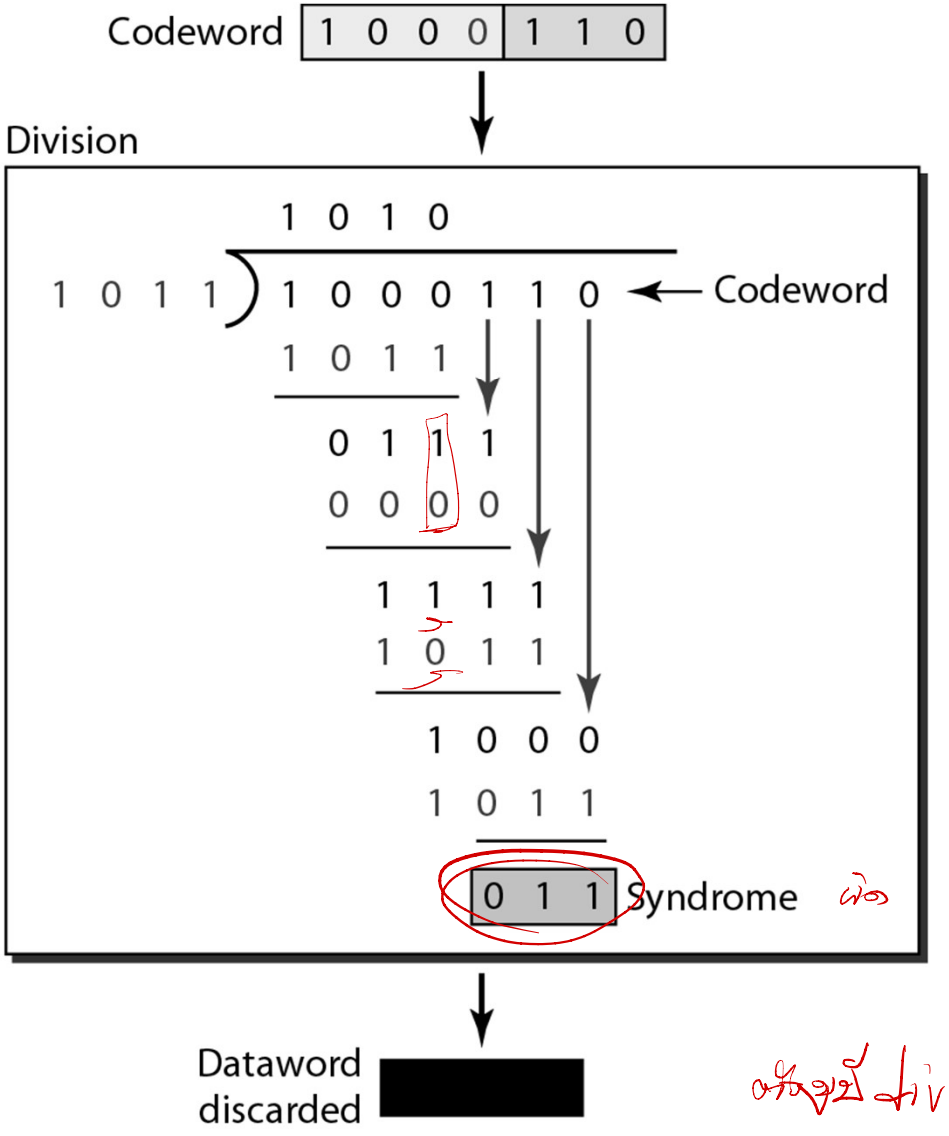


Cyclic Redundancy Check : Encoder

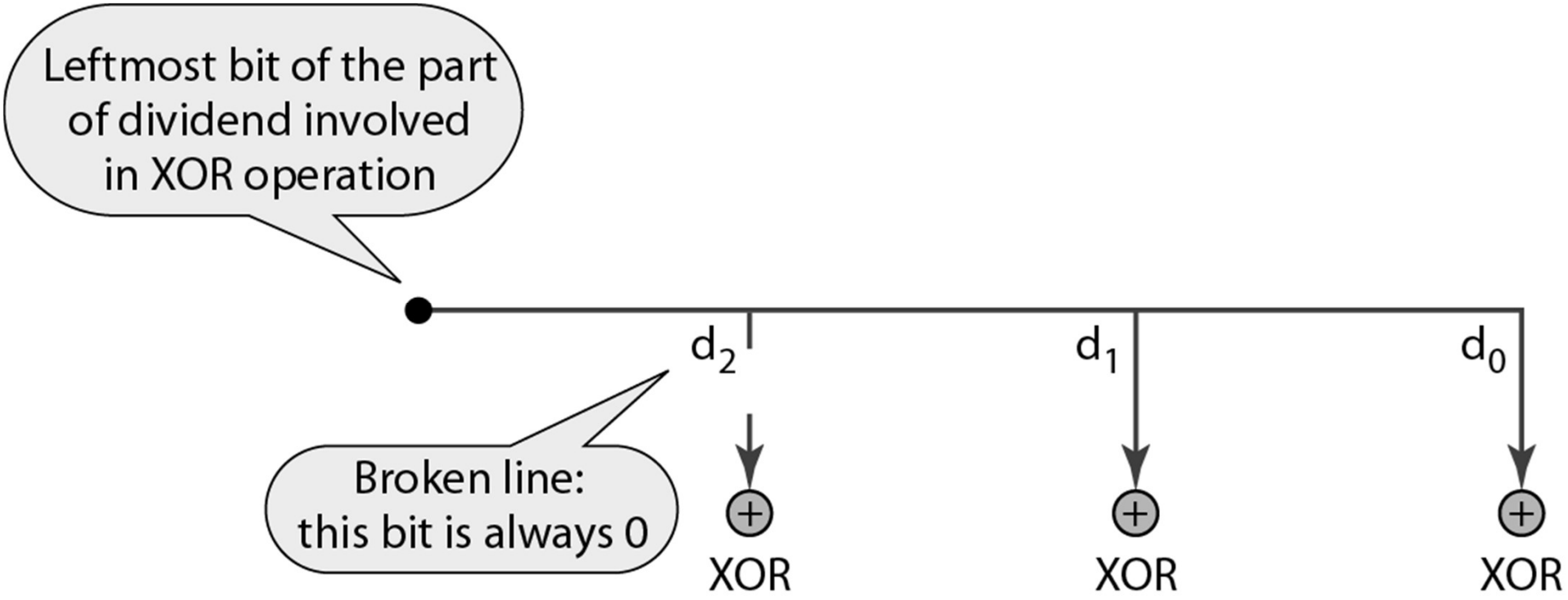


Cyclic Redundancy Check : Decoder

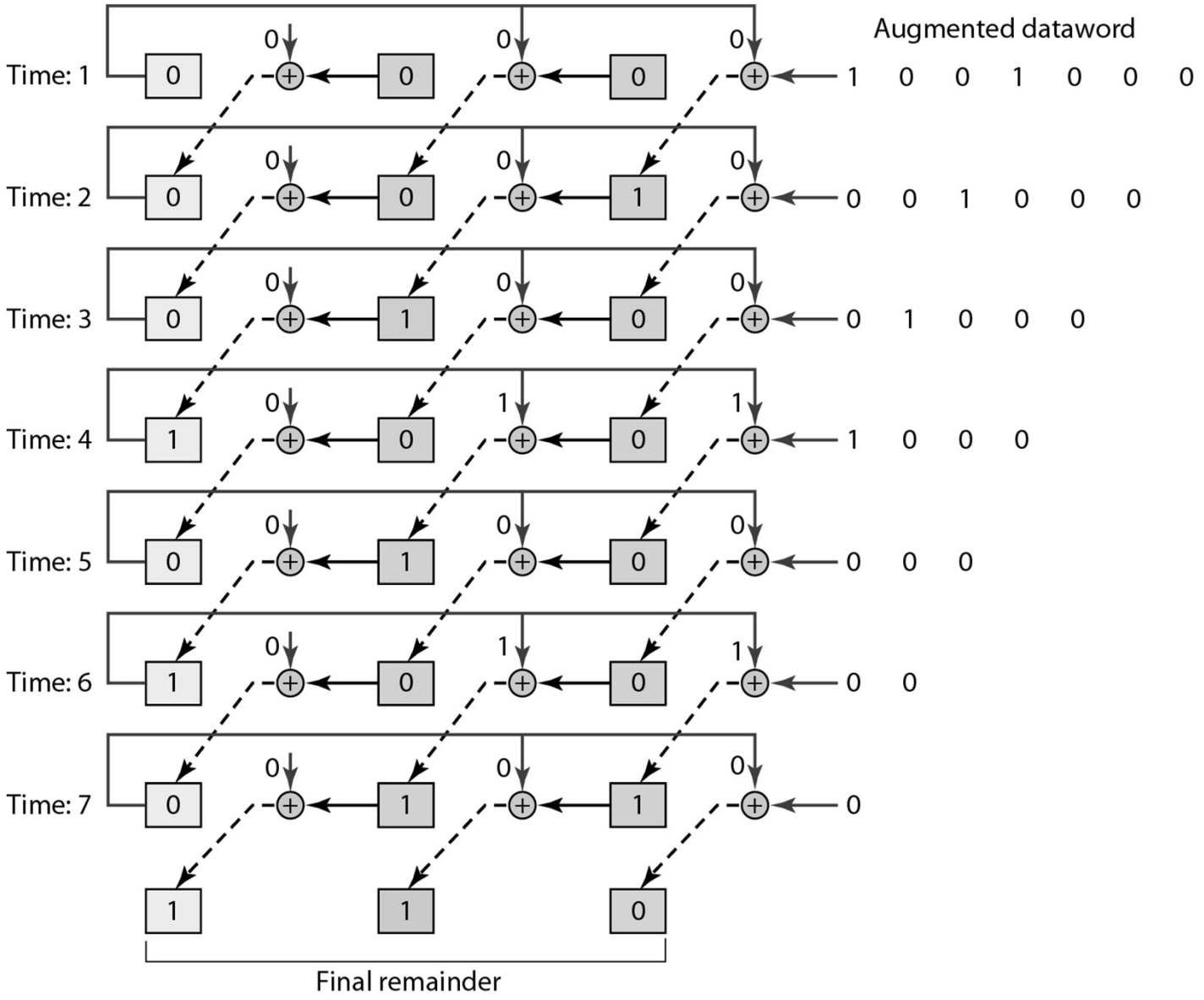




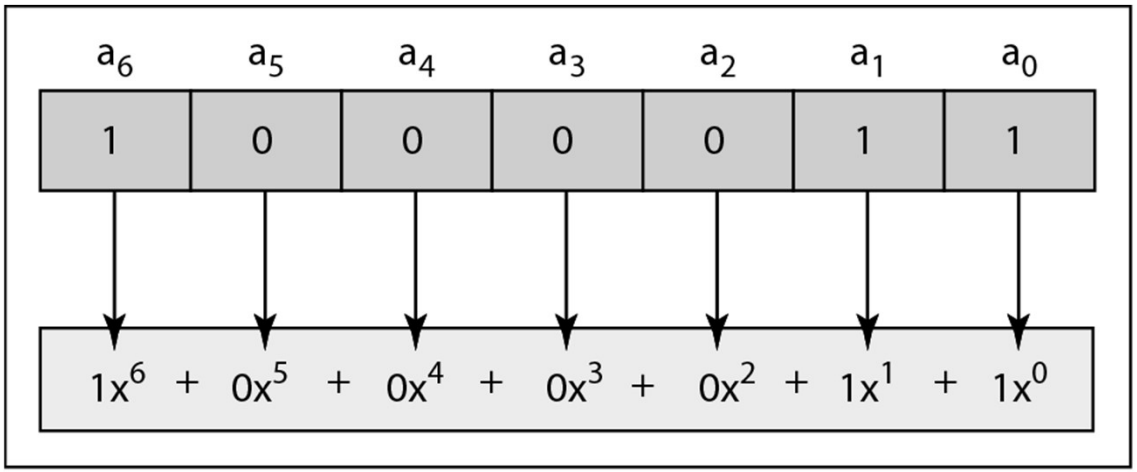
Hardware Implementation



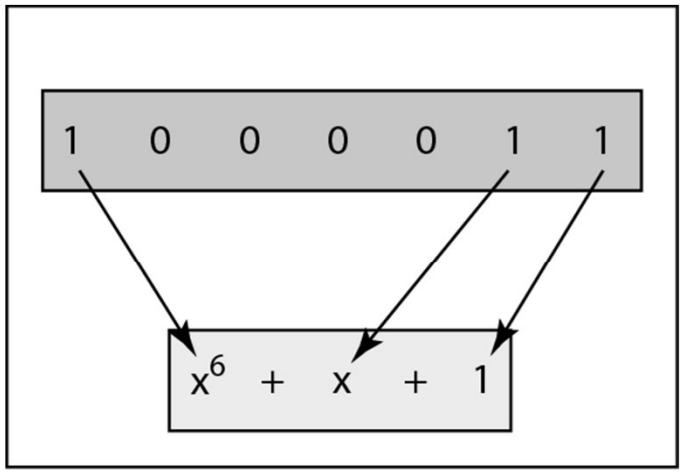
Hardware Implementation



Polynomials



a. Binary pattern and polynomial



b. Short form

Polynomials

- Dataword : $D(x)$
- Generator : $G(x)$
- Codeword : $C(x) = x^g D(x) \oplus R(x)$

$$\frac{x^g D(x)}{G(x)} = Q(x) \oplus \frac{R(x)}{G(x)}$$

$$S(x) = \frac{x^g D(x) \oplus R(x) \oplus E(x)}{G(x)}$$

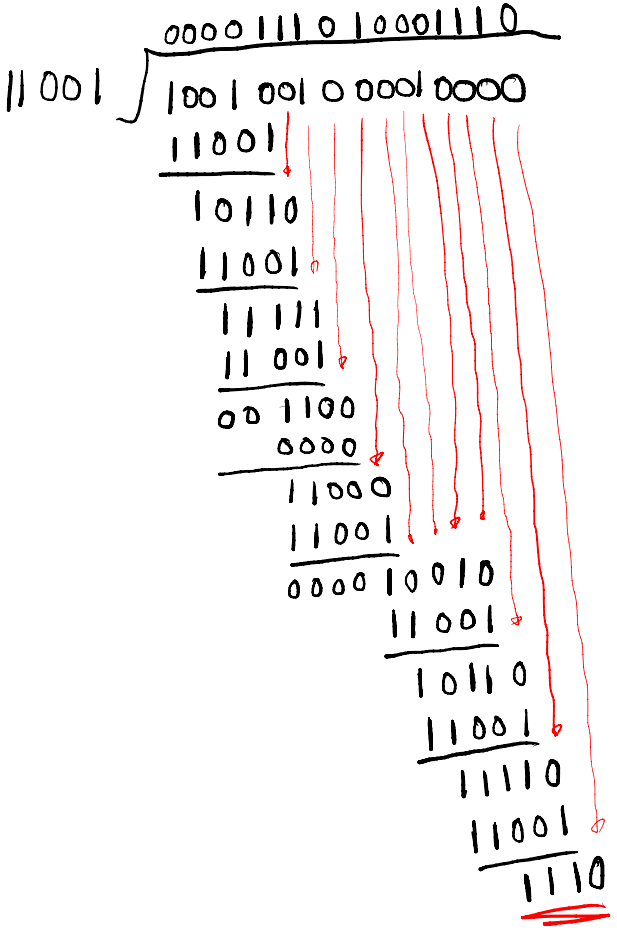
$$= \frac{x^g D(x)}{G(x)} \oplus \frac{R(x)}{G(x)} \oplus \frac{E(x)}{G(x)}$$

$$= Q(x) \oplus \frac{R(x)}{G(x)} \oplus \frac{R(x)}{G(x)} \oplus \frac{E(x)}{G(x)}$$

Data word: $D(x)$ 1001 0010 0001

$G(x) : x^4 + x^3 + 1 \longrightarrow 11001$

$$\frac{x^8 D(x)}{G(x)} = \frac{1001\ 0010\ 0001\ 0000}{11001}$$



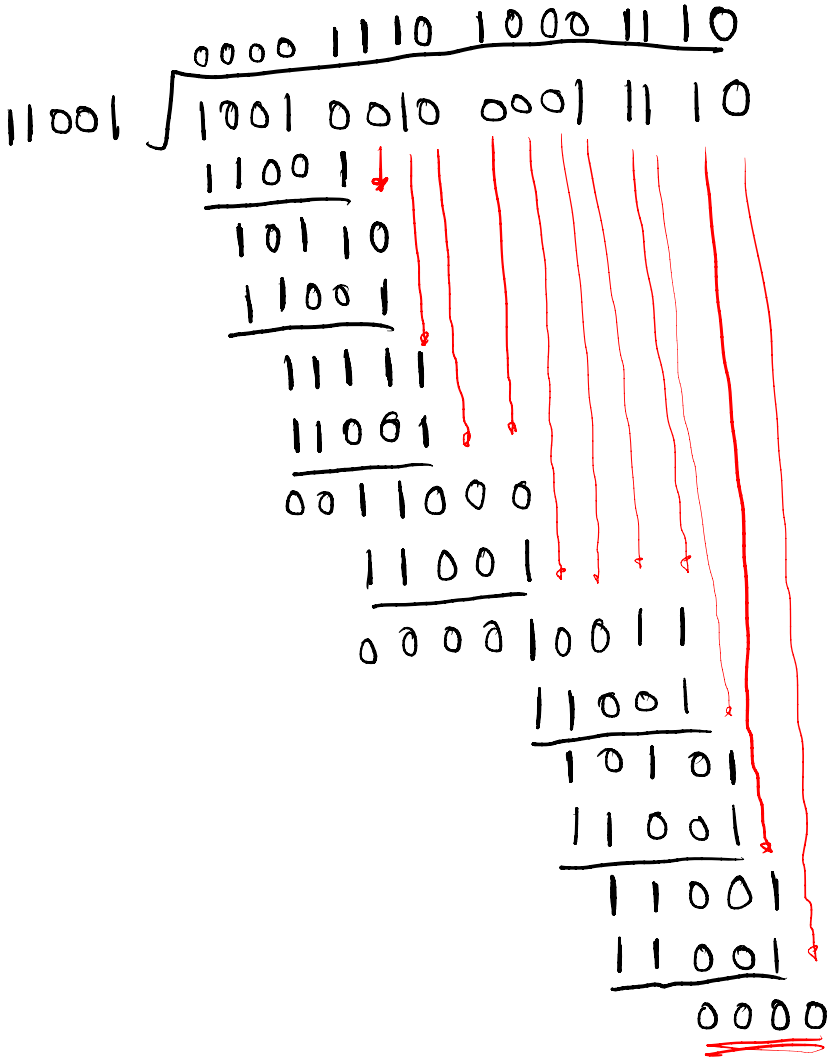
$R(x) = 1110$

$C(x) = 1001\ 0010\ 0001\ 1110$

6301092I ศุภกร ทองย้อ

$C(x) = 1001\,0010\,0001\,1110$

$G(x) : x^4 + x^3 + 1 \longrightarrow 11001$



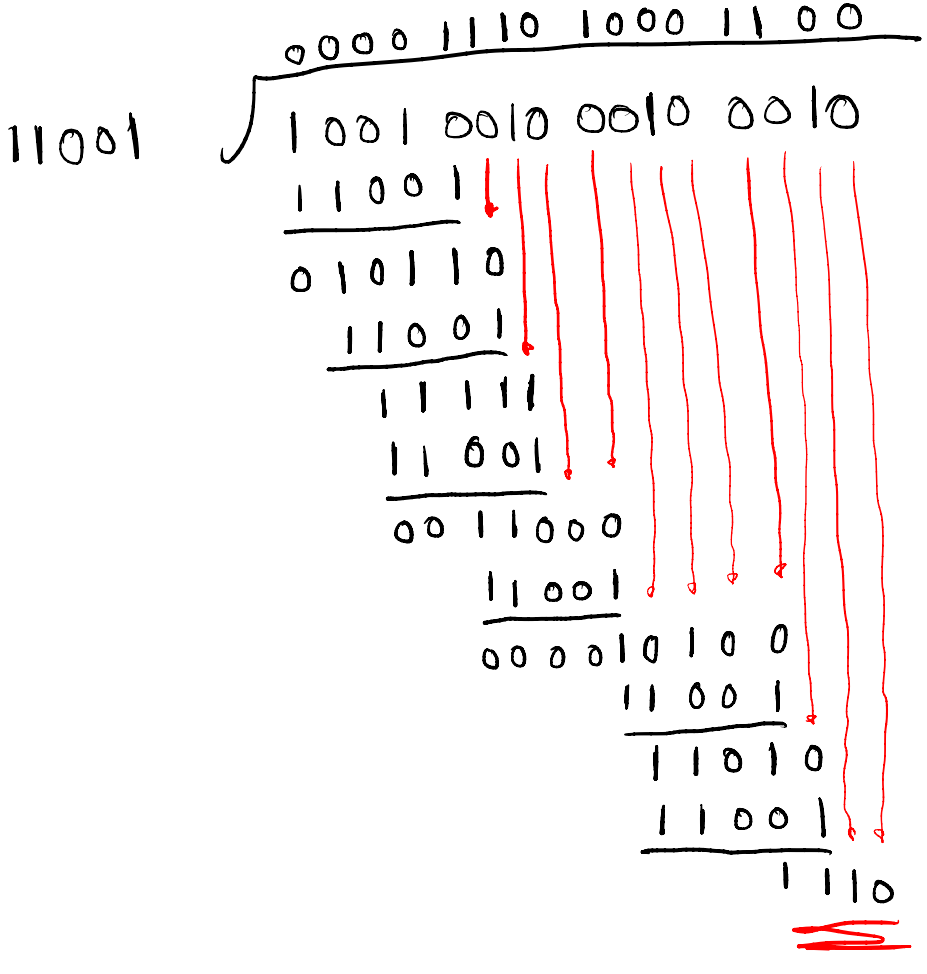
$R(x) = 0000$

6301092I ศุภกร ทองย้อ

$C(x) = 1001\ 0010\ 0001\ 1110$

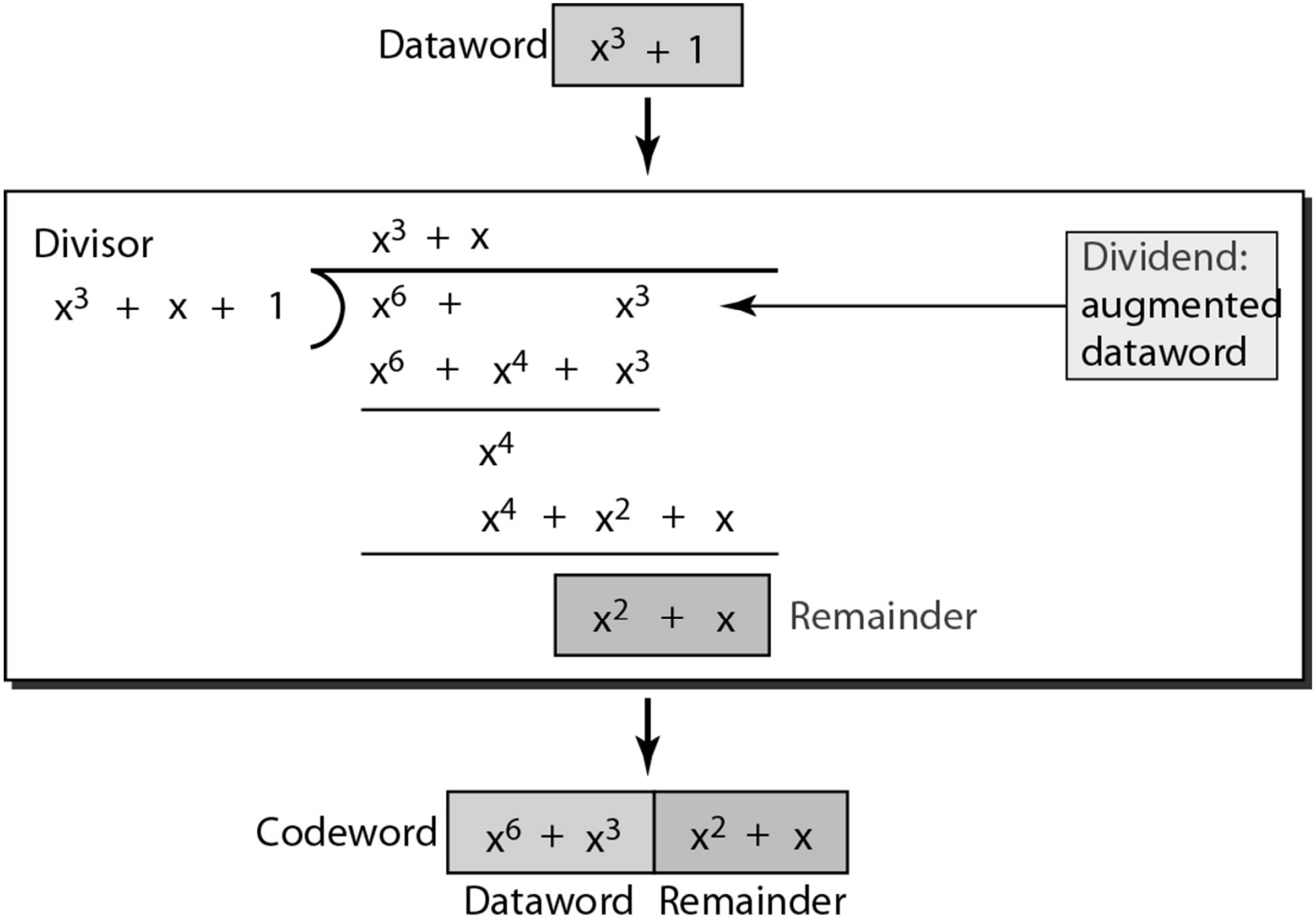
$C(x) + E(x) = 1001\ 0010\ 0010\ 0010$

$G(x) : x^4 + x^3 + 1 \longrightarrow 11001$



$R(x) = 1110$, เกิด Error

Polynomials



Cyclic Code Analysis

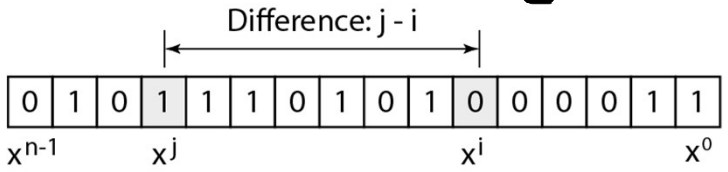
- The divisor in a cyclic code is normally called the generator polynomial or simply the generator.
- In a cyclic code,
  - If $S(x) \neq 0$, one or more bits is corrupted.
  - If $S(x) = 0$, either
    - a. No bit is corrupted. Or
    - b. Some bits are corrupted, but the decoder failed to detect them.

Cyclic Code Analysis

- In a cyclic code, those $E(x)$ errors that are divisible by $G(x)$ are not caught.
- Received codeword $\frac{C(x)+E(x)}{G(x)} = \frac{C(x)}{G(x)} + \frac{E(x)}{G(x)}$
- The first part is by definition divisible the second part will determine the error. If "0" conclusion -> no error occurred.
- Note: that could mean that an error went undetected.

Cyclic Code Analysis

- Single-Bit Error
  - If the generator has more than one term and the coefficient of x^0 is 1, all single errors can be caught.
- Two Isolated Single-Bit Errors



  - If a generator cannot divide $x^t + 1$ (t between 0 and n - 1), then all isolated double errors can be detected.
- Odd Numbers of Errors
  - A generator that contains a factor of $x + 1$ can detect all odd-numbered errors.

Cyclic Code Analysis

- Burst Errors
  - All burst errors with $L \leq r$ will be detected
  - All burst errors with $L = r + 1$ will be detected with probability $1 - \left(\frac{1}{2}\right)^{r-1}$
  - All burst errors with $L > r + 1$ will be detected with probability $1 - \left(\frac{1}{2}\right)^{r}$

Cyclic Code Analysis

- A good polynomial generator needs to have the following characteristics:
  - It should have at least two terms
  - The coefficient of the term x^0 should be 1
  - It should not divide $x^t + 1$, for t between 2 and $n - 1$
  - It should have the factor $x + 1$

Standard polynomials

Name	*Polynomial*	*Application*
CRC-8	$x^8 + x^2 + x + 1$	ATM header
CRC-10	$x^{10} + x^9 + x^5 + x^4 + x^2 + 1$	ATM AAL
CRC-16	$x^{16} + x^{12} + x^5 + 1$	HDLC
CRC-32	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$	LANs

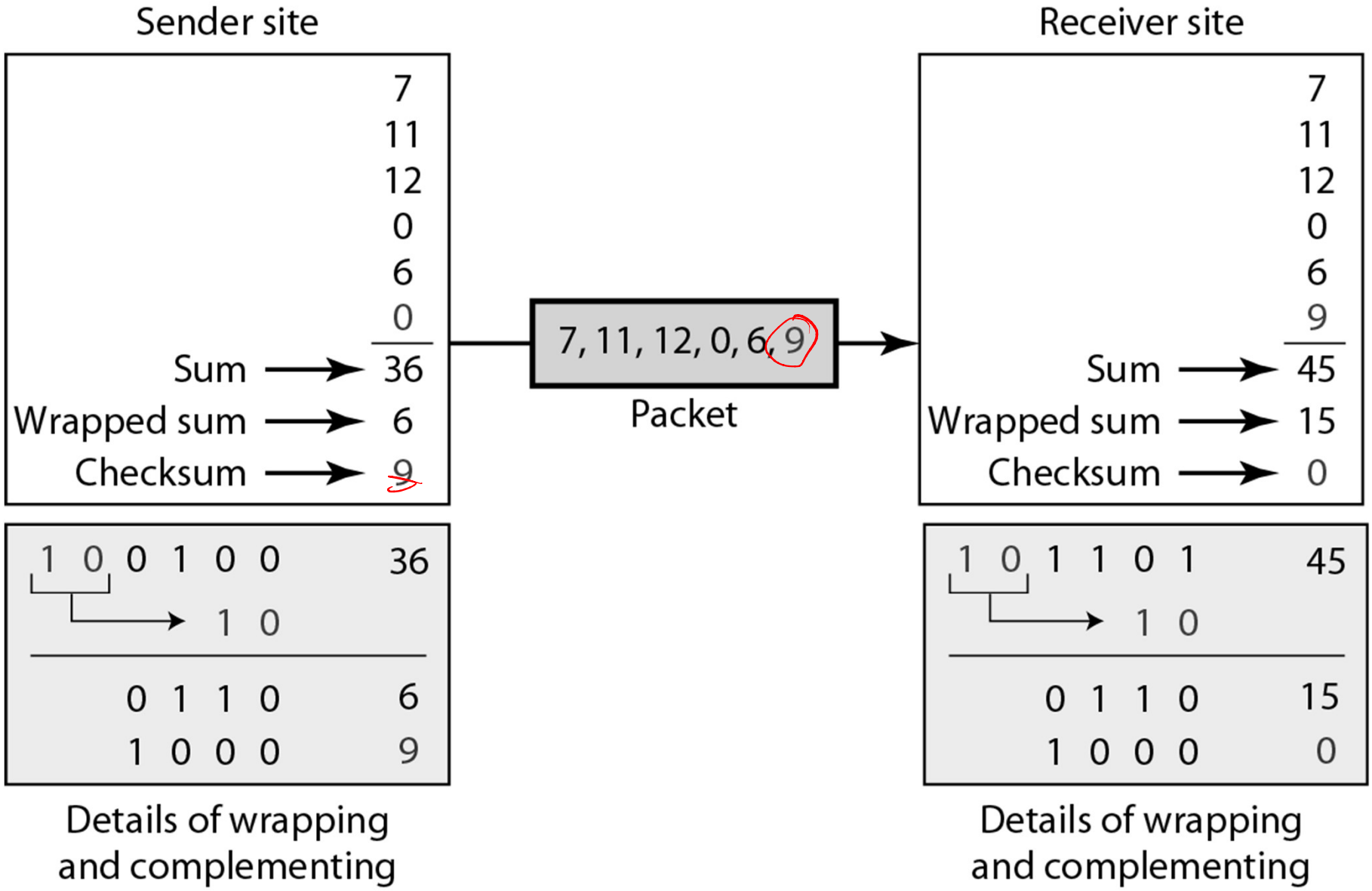
CHECKSUM

- Idea
- One's Complement
- Internet Checksum

Idea (for 4-bit number)

- Data 7, 11, 12, 0
  - Sum = 36
    - (7, 11, 12, 0, 6) => (7, 11, 12, 0, 6, 36)
  - Checksum : negative sum (complement)
    - (7, 11, 12, 0, 6) => (7, 11, 12, 0, 6, -36)

Idea (for 4-bit number)



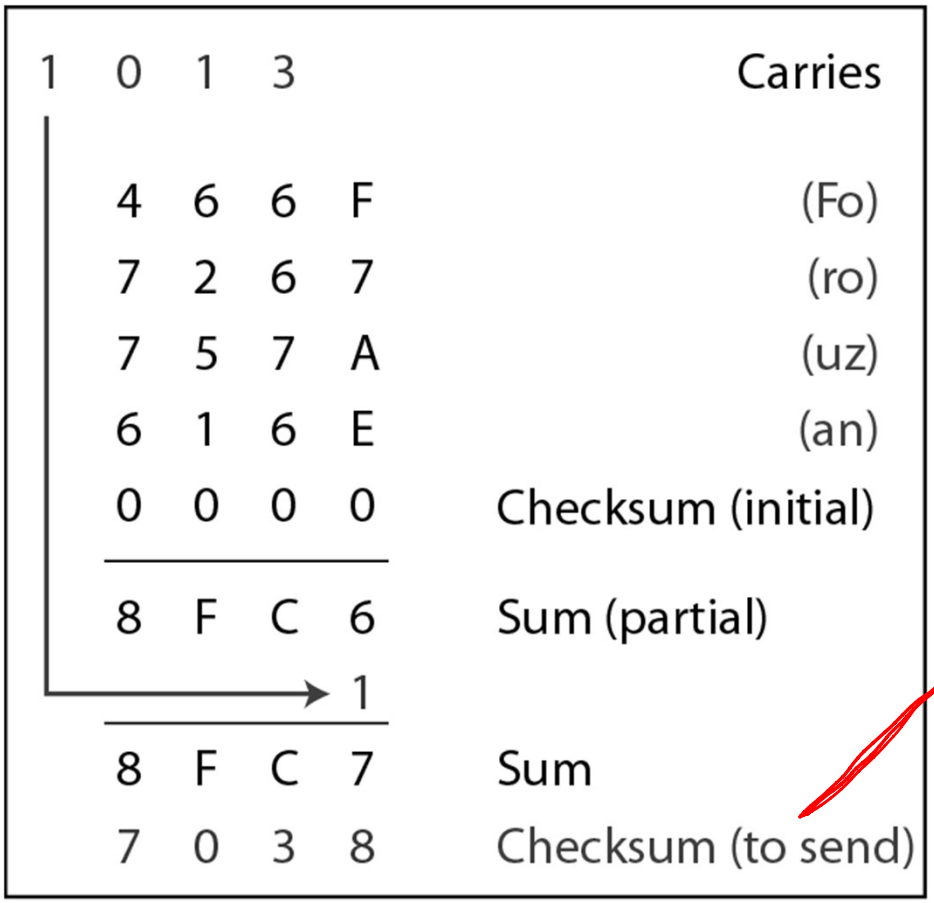
Internet Checksum

- Sender site:
  - The message is divided into 16-bit words.
  - The value of the checksum word is set to 0.
  - All words including the checksum are added using one's complement addition.
  - The sum is complemented and becomes the checksum.
  - The checksum is sent with the data.

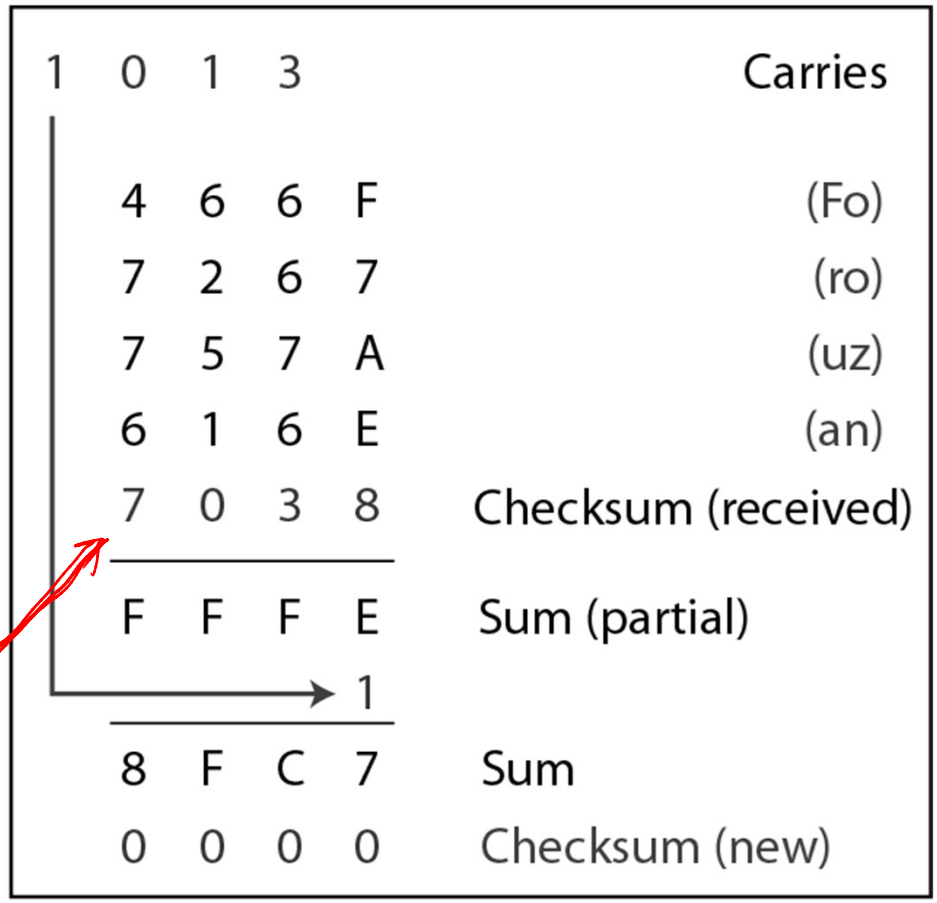
Internet Checksum

- Receiver site:
  - The message (including checksum) is divided into 16-bit words.
  - All words are added using one's complement addition.
  - The sum is complemented and becomes the new checksum.
  - If the value of checksum is 0, the message is accepted; otherwise, it is rejected.

Internet Checksum



a. Checksum at the sender site



a. Checksum at the receiver site