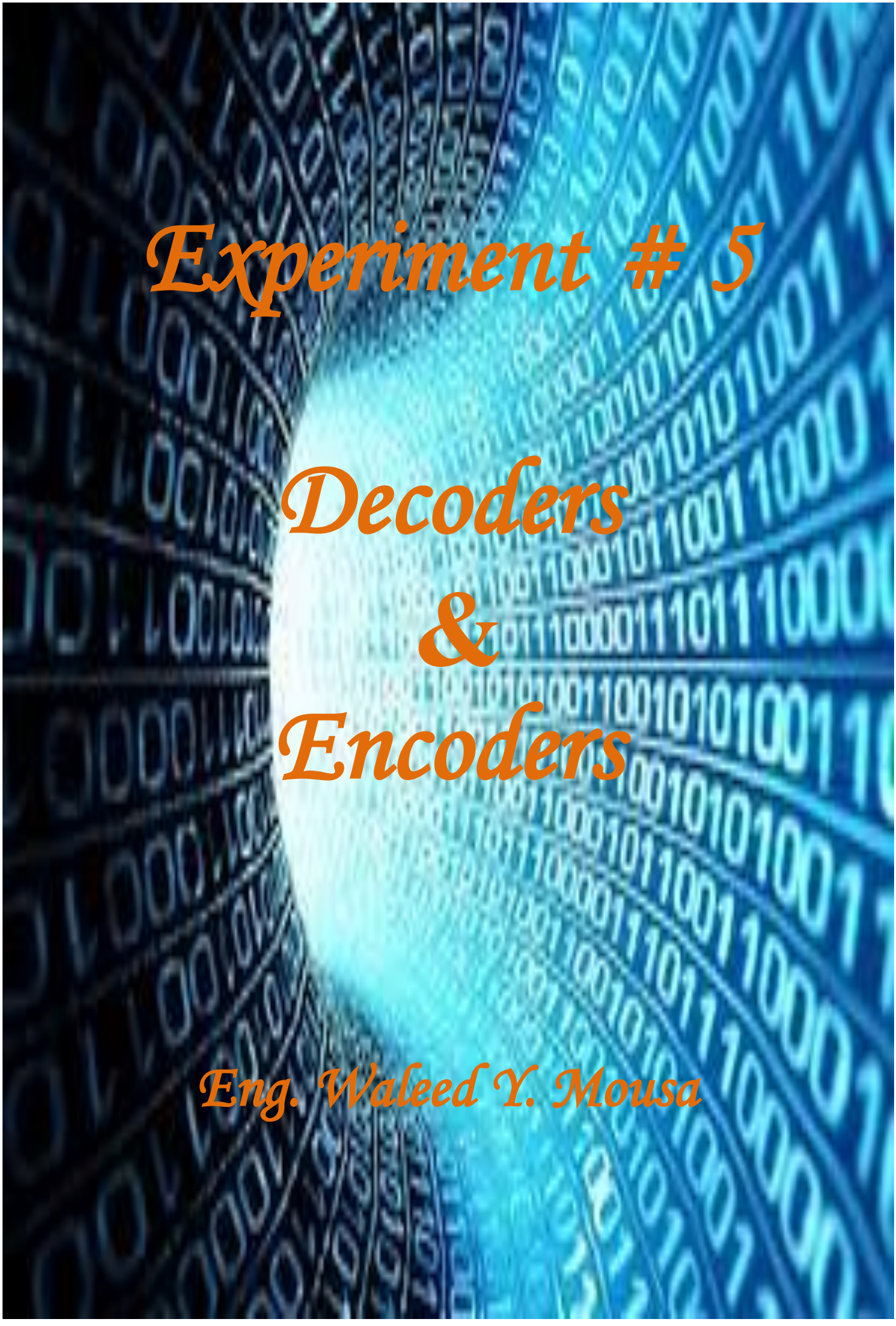


# Experiment # 5

# Decoders & Encoders

Eng. Waleed Y. Mousa

## 1. Objectives:

1. Understanding the construction and operational principles of digital decoders and encoders.

## 2. Theory:

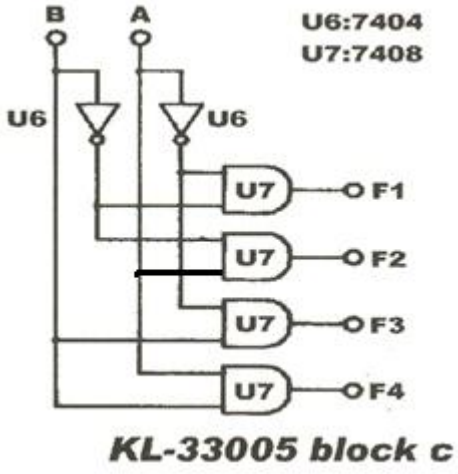
❖ **Decoders:**

A decoder is a logic circuit that will detect the presence of a specific binary number or word. The input to the decoder is a parallel binary number and the output is a binary signal that indicates the presence or absence of that specific number.
It is a combinational circuit that converts binary information from n input lines to a maximum of $2^n$ unique output lines.

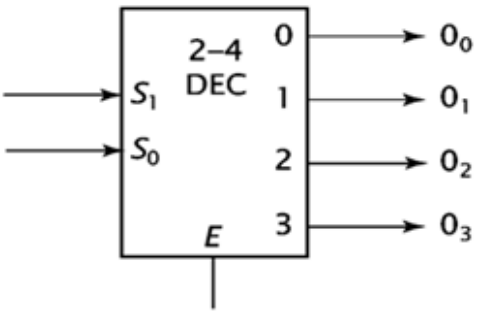
- **2-to-4 decoder:**

| Inputs | | Outputs | | | |
|---|---|---|---|---|---|
| **B** | **A** | **F1** | **F2** | **F3** | **F4** |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 |

o F1 = $\bar{A}.\bar{B}$.
o F2 = $A.\bar{B}$.
o F3 = $\bar{A}.B$.
o F4 = A.B.

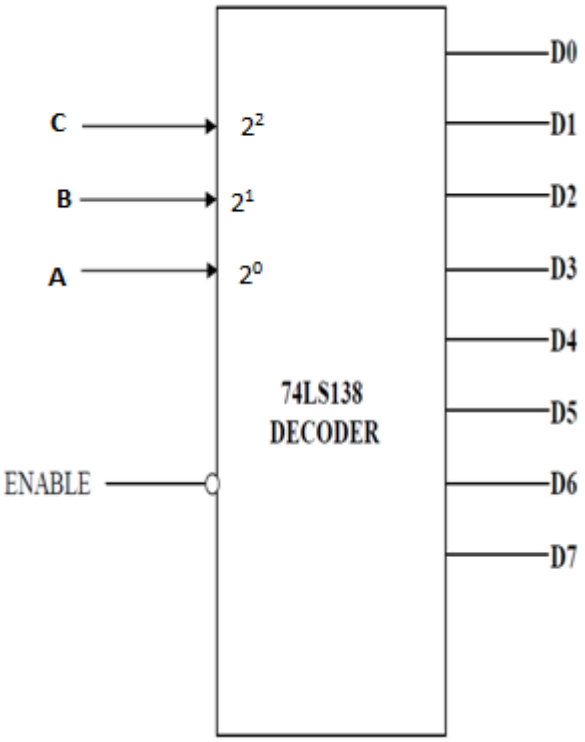


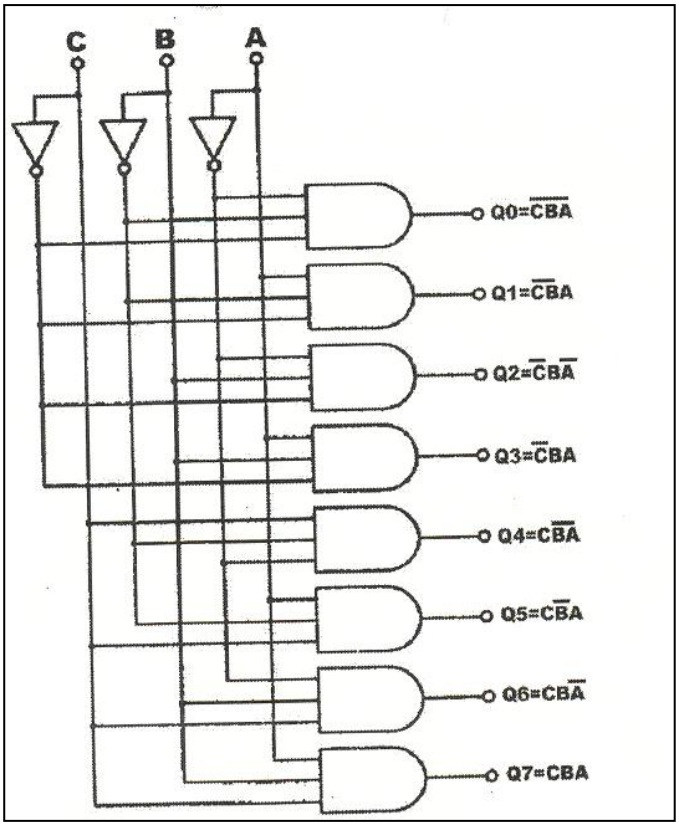
KL-33005 block c




| $S_1$ | $S_0$ | $E$ | $0_0$ | $0_1$ | $0_2$ | $0_3$ |
|---|---|---|---|---|---|---|
| X | X | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 |

- **3-to-8 decoder:**



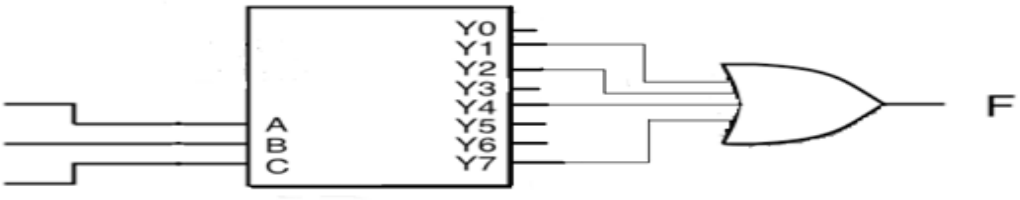

**Example:**

Implement the following truth table using a decoder and OR gate.

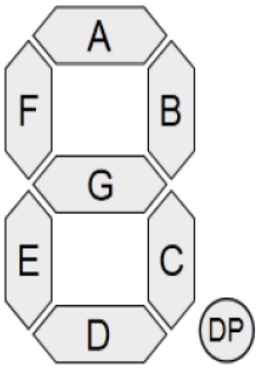
| Inputs | | | Output |
|---|---|---|---|
| A | B | C | F |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

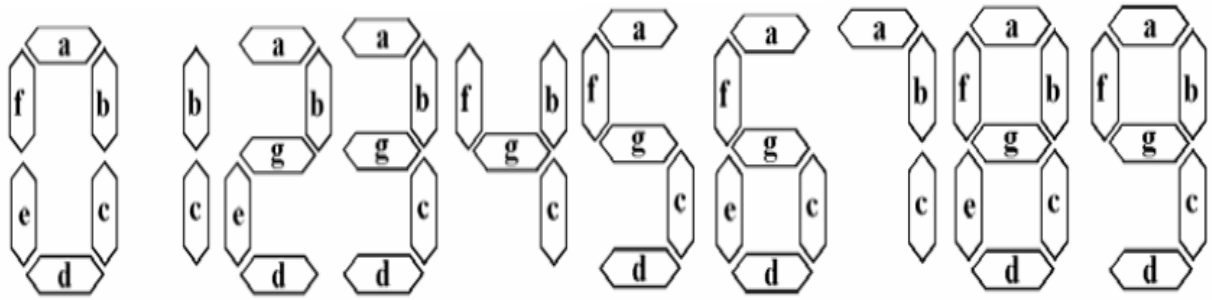
Solution:

$$F(A,B,C) = \sum(1,2,4,7)$$

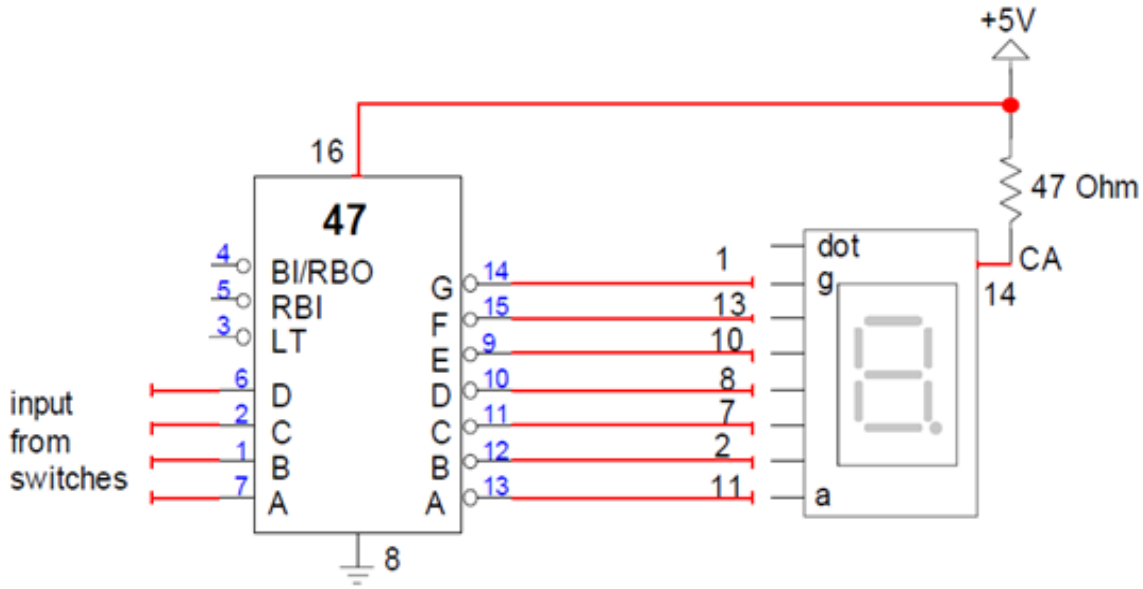


- **BCD-to-Seven Segment decoder:**
  A seven segment LED display contains 7 LEDs. Each LED is called a segment and they are identified as (a, b, c, d, e, f, g) segments.

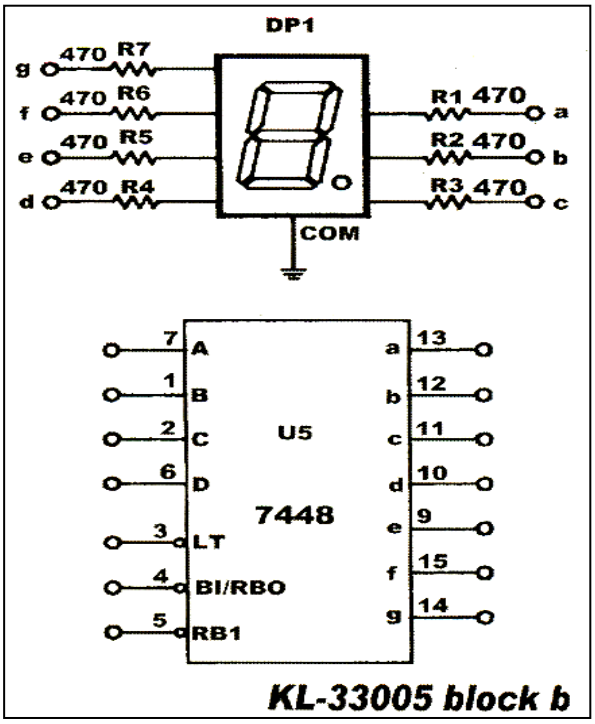


For example if decimal 9 is to be displayed a, b, c, d, f, g must be 0 and the others must be 1 (For common anode type display units), if decimal 5 is to be displayed then a, f, g, c, d must be 0 and the others must be 1.



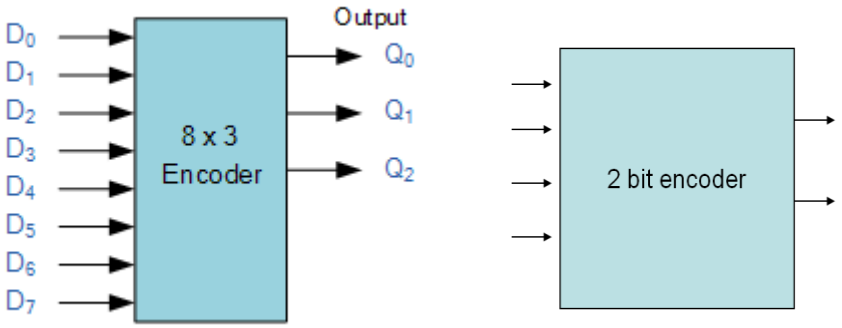

BCD-to-Seven Segment Decoder and 7-segment display



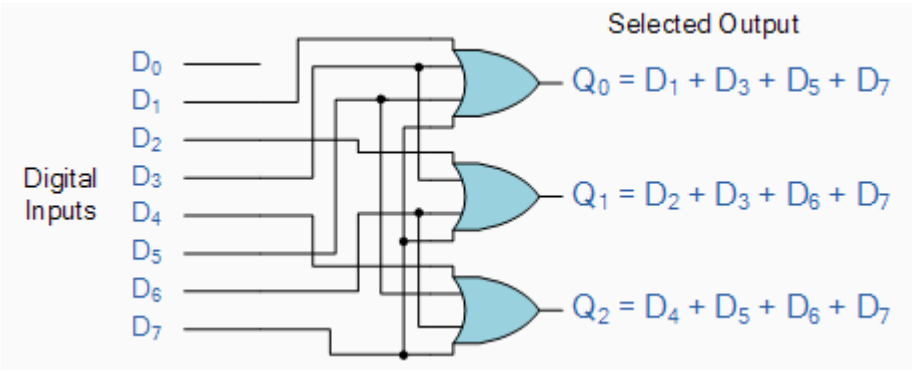

❖ **Encoders:**

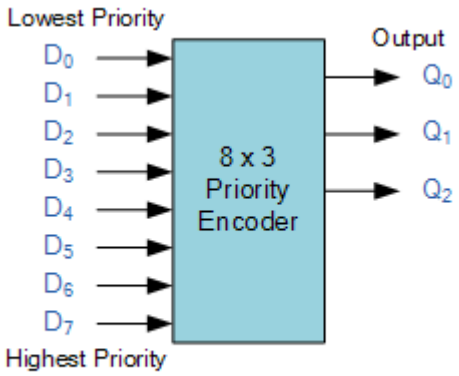
An encoder is a device, circuit, transducer, software program, algorithm or person that converts information from one format or code to another, for the purposes of standardization, speed, secrecy, security, or saving space by shrinking size.
Encoder is a digital circuit that performs the inverse operation of a decoder, generates a unique binary code from several input lines.
Generally encoders produce 2-bit, 3-bit or 4-bit code.

- n bit encoder has $2^n$ input



| Inputs | | | | | | | | Outputs | | |
|---|---|---|---|---|---|---|---|---|---|---|
| D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Q2 | Q1 | Q0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |




| Inputs | | | | | | | | Outputs | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ | $Q_2$ | $Q_1$ | $Q_0$ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | x | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | x | x | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 1 | x | x | x | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | x | x | x | x | 1 | 0 | 0 |
| 0 | 0 | 1 | x | x | x | x | x | 1 | 0 | 1 |
| 0 | 1 | x | x | x | x | x | x | 1 | 1 | 0 |
| 1 | x | x | x | x | x | x | x | 1 | 1 | 1 |

X = dont care

## 3. Lab Work:

### Part 1:

1) Use model KL-33005 block C to construct 2-to-4 decoder, connect inputs A and B to data switches SW0 and SW1, connect outputs F1, F2, F3, F4 to LEDs L0,L1, L2, L3 respectively.
Then test the results.

## Part 2

1) Use KL-33005 block b to construct BCD-to-Seven Segment, connect inputs A, B, C and D to data switches, connect the input (LT) to DIP1.0 and set it to logic 1, connect outputs of BCD to Seven Segment and from Seven Segment to the Leds. Then complete this table.

| **INPUTS** | | | | **7- Segments** | | | | | | | **Display number** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **A** | **B** | **C** | **D** | **a** | **b** | **c** | **d** | **e** | **f** | **g** | |
| 0 | 0 | 0 | 0 | | | | | | | | |
| 0 | 0 | 0 | 1 | | | | | | | | |
| 0 | 0 | 1 | 0 | | | | | | | | |
| 0 | 0 | 1 | 1 | | | | | | | | |
| 0 | 1 | 0 | 0 | | | | | | | | |
| 0 | 1 | 0 | 1 | | | | | | | | |
| 0 | 1 | 1 | 0 | | | | | | | | |
| 0 | 1 | 1 | 1 | | | | | | | | |
| 1 | 0 | 0 | 0 | | | | | | | | |
| 1 | 0 | 0 | 1 | | | | | | | | |

## 4. Exercises:

1) Draw 3-to-8 decoder block without enable. Then find the truth table.

2) Design 3-to-8 decoder using tow 2-to-4 decoders with enables. Then find the truth table.

3) Design a Full Adder using decoder and OR gates.