**EXPERIMENT NO**

**IMPLEMENTATION OF** **ENCODER** **AND** **DECODER** **USING** **IC** **74138** **&** **74148**

**COMPONENTS**

1. Digital Logic Trainer

2. IC 74138 IC 74148

**RELEVANT** **THEORY** **TOPICS**

Combinational logic Decoders, Encoders (Refer Article 4.10, 4.11of Digital design, 4𝑡ℎ Edition by Morris Mano)

**OBJECTIVE**

To study Encoder and Decoder.

**THEORY**

An encoder circuit has more input lines and fewer output lines.

A decimal to BCD encoder (10 line to 4 line) will convert (at any one time) one active input out of ten to a BCD code output.

An octal-to-binary encoder (8 line to 3 line) will convert (at any one time) one-of-eight inputs to a binary code output

A decodercircuitfewinputlinesand more output lines.

A binary-to-octal decoder converts 3 binary bits into 8 outputs (only one which will be active at one time)

A BCD decoder converts a 4-bit BCDcodeoninto 10 output outputs (only one which will be active at one time).

A hexadecimal decoder converts a 4-bit binary code on the input to a 1-of-16 output.

Decoders are often used in microprocessor systems to decode the address information from the microprocessor in order to select the correct memory chip.

**TASK** **1:** **ENCODER**

IC 74148 is a 8- Line-to- 3-line (octal to binary) Priority Encoder. It has 8 Inputs (0-7), an Enable Input **EI**, an Enable Output **EO**, 3 Output (A0-A2), and a **Gs** Output. Details as under:

* **A0-A2** outputs reflect a code that is equal to the highest valued active input.
* **Gs** **output** goes **low** any time any of the input goes low (this low signal is Used for interrupt request to CPU, when connected for the purpose).
* **EI** **and** **EO** are used for cascading more than one 74148 together.

Functional block diagramof IC 74148isattached. Connection diagramandTruth table is shown below:

|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| **EI** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **A2** | **A1** | **A0** | **Gs** | **Eo** |
| 0 | X | X | x | x | x | X | X | 0 |  |  |  | 0 | 1 |
| 0 | X | X | x | x | x | X | 0 | 1 |  |  |  | 0 | 1 |
| 0 | X | X | x | x | x | 0 | 1 | 1 |  |  |  | 0 | 1 |
| 0 | X | X | x | x | 0 | 1 | 1 | 1 |  |  |  | 0 | 1 |
| 0 | X | X | x | 0 | 1 | 1 | 1 | 1 |  |  |  | 0 | 1 |
| 0 | X | X | 0 | 1 | 1 | 1 | 1 | 1 |  |  |  | 0 | 1 |
| 0 | X | 0 | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  | 0 | 1 |

Wire the circuit as per figure above and fill in the blanks in the truth table.

**TASK** **2:** **DECODERS:**

IC74138 has beenusedasadecoder.

It has **3**-SelectInputsand8-DataOutputs. Functional block diagram of IC 74138 is attached. **Note** that the IC 74138 has **Enable** **Inputs** which we will not use during decoder operation, therefore we will keep G1 as high and G2A and G2B as low so that the output of Enable gate remains always high and does not interfere with our desired result. Also **note** that the **output** **of** **the** **IC** **is** **active** **low**. Connection diagram and Truth table of the IC 74138 when used as decoder is shown below:

Note: Output of the IC 74138 is active low, so the output line having a Zero

In the Truth Table will be selected

|  |  |
| --- | --- |
| **SELECT** **INPUT** | **DATA** **OUTPUT** |
| **C** | **B** | **A** | **O7** | **O6** | **O5** | **O4** | **O3** | **O2** | **O1** | **O0** |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Wire the circuit as per figure 1 above and verify the results.

**Observations/Comments/Explanation of Results:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ASSIGNMENT

P1. Design priority encoder.

P2. Implement full adder and full subtractor using decoder.