

**Faculty of Engineering and Technology**

**Department of Electrical and Computer Engineering**

**ENCS 211**

**Digital Electronics and Computer Organization Lab**

**Experiment #3**

**Comparators, Adders and Subtractors**

**Student’s names: Ahmad Salah**

**Student’s id’s: 1130083**

1. **Abstract:**

In this experiment we will show you how to construct a Decoder, Encoder, Multiplexers and Demultiplixers.

We implement them using a basic logic gates, IC, KL’s and Multimeter.

***2) Theory:***

***\* Encoder:***

An encoder is a digital circuit that perform inverse operation of a decoder. An encoder has 2^n input lines and n output lines. In encoder the output lines generates the binary code corresponding to the input value , In encoder it is assumed that only one input has a value of one at any given time otherwise the circuit is meaningless. It has an ambiguila that when all inputs are zero the outputs are zero. The zero outputs can also be generated when D0 = 1.

***\* Decoder:***

A decoder is a multiple input multiple output logic circuit which converts coded input into coded output where input and output codes are different. The input code generally has fewer bits than the output code. Each input code word produces a different output code word i.e. there is one to one mapping can be expressed in truth table. In the block diagram of decoder circuit the encoded information is present as n input producing 2^n possible outputs. 2^n output values are from 0 through out 2n–1.

***\* Multiplexer:***

Multiplexer means transmitting a large number of information units over a smaller number of channels or lines. A digital multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line. The selection of a particular input line is controlled by a set of selection lines. Normally there are 2^n input line and n selection lines whose bit combination determine which input is selected.

***\* Demultiplexer:***

The function of Demultiplexer is in contrast to multiplexer function. It takes information from one line and distributes it to a given number of output lines. For this reason, the demultiplexer is also known as a data distributor. Decoder can also be used as demultiplexer. In the 1: 4 demultiplexer circuit, the data input line goes to all of the AND gates. The data select lines enable only one gate at a time and the data on the data input line will pass through the selected gate to the associated data output line.

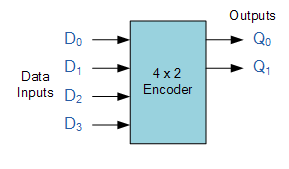
***\* Procedure:***

3.1) Encoder:

3.1.1) 4-to-2-Line Encoder:

This encoder has 4 inputs and 2 outputs, this encoder gives for each bit of the inputs a special value, we have 2 outputs, and this gives 4 input base on inputs number = 2^2,

When A=1 its code (0 0) , B=1 🡪 (0 1), C=1 🡪(1 0) , and when D = 1 🡪 (11), these outputs are shown when the rest of the inputs equal zero, if any other input has 1, then the output would be (0 0).



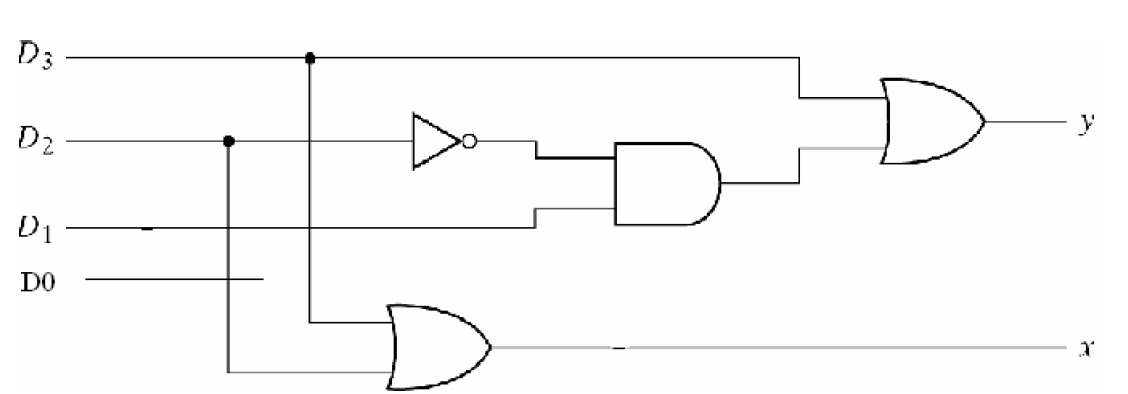
**4\*2 Encoder Design**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **F9** | **F8** |
| **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **0** | **1** | **0** | **0** |
| **0** | **0** | **1** | **0** | **0** | **1** |
| **0** | **0** | **1** | **1** | **0** | **0** |
| **0** | **1** | **0** | **0** | **1** | **0** |
| **0** | **1** | **0** | **1** | **0** | **0** |
| **0** | **1** | **1** | **0** | **0** | **0** |
| **0** | **1** | **1** | **1** | **0** | **0** |
| **1** | **0** | **0** | **0** | **1** | **0** |
| **1** | **0** | **0** | **1** | **0** | **0** |
| **1** | **0** | **1** | **0** | **0** | **0** |
| **1** | **0** | **1** | **1** | **0** | **0** |
| **1** | **1** | **0** | **0** | **0** | **0** |
| **1** | **1** | **0** | **1** | **0** | **0** |
| **1** | **1** | **1** | **0** | **0** | **0** |
| **1** | **1** | **1** | **1** | **0** | **0** |

**4\*2 Line Encoder Truth Table**

3.1.2) 4-to -2 priority encoder:

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.



**4\*2 Priority Encoder Design**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **D0** | **D1** | **D2** | **D3** | **X** | **Y** |
| **0** | **0** | **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** |  |  |
| **1** | **0** | **1** | **0** |  |  |
| **1** | **0** | **1** | **1** |  |  |
| **1** | **1** | **0** | **0** |  |  |
| **1** | **1** | **0** | **1** |  |  |
| **1** | **1** | **1** | **0** |  |  |
| **1** | **1** | **1** | **1** |  |  |