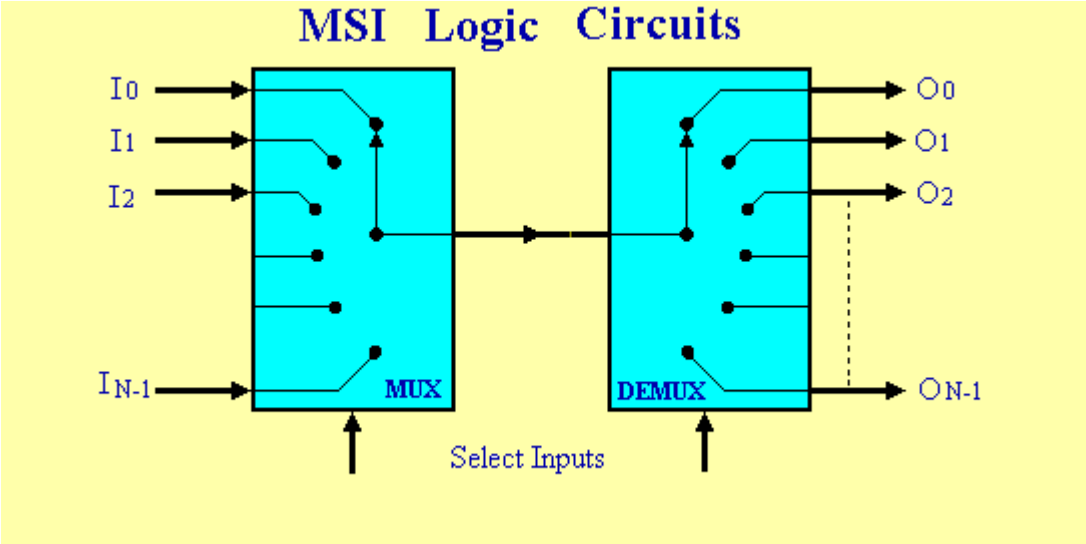
9. MSI Logic Devices



Objectives

- Analyze and use decoders and encoders in various types of circuit applications.
- Understand the operation of multiplexers and demultiplexers by analyzing several circuit applications.

Introduction

• Categorization of ICs

Digital ICs are broadly categorized based on number of gates found in the IC, as follows:

One definition is:

		gates per die
SSI	Small Scale Integration	< 25
MSI	Medium Scale Integration	25 - 150
LSI	Large Scale Integration	150 - 1000
VLSI	Very Large Integration	> 1000

Other definitions are based on the number of transistors in the die.

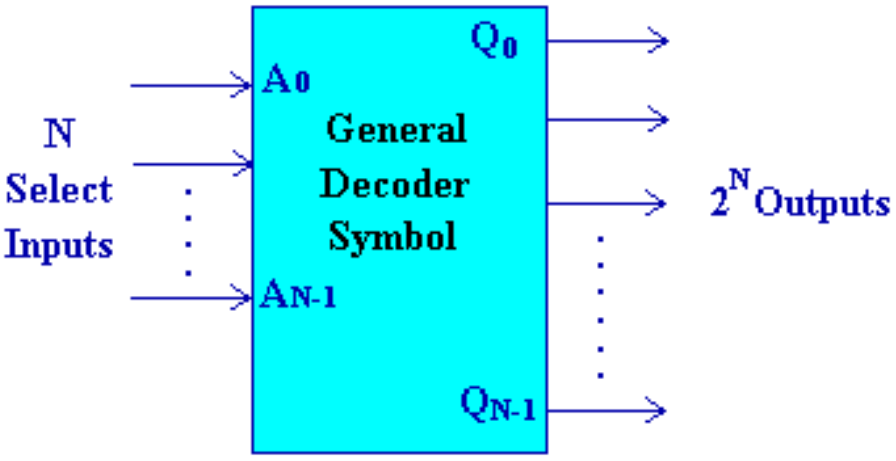
Decoders

Definition:
A decoder is a logic circuit that will convert an *N*-bit binary input code into *M* output lines.
For each of the M output lines, only *ONE* will be activated for a given input code.

If *N*, is the number if input lines, there can be up to 2^N possible output lines.

E.g., if N = 4, i.e. 4 input lines,
Then there can be up to $2^N = 16$ different possible output lines

The outputs can be either *active HIGH* or *active LOW* depending on the circuit configuration.



The decoder building block has the characteristics that for a given encoded input, *only ONE* output is true, while the remaining outputs are false.

Not all decoders utilize the 2^N possible input combinations.

E.g., In a BCD-to-decimal decoder, there are 4 select inputs giving $2^4 = 16$ possible input combinations or codes. Only ten of the sixteen available combinations or codes are used. Any code greater than 1001 (non-BCD) will not result in any of the 10 outputs being active.

Most decoder outputs are normally *active LOW*, i.e. when an output is active (or selected), it is in logic Low state.

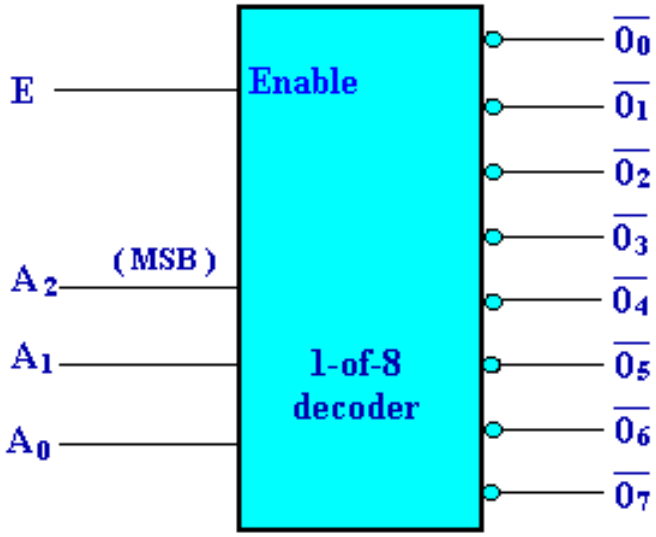
Besides select inputs, most decoders also have one or more **Enable inputs**. These inputs are used to enable or disable the whole decoder IC. They also allow several similar decoder devices to be connected together to form a bigger decoder unit.

When enabled, one of the outputs of the decoder will be active.

Conversely when disabled, *All* outputs of the decoder will be inactive regardless of the combinations applied to the select inputs.

Decoders are typically described based on the number of Select inputs to the Outputs.

E.g: For the decoder shown below,



All these four descriptions can be used to describe the same device.

3 line to 8 line decoder	Basically refers to a decoder with 3 select inputs, and 8 data outputs.
3 : 8 decoder	
1 of 8 decoder	An apt description meaning: only one of the 8 outputs will be active at any one time.
Octal to Binary decoder	Not a very appropriate description.

Question1: How many Select inputs are there in a 1-of-64 decoder?

Answer: 6 select inputs generate 2^6 or 64 codes or combinations.

Question2: What is the maximum possible number of outputs in a decoder with 4 select and 2 enable inputs?

Answer: Maximum number of outputs = 2^4 or 16 outputs.

Examples of Decoder Applications

- Decoders are typically used in Personal Computers (PC) for the purpose of decoding memory and IO systems.
- Decoders can also be used for the purpose of decoding counters and as data collectors.

3:8 Decoder

A 3:8 decoder is so called because it has 3 select inputs and 8 outputs, of which **only one** can be active at any one time.

Also called a 1-of-8 decoder, meaning that, only one of the eight outputs is active at a time.

The function table illustrates:

Select Inputs			Outputs							
A2	A1	A0	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

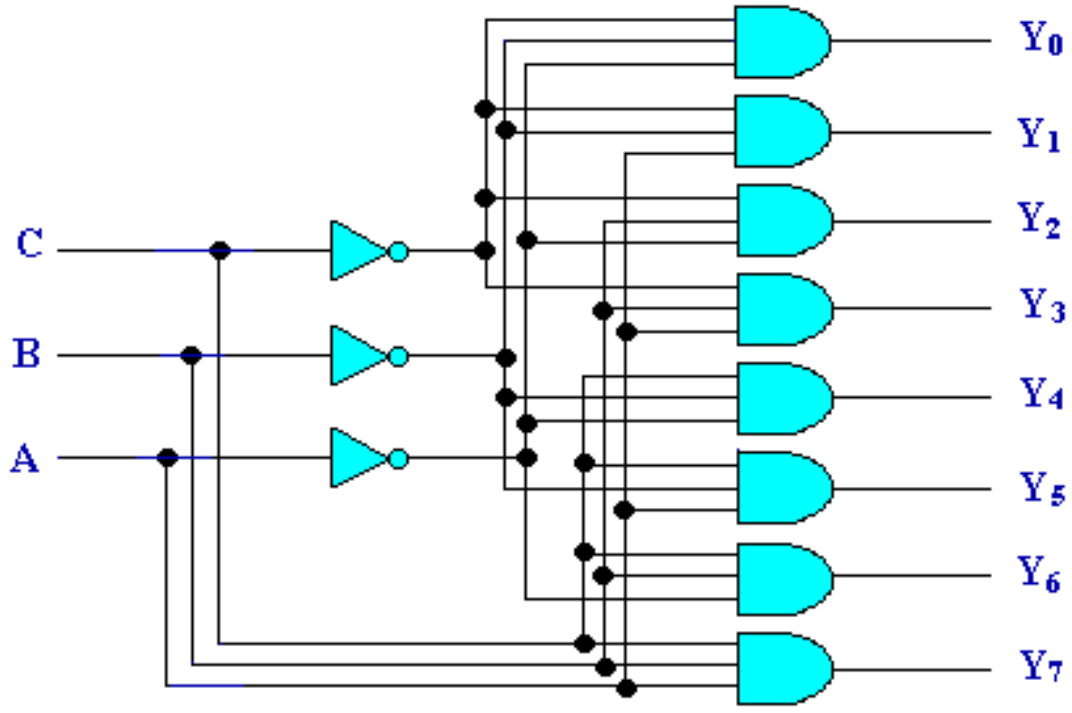
The outputs are designated using a decimal number ranging from 0 to 7, such that the output designated 0 is active (goes H in this case) when the select inputs are 000 (binary for decimal 0), output 1 is active when the select inputs are 001 (binary for decimal 1), output 2 is active when select inputs are 010 (binary for decimal 2), and so on.

From the given truth table, the Boolean expression for each output is:

$$Y_0 = \overline{A}\,\overline{B}\,\overline{C} \qquad Y_1 = A\overline{B}\,\overline{C} \qquad Y_2 = \overline{A}B\overline{C} \qquad Y_3 = AB\overline{C}$$

$$Y_4 = \overline{A}\,\overline{B}C \qquad Y_5 = A\overline{B}C \qquad Y_6 = \overline{A}BC \qquad Y_7 = ABC$$

Hence the logic circuit is a collection of AND gates:



Question: If the outputs of decoder are to be active Low, what circuit modifications are required?

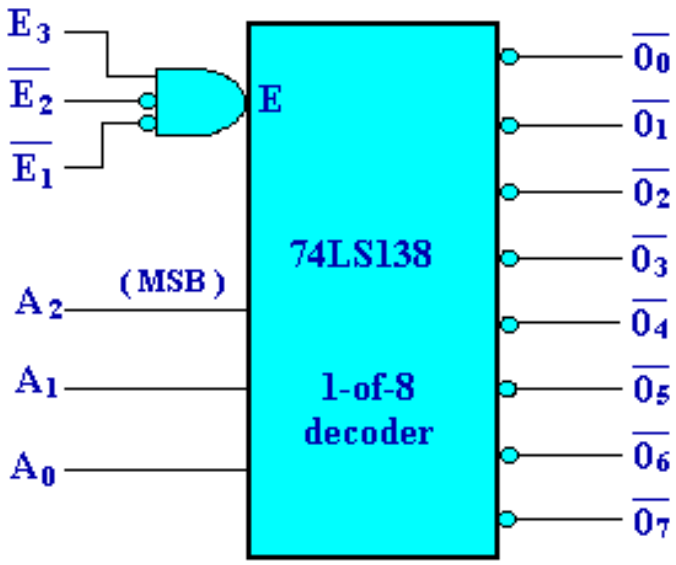
Answer: Replace each AND gate with a NAND gate.

Enable Inputs in the 74LS138

Most practical decoder ICs have one or more ENABLE inputs that are used to control the operation of the decoder.

When the ENABLE inputs are **active**, the decoders function normally, otherwise all outputs will be forced to the **inactive** level.

An Example is: The **74LS138** Decoder, shown below:



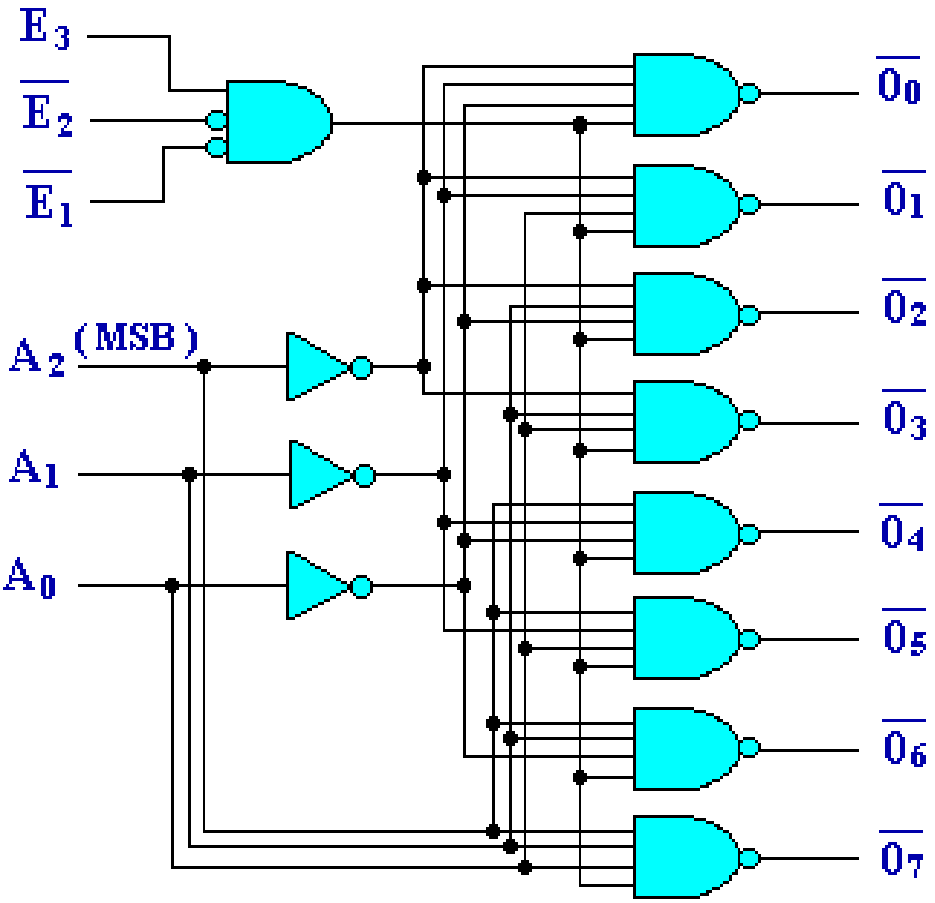
Function Table

E1	E2	E3	Outputs
0	0	1	Respond to i/p code
1	X	X	Disabled - all o/ps H
X	1	X	--------do---------
X	X	0	--------do---------

NB: When an O/P is active, the O/P goes Low, i.e O/P is active Low.

When disabled, i.e. E1, E2, E3 <> 001, all outputs will be at logic High. When enabled however, one of the eight outputs will go Low; the output that goes low being determined by the code applied to the select inputs A2, A1 and A0.

From the function table, the schematic of the 74138 decoder is as shown:



Nand gates are used to provide active Low outputs.

Question: If decoder inputs are: E1=E2=0, E3=1 and, A2,A1,A0 = 101, which output will be selected?

Answer: 101 = 5 and hence, output $\overline{O5}$ goes Low.

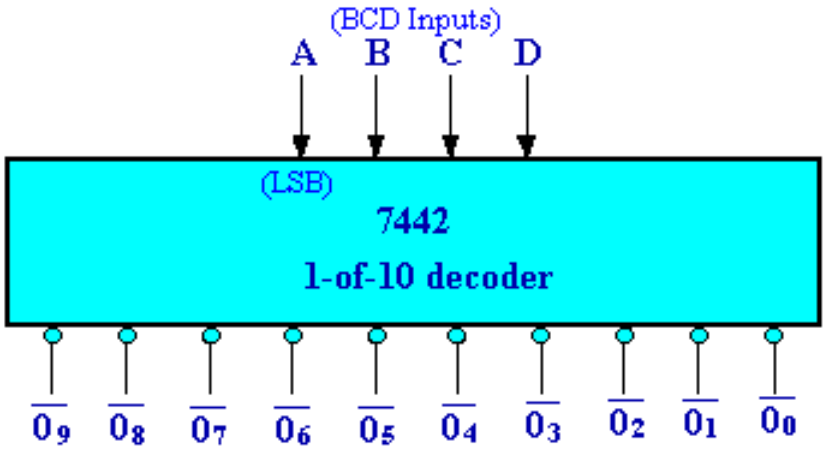
7442 BCD to Decimal Decoder

The 7442 IC is a 1-of-10 decoder.

Also known as a BCD-to-decimal decoder, i.e. it decodes BCD codes.

Application of other than the BCD codes will result in all outputs being High or in active.

For a valid BCD code input, the corresponding output will be activated. E.g., if BCD code 0011 (for decimal 3) is applied to DCBA, then output O3 will be Low.



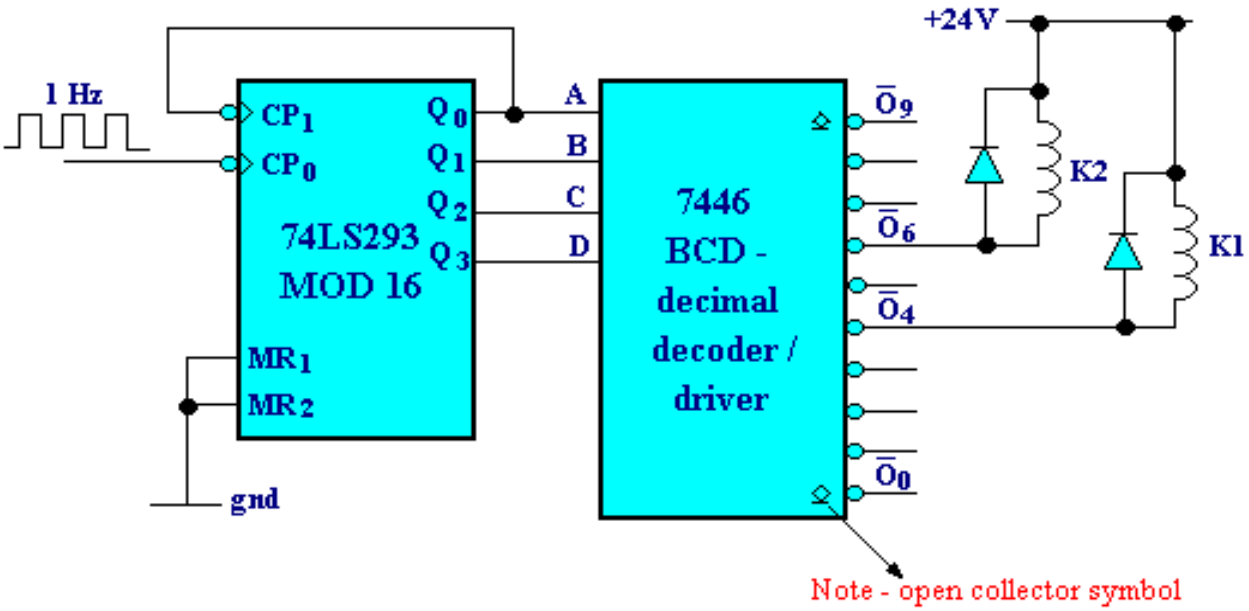
Question: Which output goes Low if the BCD code applied is 1000?

Answer: 1000 binary converts to 8 decimal and hence output O8 goes Low.

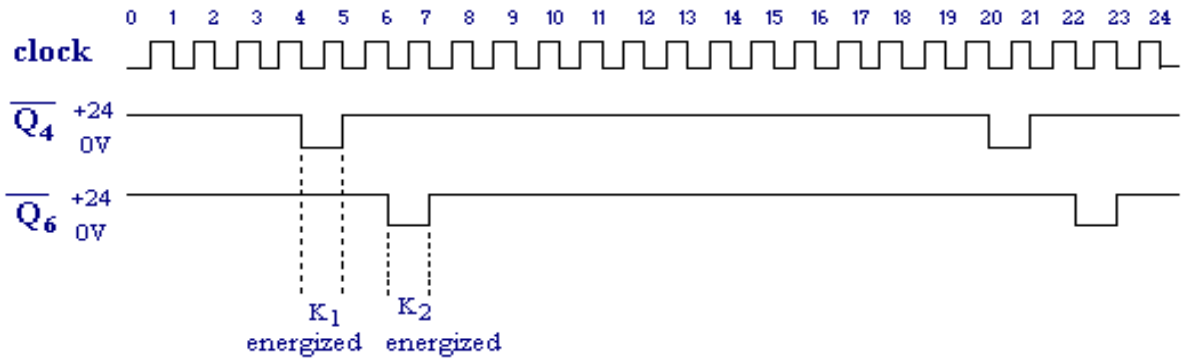
A Decoder Application

Decoders are used whenever an output or group of outputs is to be activated only on the occurrence of a specific combination of input levels.

E.g. Counter/Decoder combination used to provide timing and sequencing operations.



K2 and K1 are relays are energized whenever the Mod-16 counter outputs 0110 and 0100, respectively.

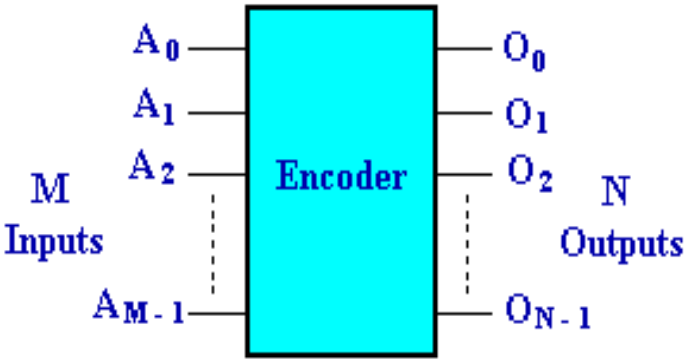


Encoders

An encoder is basically the opposite of a decoder.

A decoder detects or responds to an input code whereas an encoder is used to produce a code.

In general, an encoder has a number of input lines and produces an N-bit output code that is dependent upon which input line being activated.



NB:
Only one input is active at a time
2^N outputs $\leq$ M inputs

General Encoder Symbol

An encoder with N outputs will have up to 2^N inputs.

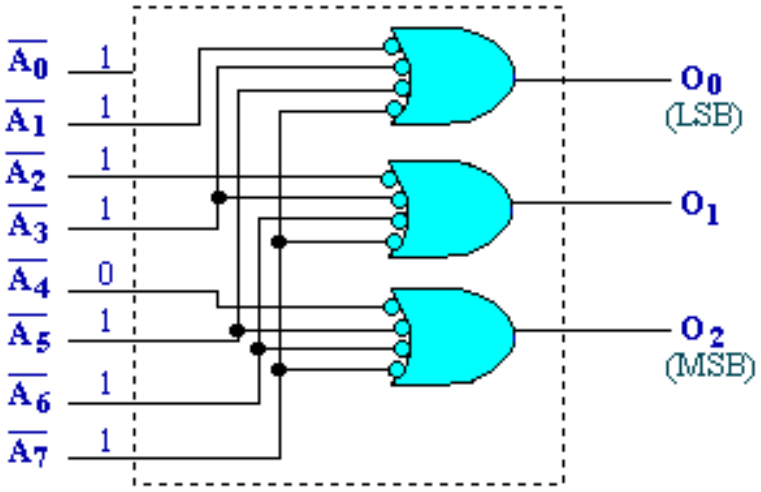
E.g. An Octal encoder, i.e. device that produce Octal codes, and has 3 outputs, will have 2^3 or 8 input lines; one input line for each code to be generated.

Encoders are described based on the number of input to output lines. E.g., an encoder with 10 input lines and 4 output lines would be described as a 10-to-4 lines encoder or 10:4 encoder.

Shown below is a hypothetical 8 lines to 3 lines encoder that is also referred to as an octal-to-binary encoder.

This encoder produces the binary code of an octal number.

E.g., to generate the binary equivalent of octal 4, input A4 must be activated, while all other inputs remain High.



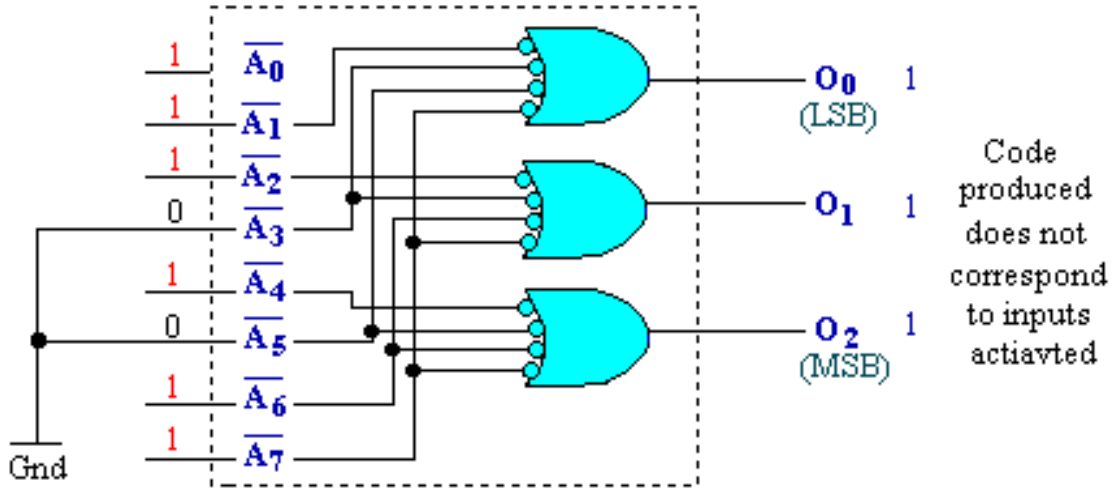
8 : 3 Encoder Example

Question1: What is binary code generated if input A4 is active while all other inputs remain High?

Answer: The code generated would be binary of 4, which is 0100.

In this simple decoder circuit however, consider what happens when inputs A3 and A4 are simultaneously activated.

In this situation, the code produced at the outputs is neither the binary equivalent of octal 3 nor octal 5 but rather the binary equivalent of octal 7! In other words an error has occurred.



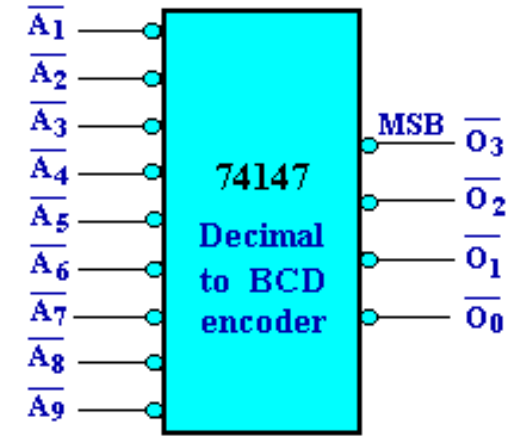
To prevent erroneous code from being produced when multiple inputs are activated at the same time, additional logic circuitry is required. This leads to special encoders that are called *Priority Encoders.*

Priority Encoders

A **priority encoder** includes additional logic to ensure that when two or more inputs are activated simultaneously, the output code will correspond to the highest-numbered input activated.

If the previous example is used, the code produced will correspond to the binary equivalent of octal 5, i.e. 101 since 5 is higher than 3.

An example is the **74147** Decimal-to-BCD Priority Encoder.



$\overline{A_1}$	$\overline{A_2}$	$\overline{A_3}$	$\overline{A_4}$	$\overline{A_5}$	$\overline{A_6}$	$\overline{A_7}$	$\overline{A_8}$	$\overline{A_9}$	$\overline{O_3}$	$\overline{O_2}$	$\overline{O_1}$	$\overline{O_0}$
1	1	1	1	1	1	1	1	1	1	1	1	1
X	X	X	X	X	X	X	X	0	0	1	1	0
X	X	X	X	X	X	X	0	1	0	1	1	1
X	X	X	X	X	X	0	1	1	1	0	0	0
X	X	X	X	X	0	1	1	1	1	0	0	1
X	X	X	X	0	1	1	1	1	1	0	1	0
X	X	X	0	1	1	1	1	1	1	0	1	1
X	X	0	1	1	1	1	1	1	1	1	0	0
X	0	1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	1	0

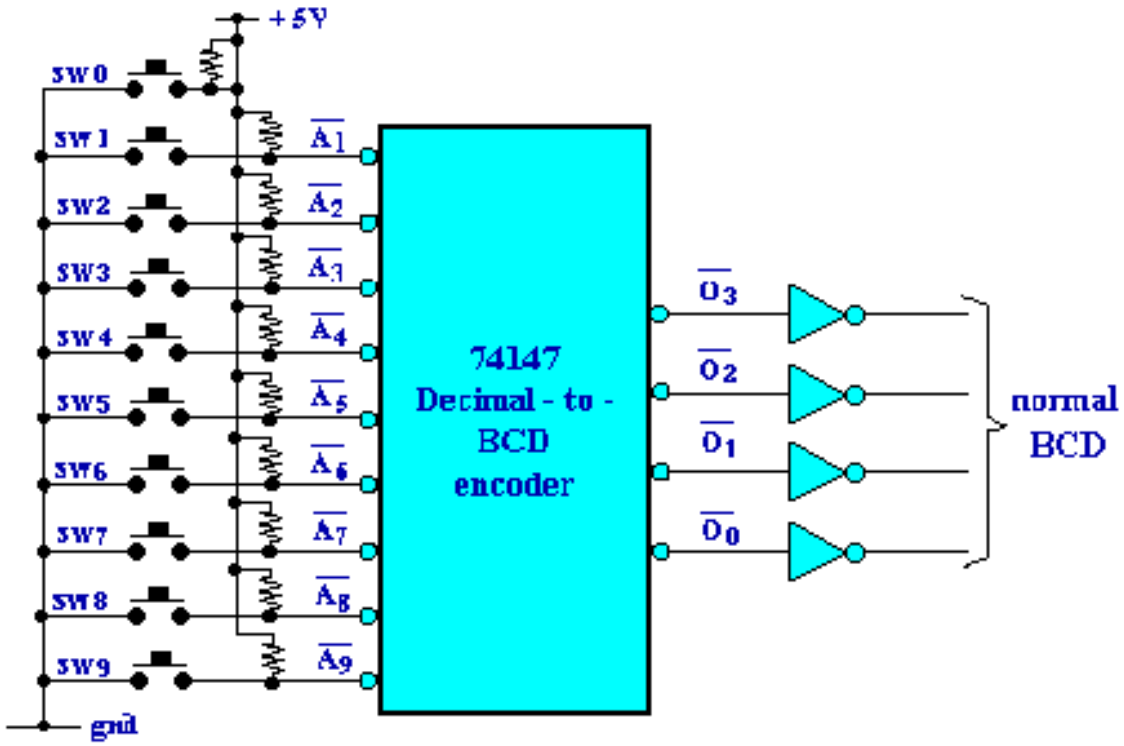
Function Table

It has 10 input lines & 4 output lines and produce an *inverted BCD code* that corresponds to the input line activated.

E.g. If inputs activated are A3, A5 and A8, the code generated will be due to input A8 since it has the higher priority (8>5>3). As the outputs are active low, the code is the inverse of 1000 or **0111**.

Question: What circuits are required if the true value of binary code is to be generated at the outputs of the 74147 priority encoder?

Answer: Inverters must be connected to each of the 4 outputs (see diagram).



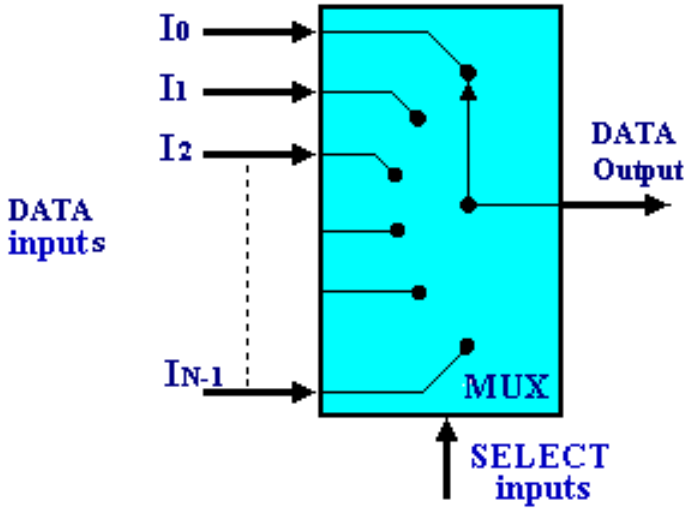
By default, each push-button switch at the inputs generates logic High if not activated.

When a switch is 'pressed', the corresponding input goes Low and the BCD code for the input is generated. E.g. depressing switch A7 generates 1000, which after inversion by the NOT gates yields 0111 or BCD code for 7 decimal.

Multiplexers - Data Selectors

A **multiplexer** (abbr. Mux) or **data decoder** is a logic circuit that accepts several data inputs and allows only **one** of them to be routed to the output.

The routing of the desired data input to the output is controlled by **SELECT** or **ADDRESS** inputs.



Conceptually, a multiplexer is equivalent to a rotary switch with several inputs but only one output. At any one time, only one input is connected to the output.

The selection of which input is to be connected to the output is determined by the data applied to SELECT inputs.

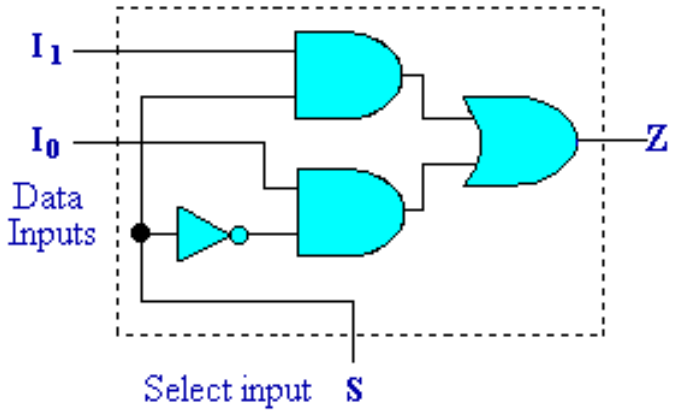
For an N-bit SELECT input, a multiplexer can have up to 2^N data inputs.

Two-Input Multiplexer

The most basic multiplexer is a 2-input to 1-output device.

This has 2 data inputs and one data output. To select one of two data inputs, one Select bit denoted as S, is required.

A mux is basically an inhibit circuit such that at any one time, only *one* of the data inputs is enabled while the other input is inhibited or disabled.



Select S	Output Z
0	Z = I0
1	Z = I1

Function Table of 2-to-1 Multiplexer

In the 2-to-1 mux shown, when S = 0, data input I0 is enabled and data input I1 is inhibited or disabled. Output Z = I0.

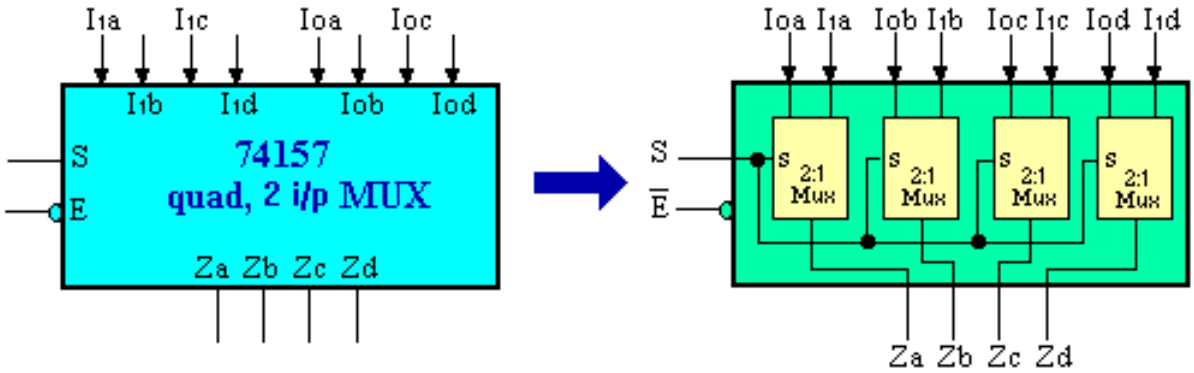
Conversely when S = 1, output Z takes on the logic value of I1.

Quad 2-input Mux

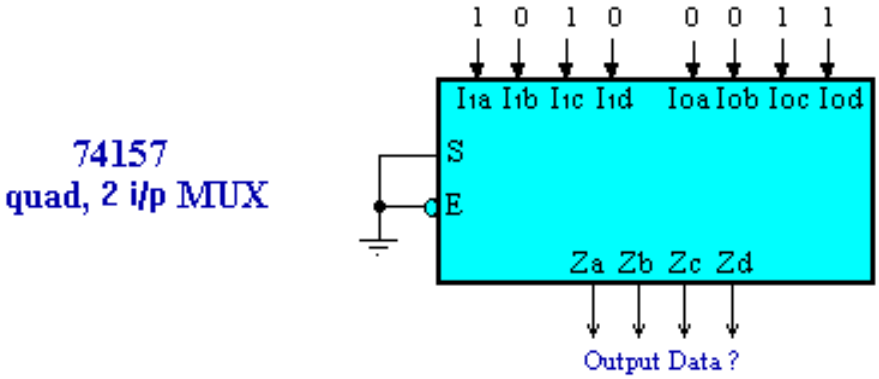
Example: The 74157 Quad 2-input Multiplexer.

The IC consists of four units of 2-to-1 multiplexer that are basically independent except that the Select input S and Enable input E are common.

When enabled, the Z outputs will take on the logic levels present at the corresponding data inputs of each multiplier.



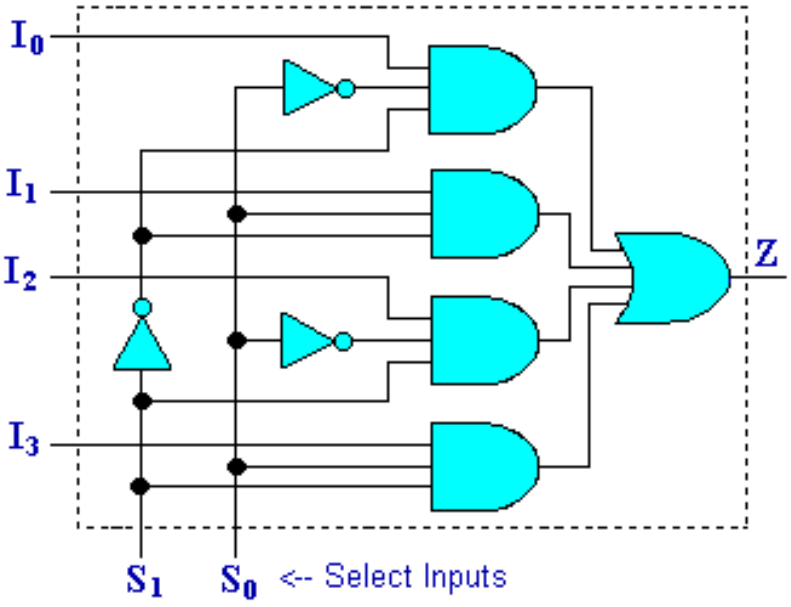
Question: What will be logic levels at Za Zb Zc Zd if the input levels are as shown in the diagram below?



Answer: With S=0, all IO inputs are selected. Thus Za Zb Zc Zd = 0 0 1 1

Four-Input Multiplexer

A 4-input mux has 2 select inputs S1 S0 to determine which of 4 data inputs (I3 I2 I1 I0) is connected to the single output Z.



S1	S0	Output Z
0	0	I0
0	1	I1
1	0	I2
1	1	I3

Function table of a Four-to-One Mux

Question: If inputs I0=L, I1=H, I2=L, I3=H, what will be the logic level present at output Z if S1 and S0 are both logic High?

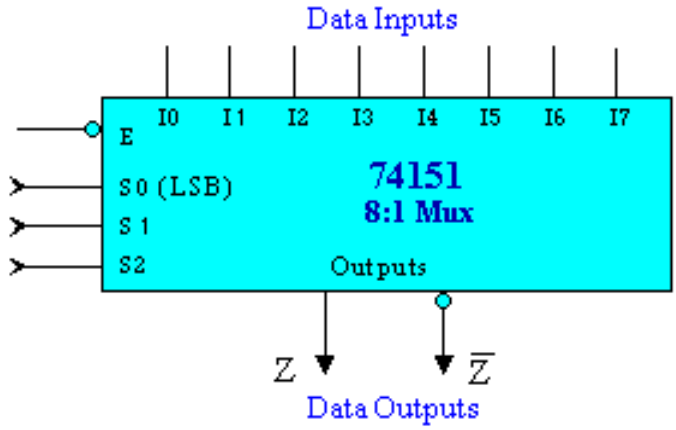
Answer: If S1 and S0 are both High, i.e. 1 1, input I3 is selected and therefore output Z = I3 = H.

Eight-Input Multiplexer – 74151

This is an 8-input multiplexer with an enable input and 2 data outputs (true and inverted).

Three select inputs are required to give the eight combinations needed to select one of the eight data inputs for connection to the outputs.

The enable input E adds flexibility to the device as it allows multiple 74151s to be connected together to form a larger multiplexer.



S2	S1	S0	Z
0	0	0	I0
0	0	1	I1
0	1	0	I2
0	1	1	I3
1	0	0	I4
1	0	1	I5
1	1	0	I6
1	1	1	I7

Table shown with Enable E at Low

Question: If E=1 and S2 S1 S0 = 101, what is the logic levels at output Z? What would be the new level at if E=0 for the same select inputs?

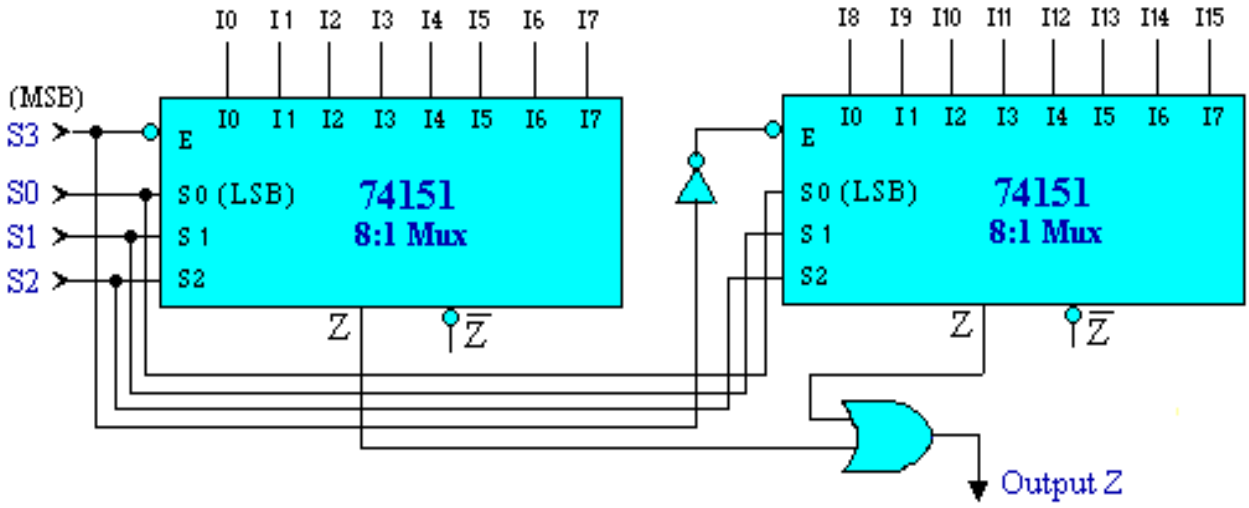
Answer: If E=1, the Mux is disabled and Z=0 regardless of the inputs.
If E=0, then Z=I5, i.e. logic level at I5

In general, the code for selecting a specific input is the binary equivalent of the input variable subscript.

E.g. To select I6, the code required at S2 S1 S0 inputs is the binary of 6_{10} or 110_2.

Question: Given two 74151 ICs and 2 other logic gates, how would you connect the multiplexers and logic gates to form a 16-to-1 multiplexer?

Answer: Shown below.



Two 74151s are required to produce the 16 data inputs. The Enable input is used to generate the fourth Select input S3. An inverter is connected between the two enable lines so that when the left IC is enabled, the right IC would be disabled and vice versa. The two outputs are combined through a 2-input OR gate to produce the single output line required.

Multiplexer Applications

Multiplexer applications are many and varied.

Multiplexers are typically used in data communication circuits for data selection and routing.

Being very flexible devices, multiplexers can also be used for:

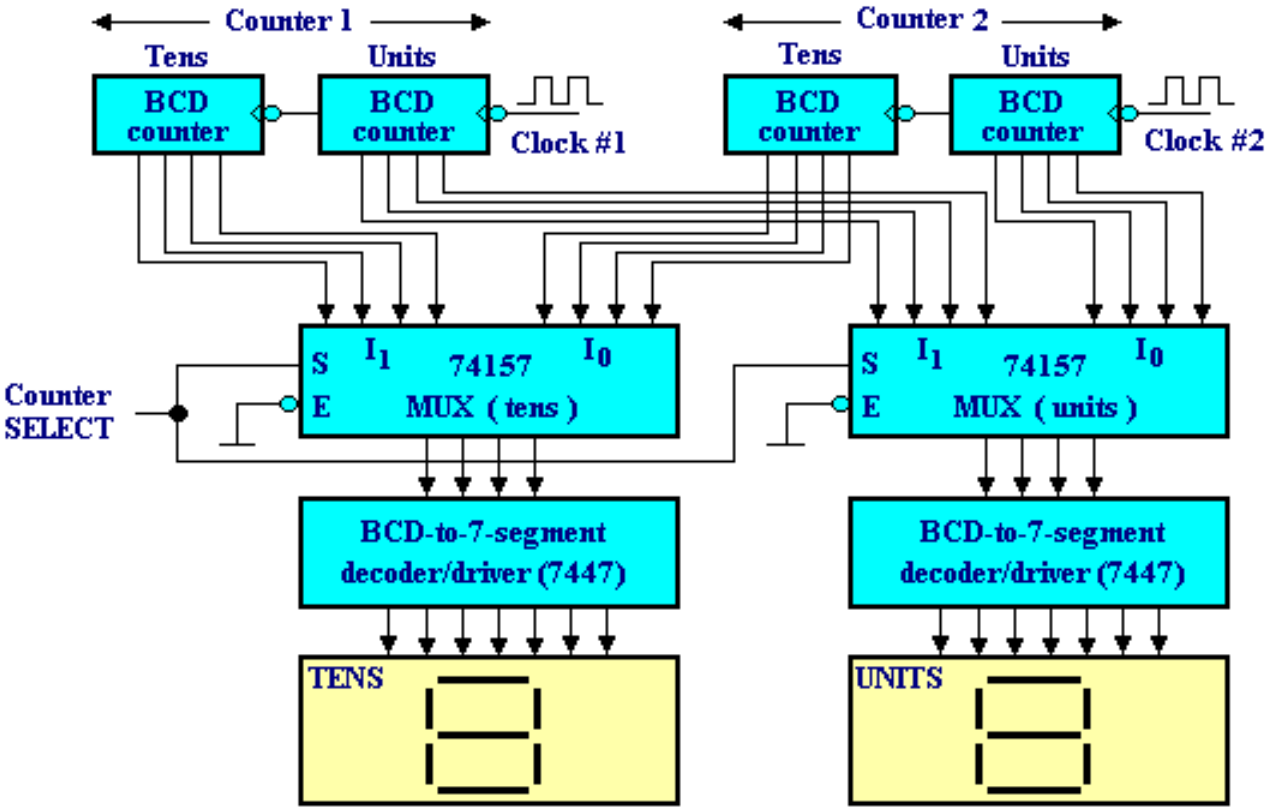
- logic function generation
- waveform generation
- parallel-to-serial conversion
- operation sequencing
- etc.

Some application examples involving multiplexers, in particular, logic function generation, are as follows:

Example 1

Clock Multiplexing:

Using two 74157 multiplexers to select and display the contents of either of two BCD counters.



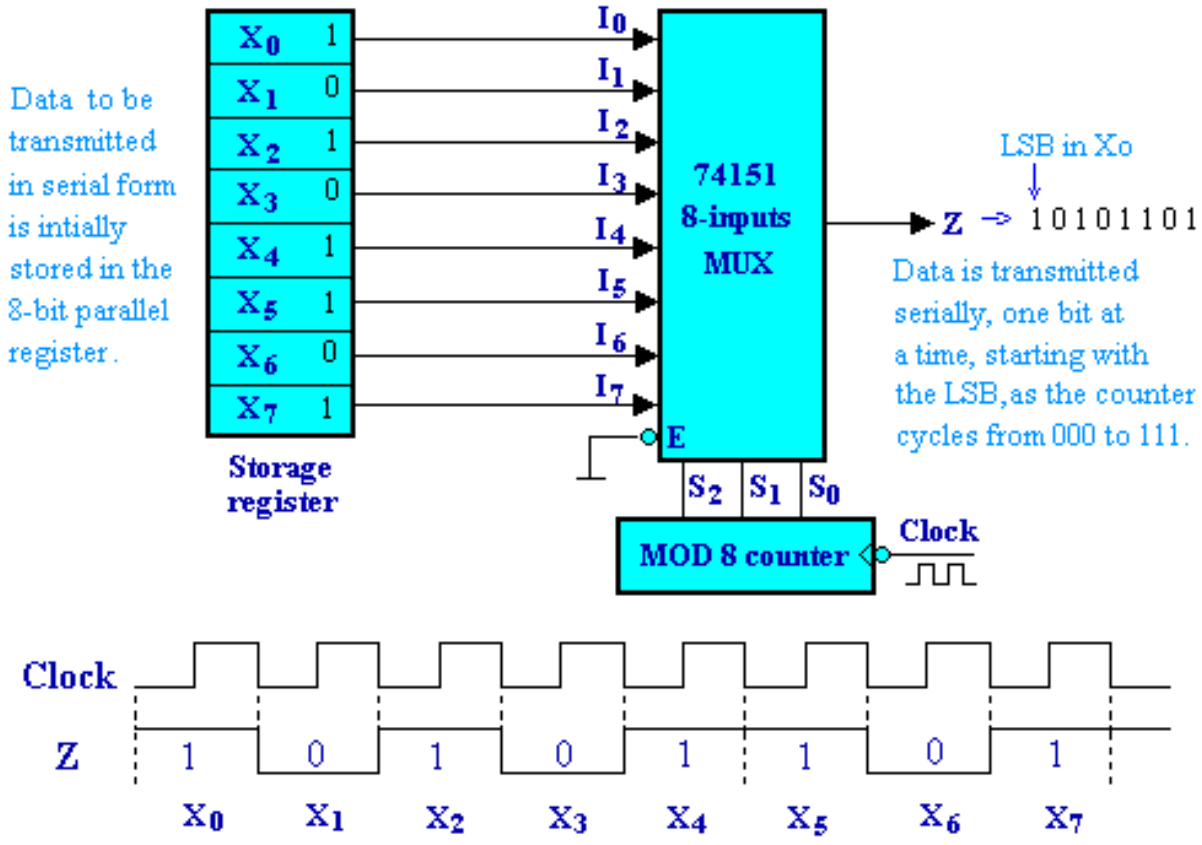
With Counter Select = 0, I0 inputs are selected and hence Counter 2 outputs are displayed in the 7-segment displays

With Counter Select = 1, Counter 1 outputs will be selected instead.

Example 2

Parallel-to-Serial Conversion:

When data is transmitted over relatively long distances, serial transmission is used. Parallel data in a computer need to be converted to the serial form.



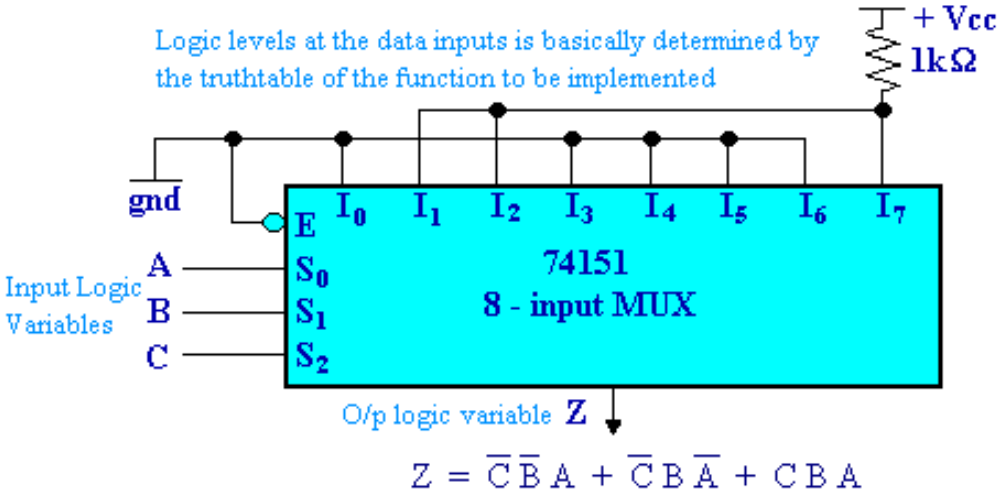
As the mod-8 counter cycles from 000 to 111, each input at the 74151 is selected in turn & hence the date in the storage register is transmitted to the multiplexer output, one bit at a time.

Example 3

Logic Function Generation:

Multiplexers can be used to implement logic functions directly from truth table without the need for simplification.

When used for this purpose, the logic variables are connected to the select inputs and each data input is connected permanently HIGH or LOW as specified in the truth table.



C	B	A	Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

E.g., For CBA = 000, input I0 is selected.
From the truth table, a logic 0 is required at Z. Thus I0 is grounded.

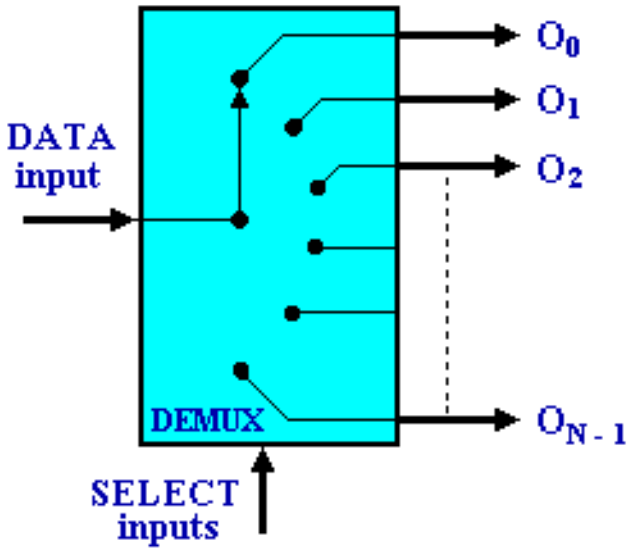
For CBA = 001, input I1 is selected.
From the truth table, a logic 1 is required at Z. Therefore, I1 is connected to + Vcc.

Etc.

Demultiplexers (Data Distributors)

A demultiplexer (abbr. Demux) performs the reverse operation of the multiplexer.

It takes a single input and route it to one of the several outputs.



NB: The Demultiplexer (DeMux) is basically the reverse of a Multiplexer. Conceptually, it is equivalent to a rotary switch with one input and several outputs. At any one time, the data input is connected to one of the outputs. Which output is connected to the data input is determined by the cobination applied at the Select inputs.

For an N-bit SELECT input, a demultiplexer can route a single data input to one of 2^N data outputs.

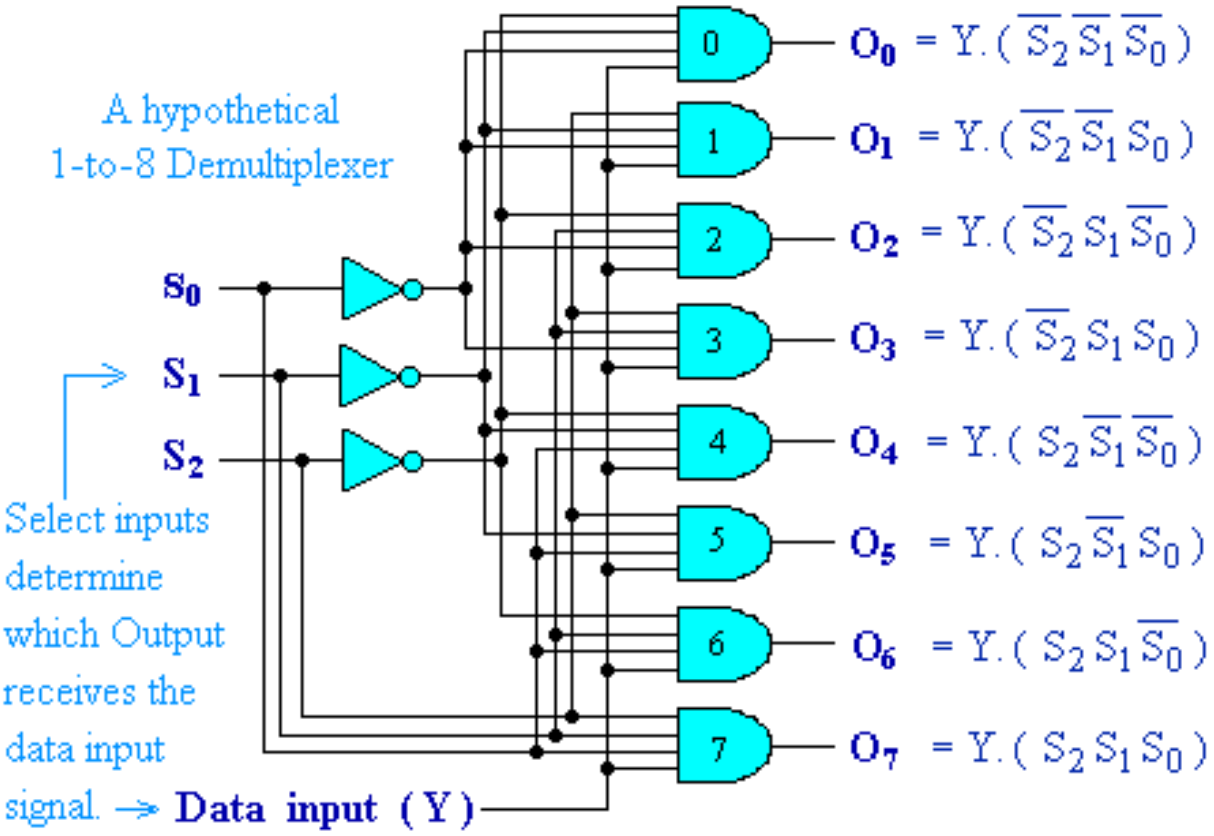
E.g., a 4-bit select Demux can have up to 2^4 or 16 outputs.

A 6-bit select Demux will have up to 2^6 or 64 outputs.

1-to-8 demultiplexer

A simple 1-to-8 Demultiplexer (DeMux) is shown below.

Such a DeMux would have 3 Select inputs to determine which one of the 8 outputs is connected to the single data input.

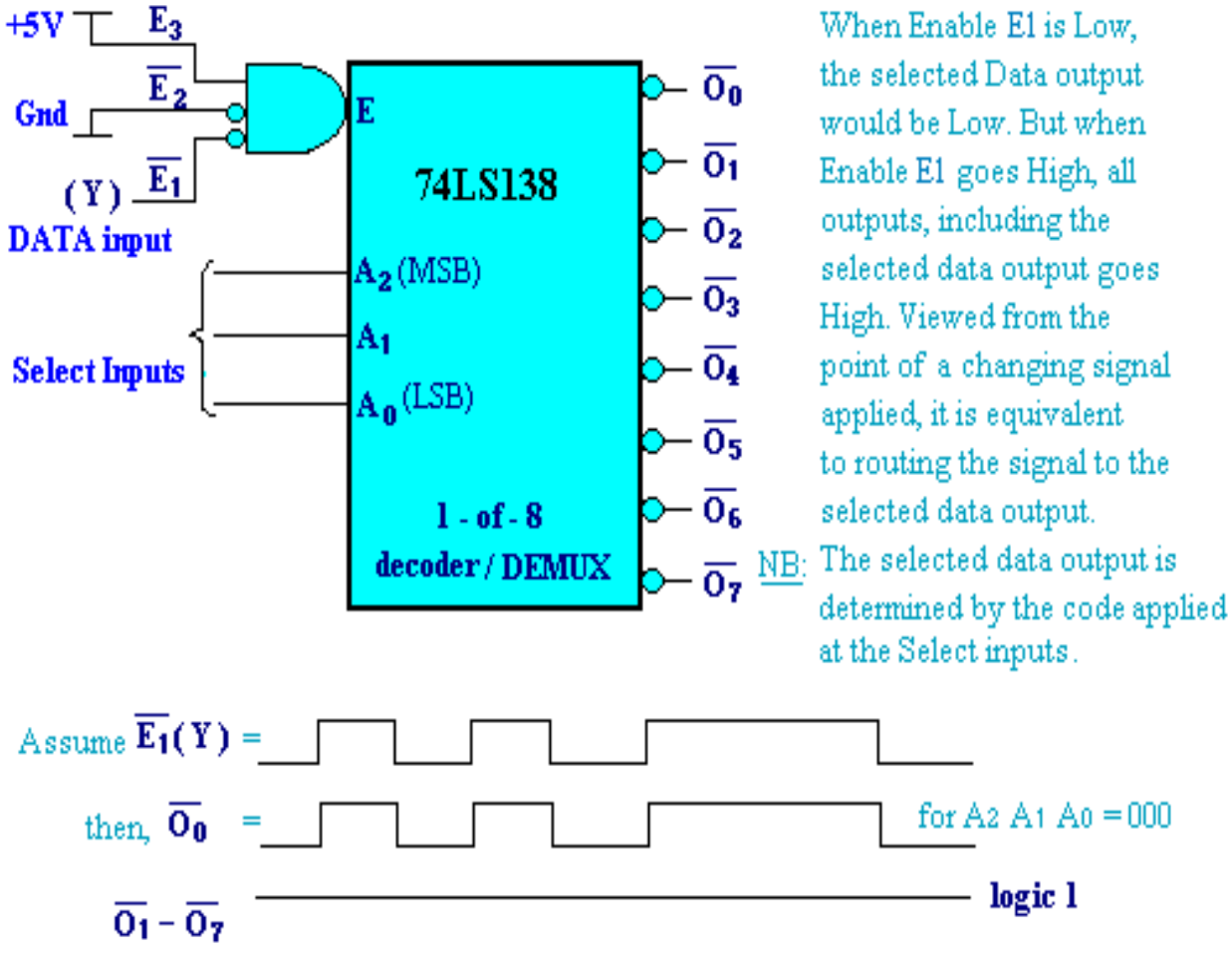


E.g., when S2 S1 S0 = 0 1 1, only AND gate 3 is enabled, while all other AND gates are disabled. Hence data input Y is routed to output 03; all other outputs remain 0.

Using A Decoder as Demultiplexer

A decoder with an **ENABLE** input is aptly suited to be used as a demultiplexer. In this instance, the ENABLE input is used as the single data input of the demultiplexer.

An Example: Using 74LS138 decoder as a Demux



Unused enable inputs E3 and E2 are tied to the correct logic levels to enable the AND gate.