**Lab 7**

**Objectives:**

* To learn the implementation of Boolean function using multiplexer
* To learn how to implement Multiplexers using decoders

**2-to-4 line decoders:**

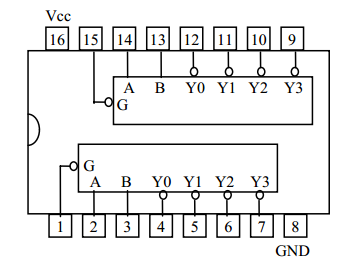
74LS139 IC contains two fully independent 2-to-4 line decoders with active low enables. The function table and connection diagram for this IC are shown below:

**Function Table:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Enable** | **Selection Inputs** | | **Outputs** | | | |
| **G** | **B** | **A** | **Y0** | **Y1** | **Y2** | **Y3** |
| H | X | X | H | H | H | H |
| L | L | L | L | H | H | H |
| L | L | H | H | L | H | H |
| L | H | L | H | H | L | H |
| L | H | H | H | H | H | L |

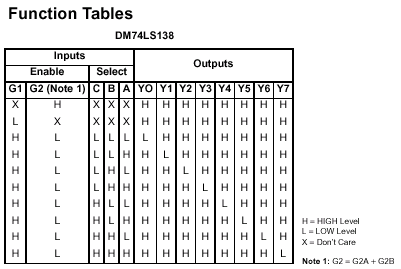
H= Logic High, L= Logic Low, X= Don’t Care

**Connection Diagram:**

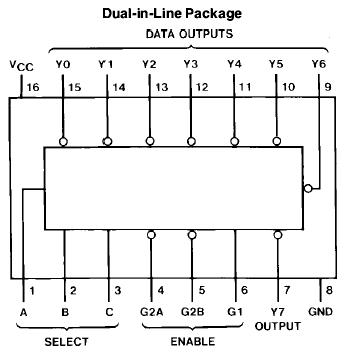
****

**3-to-8 line decoders:**

74LS138 IC contains 3-to-8 line decoder. The function table and connection diagram for this IC are shown below:

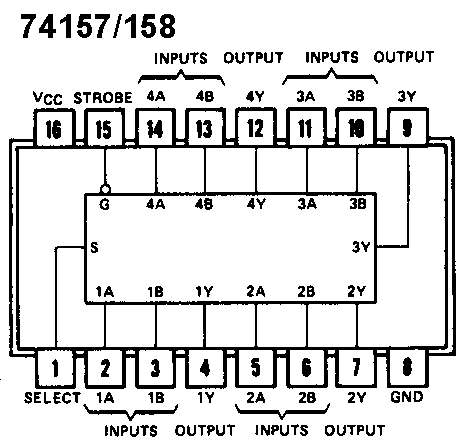
****

**Connection Diagram:**



**2x1 MUX:**

74LS157 IC is a dual 4x1 MUX with active low enable.

****

**4x1 MUX:**

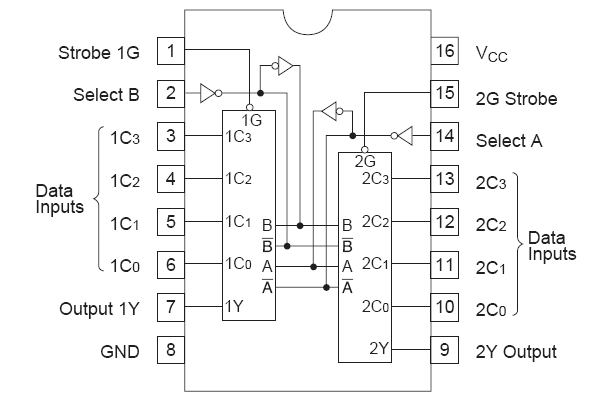
74LS153 IC is a dual 4x1 MUX with active low enables. Two 4x1 MUXs with common selection pins but independent inputs and independent outputs is known as dual 4x1 MUX. The function table and connection diagram for this IC are shown below:

**Function Table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Strobe (Enable)** | **Selection Inputs** | | **Data Inputs** | | | | **Output** |
| **G** | **B** | **A** | **C0** | **C1** | **C2** | **C3** | **Y** |
| **H** | **X** | **X** | **X** | **X** | **X** | **X** | **L** |
| **L** | **L** | **L** | **L** | **X** | **X** | **X** | **L** |
| **L** | **L** | **L** | **H** | **X** | **X** | **X** | **H** |
| **L** | **L** | **H** | **X** | **L** | **X** | **X** | **L** |
| **L** | **L** | **H** | **X** | **H** | **X** | **X** | **H** |
| **L** | **H** | **L** | **X** | **X** | **L** | **X** | **L** |
| **L** | **H** | **L** | **X** | **X** | **H** | **X** | **H** |
| **L** | **H** | **H** | **X** | **X** | **X** | **L** | **L** |
| **L** | **H** | **H** | **X** | **X** | **X** | **H** | **H** |

H= Logic High, L= Logic Low, X= Don’t Care

**Connection Diagram:**



**Half Adder:**

Half adder is a logic circuit that performs binary addition of two 1-bit numbers. It generates two outputs namely ‘**Sum**’ and ‘**Carry**’.

**Truth Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **Carry** | **Sum** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

**Boolean Expressions of Outputs:**

**Full Adder:**

Full adder is a logic circuit that performs binary addition of two 2-bit numbers. It generates two outputs namely ‘**Sum**’ and ‘**Carry**’.

**Truth Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **Carry** | **Sum** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

**Boolean Expressions of Outputs:**

or

**Lab Tasks:**

**Question 1(Hardware implementation)**

Implement 4x1 mux using one 2x4 decoder, four NOT, four ANDs and three OR gates.

**Question 2:(Hardware implementation)**

Implement the following function using 4x1 mux

F (X, Y, Z) = m1 + m2 + m6 + m7  
**Question 3:( Logicworks)**

Implement Dual 2x1 MUX Using 2x1 MUX(s)only

**Question 4:**

Implement full adder on logic works.