វិទ្យាស្ថានបច្ចេកវិទ្យាកម្ពុជា
INSTITUTE OF TECHNOLOGY OF CAMBODIA

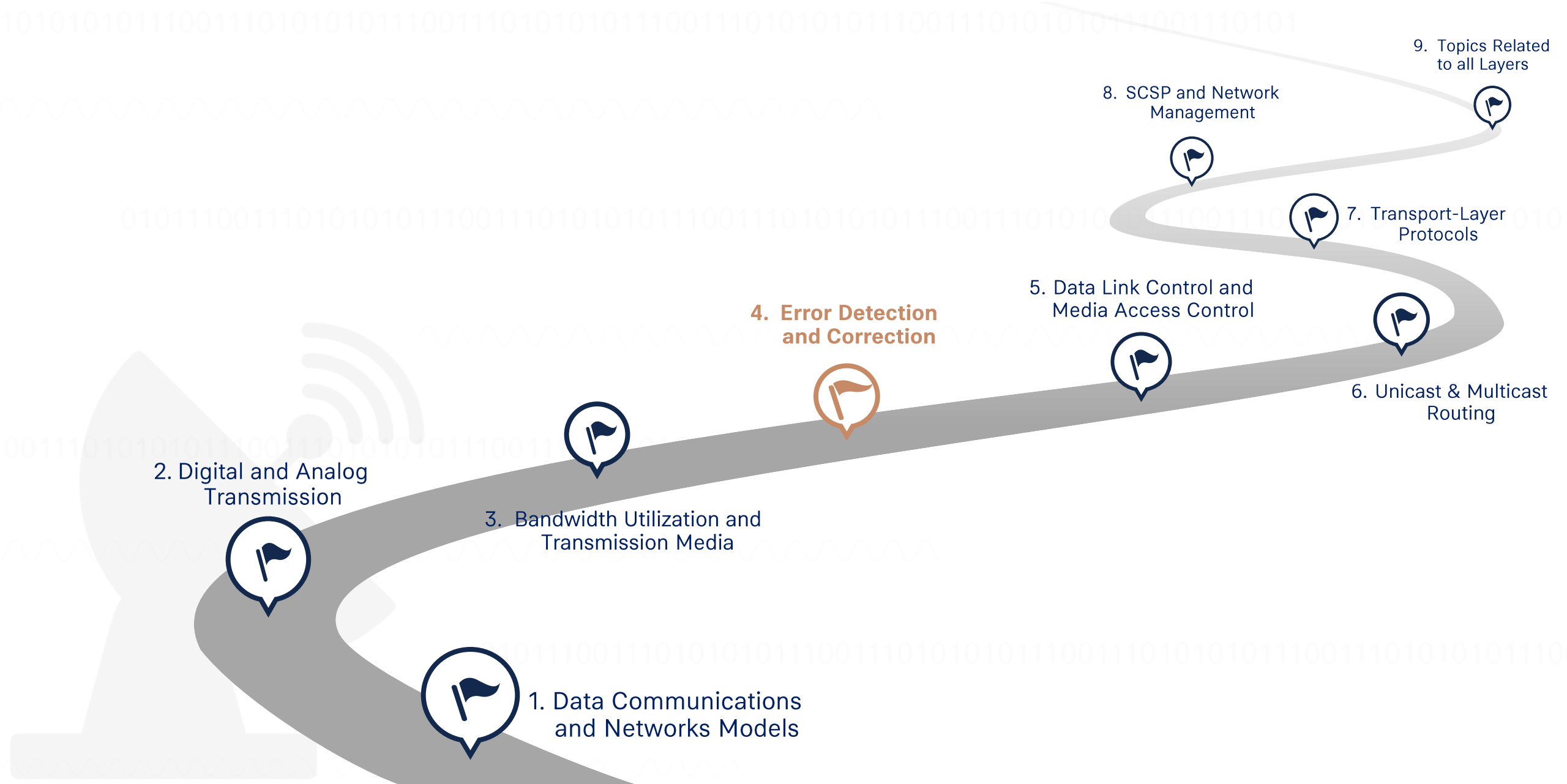
Department of Information and Communication Engineering

ERROR DETECTION AND CORRECTION

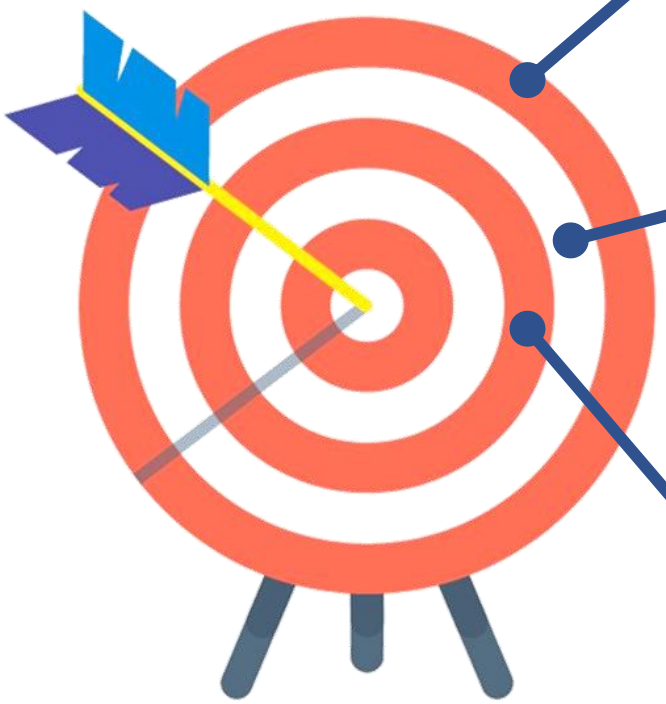
TEL-L04

Subject : Telecommunications
Instructor : Mr. Chea Socheat
E-mail : chea.socheat@itc.edu.kh
Phone : (+855) 96 421 6150

LEARNING MAP



LEARNING OBJECTIVES



Introduces types of errors, concept of redundancy, and error detection and correction: Block Coding, Hamming Distance.

Discusses cyclic codes and subset of cyclic code, CRC, to shows how it can implemented in hardware and represented by polynomials

Discusses checksums and forward error correction, it shows how Hamming distance can be used to achieved the same goal: XOR, interleaving chunk, etc.

LECTURE CONTENTS

01 Data-Link Layer

02 Link-Layer Addressing

03 Error Detection and Correction

04 Block Coding

05 Cyclic Codes

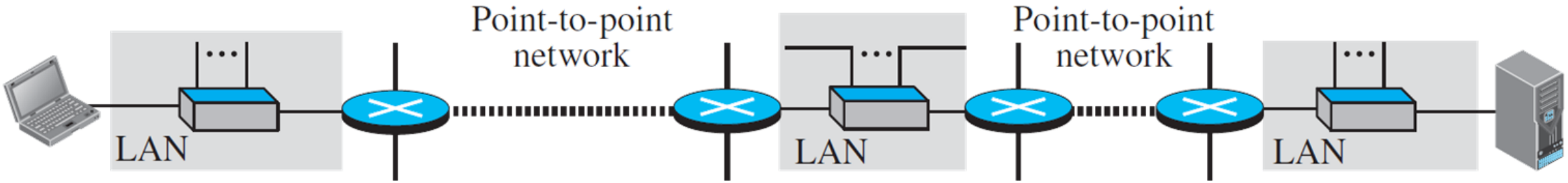
06 Checksum

07 Forward Error Correction

08 Summary & Homework

Nodes and Link

- Communication at the data-link layer is node-to-node; where a data unit from one point in the internet needs to pass through many networks (LANS and WANs) to reach another point (connected by routers).

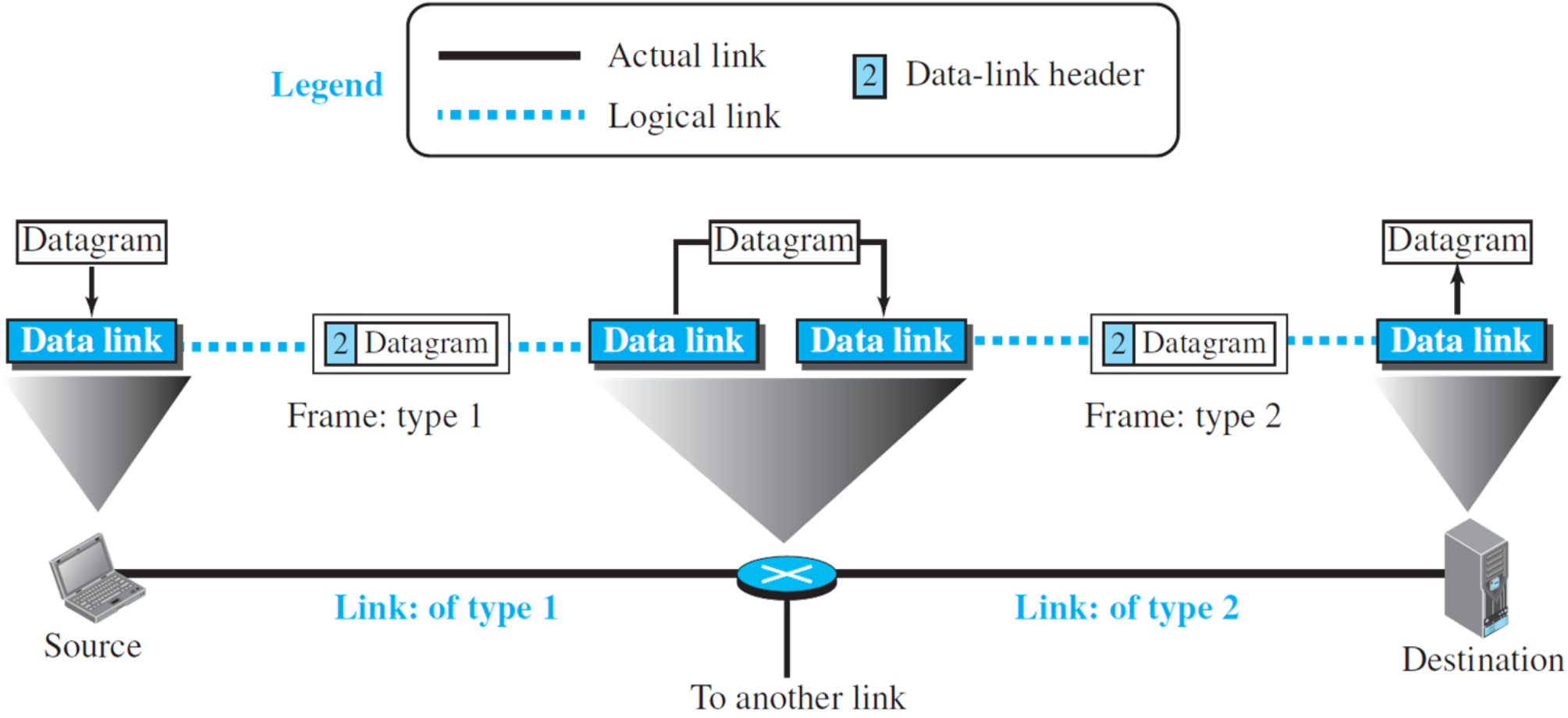


a. A small part of the Internet

b. Nodes and links

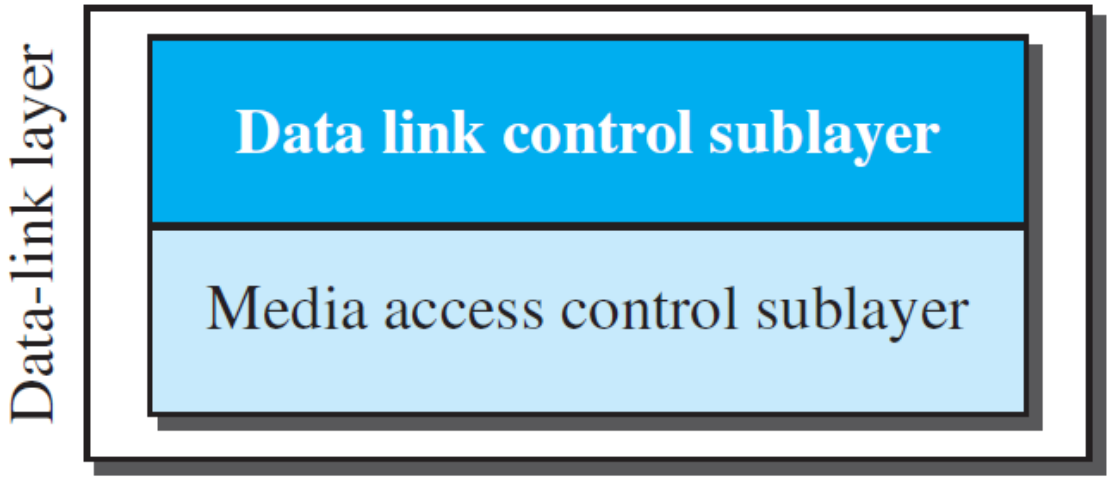
Services

- Data-link layer is located between the physical and network layers. It provides services as **Framing**, **Addressing**, **Flow Control**, **Error Control**, and **Congestion Control**.



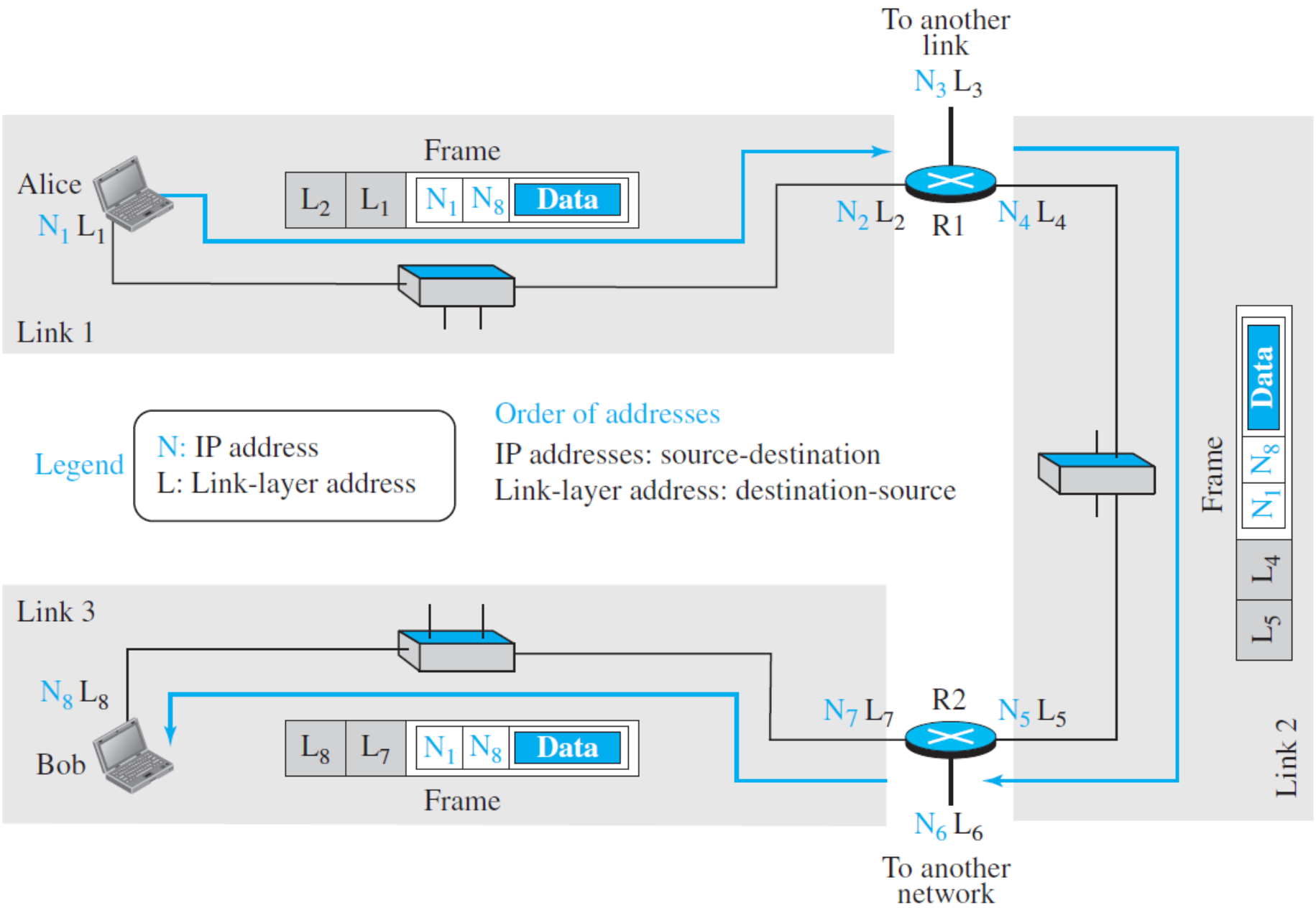
Categories and Sublayers

- Data-link layer controls how the medium is used by use the whole capacity of medium. It has two categories: **point-to-point link** and/or **broadcast link**.
- Functionality and services provided by the data-link layer, can divide into two sublayers: **Data Link Control (DLC)** and **Media Access Control (MAC)**.



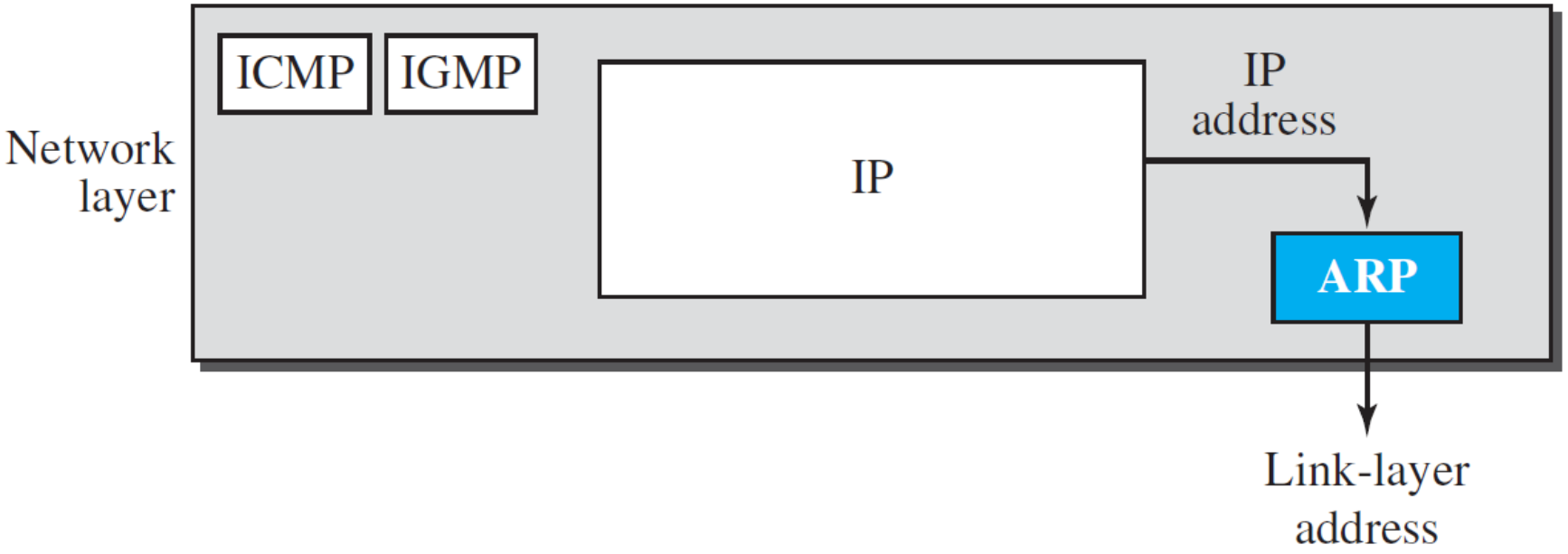
a. Data-link layer of a broadcast link

b. Data-link layer of a point-to-point link


To another link
N_3 L_3
Frame
Alice
N_1 L_1
L_2 L_1 N_1 N_8 Data
N_2 L_2
R1
N_4 L_4
Link 1
Legend
N: IP address
L: Link-layer address
Order of addresses
IP addresses: source-destination
Link-layer address: destination-source
Frame
L_5 L_4 N_1 N_8 Data
Link 3
N_8 L_8
Bob
L_8 L_7 N_1 N_8 Data
Frame
N_7 L_7
R2
N_5 L_5
N_6 L_6
To another network
Link 2

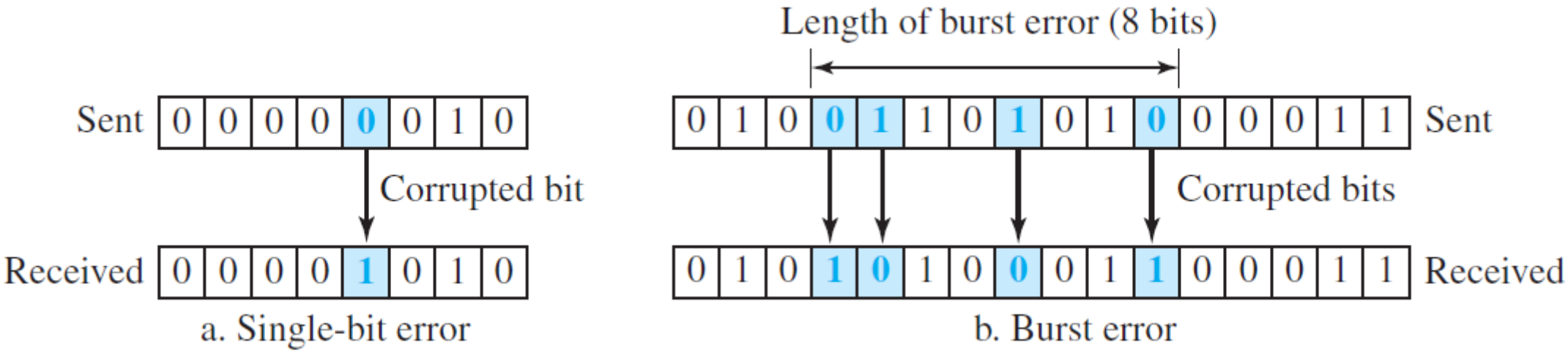
Address Resolution Protocol (ARP)

- Some link-layer protocols define three types of addresses: **Unicast Address**, **Multicast Address**, and **Broadcast Address**.
- ARP becomes helpful when IP address of the next node is not helpful in moving a frame through a link. ARP accepts an IP address from the IP protocol, maps the address to the corresponding link-layer address, and passes it to the data-link layer.



Types of Errors

- Whenever bits flow from one point to another, they are subject to unpredictable changes because of interference that can change the shape of signal.
- There are two types of errors: single-bit error and burst error.



Redundancy

- To be able to detect and/or correct errors, we need to send extra bits with our data, called redundancy. Redundant bits allows the receiver to detect and/or correct corrupted bits.

Detection vs Correction

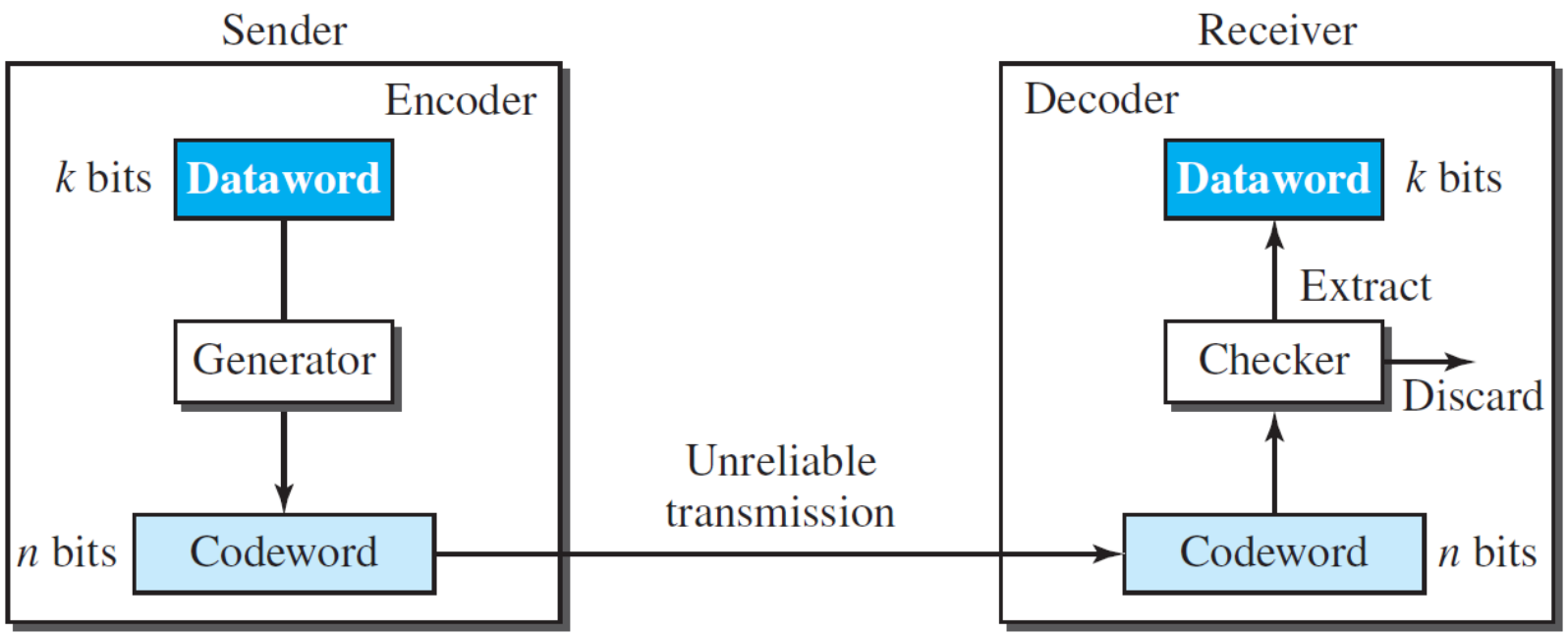
- In **error detection**, we are only looking to see if any error has occurred. The answer is a simple yes or no and not even interested in the number of corrupted bits.
- In **error correction**, we need to know the exact number of bits that are corrupted and their location in the message. The number of errors and the size of the message are important factors.

Coding

- Sender adds redundant bits through a process that creates a relationship between the redundant bits and the actual data bits. Receiver checks the relationships between the two sets of bits to detect errors. In coding schemes divided into two categories: Block Coding and Convolution Coding.

Error Detection

- If the following two conditions are met, the receiver can detect a change in the original codeword: 1) receiver has a list of valid codeword; 2) original codeword has changed to an invalid one.



Example

Let us assume that $k = 2$ and $n = 3$. Table below shows the list of datawords and codewords. We will see how to derive a codewords from a dataword.

Dataword	*Codeword*	*Dataword*	*Codeword*
00	000	10	101
01	011	11	110

Assume the sender encodes the datawords 01 as 011 and sends it to the receiver. Consider the following cases:

1. Receiver receives 011. It's a valid codeword. Receiver extracts the dataword 01 from it.
2. Corrupted, 111 is received. This is not a valid codeword and is discarded.
3. Corrupted, 000 is received and valid codeword, but not correct.

Note: An error detecting code can detect only the types of errors for which it is designed; other types of errors may remain undetected.

Parity-Check Code

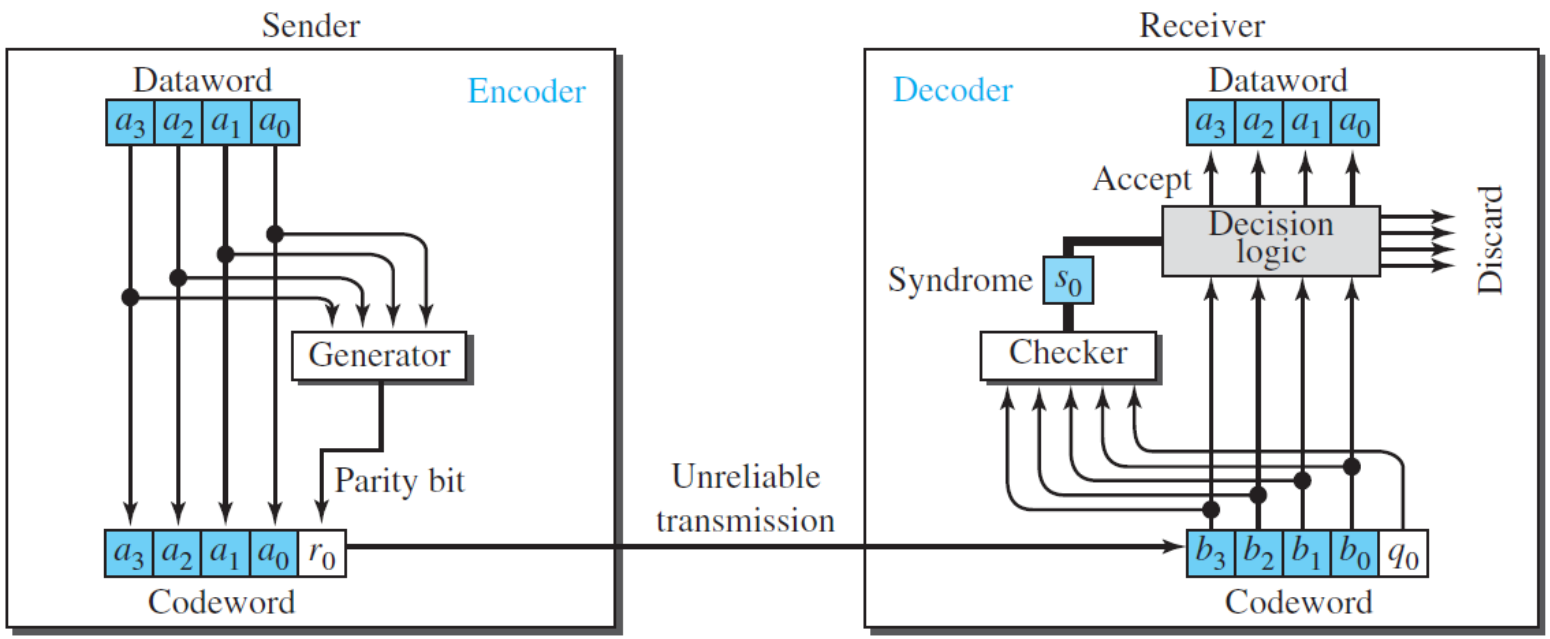
- Generator takes a copy of 4 bits dataword ($\mathbf{a_0}, \mathbf{a_1}, \mathbf{a_2}$, and $\mathbf{a_3}$) and generates a parity bit r_0

$$r_0 = a_3 + a_2 + a_1 + a_0$$

- Syndrome is 0 when the number of 1s in the received codeword is even; otherwise, it is 1.

$$s_0 = b_3 + b_2 + b_1 + q_0$$

- If syndrome is 0, codeword is accepted as the dataword; if syndrome is 1, codeword is discarded.



Encoder and decoder for simple parity-check

Dataword	*Codeword*	*Dataword*	*Codeword*
0000	**00000**	1000	**10001**
0001	**00011**	1001	**10010**
0010	**00101**	1010	**10100**
0011	**00110**	1011	**10111**
0100	**01001**	1100	**11000**
0101	**01010**	1101	**11011**
0110	**01100**	1110	**11101**
0111	**01111**	1111	**11110**

Hamming Distance

- Hamming distance between two words is the number of differences between the corresponding bits; between two words x and y as $d(x, y)$.
- To guarantee the detection of up to s error in all cases, the minimum Hamming distance in a block must be $d_{\min} = s + 1$.

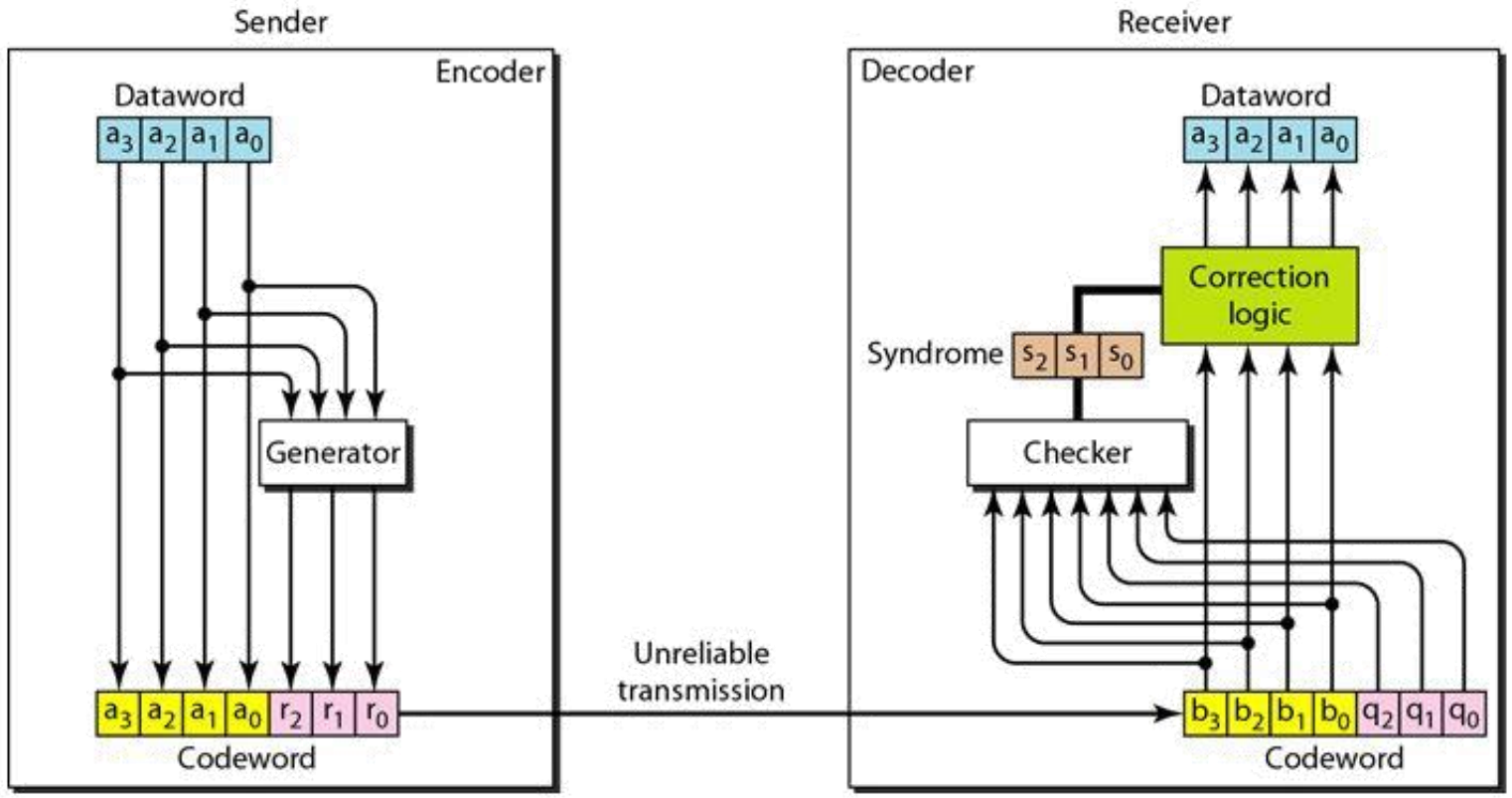
Example

Let find the Hamming distance between two pairs of word

1. The Hamming distance $d(000,001)$
2. The Hamming distance $d(10101,11110)$
3. d(00000,01101) =

Hamming Code

- Hamming code is a set of error correction codes that can be used to detect and correct the errors that can occur when the data moved or stored from the sender to the receiver.



Hamming code encoder and decoder

Hamming Code

- The process of encoding a message by using Hamming code:

1) Calculate the total number of parity bits. Assume that, $\boldsymbol{k}$-bits is number of a message; $\boldsymbol{r}$ is the number of parity bit; then the codeword $\boldsymbol{n} = \boldsymbol{k} + \boldsymbol{r}$. Thus, the number of redundant bits can be calculated using the formula:

$$2^r \geq k + r + 1 \quad \text{where } r = \{1, 2, 3, \dots, r\}$$

2) Determining the position of parity bits; these parity bits are places at the position which correspond to the power of 2.
3) Calculating the values of parity bits; to ensure that the total number of 1's in data is even or odd which used for error detection. There are two types of parity bits:

   a) **Even parity bit**: In the case of even parity, for a given set of bits, the number of 1's are counted. If that count is odd, the parity bit value is set to 0.

   b) **Odd Parity bit**: In the case of odd parity, for a given set of bits, the number of 1's are counted. If that count is even, the parity bit value is set to 1.

Example

Assume the sender sends the dataword 1011. The number of bits in these dataword $k = 4$ bits $(a_3 a_2 a_1 a_0)$.

1) Calculate the number of parity bits r:

$2^r \geq k + r + 1$ with $r = 3;\ k = 4$; then

$2^3 \geq 4 + 3 + 1$ or $8 \geq 8$ (True)

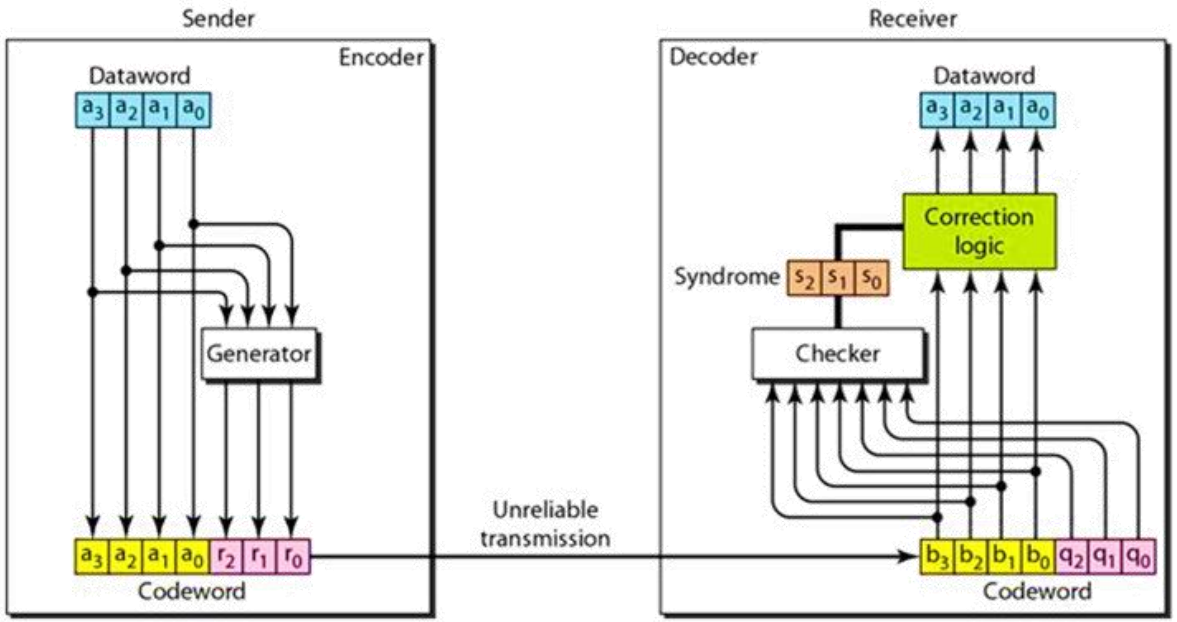
2) Position of parity bits with $r = \{0, 1, 2\}$

			2^2		2^1	2^0
7	**6**	**5**	**4**	**3**	**2**	**1**
a_3	a_2	a_1	r_2	a_0	r_1	r_0

3) Calculate the parity bits: $r_0 = a_0, a_1, a_3 = 111 = 1$ (odd);

$r_1 = a_0, a_2, a_3 = 101 = 0$ (even); $r_2 = a_1, a_2, a_3 = 101 = 0$ (even)

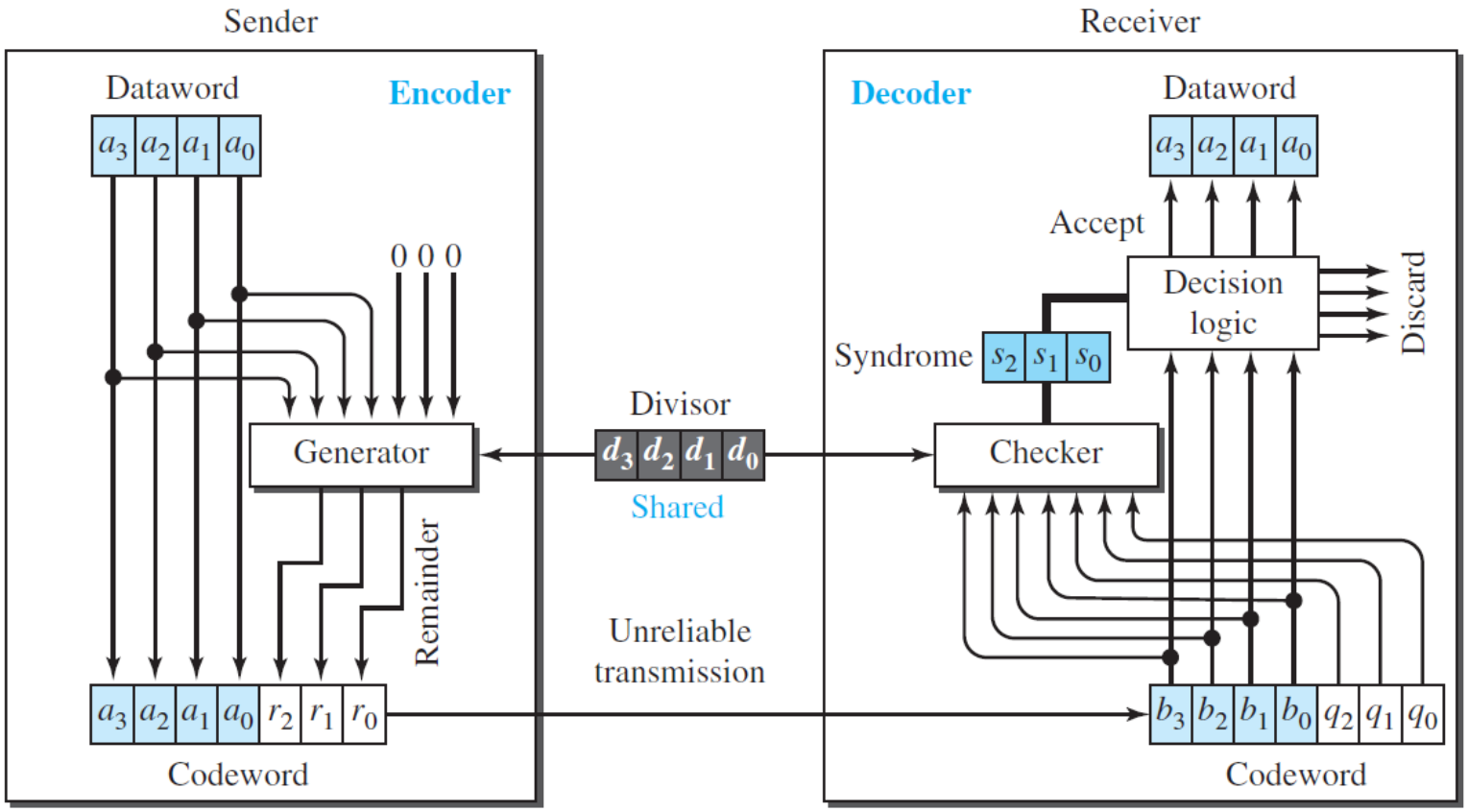
The message generated $(k + r) = 1011001$



Dec	r_2	r_1	r_0
1 ➔	0	0	1
2 ➔	0	1	0
3 ➔	0	1	1
4 ➔	1	0	0
5 ➔	1	0	1
6 ➔	1	1	0
7 ➔	1	1	1

- Cyclic codes are special linear block codes with one extra property. In a cyclic code, if a codeword is cyclically shifted, the result is another codeword.
- if we call the bits in the first word a_0 to a_6, and the bits in the second word $b_0$to b_6, we can shift the bits by using the following:

$$b_1 = a_0 \; ; \; b_2 = a_1 \; ; \; b_3 = a_2 \; ; \; b_4 = a_3; \; b_5 = a_4; \; b_6 = a_5; \; b_0 = a_6$$

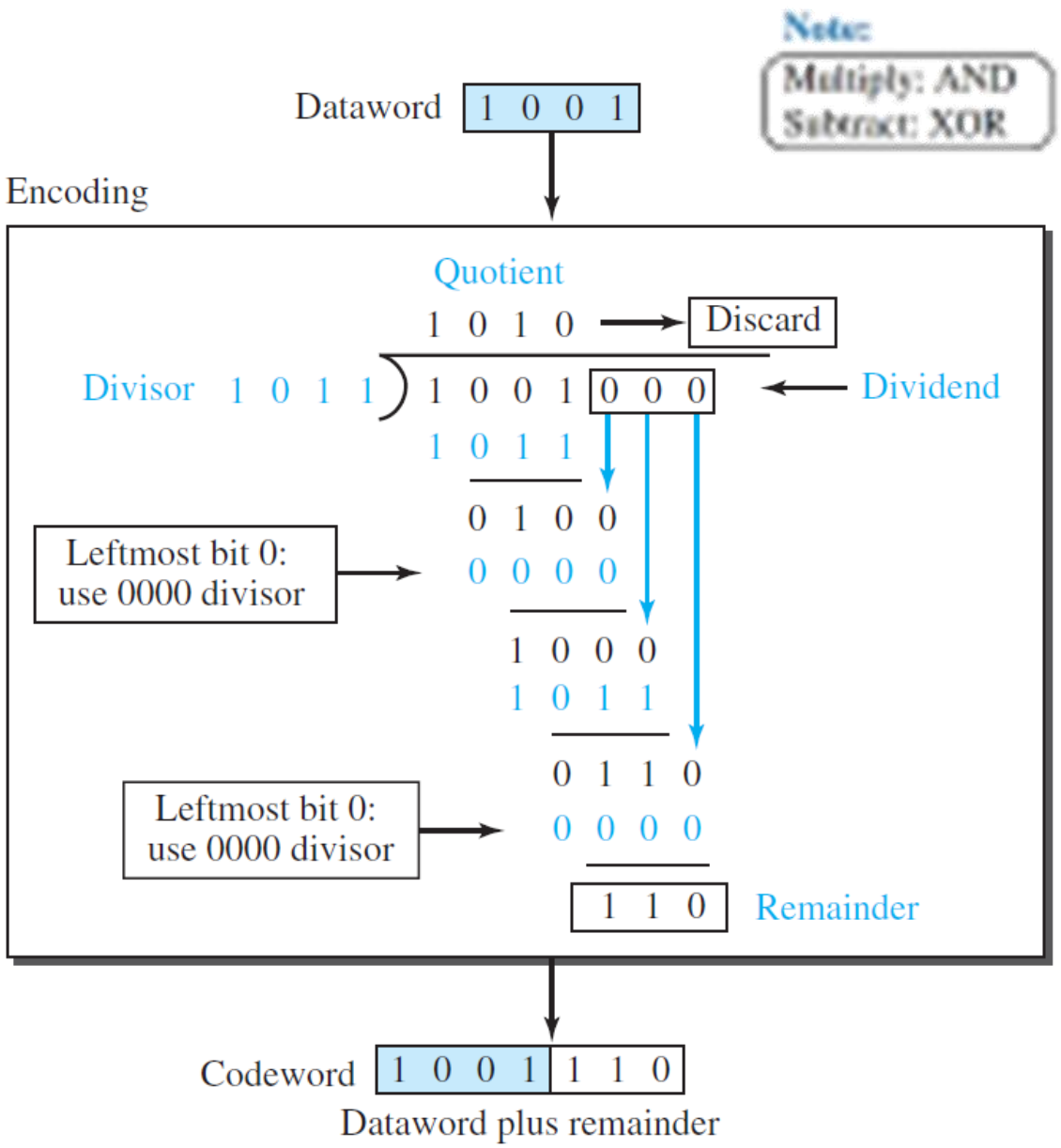


Cyclic Redundancy Check (CRC)

- In the encoder, dataword has k bits (4 here); codeword has n bits (7 here). The size of the dataword is augmented by adding $n-k$ (3 here) 0s to the right-hand side of the word.
- Generator uses a divisor of size $n-k+1$ (4 here). Generator divides the augmented dataword by the divisor. The quotient of the division is discarded; the remainder ($r_2, r_1,$ and r_0) is appended to the dataword to create the codeword.
- The decoder receives the codeword. A copy of all n bits is fed to the checker, which is a replica of the generator. The remainder produced by the checker is a syndrome of $n-k$ (3 here) bits, which is fed to the decision logic analyzer.
- The analyzer has a simple function. If the syndrome bits are all 0s, the 4-leftmost bits are accepted as the dataword; otherwise, the 4 bits are discarded (error).

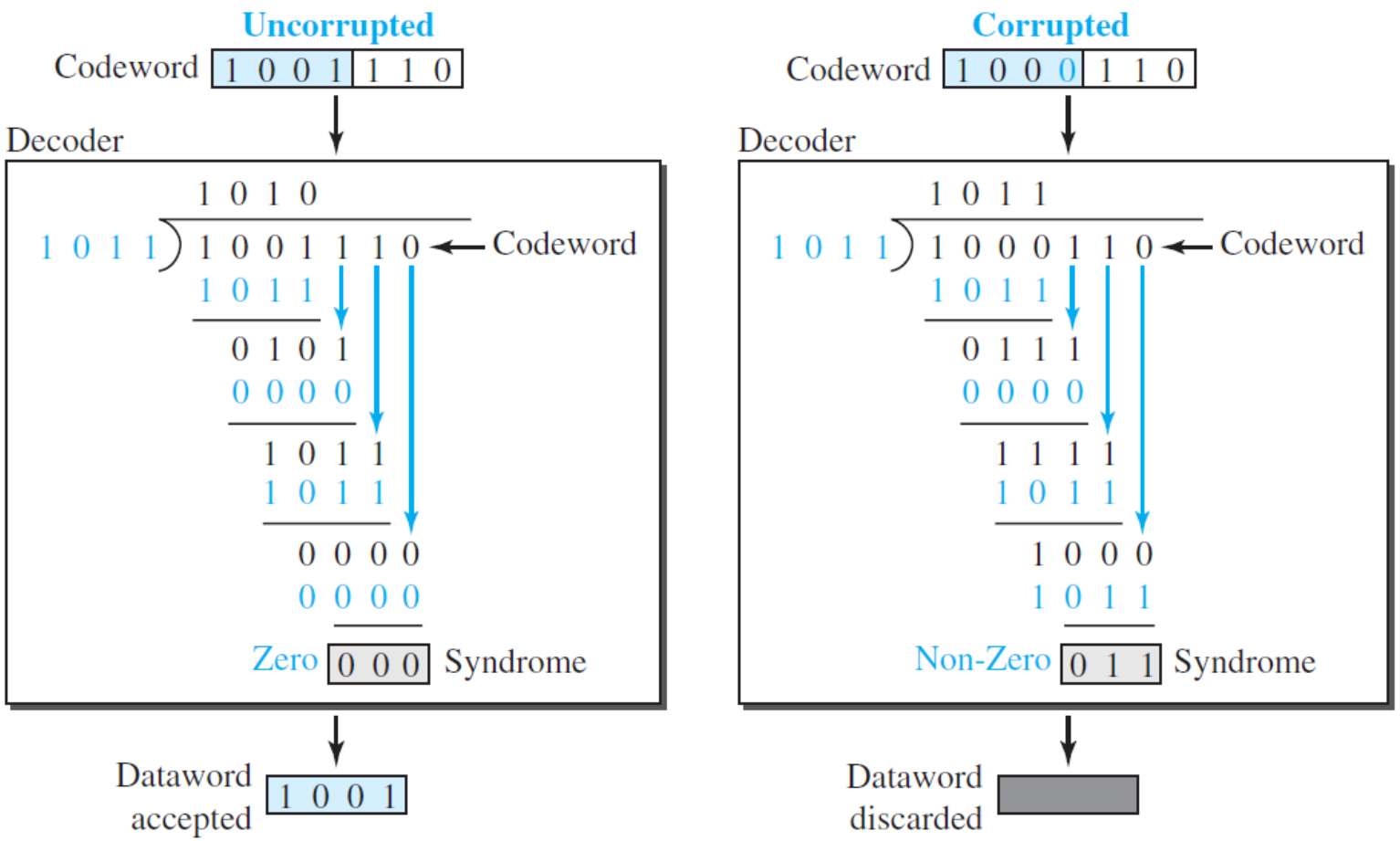
CRC - Encoder

- The encoder takes a dataword and augments it with $n - k$ number of 0s then divides the augmented dataword by the divisor.
- The result of the XOR operation (remainder) is 3 bits (in this case), which is used for the next step after 1 extra bit is pulled down to make it 4 bits long.
- When there are no bits left to pull down, we have a result. The 3-bit remainder forms the check bits $(r_2, r_1, \text{and } r_0)$. They are appended to the dataword to create the codeword.



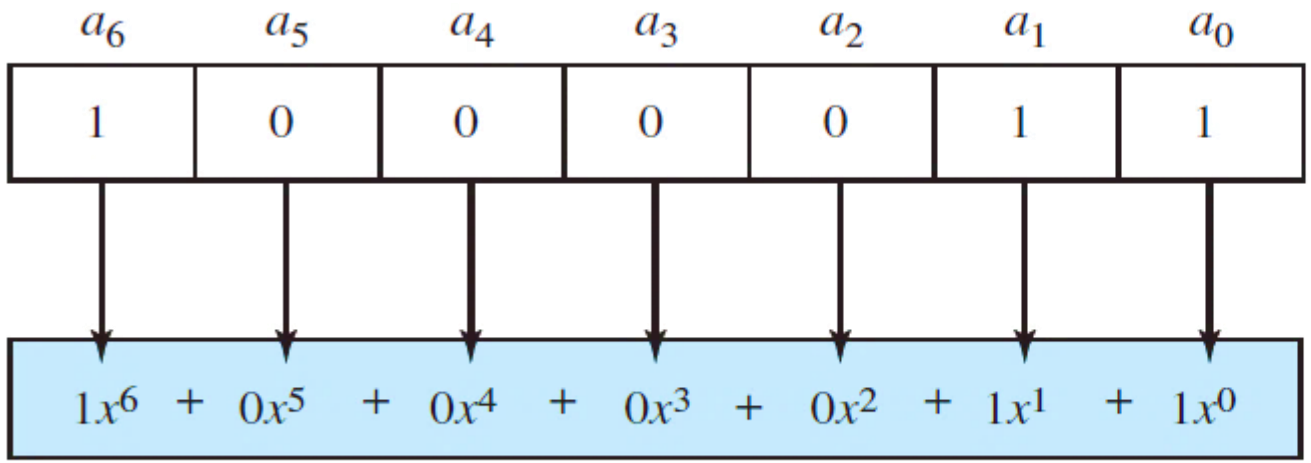
CRC - Decoder

- Decoder does the same division process as the encoder. The remainder of division is the syndrome. If the syndrome is all 0s, there is no error and the dataword is separated from the received codeword and accepted. Otherwise, everything is discarded.

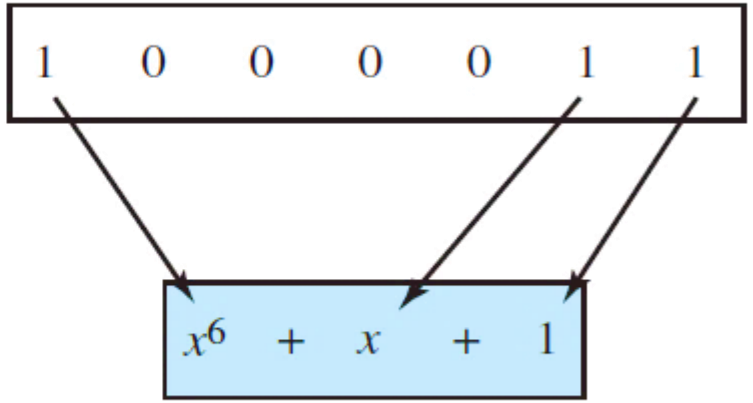


Polynomials

- A better way to understand cyclic codes and how they can be analyzed is to represent them as polynomials. A pattern of 0s and 1s can be represented as a polynomial with coefficients of 0 and 1. The power of each term shows the position of the bit; the coefficient shows the value of the bit.
- The degree of a polynomial is the highest power in the polynomial. For example, the degree of the polynomial $x^6 + x + 1$ is 6.



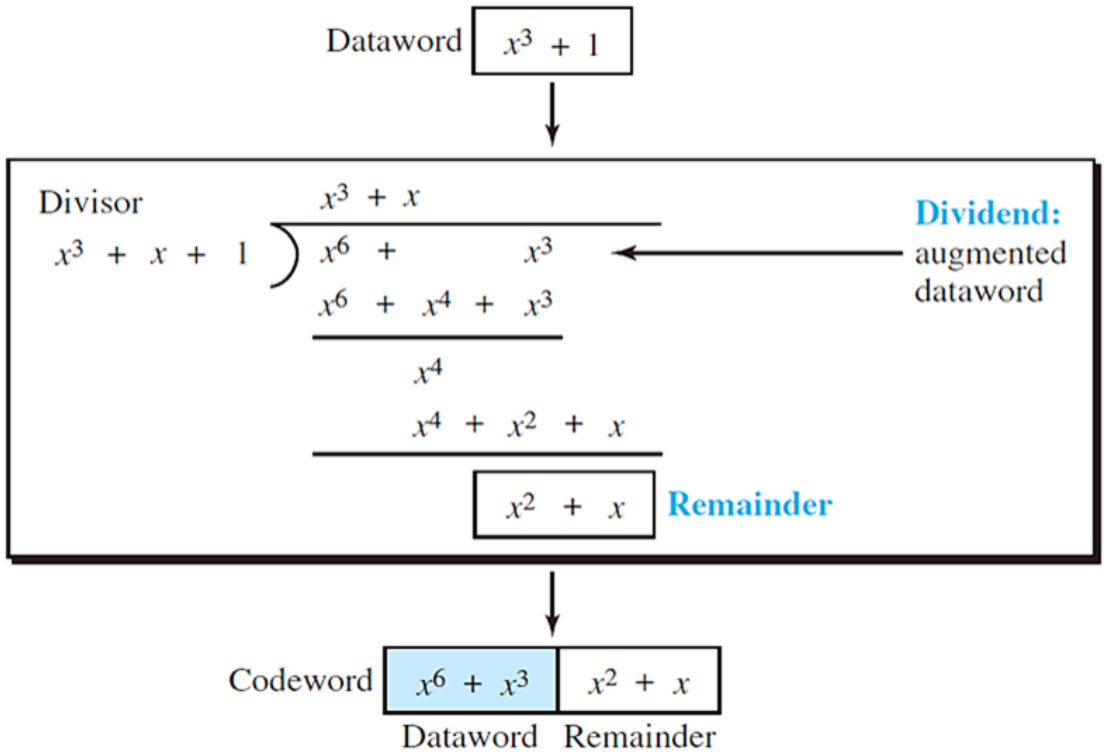
a. Binary pattern and polynomial



b. Short form

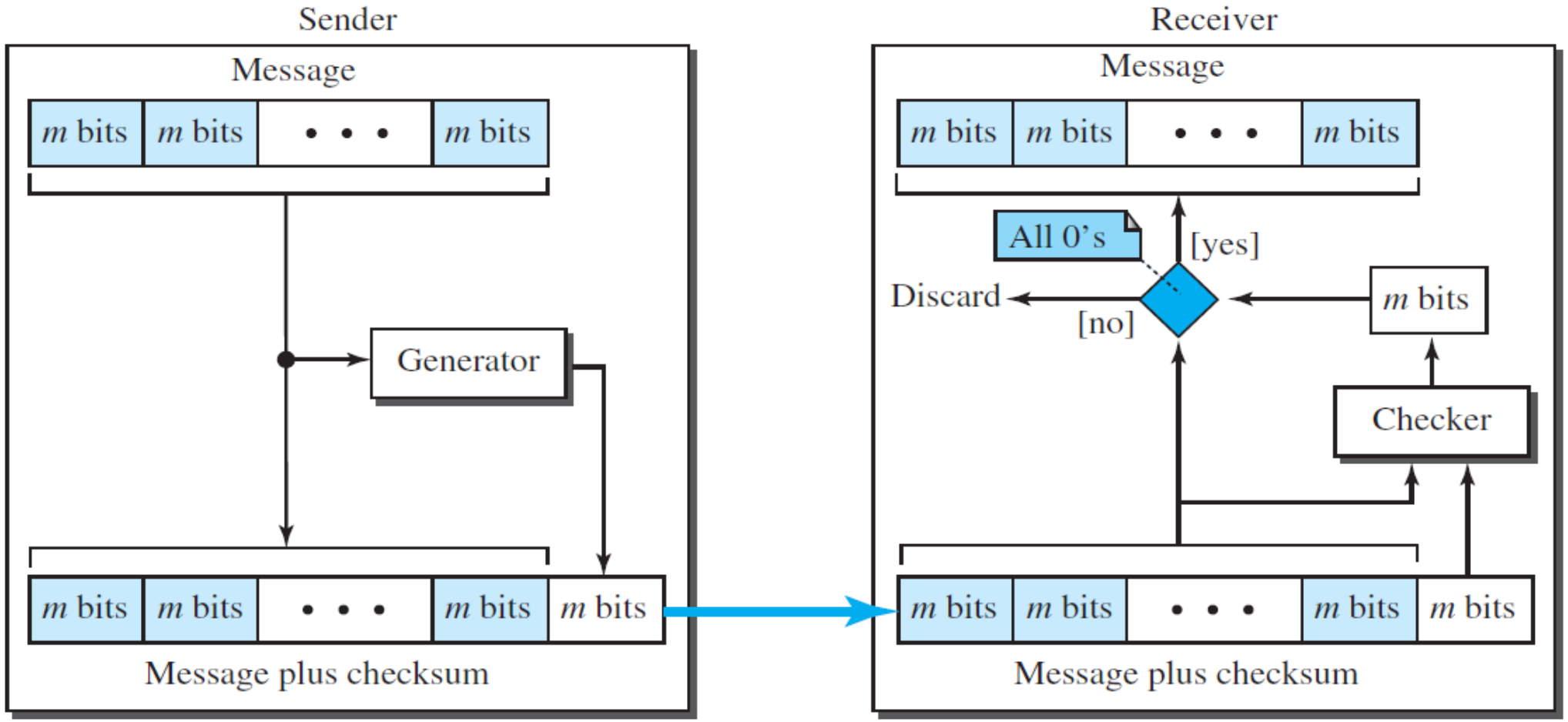
Cyclic Code Encoder Using Polynomials

- Dataword 1001 is represented as $x^3 + 1$; divisor 1011 is represented as $x^3 + x + 1$.
- To find the augmented dataword, we have left-shifted dataword 3 bits; the result is $x^6 + x^3$.
- In polynomial representation, the divisor is normally referred to as the generator polynomial.
- The divisor in a cyclic code is normally called the generator polynomial or simply the generator.



Name	*Polynomial*	*Used in*
CRC-8	$x^8 + x^2 + x + 1$ **100000111**	ATM header
CRC-10	$x^{10} + x^9 + x^5 + x^4 + x^2 + 1$ **11000110101**	ATM AAL
CRC-16	$x^{16} + x^{12} + x^5 + 1$ **10001000000100001**	HDLC
CRC-32	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ **100000100110000010001110110110111**	LANs

- Checksum is an error-detecting technique that can be applied to a message of any length. In the Internet, the checksum technique is mostly used at the network and transport layer rather than the data-link layer.

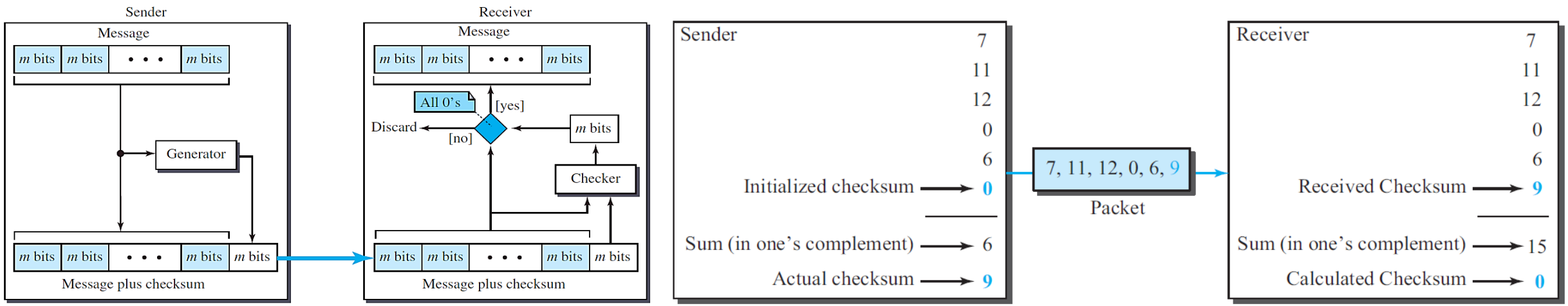


Concept

- At the source, the message is first divided into m-bit units. The generator then creates an extra m-bit unit called checksum. At destination, checker creates a new checksum from the combination of message and sent checksum. If new checksum is all 0s, message is accepted; otherwise, message is discarded.
- In one's complement arithmetic, complement of a number is found by completing all bits (changing all 1s to 0s and all 0s to 1s), the same as subtracting the number from $2^m - 1$.

Sender	*Receiver*
1. The message is divided into 16-bit words.	1. The message and the checksum are received.
2. The value of the checksum word is initially set to zero.	2. The message is divided into 16-bit words.
3. All words including the checksum are added using one's complement addition.	3. All words are added using one's complement addition.
4. The sum is complemented and becomes the checksum.	4. The sum is complemented and becomes the new checksum.
5. The checksum is sent with the data.	5. If the value of the checksum is 0, the message is accepted; otherwise, it is rejected.

Example: Suppose the message is a list of five 4-bit numbers that we want to send to a destination. The set of numbers is (7, 11, 12, 0, 6).



- Sum of original number is 36 or (7, 11, 12, 0, 6, 36); in binary is $(100100)_2$. Sender adds all five number in one's complement to get $(10)_2 + (0100)_2 = (0110)_2 = (6)_{10}$; Then, complements the result to get the checksum $(1001)_2 = (9)_{10}$.
- Sender send five data number and checksum (7, 11, 12, 0, 6, 9). If there is no corruption in transmission, receiver receives (7, 11, 12, 0, 6, 9) and adds them in one's complement to get 15. The receiver complements 15 to get 0.

Using Hamming Distance

- To correct the error or reproduce the packet immediately, several schemes have been designed and used in this case that are collectively referred to as **FEC** techniques.
- Hamming distance between the received codeword and the sent codeword is the number of bits that are corrupted during transmission. The Hamming distance between two words is the number of differences between corresponding bits.

Example: Suppose that we want to transmit 2-bits sequence into 5-bits codeword.

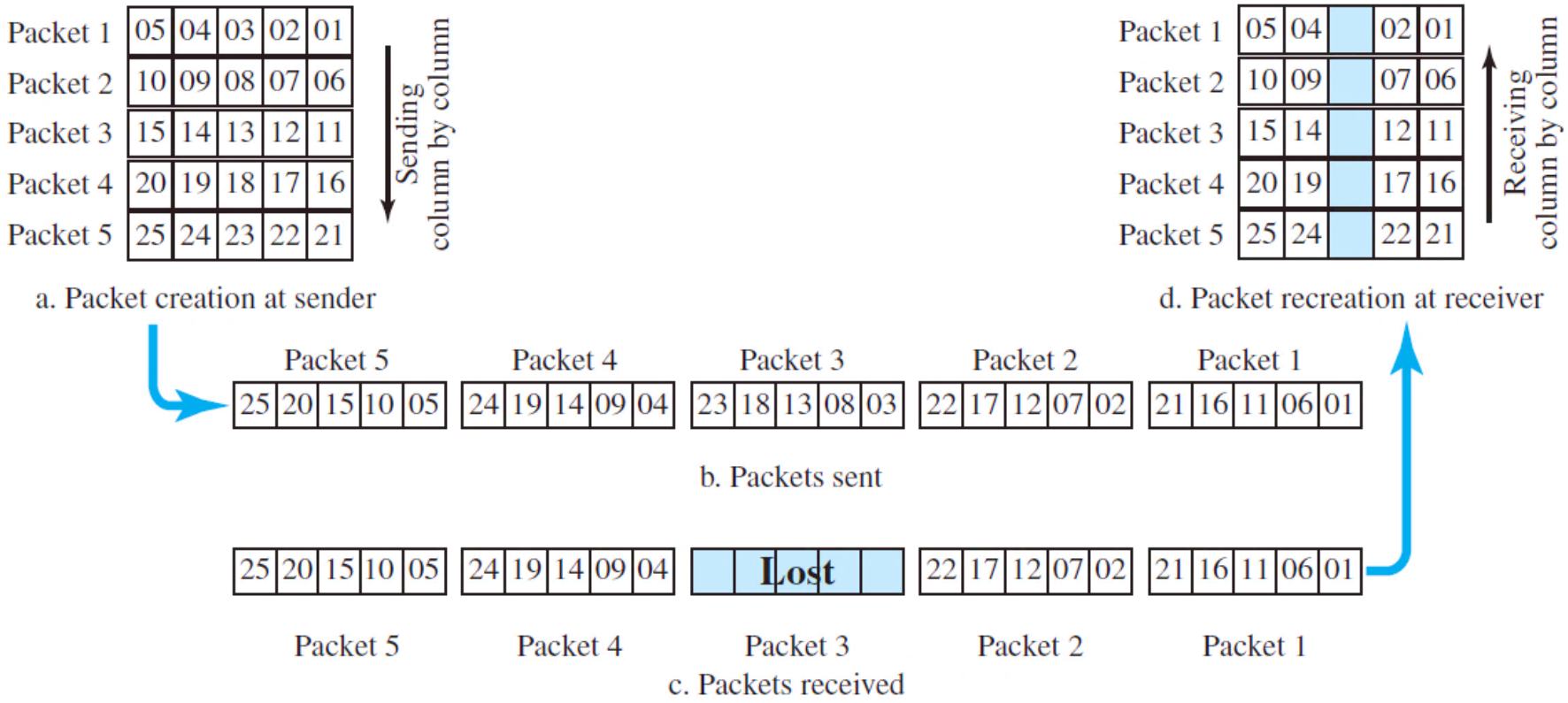
10	01	00	11
10011	11100	00000	01111
10011	10100	00000	01101
	11100		01111
10	01	00	11

Encoded Bit Sequence	Corresponding Codeword
00	00000
01	11100
10	10011
11	01111

Hamming Distance	Hamming Distance
2	3
1	2
3	4
4	1

Chunk Interleaving

- Another way to achieve FEC in multimedia is to allow some small chunks to be missing at the receiver. All chunks are not belonging to the same packet be missing; only, one chunk be missing in each packet. If the packet is lost, we miss only one chunk in each packet, which is normally acceptable in multimedia communication.



6. SUMMARY

Internet is made of many hosts, networks, and connecting devices such as routers. Hosts and connecting devices are referred to as nodes; networks are referred to as links.

Data-link layer is responsible for the creation and delivery of a frame to another node; packetizing (framing), flow control, error control, and congestion control along the link.

Data can be corrupted during transmission and some applications require that errors be detected and corrected: single bit error and burst error.

Hamming distance between two words is the number of differences between corresponding bits; minimum Hamming distance is the smallest Hamming distance between all possible pairs in a set of words.

Cyclic codes are special linear block codes with one extra property. If a codeword is cyclically shifted (rotated), the result is another codeword.

7. HOMEWORK

1. Distinguish between communication at the network layer and communication at the data-link layer.
2. Why does a host or router need to run ARP program all of the time in the background?
3. In a block code, a dataword is 20 bits and the corresponding codeword is 25 bits. What are the value of k, r, and n according to the definitions? How many redundant bits are added to each dataword?
4. What is the Hamming distance for each of the following codewords?

a. $d\,(10000, 00000)$
b. $d\,(10101, 10000)$
c. $d\,(00000, 11111)$
d. $d\,(00000, 00000)$

5. Apply the following operations on the corresponding polynomials:

a. $(x^3 + x^2 + x + 1) + (x^4 + x^2 + x + 1)$
b. $(x^3 + x^2 + x + 1) - (x^4 + x^2 + x + 1)$
c. $(x^3 + x^2) \times (x^4 + x^2 + x + 1)$
d. $(x^3 + x^2 + x + 1) / (x^2 + 1)$

THANK YOU!

ANY QUESTION?