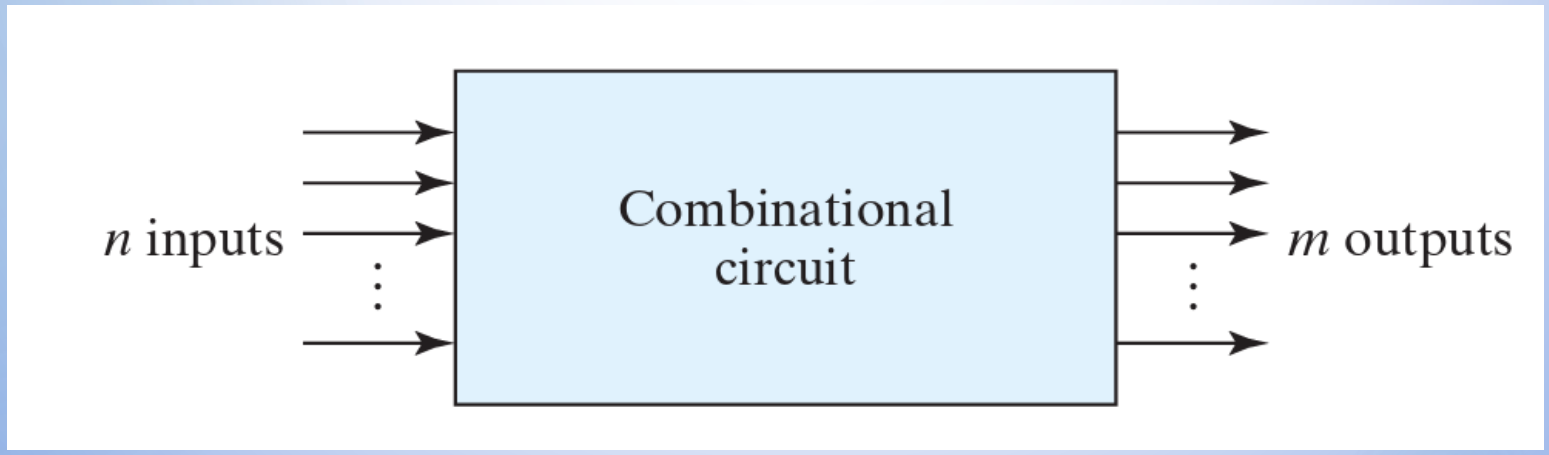
CSE 2105
Digital Logic Design

COMBINATIONAL LOGIC CIRCUIT

Combinational Logic circuit

- A combinational circuit consists of an interconnection of logic gates. The output signal of a combinational logic circuit depends on the present values of the signals at their inputs.
- A block diagram of a combinational circuit is shown in the figure. The *n* input binary variables come from an external source; the *m* output variables are produced by the internal combinational logic circuit and go to an external destination. Each input and output variable exists physically as an analog signal whose values are interpreted to be a binary signal that represents logic 1 and logic 0.



Designing a Combinational Logic circuit

The design of a combinational logic circuit involves the following steps:

1. First, understand the problem.
2. From the specifications of the circuit, determine the required number of inputs and outputs and assign a symbol to each.
3. Derive the truth table that defines the required relationship between inputs and outputs.
4. Obtain the simplified Boolean functions for each output as a function of the input
5. variables.
6. Draw the logic diagram and verify the correctness of the design (manually or by simulation).

Encoder

- An encoder is a digital circuit has 2^n (or fewer) input lines and n output lines. The output lines generate the binary code corresponding to the input value. An example of an encoder is the octal-to-binary encoder.
- Among the input lines of the encoder, only one is activated at a given time, and it produces an N-bit output code, depending on which input is activated.

Truth table of Octal to Binary Encoder

Input								Output		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	X	Y	Z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

Octal to Binary Encoder

Boolean output functions:

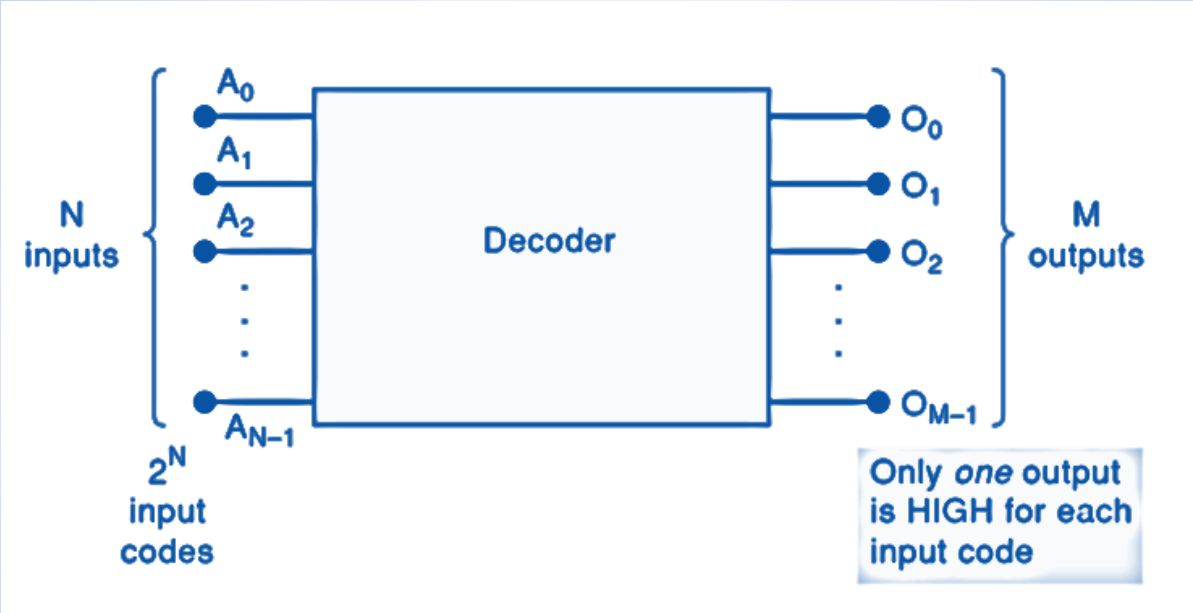
- $z = D_1 + D_3 + D_5 + D_7$
- $y = D_2 + D_3 + D_6 + D_7$
- $x = D_4 + D_5 + D_6 + D_7$

The encoder can be implemented with three OR gates.

Decoder

- A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines.
- In other words, a decoder circuit looks at its inputs, determines which binary number is present there, and activates the one output that corresponds to that number; all other outputs remain inactive.
- The diagram for a general decoder is shown with N inputs and M outputs. Because each of the N inputs can be 0 or 1, there are 2^N possible input combinations or codes. For each of these input combinations, only one of the M outputs will be active (HIGH); all the other outputs are LOW.

Decoder



Some decoders do not utilize all of the 2^N possible input codes but only certain ones. For example, a BCD-to-decimal decoder has a four-bit input code and ten output lines that correspond to the ten BCD code groups 0000 through 1001. Decoders of this type are often designed so that if any of the unused codes are applied to the input, none of the outputs will be activated.

Applications of Decoder

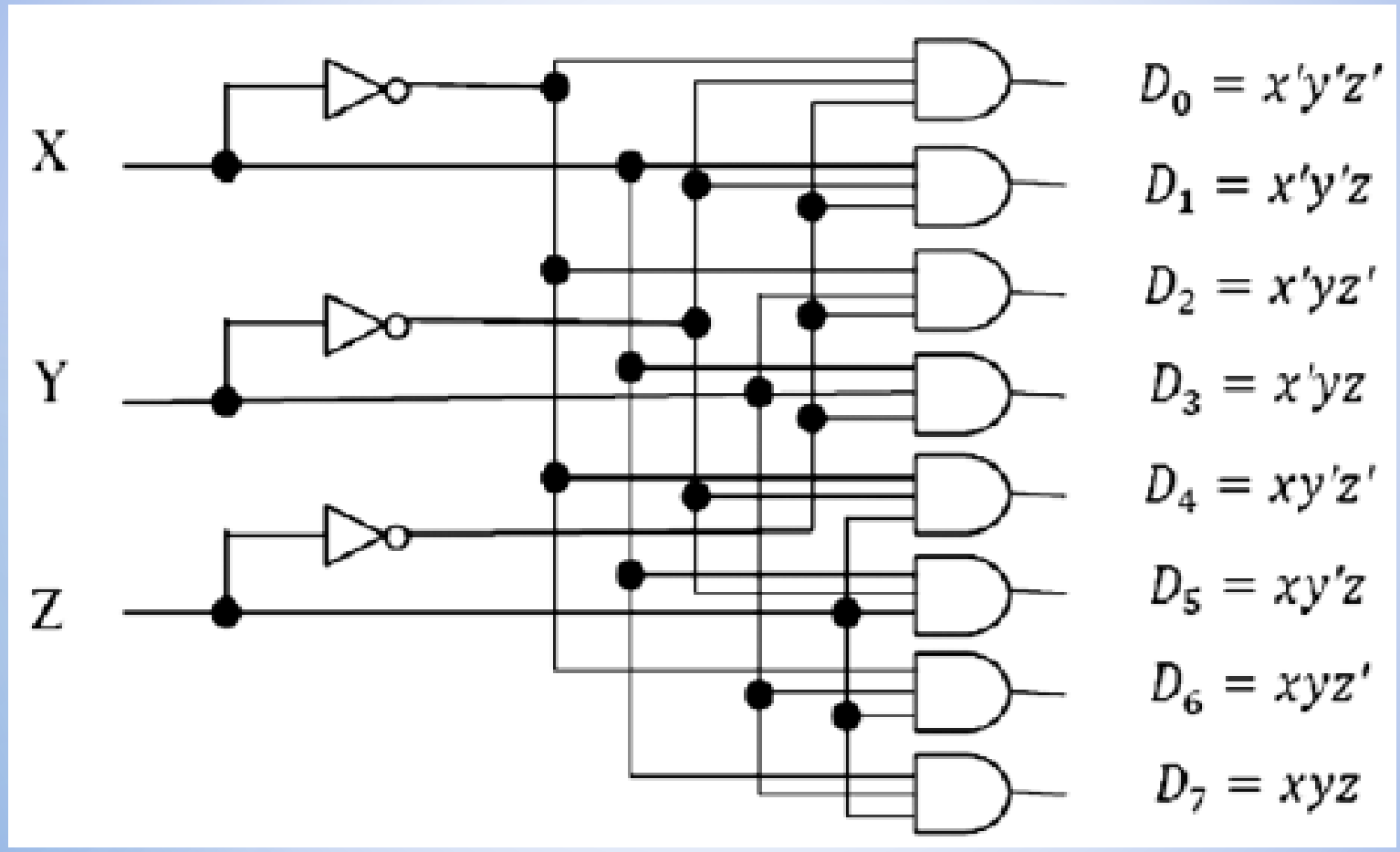
- Decoders are used for code conversions.
- Decoders are extensively used in memory systems of computers.
- Decoders are also used in data routing applications where very short propagation delay is required.
- Decoder may also be used for timing or sequencing purposes.
- Decoders are also utilized to turn on and off digital devices at a specific time.

Binary to Octal Decoder

Truth Table of a Three-to-Eight-Line Decoder

Inputs			Outputs							
x	y	z	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
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

Logic Diagram of Decoder



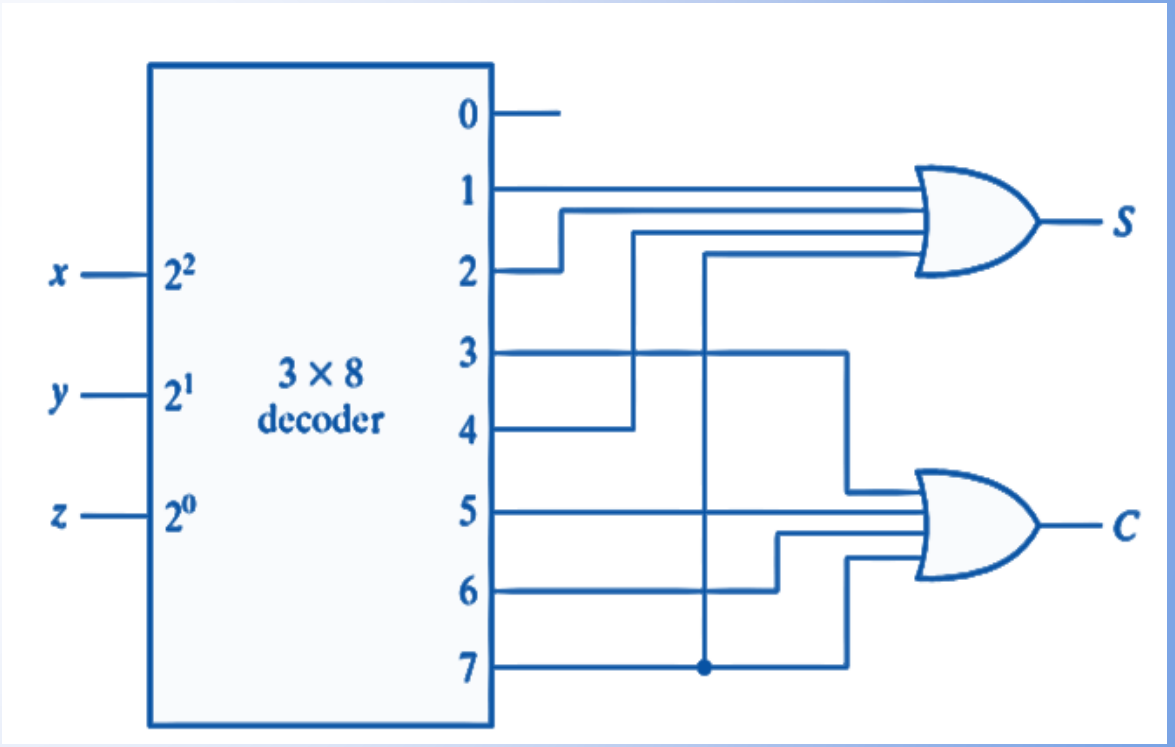
Combinational Logic Implementation

- A decoder provides the 2^n minterms of n input variables.
- Each asserted output of the decoder is associated with a unique pattern of input bits. Since any Boolean function can be expressed in sum-of-minterms form, a decoder that generates the minterms of the function, together with an external OR gate that forms their logical sum, provides a hardware implementation of the function.
- In this way, any combinational circuit with n inputs and m outputs can be implemented with an n -to-2^n -line decoder and m OR gates.
- A decoder is then chosen such that it generates all the minterms of the input variables. The inputs to each OR gate are selected from the decoder outputs according to the list of minterms of each function.

Combinational Logic Implementation

- From truth table of the full adder, we obtain the functions for the combinational circuit in sum-of-minterms form:
- S(x, y, z) = (1, 2, 4, 7)
- C(x, y, z) = (3, 5, 6, 7)

Since there are three inputs and a total of eight minterms, we need a three-to-eight-line decoder.
The implementation is shown in the figure
Decoder generates the eight minterms for x, y, and z.
The OR gate for output S forms the logical sum of minterms 1, 2, 4, and 7.
The OR gate for output C forms the logical sum of minterms 3, 5, 6, and 7.

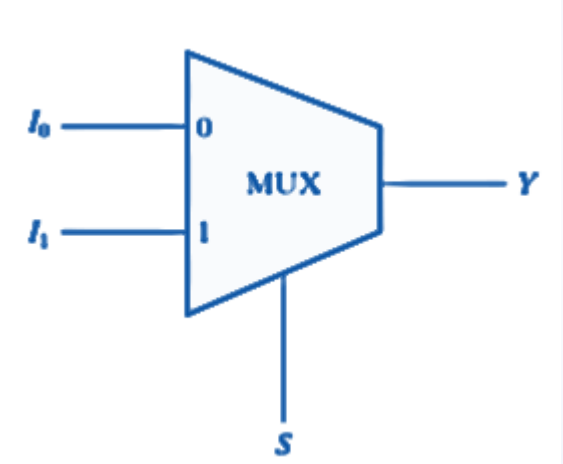


Multiplexer

- A 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 2n input lines and n selection lines whose bit combinations determine which input is selected.
- A two-to-one-line multiplexer connects one of two 1-bit sources to a common destination, as shown in the figure. The circuit has two data input lines, one output line, and one selection line S . When S = 0, the upper AND gate is enabled and I_0 has a path to the output. When S = 1,the lower AND gate is enabled and I_1 has a path to the output. The multiplexer acts like an electronic switch that selects one of two sources.

Multiplexer

- The block diagram of a multiplexer is sometimes depicted by a wedge-shaped symbol, as shown in the figure. It suggests visually how a selected one of multiple data sources is directed into a single destination. The multiplexer is often labeled “MUX” in block diagrams.



- For example, a modern home stereo system may have a switch that selects music from one of four sources: a cassette tape, a compact disc (CD), a radio tuner, or an auxiliary input such as audio from a VCR or DVD. The switch selects one of the electronic signals from one of these four sources and sends it to the power amplifier and speakers. In simple terms, this is what a multiplexer (MUX) does: it selects one of several input signals and passes it on to the output.

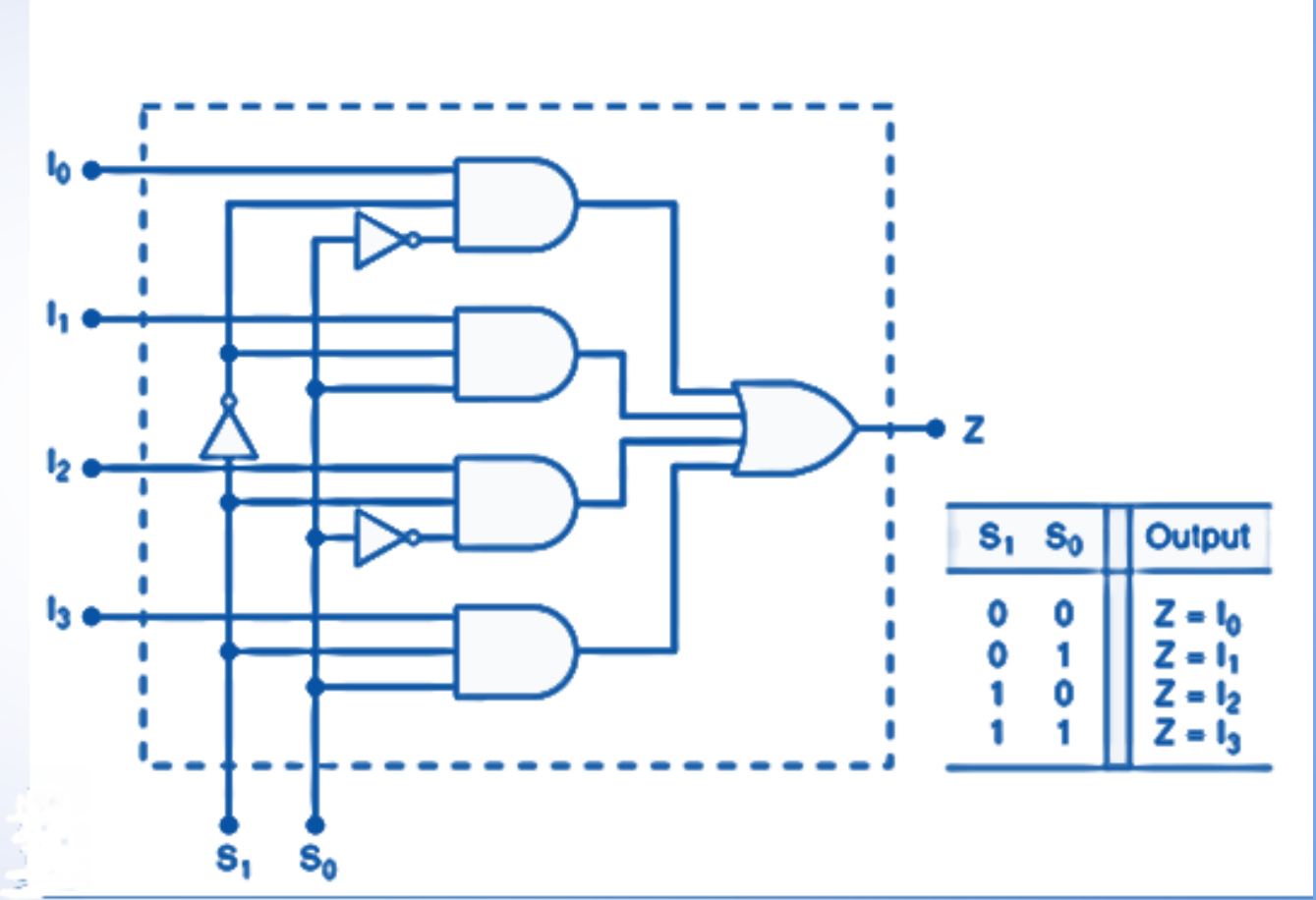
Four-Input Multiplexer

- Here, four inputs are selectively transmitted

to the output according to the four possible

combinations of the S_1, S_0 select inputs. Each data

input is gated with a different combination of select input levels. I_0 is gated with S_1 and S_0 so that I_0 will pass through its AND gate to output Z only when $S_1 = 0$ and $S_0 = 0$. The table in the figure gives the outputs for the other three input-select codes.



S_1	S_0	Output
0	0	$Z = I_0$
0	1	$Z = I_1$
1	0	$Z = I_2$
1	1	$Z = I_3$

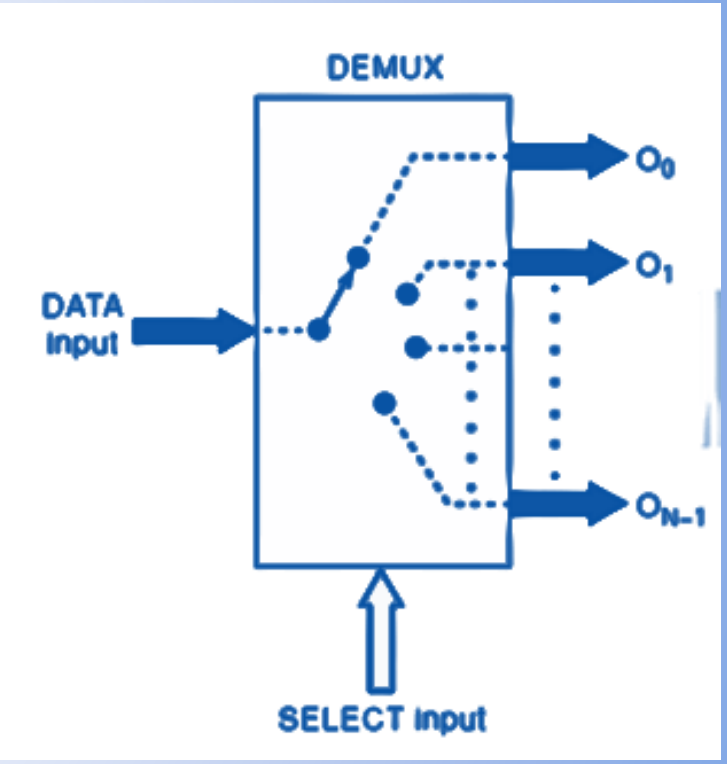
Application of Multiplexer

Multiplexer circuits find numerous and varied applications in digital systems of all types. These applications include-

- data selection,
- data routing,
- operation sequencing,
- parallel-to-serial conversion,
- waveform generation and
- logic-function generation.

Demultiplexer

- A multiplexer takes several inputs and transmits one of them to the output. A demultiplexer (DEMUX) performs the reverse operation: it takes a single input and distributes it over several outputs.
- Figure shows the functional diagram for a digital demultiplexer.
- The large arrows for inputs and outputs can represent one or more lines. The select input code determines to which output the DATA input will be transmitted. In other words, the demultiplexer takes one input data source and selectively distributes it to 1 of N output channels just like a multiposition switch.

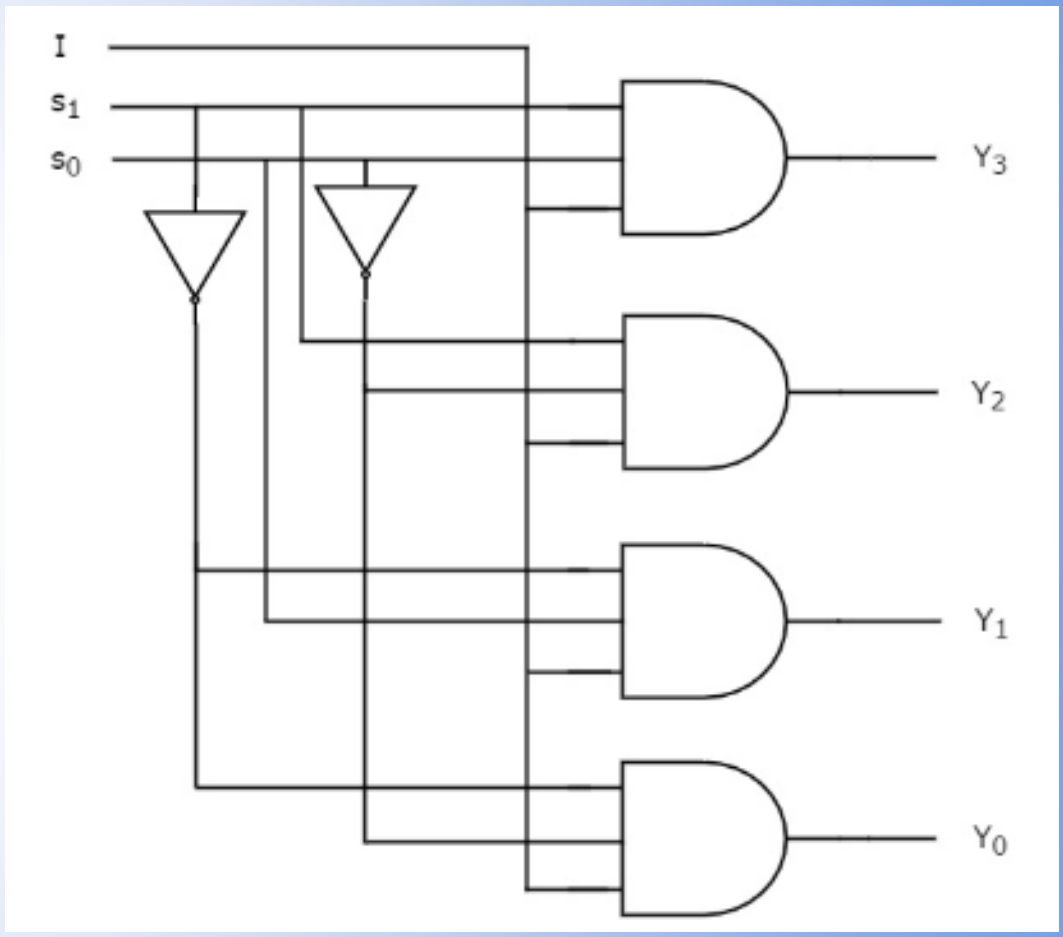


1 to 4 line Demultiplexer

Selection Input		Outputs			
S_1	S_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	**I**
0	1	0	0	**I**	0
1	0	0	**I**	0	0
1	1	**I**	0	0	0

From the above Truth table, we can directly write the Boolean functions for each output as

- $Y_0 = S_1'S_0'I$
- $Y_1 = S_1'S_0I$
- $Y_2 = S_1S_0'I$
- $Y_3 = S_1S_0I$



Boolean Function Using Demultiplexer

- First, find the number of input variables. If it's 3 variable functions, then we need a 1:8 multiplexer. If it's an n variable function, we require a $1:2^n$ multiplexer.
- Put the variables as selector lines of the multiplexer. A $1:2^n$ multiplexer will have n selector lines.
- Now, from the truth table of the function, find the minterms and grab the corresponding output lines of the demultiplexer, and put them into an OR gate. This makes sure that whenever any minterm of the function is high, the output is high.

Full Adder Using Demultiplexer

- Full Adder is a combinatorial circuit that computes the sum and carries out two input bits and an input carry. So it has three inputs – the two bits A and B, and the input carry C_{in}, and two outputs – sum, S and output carry, C_{out}.
- Truth table of full adder is given below:

Input			**Outputs**	
A	B	Carry in, C_{in}	Sum, S	Carry out, C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Full Adder Using Demultiplexer

From the truth table, we can write

- $S(A, B, C_{in}) = \sum(1,2,4,7)$
- $C_{out}(A, B, C_{in}) = \sum(3,5,6,7)$

which is,

- $S(A, B, C_{in}) = (A'B' C_{in}) + (A'B C_{in}') + (AB' C_{in}') + (AB C_{in})$
- $C_{out}(A, B, C_{in}) = (A'B C_{in}) + (AB' C_{in}) + (AB C_{in}') + (AB C_{in})$

Full Adder Using Demultiplexer

- We have two outputs and therefore two functions S and C_{out}. Clearly, we need to use a 1:8 demultiplexer.
- Firstly, we have to set the input line = 1.
- Using the above steps, we see that for S: we need to put line numbers 1, 2, 4, and 7 of the demultiplexer to an OR gate. For the C-out: we have an OR gate, the lines 3, 5, 6, and 7.