Digital Logic Design
(EL-227)
LABORATORY MANUAL
Spring-2021

LAB 08
Binary Encoder

STUDENT NAME	ROLL NO	SEC

INSTRUCTOR SIGNATURE& DATE

MARKS AWARDED: /03

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES (NUCES), KARACHI

Date: 26-March-2021

Lab Session 08:Binary Decoder

OBJECTIVES:

After completing this lab, you would be able to know

- To study the basic operation and design of the Encoder circuits
- Explain the working principle of 3-8 line Octal to Binary Encoding
- Understand the usage of Priority Encoder

APPARATUS:

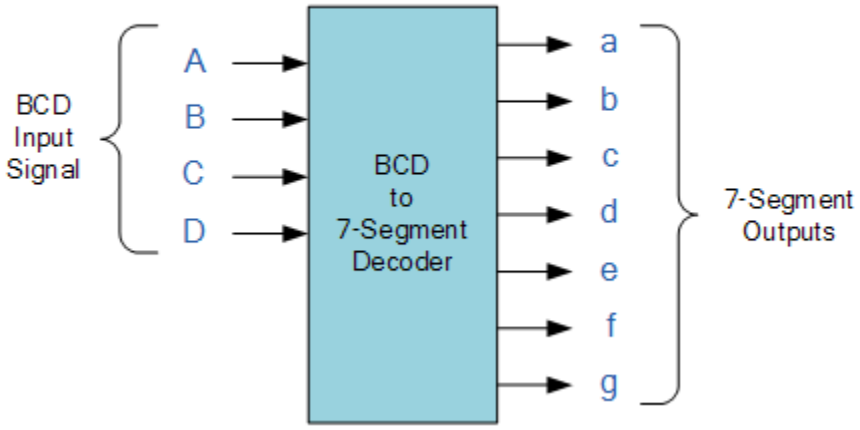
- Logics Trainer, Logics works

Revision decoder lab 7 (before mid)

BCD to 7-Segment Display Decoders

A binary coded decimal (BCD) to 7-segment display decoder such as the TTL 74LS47 or 74LS48, have 4 BCD inputs and 7 output lines, one for each LED segment. This allows a smaller 4-bit binary number (half a byte) to be used to display all the denary numbers from 0 to 9 and by adding two displays together, a full range of numbers from 00 to 99 can be displayed with just a single byte of eight data bits.

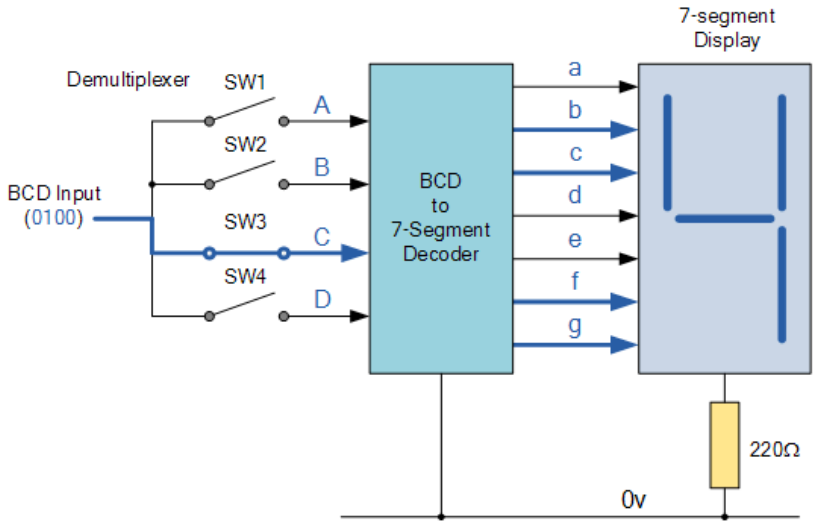
BCD to 7-Segment Decoder



The use of **packed** BCD allows two BCD digits to be stored within a single byte (8-bits) of data, allowing a single data byte to hold a BCD number in the range of 00 to 99.

An example of the 4-bit BCD input (0100) representing the number “4” is given below.

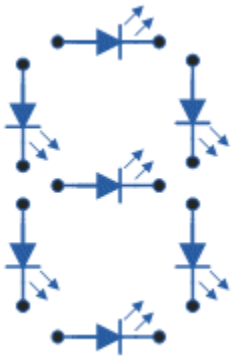
Display Decoder Example No1



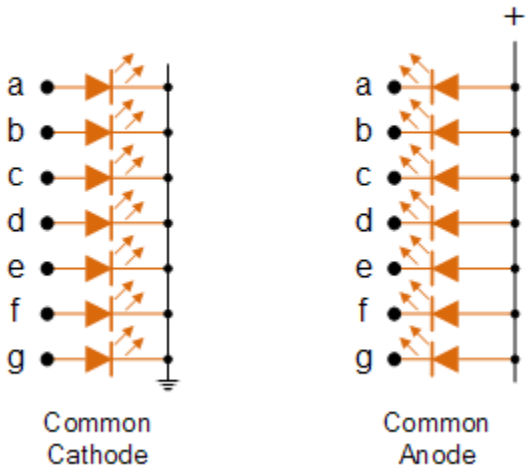
In practice current limiting resistors of about 150Ω to 220Ω would be connected in series between the decoder/driver chip and each LED display segment to limit the maximum current flow. There are different display decoders and drivers available for the different types of available displays, either LED or LCD. For example, the 74LS48 for common-cathode LED types, the 74LS47 for common-anode LED types, or the CMOS CD4543 for liquid crystal display (LCD) types.

Liquid crystal displays (LCD´s) have one major advantage over similar LED types in that they consume much less power and nowadays, both LCD and LED displays are combined together to form larger Dot-Matrix Alphanumeric type displays which can show letters and characters as well as numbers in standard Red or Tri-colour outputs.

LED in logicwork



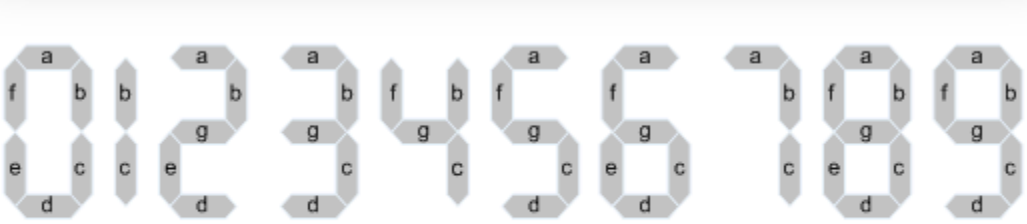
Common Cathode and Common Anode Format



7447

BCD to Seven Segment Driver

7447 IC is particularly used to drive common-anode Seven Segment displays. Its input is a BCD number and output drives a seven segment display.



7-Segment Display Elements for all Numbers.

Encoder

Introduction:

An encoder is a combinational circuit that performs the inverse operation of a decoder. An encoder has a maximum of 2^n input lines and n output lines. The encoder generates binary code at its output lines that represents which input line is active at a given time. In encoder, it is assumed that only one input is active high at a time, if more than one inputs are high simultaneously then ambiguous output is generated. In order to resolve this ambiguity, there must be some input priority function to ensure that only one input is encoded at a time.

A priority encoder is a combinational circuit that encodes the input using priority function i.e. if more than one inputs are high simultaneously than the input having the highest priority will take precedence. Each input line is assigned priority. The most significant input line may be given highest priority and least significant input line the lowest or vice versa. The priority encoder has an additional output to ensure that at least one input line is active high and the binary code at the output lines is valid. Figure 7-2 shows the block diagram of $2^n xn$ priority encoder.

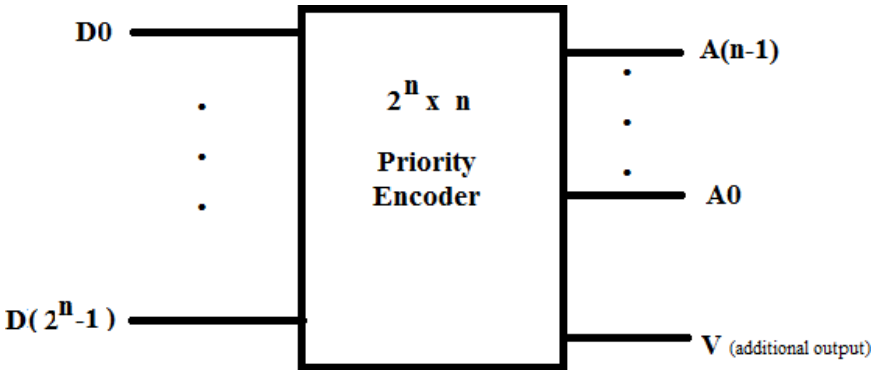


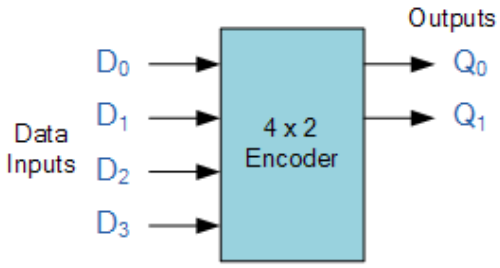
Figure 1 2x4 Encoder Block Diagram

Unlike a multiplexer that selects one individual data input line and then sends that data to a single output line or switch, **Digital Encoder** more commonly called a **Binary Encoder** takes ALL its data inputs one at a time and then converts them into a single encoded output. So we can say that a binary encoder, is a multi-input combinational logic circuit that converts the logic level "1" data at its inputs into an equivalent binary code at its output.

Generally, digital encoders produce outputs of 2-bit, 3-bit or 4-bit codes depending upon the number of data input lines. An "n-bit" binary encoder has 2^n input lines and n-bit output lines with common types that include 4-to-2, 8-to-3 and 16-to-4 line configurations.

The output lines of a digital encoder generate the binary equivalent of the input line whose value is equal to "1" and are available to encode either a decimal or hexadecimal input pattern to typically a binary or "B.C.D" (binary coded decimal) output code.

4-to-2 Bit Binary Encoder



Inputs				Outputs	
D_3	D_2	D_1	D_0	Q_1	Q_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1
0	0	0	0	x	x

Disadvantage

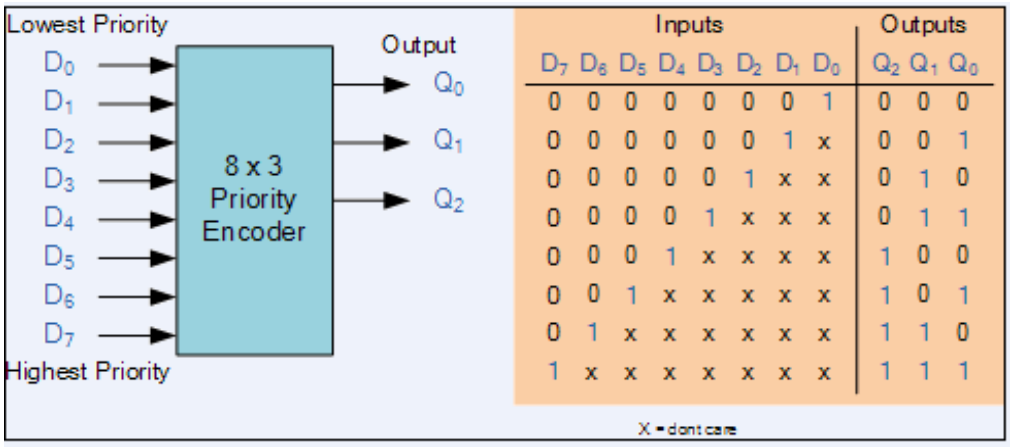
One of the main disadvantages of standard digital encoders is that they can generate the wrong output code when there is more than one input present at logic level "1". For example, if we make inputs D_1 and D_2 HIGH at logic "1" both at the same time, the resulting output is neither at "01" or at "10" but will be at "11" which is an output binary number that is different to the actual input present. Also, an output code of all logic "0"s can be generated when all of its inputs are at "0" OR when input D_0 is equal to one.

One simple way to overcome this problem is to "Prioritise" the level of each input pin. So if there is more than one input at logic level "1" at the same time, the actual output code would only correspond to the input with the highest designated priority. Then this type of digital encoder is known commonly as a **Priority Encoder** or **P-encoder** for short.

Priority Encoder

The **Priority Encoder** solves the problems mentioned above by allocating a priority level to each input. The *priority encoders* output corresponds to the currently active input which has the highest priority. So when an input with a higher priority is present, all other inputs with a lower priority will be ignored.

The priority encoder comes in many different forms with an example of an 8-input priority encoder along with its truth table shown below.



From this truth table, the Boolean expression for the encoder above with data inputs D_0 to D_7 and outputs Q_0, Q_1, Q_2 is given as:

$$Q_0 = \sum(1, 3, 5, 7)$$

$$Q_0 = \sum\left(\overline{D}_7\overline{D}_6\overline{D}_5\overline{D}_4\overline{D}_3\overline{D}_2D_1 + \overline{D}_7\overline{D}_6\overline{D}_5\overline{D}_4D_3 + \overline{D}_7\overline{D}_6D_5 + D_7\right)$$

$$Q_0 = \sum\left(\overline{D}_6\overline{D}_4\overline{D}_2D_1 + \overline{D}_6\overline{D}_4D_3 + \overline{D}_6D_5 + D_7\right)$$

$$Q_0 = \sum\left(\overline{D}_6\left(\overline{D}_4\overline{D}_2D_1 + \overline{D}_4D_3 + D_5\right) + D_7\right)$$

Output Q_1

$$Q_1 = \sum(2, 3, 6, 7)$$

$$Q_1 = \sum\left(\overline{D}_7\overline{D}_6\overline{D}_5\overline{D}_4\overline{D}_3D_2 + \overline{D}_7\overline{D}_6\overline{D}_5\overline{D}_4D_3 + \overline{D}_7D_6 + D_7\right)$$

$$Q_1 = \sum\left(\overline{D}_5\overline{D}_4D_2 + \overline{D}_5\overline{D}_4D_3 + D_6 + D_7\right)$$

$$Q_1 = \sum\left(\overline{D}_5\overline{D}_4(D_2 + D_3) + D_6 + D_7\right)$$

Output Q_2

$$Q_2 = \sum(4, 5, 6, 7)$$

$$Q_2 = \sum\left(\overline{D}_7\overline{D}_6\overline{D}_5D_4 + \overline{D}_7\overline{D}_6D_5 + \overline{D}_7D_6 + D_7\right)$$

$$Q_2 = \sum(D_4 + D_5 + D_6 + D_7)$$

Then the final Boolean expression for the priority encoder including the zero inputs is defined as:

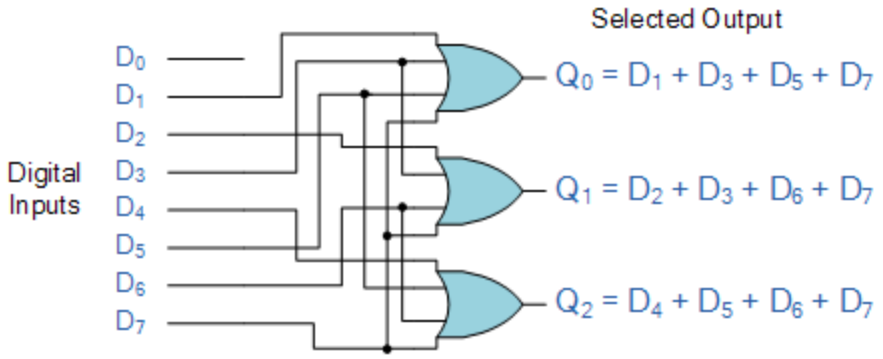
Priority Encoder Output Expression

$$Q_0 = \sum\left(\overline{D}_6\left(\overline{D}_4\overline{D}_2D_1 + \overline{D}_4D_3 + D_5\right) + D_7\right)$$

$$Q_1 = \sum\left(\overline{D}_5\overline{D}_4(D_2 + D_3) + D_6 + D_7\right)$$

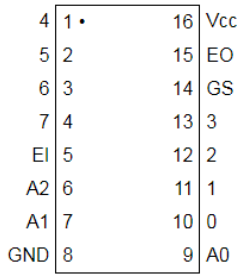
$$Q_2 = \sum(D_4 + D_5 + D_6 + D_7)$$

Digital Encoder using Logic Gates

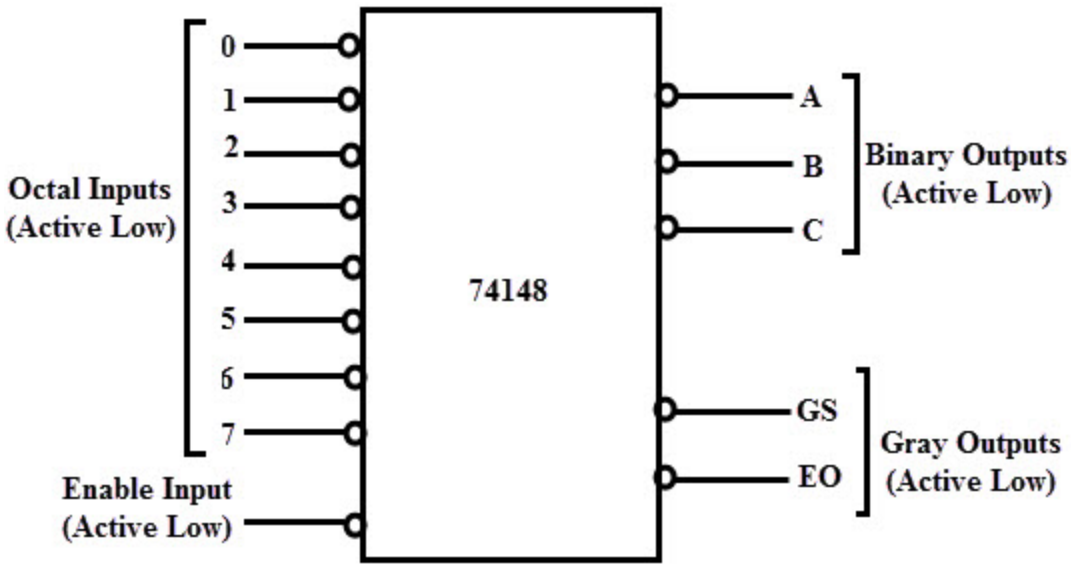


74148 (8 x 3) Octal to Binary Priority Encoder

Priority encoders are available in standard IC form and the TTL 74LS148 is an 8-to-3 bit priority encoder which has eight active LOW (logic “0”) inputs and provides a 3-bit code of the highest ranked input at its output.



Pin	Symbol	Description
1	4	decimal data input (active low)
2	5	decimal data input (active low)
3	6	decimal data input (active low)
4	7	decimal data input (active low)
5	EI	enable input (active low)
6	A2	binary address output (active low)
7	A1	BCD address output (active low)
8	GND	ground
9	A0	binary address output (active low)
10	0	decimal data input (active low)
11	1	decimal data input (active low)
12	2	decimal data input (active low)
13	3	decimal data input (active low)
14	GS	goes low when EI is low and any input is low
15	EO	goes high when EI is low and any input is low (EO = $\overline{GS}$)
16	Vcc	supply voltage



Function of various pins of these ICs is described below:

1. A, B, C, D: Active high inputs representing BCD digits (D being the MSB).
2. OA through OG: Active low outputs to drive segments although of the display.
3. RBI: Ripple Blanking Input. Turns off all the segments if kept low, provided that LT is kept high and all other inputs(A,B,C,D,BI)are kept low. Should be kept high otherwise.
4. BI/RBO: Wire-AND logic serving as a Blanking Input and/or Ripple Blanking Output.
5. BI: Turns off all the segments if low.

6. RBO: Goes to a low level (response condition) along with other outputs,when RBI and inputs A, B, C, and Dare low with LT input at high level.
7. LT: Lamp Test input . Tests whether all segments are working or not. Illuminates all segments if kept low, provided that high. Should be kept high otherwise.
8. VCC and GND: Supply connections lines.

You are required to display the outputs of a 74148 encoder IC on a seven segment display. The circuit is given in figure,

1. Make connections as shown in figure
2. Select any input from 74148IC and observe the corresponding decimal code being displayed on the seven segments.

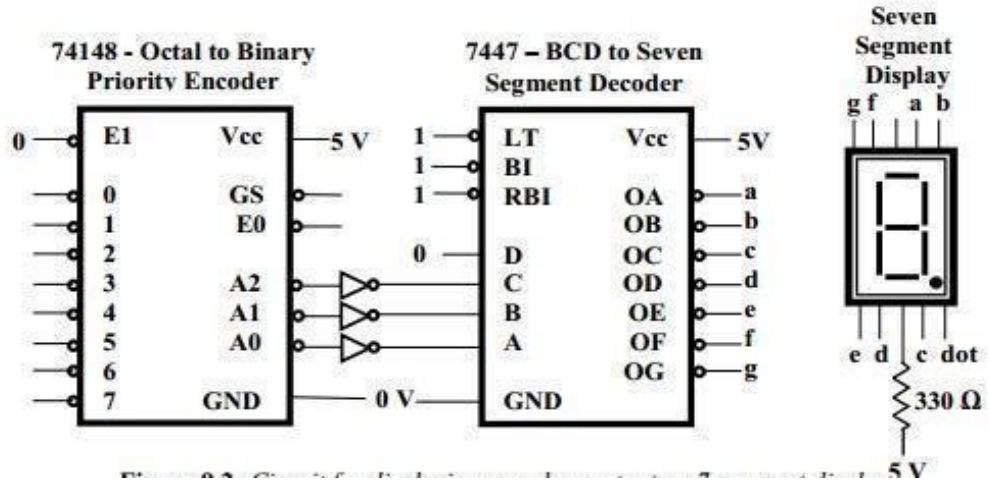


Figure3 8*3 Circuit of Encoder and Segment display

Note :

Please complete lab 07 tasks and lab 8 task in the software and submit one pdf file in which you need to place all the snips of the lab tasks.

Lab Task#1

Implement Octal to Binary Priority Encoder on Bread Board by Using 74148 IC.

INPUTS								OUTPUTS		
D7	D6	D5	D4	D3	D2	D1	D0	Z2	Z1	Z0

Lab Task#2

Implement Decimal to BCD Priority Encoder by Using 74147 IC.

	Decimal Inputs									Outputs			
E1'	1'	2'	3'	4'	5'	6'	7'	8'	9'	Q3'	Q2'	Q1'	Q0'