Reliability of Memory Systems

- **Error Detecting Systems:**

  It is the process of encountering errors, resulting from operational deficiencies or noise.

- **Error Detecting Code (EDC):**

  A code that is considered to detect an error in data,

Outline

- The need for error detection and correction
- How simple parity check can be used to detect error
- How Hamming code is used to detect and correct error

Introduction

- Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission.
- Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination.

Introduction

- Two basic strategies for dealing with errors
  - One way is
    - to include enough redundant information (extra bits are introduced into the data stream at the transmitter on a regular and logical basis)
    - along with each block of data sent to enable the receiver to deduce what the transmitted character must have been
  - Other way is
  - to include only enough redundancy to allow the receiver to deduce that error has occurred, but not which error has occurred and the receiver asks for a retransmission

Introduction

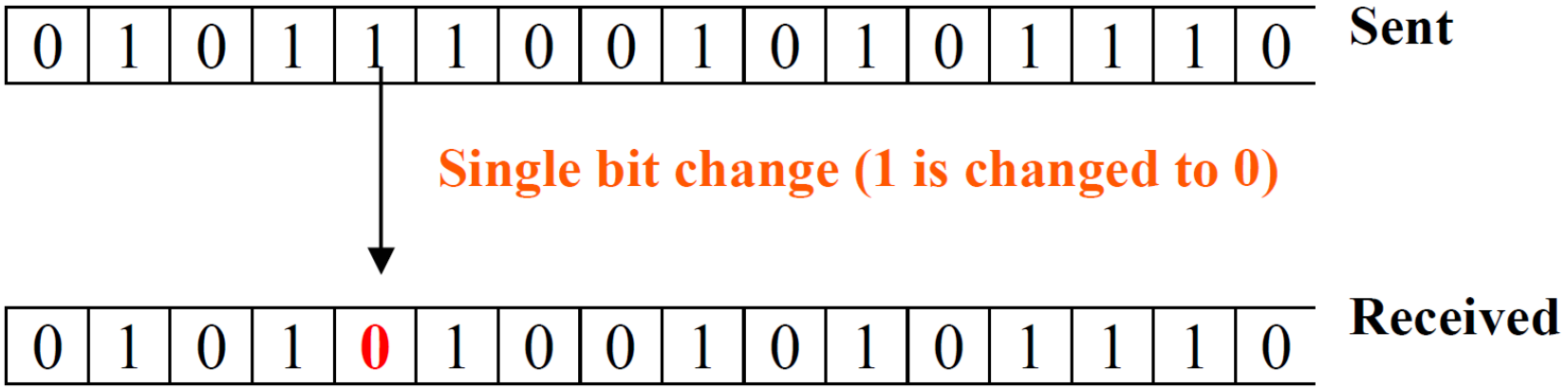
- The former strategy uses **Error-Correcting Codes**
- and latter uses **Error-detecting Codes**.

Types of errors

- Single-bit error
- Burst error
- interferences can change the timing and shape of the signal.
- If the signal is carrying binary encoded data, changes can alter the meaning of the data.

Single-bit Error

- The term single-bit error means that only one bit of given data unit (such as a byte, character, or data unit) is changed from 1 to 0 or from 0 to 1

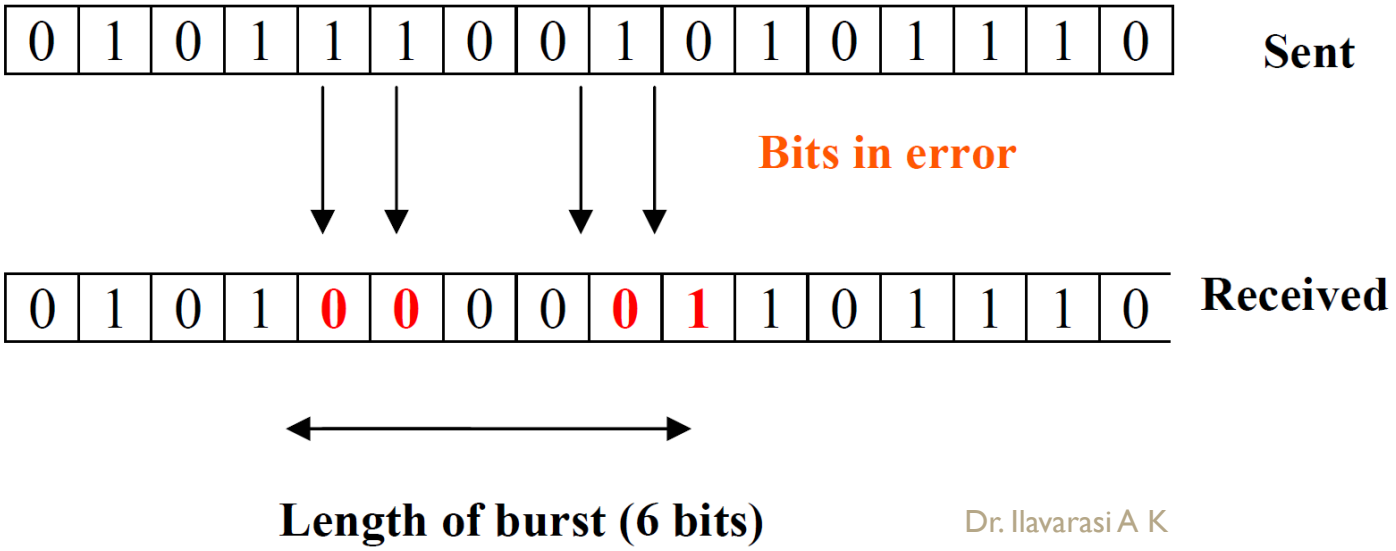


Single-bit Error

- Single bit errors are least likely type of errors in serial data transmission.
- To see why, imagine a sender sends data at 10 Mbps.
- This means that each bit lasts only for 0.1 μs (micro-second).
- For a single bit error to occur noise must have duration of only 0.1 μs (micro-second), which is very rare.
- However, a single-bit error can happen if we are having a parallel data transmission.
- For example, if 16 wires are used to send all 16 bits of a word at the same time and one of the wires is noisy, one bit is corrupted in each word.

Burst Error

- Two or more bits in the data unit have changed from 0 to 1 or vice-versa
- Doesn't means that error occurs in consecutive bits
- The length of the burst error is measured from the first corrupted bit to the last corrupted bit.
- Some bits in between may not be corrupted.



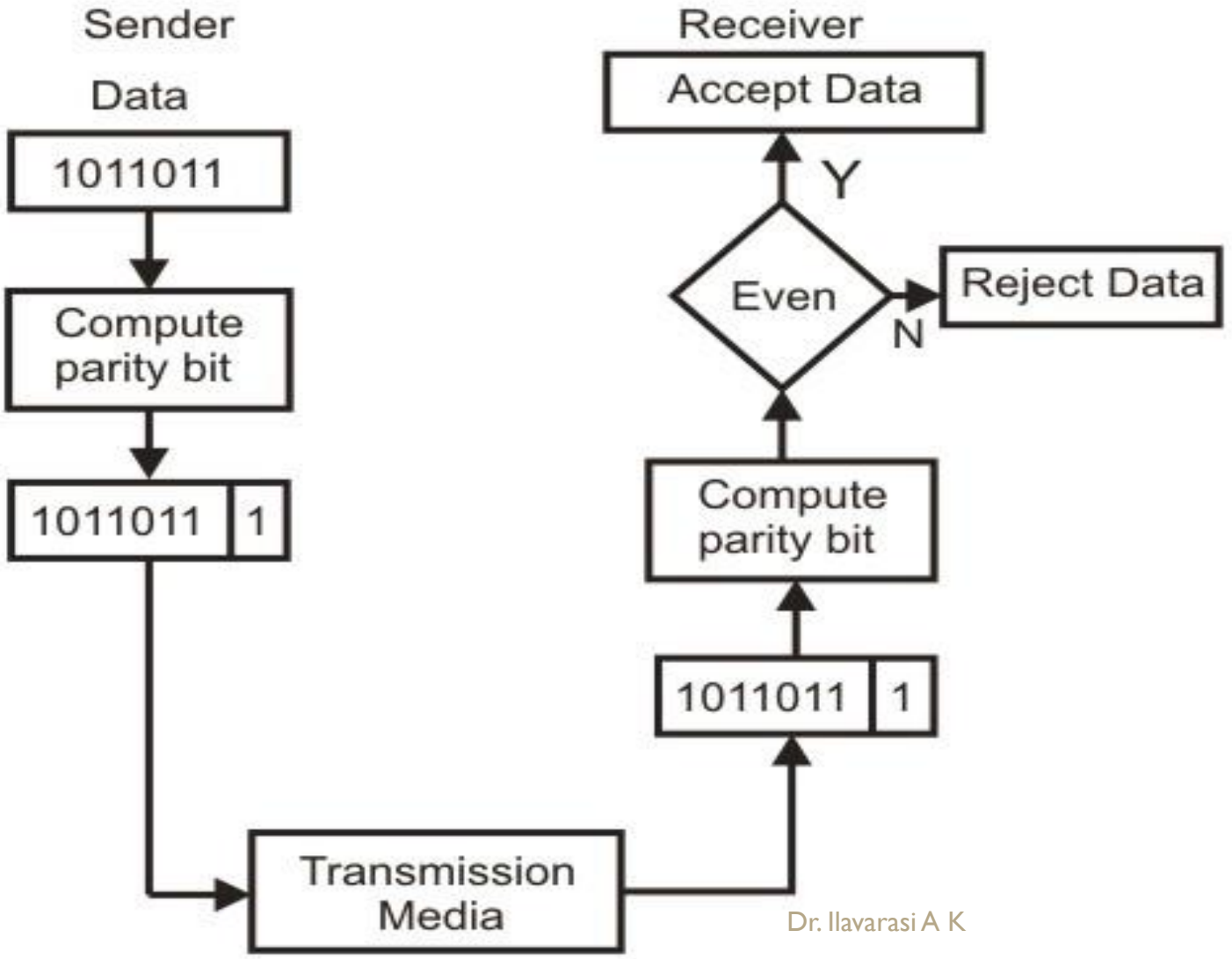
Burst Error

- Likely to happen in serial transmission.
- The duration of the noise is normally longer than the duration of a single bit, which means that the noise affects data; it affects a set of bits
- The number of bits affected depends on the data rate and duration of noise.

Error Detecting Codes

- Basic approach used for error detection is the use of redundancy, where additional bits are added to facilitate detection and correction of errors.
- Popular:
  - Simple Parity check
  - Two-dimensional Parity check
  - Hamming code

Simple Parity Checking or One-dimension Parity Check



Simple Parity Checking or One-dimension Parity Check

- common and least expensive mechanism for error-detection is the simple parity check
- redundant bit called **parity bit**, is appended to every data unit so that the number of 1s in the unit
- Even Parity : In this method,*parity bit* of 1 is added to the block if it contains an odd number of 1's and 0 is added if it contains an even number of 1's.
- At the receiving end the parity bit is computed from the received data bits and compared with the Sent parity bit

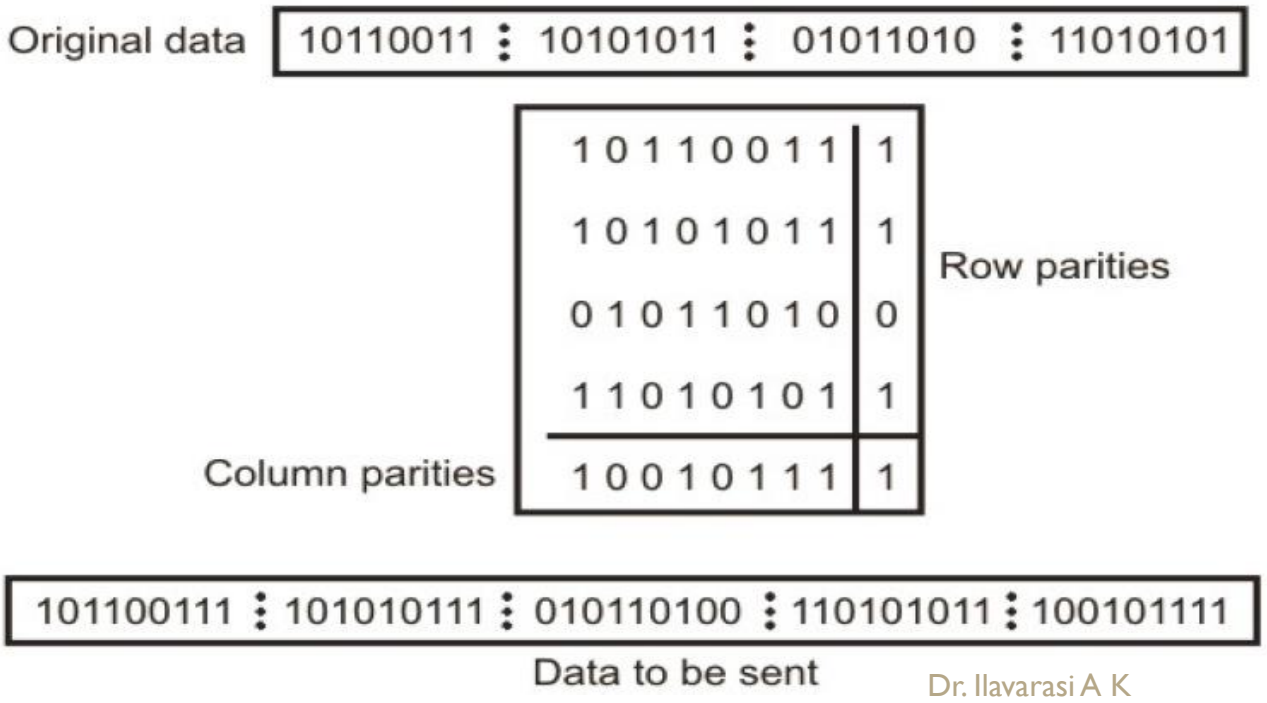
Possible 4-bit data words and corresponding code words

Decimal value	Data Block	Parity bit	Code word
0	0000	0	00000
1	0001	1	00011
2	0010	1	00101
3	0011	0	00110
4	0100	1	01001
5	0101	0	01010
6	0110	0	01100
7	0111	1	01111
8	1000	1	10001
9	1001	0	10010
10	1010	0	10100
11	1011	1	10111
12	1100	0	11000
13	1101	1	11011
14	1110	1	11101
15	1111	0	11110

An observation of the table reveals that to move from one code word to another, at least two data bits should be changed.
Hence these set of code words are said to have a minimum distance (*hamming distance*) of 2
A single parity check code can detect only odd number of errors in a code word.

2D Parity Check

- Parity check bits are calculated for each row, which is equivalent to a simple parity check bit
- Parity check bits are also calculated for all columns then both are sent along with the data.



Reliability of Memory Systems: Continued…

- **Error Correcting Systems:**

  It is the process of discovery of errors, as well as the restoration of the original data.

- **Error Correction Code:**

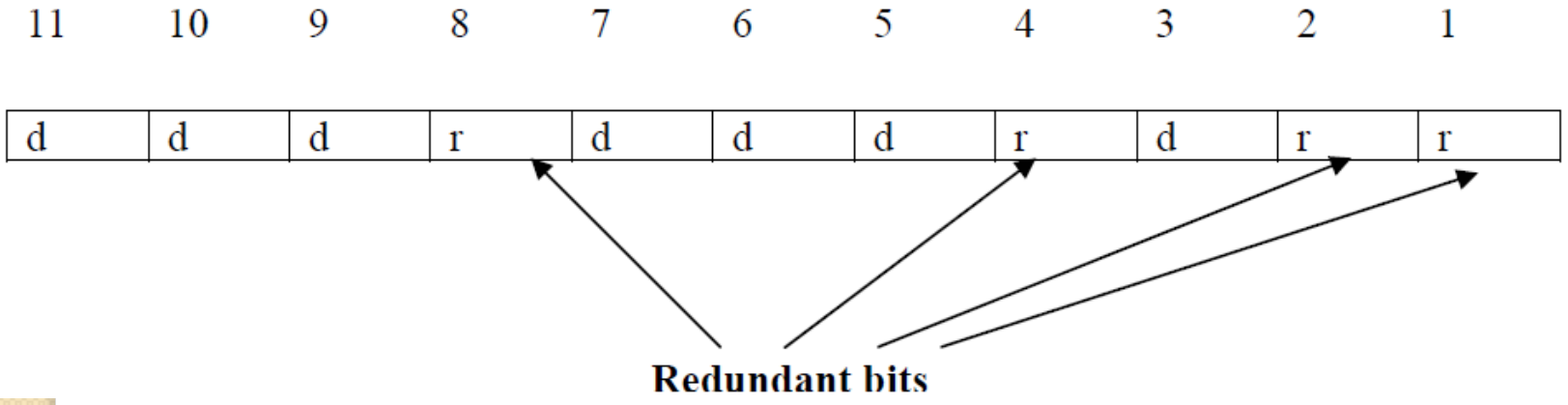
  If any error occurs, then code is used to detect and correct the errors.

Single-bit error correction(SEC)

- For correcting an error one has to know the exact position of error, i.e. exactly which bit is in error
- To calculate the numbers of redundant bits (r) required to correct d data bits, let us find out the relationship between the two.
- So we have (d+r) as the total number of bits, which are to be transmitted; then r must be able to indicate at least d+r+1 different values.
- Of these, one value means no error, and remaining d+r values indicate error location of error in each of d+r locations.
- So, d+r+1 states must be distinguishable by r bits, and r bits can indicates 2^r states.
- Hence, 2^r must be greater than d+r+1.; **2^r >= d+r +1**

Error Detection

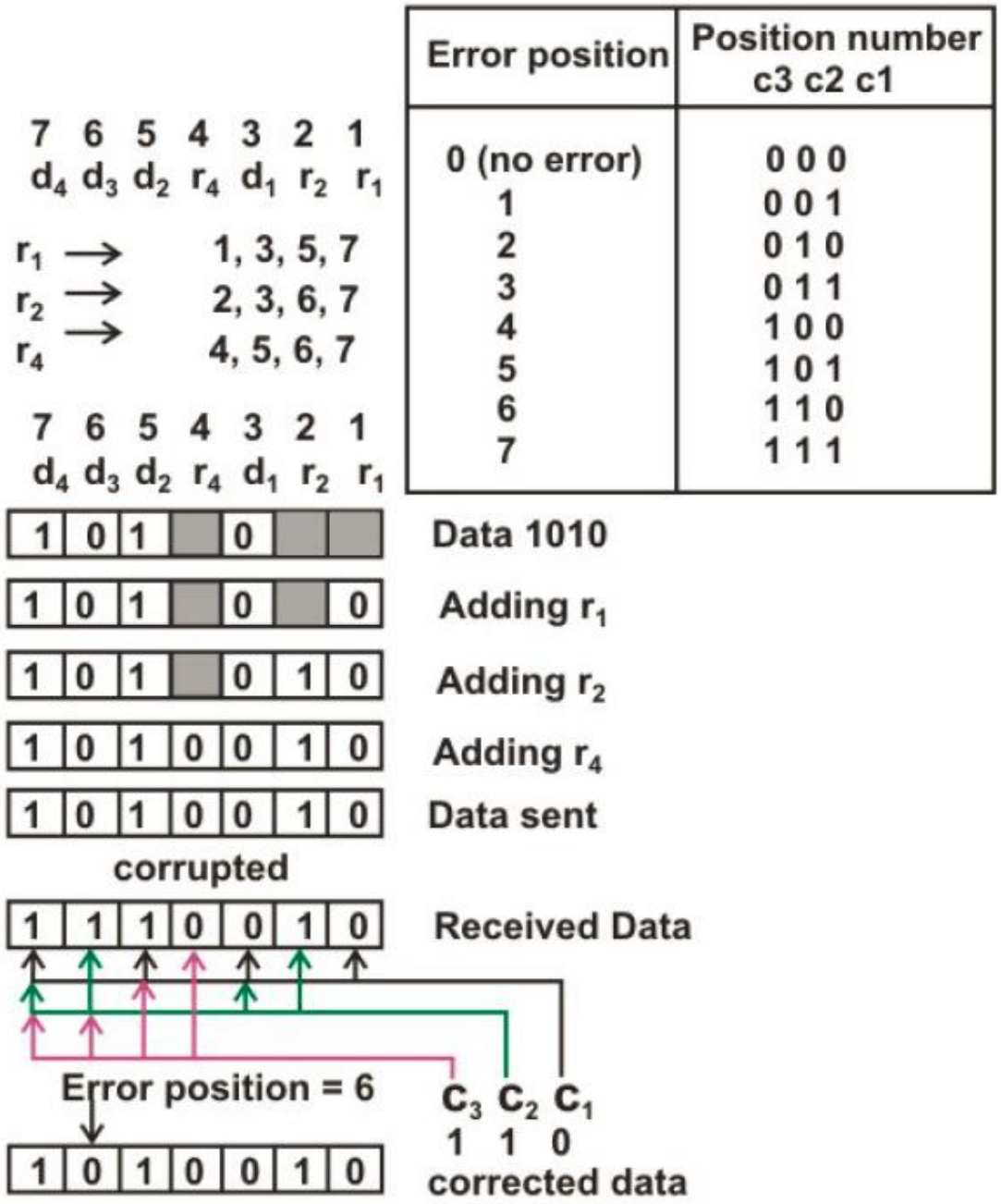
- if d is 7, then the smallest value of r that satisfies the above relation is 4. So the total bits, which are to be transmitted is 11 bits (d+r = 7+4 =11).



Positions of redundancy bits in hamming code

Error Detection

- Basic approach for error detection by using Hamming code is as follows:
  - To each group of m information bits k parity bits are added to form (m+k) bit code as shown in Fig. 3.2.8.
  - Location of each of the (m+k) digits is assigned a decimal value.
  - The k parity bits are placed in positions 1, 2, …, 2k-1 positions. K parity checks are performed on selected digits of each codeword.
  - At the receiving end the parity bits are recalculated. The decimal value of the k parity bits provides the bit-position in error, if any.



Error position	Position number c3 c2 c1
0 (no error)	0 0 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

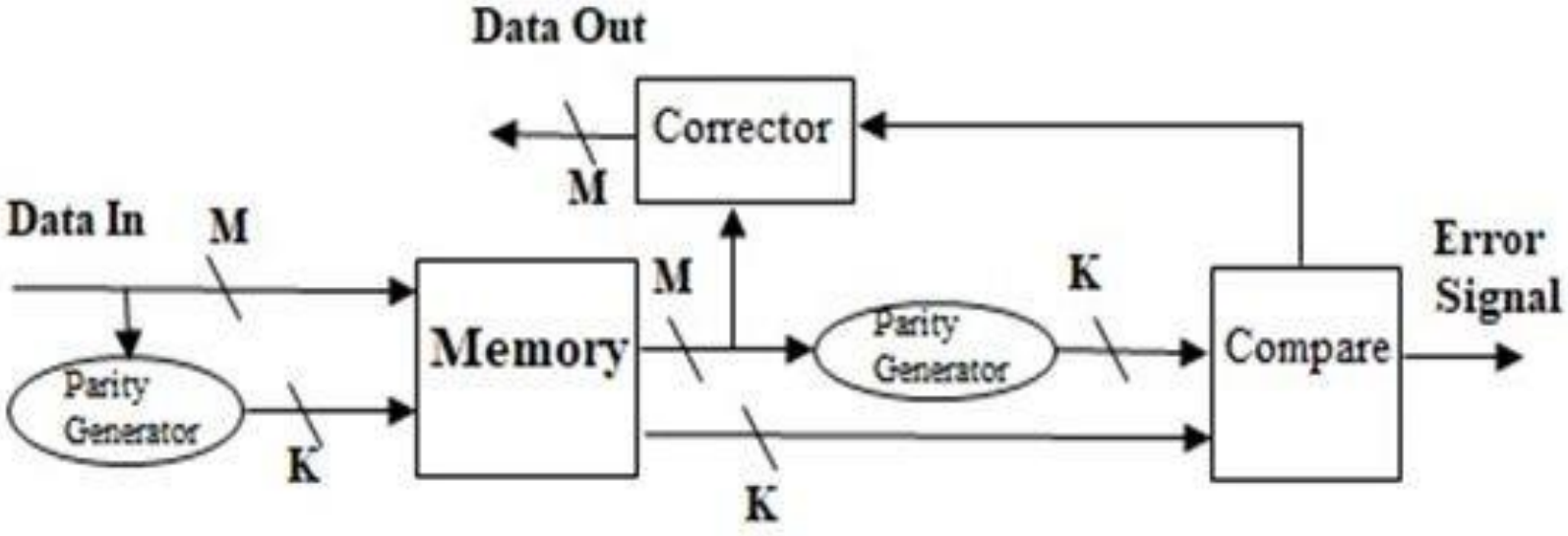
Use of Hamming code for error correction for a 4-bit data

Error Correction

- hamming code is used for correction for 4-bit numbers (d4d3d2d1) with the help of three redundant bits (r3r2r1).
- For the example data 1010, first r1 (0) is calculated considering the parity of the bit positions, 1, 3, 5 and 7.
- Then the parity bits r2 is calculated considering bit positions 2, 3, 6 and 7.
- Finally, the parity bits r4 is calculated considering bit positions 4, 5, 6 and 7 as shown.
- If any corruption occurs in any of the transmitted code 1010010, the bit position in error can be found out by calculating r3r2r1 at the receiving end.
  - For example, if the received code word is 1110010, the recalculated value of r3r2r1 is 110, which indicates that bit position in error is 6, the decimal value of 110.

Hamming Code Architecture

The following general architecture generates a Double Error Detection (DED) and Single-Error Correcting (SEC) code for any number of bits.



Hamming Code Operational Procedure

- Data are to be read into memory.
- A function f (Parity Generator), is applied on the data to yield a code.
- Parity code (K bits) and the data (M bits) are deposited in memory.
- While reading out the word, a new set of K parity code bits is produced from the M data bits.
- Finally, compare new parity code with old parity code bits.
- **Note:** For M data bits or K check bits single error correction hamming code must have **(2^K) - 1 >= M + K**

Arrangement of Hamming code

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Parity bit					P8				P4		P2	P1

Check (Parity) Bits Calculations

P1 = 3⊕5⊕7⊕9⊕11 (bit positions)

P2 = 3⊕6⊕7⊕10⊕11 (bit positions)

P4 = 5⊕6⊕7⊕12 (bit positions)

P8 = 9⊕10⊕11⊕12 (bit positions)

Example:

Generate hamming code (Even parity) for a 8-bit word – 00111001.

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Hamming Code	0	0	1	1	P8	1	0	0	P4	1	P2	P1

$P1 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$

$P2 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$

$P4 = 0 \oplus 0 \oplus 1 \oplus 0 = 1$

$P8 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$

Example:

Detection of error using hamming code

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Hamming Code	0	0	1	1	P8=0	1	0	0	P4=1	0	P2=1	P1=1

P1 = 0⊕0⊕1⊕1⊕0 = 0

P2 = 0⊕0⊕1⊕1⊕0 =0

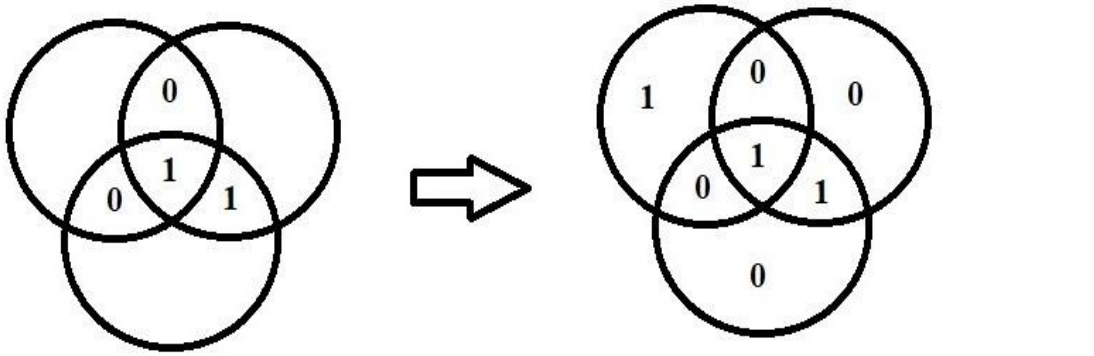
P4= 0⊕0⊕1⊕0 = 1

P8 = 1⊕1⊕0⊕0 = 0

P8	P4	P2	P1
0	1	1	1
⊕ 0	1	0	0
0	0	1	1

Hamming code technique illustrated by using Venn diagram

Hamming Encoding:

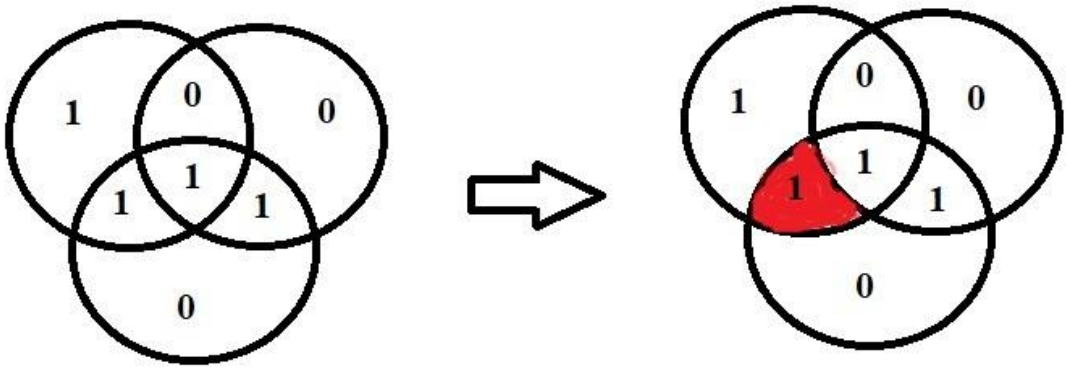


4 bit Data

Hamming Code
(Data + Parity bits)
4 bits(Data) + 3 bits (Parity) = 7 bits

Hamming coding technique is illustrated by using Venn diagram

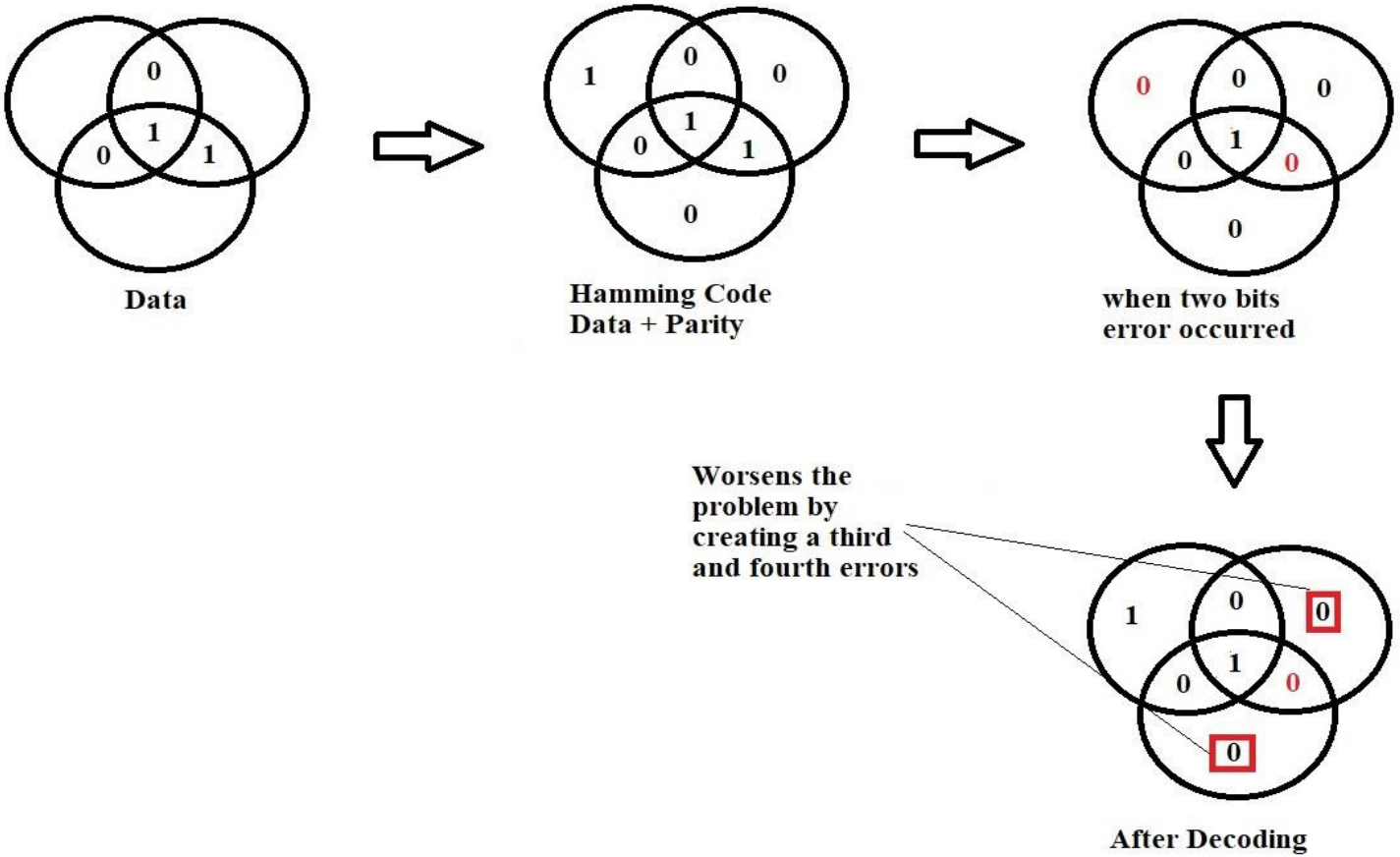
Error
Detection:



Hamming Code
(Data + Parity bits)
4 bits(Data) + 3 bits (Parity) = 7 bits

Error Detection using Parity bits

Hamming SEC-DEC Code



Assignment:

1)Find the number odd parity or check bits needed when we have 2048 bit input word? Solution: input word = 2048 bits = 2^12 =12 bits.

2)Suppose an 8-bit input data is 11101010. Determine the odd parity or check bits?

3)Suppose an 8-bit word is 00111101, Determines the even parity bits, hamming code word and illustrate how to correct when a single bit error occurs?

Assignment: - Solutions

1)Find the number odd parity or check bits needed when we have 2048 bit input word? Solution: input word = 2048 bits = 2^12 =12 bits.

2)Suppose an 8-bit input data is 11101010. Determine the odd parity or check bits? Solution:

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Hamming Code	D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
Input Data	1	1	1	0		1	0	1		0		

P1 = D3⊕D5⊕D7⊕D9⊕ D11 = 0⊕1⊕1⊕0⊕1 = 0

P2 = D3⊕D6⊕D7⊕D10⊕ D11 = 0⊕0⊕1⊕1⊕1 =0

P4= D5⊕D6⊕D7⊕ D12 = 1⊕0⊕1⊕1 = 0

P8 = D9⊕D10⊕D11⊕ D12 = 0⊕1⊕1⊕1 = 0

Assignment: - Solutions (Continued..)

3) Suppose an 8-bit word is 00111101, Determines the even parity bits, hamming code word and illustrate how to correct the error if the received word is 00111111.

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Hamming Code	D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
Input Data	0	0	1	1		1	1	0		1		

P1 = D3⊕D5⊕D7⊕D9⊕ D11 = 1⊕0⊕1⊕1⊕0 = 1

P2 = D3⊕D6⊕D7⊕D10⊕D11 = 1⊕1⊕1⊕1⊕0 =0

P4= D5⊕D6⊕D7⊕D12 = 0⊕1⊕1⊕0 = 0

P8 = D9⊕D10⊕D11⊕D12 = 1⊕1⊕0⊕0 = 0

Supposed error occurred in fifth position that is D5 got 1 instead of 0 then calculate new even parity.

P1 = D3⊕D5⊕D7⊕D9⊕ D11 = 1⊕1⊕1⊕1⊕0 = 0

P2 = D3⊕D6⊕D7⊕D10⊕D11 = 1⊕1⊕1⊕1⊕0 =0

P4= D5⊕D6⊕D7⊕D12 = 1⊕1⊕1⊕0 = 1

P8 = D9⊕D10⊕D11⊕D12 = 1⊕1⊕0⊕0 = 0

Syndrome word

P8 P4 P2 P1

0 0 0 1

⊕ 0 1 0 0

0 1 0 1

Error at 5th location

References:

1. David A. Patterson and . John L. Hennessy "Computer Organization and Design-The Hardware/Software Interface" 5th edition, Morgan Kaufmann, 2011.

2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization, Mc Graw Hill, Fifth edition ,Reprint 2011.

3. W. Stallings, Computer organization and architecture, Prentice-Hall, 8th edition, 2009