Combinational Circuits

➢ **Definition**

A **Combinational Circuit** is a digital circuit where the **output depends only on the present inputs** and **not on past inputs (no memory)**.

It is built using **logic gates** (AND, OR, NOT, NAND, NOR, XOR, XNOR).

➢ **Characteristics**

1. Output depends **only on present inputs**.
2. No storage elements (no memory).
3. Faster operation compared to sequential circuits (no clock needed).
4. Designed using **Boolean algebra** or **truth tables**.
5. Can be represented using **K-maps** for simplification

Half Adder

Half Adder is a combinational logic circuit that is designed by connecting one EX-OR gate and one AND gate. The half-adder circuit has two inputs: A and B, which add two input digits and generate a carry and a sum.

The output obtained from the EX-OR gate is the sum of the two numbers while that obtained by AND gate is the carry. There will be no forwarding of carry addition because there is no logic gate to process that. Thus, this is called the Half Adder circuit.

Logical Expression of Half Adder

The Logical Expression for half added is given as

$Sum = A \oplus B$

$Carry = A.B$

Truth Table of Half Adder

The Truth Table for Half Added is Given as

Truth Table			
Input		Output	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full Adder

Full Adder is the circuit that consists of two EX-OR gates, two AND gates, and one OR gate. Full Adder is the adder that adds three inputs and produces two outputs which consist of two EX-OR gates, two AND gates, and one OR gate. The first two inputs are A and B and the third input is an input carry as C-IN. The output carry is designated as C-OUT and the normal output is designated as S which is SUM.
The equation obtained by the EX-OR gate is the sum of the binary digits. While the output obtained by AND gate is the carry obtained by addition.

Logical Expression of Full Adder

Given Below is the Logical Expression of Full Adder

$SUM = (A \oplus B) \oplus Cin$

$CARRY\text{-}OUT = A.B + Cin\ (A \oplus B)$

Truth Table of Full Adder

Given Below is the truth Table of Full Adder

Input			Output	
A	B	Cin	Sum	Carry
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

Sum and Carry Operation

- **Sum (S)** : It results from the XOR gate, which is a logic gate that adds two or more bits together in the same way that you add in base 2 with no acknowledgement of carry from the previous bit.
- **Carry (C or Cout):** It is the output of the AND operation in the case of the Half Adder or both AND and OR Operations in the case of the Full Adder to indicate that a '1' has to be carried over to the next bit position.

Advantages and Disadvantages

Advantages of Half Adder

- Flexible and easy when it comes to design.
- Involves the use of fewer logic gates thus, is cheaper.

Disadvantages of Half Adder

- Fails to process a carry input from the previously added numbers.
- Restricted to the addition of only two bits.

Advantages of Full Adder

- Can add 3 bits, it includes one carry input and a carry output, which can perform more elaborate computations.
- It can be cascaded to produce adders for a number of bit additions which makes it suitable for multi bit arithmetic.

Disadvantages of Full Adder

- Complex and needs more gates, hence making the design more complicate and expensive.

- Yeah man, slightly slower because normally 2 gate process are used instead of 1.

Applications

Applications of Half Adder

- Arithmetic operations like addition, subtraction, and multiplication in low level dynamic circuits.
- Three types of rectifiers: half-wave, full-wave, and full-wave with a center tapped secondary. Used in small integration circuits.

Applications of Full Adder

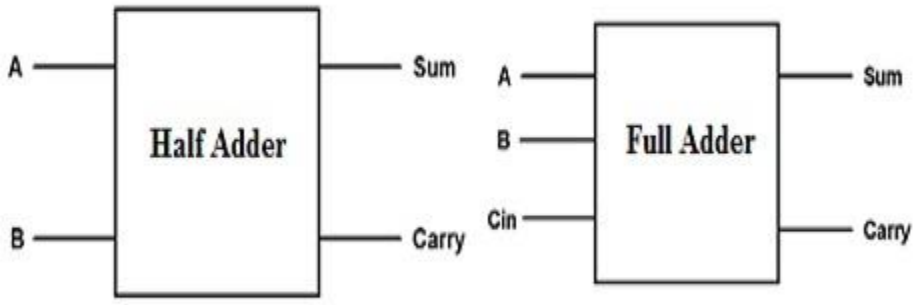
- Carry-look ahead adders in digital processors that utilize multi-bit binary addition.
- Present in the arithmetic logic units (ALU) and other complicated digital systems.

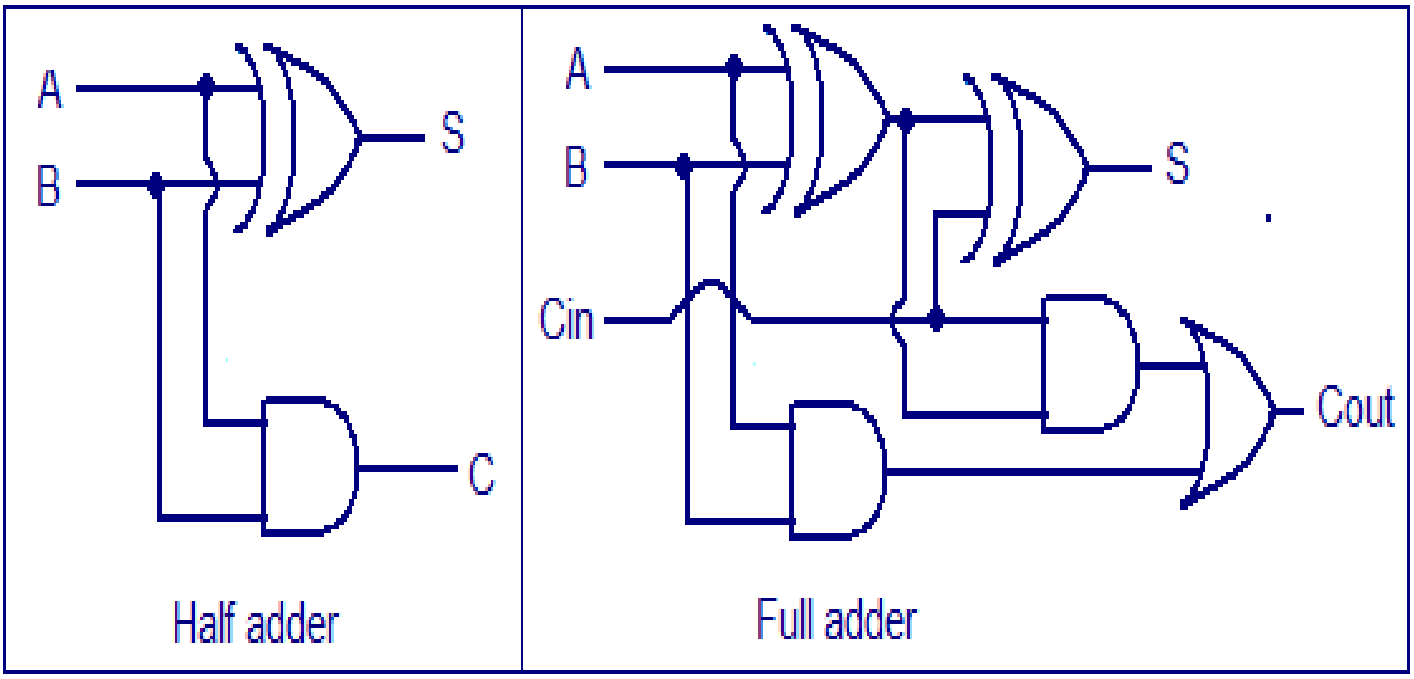
Difference between Half Adder and Full Adder

Given Below is the Difference between the Half adder and Full adder

Parameters	Half Adder	Full Adder
Description	Half Adder is a combinational logic circuit that adds two 1-bit digits. The half adder produces a sum of the two inputs.	A full adder is a combinational logic circuit that performs an addition operation on three one-bit binary numbers. The full adder produces a sum of the three inputs and carry value.
Previous carry	The previous carry is not used.	The previous carry is used.
Inputs	In Half adder, there are two input bits (A, B).	In full adder, there are three input bits (A, B, C-in).
Outputs	The generated output is of two bits-Sum and Carry from the input of 2 bits.	The generated output is of two bits-Sum and Carry from the input of 3 bits.
Used as	A half adder circuit cannot be used in the same way as a full adder circuit.	A full adder circuit can be used in place of a half adder circuit.

Parameters	Half Adder	Full Adder
Feature	It is simple and easy to implement	The design of a full adder is not as simple as a half adder.
Logical Expression	Logical Expression for half adder is : S=a⊕b ; C=a*b.	Logical Expression for Full adder is : S=a⊕b⊕Cin; Cout=(a*b)+(Cin*(a⊕b)).
Logic gates	It consists of one EX-OR gate and one AND gate.	It consists of two EX-OR, two AND gates, and one OR gate.
Applications	It is used in Calculators, computers, digital measuring devices, etc.	It is used in Multiple bit addition, digital processors, etc.
Alternate name	There is no alternate name for half adder.	Full adder is also known as ripple-carry adder.

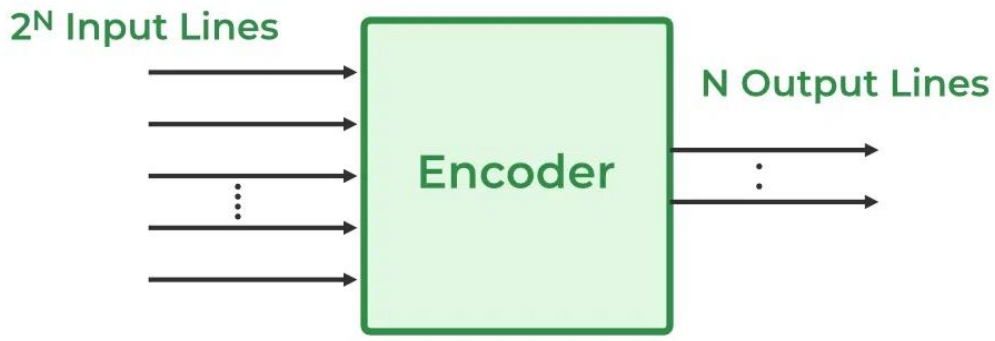




Encoders and Decoders

1. Encoder

- **Definition**:
  An **encoder** is a combinational logic circuit that converts **2^n input lines into n output lines**.
  - It performs the reverse operation of a decoder.
  - It reduces multiple input lines into fewer output lines by **encoding** the input signal into a binary code.
- **General Form**:
  - **Inputs**: 2^n
  - **Outputs**: n
    Example: A **4-to-2 encoder** has 4 inputs (D_0–D_3) and 2 outputs (Y_0, Y_1).
- **Working Principle**:
  - Only one input line is active at a time.
  - The output is the binary code corresponding to the active input line.



INPUTS				OUTPUTS	
Y3	**Y2**	**Y1**	**Y0**	**A1**	**A0**
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Logical expression for A1 and A0:

```
A1 = Y3 + Y2
A0 = Y3 + Y1
```

- **Applications**:
  - Keyboards (detect which key is pressed).
  - Data compression in digital systems.
  - Multiplexing and memory addressing.

2. Decoder

- **Definition**:
  A **decoder** is a combinational logic circuit that converts **n input lines into 2^n output lines**.
  - It performs the reverse operation of an encoder.
  - It is used to **decode binary information** into a specific output line.
- **General Form**:
  - **Inputs**: n
  - **Outputs**: 2^n
    Example: A **2-to-4 decoder** has 2 inputs (A_0, A_1) and 4 outputs (D_0–D_3).
- **Working Principle**:
  - Based on the binary input, one of the outputs becomes **active (logic 1)** while all others remain **inactive (logic 0)**.
- **Example (2-to-4 Decoder Truth Table)**:

2-to-4 Decoder

1. Definition

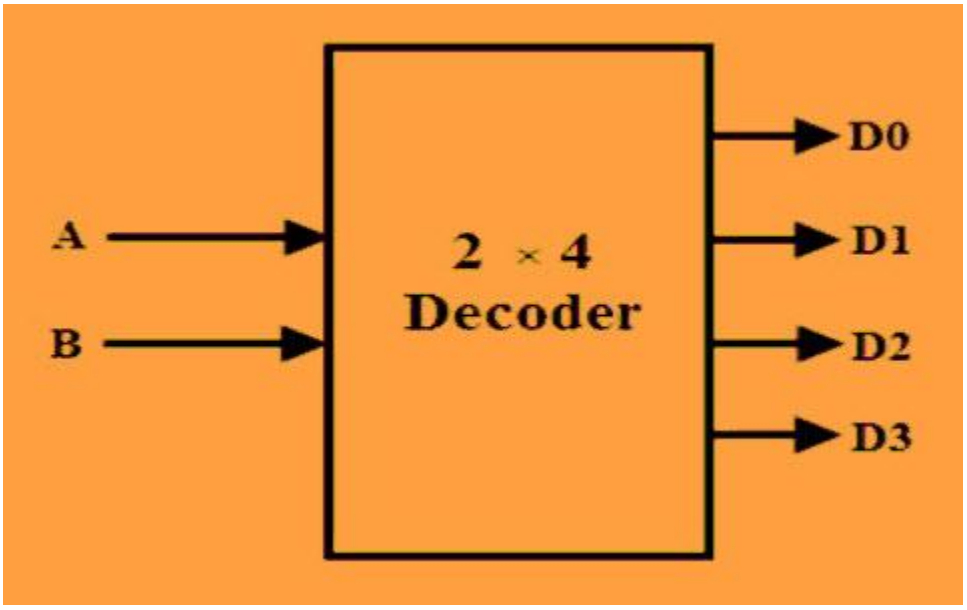
A 2-to-4 decoder is a combinational circuit with:

- **Inputs**: 2 (say A1, A0)
- **Outputs**: 4 (say D0, D1, D2, D3)
- **Operation**: Based on the binary value of inputs, only **one output** is active (logic 1) at a time, and others are 0.

2. Truth Table

A1	A0	D0	D1	D2	D3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

- When inputs are **00 → D0 is active**
- When inputs are **01 → D1 is active**
- When inputs are **10 → D2 is active**
- When inputs are **11 → D3 is active**



- **Applications**:
  - Memory addresses decoding.
  - Instruction decoding in CPUs.
  - Seven-segment display drivers.
  - De multiplexing

3. Difference between Encoder and Decoder

Feature	Encoder	Decoder
Definition	Converts 2^n inputs → n outputs	Converts n inputs → 2^n outputs
Function	Encoding (compression)	Decoding (expansion)
Direction	Many inputs to fewer outputs	Fewer inputs to many outputs
Example	4-to-2 Encoder	2-to-4 Decoder
Operation	Outputs binary code for active input	Activates a single output line for binary input
Use case	Data compression, input recognition	Memory decoding, display drivers

4. Advantages and Disadvantages

Encoder

Advantages:

- Reduces the number of data lines (compression).
- Efficient for large input devices (e.g., keyboards).
- Saves hardware by reducing circuit complexity.

Disadvantages:

- Only one input should be active at a time; otherwise, errors occur.
- Requires additional logic (like priority encoders) to handle multiple inputs.

Decoder

Advantages:

- Expands binary information into distinct outputs.
- Essential for memory addressing and display systems.
- Simple to design using logic gates.

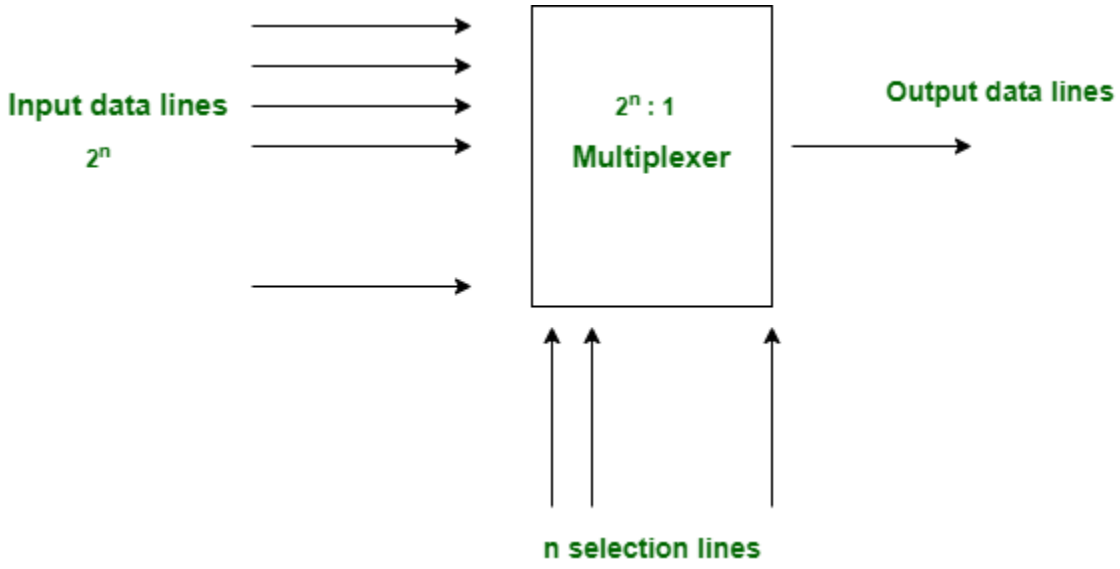
Disadvantages:

- Increases hardware complexity (more outputs → more gates).
- Power consumption is higher for larger decoders.
- Limited scalability for very large input sizes.
-

- **Encoder** = "Compresses inputs into binary code."
- **Decoder** = "Expands binary code into unique outputs."

MULTIPLEXER AND DEMULTIPLEXER

A Multiplexer (MUX) and a Demultiplexer (DEMUX) are essential digital circuits in communication systems, performing opposite functions. A multiplexer combines multiple input signals into a single output, while a demultiplexer takes a single input signal and routes it to one of many output lines.



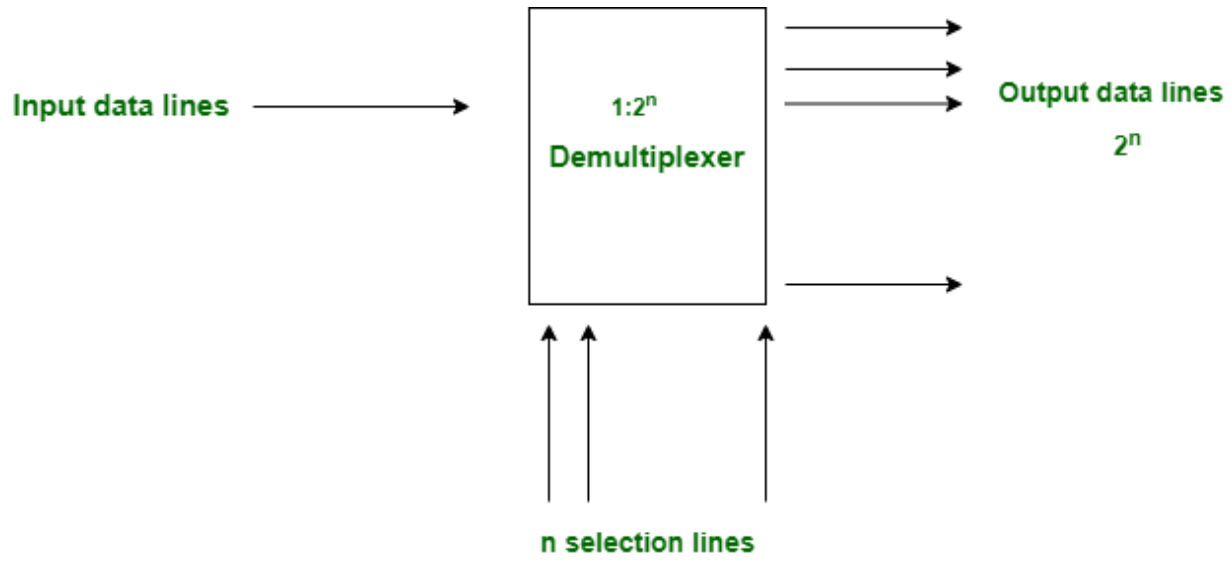
Advantages of Multiplexer (MUX)

- **Reduces the number of data lines**: MUX allows multiple signals to share a single communication line, saving space and resources.
- **Simpler Design**: It simplifies the system by reducing the number of data channels needed.

- **Efficient Use of Resources**: MUX optimizes the use of communication channels or buses.
- **Flexible Design**: MUX can be used for different types of data and communication formats.
- **Compact Systems**: It helps make systems smaller and simpler by reducing the number of connections.
- **Reduces Errors**: Fewer data lines mean less chance of interference or noise.

Disadvantages of Multiplexer (MUX)

- **Complex Control**: It needs extra circuits to select which input is sent, making the design more complicated.
- **Limited Inputs**: The number of inputs depends on the number of control lines (e.g., 2 control lines allow only 4 inputs).
- **Propagation Delay**: Switching between inputs can introduce delays in high-speed systems.
- **Higher Power Use**: Larger multiplexers with more inputs can consume more power.
- **Signal Loss**: Some signal degradation may occur when combining multiple signals.



Advantages of Demultiplexers (DEMUX)

1. **Efficient Data Distribution**: Routes one input signal to multiple outputs effectively.
2. **Reduced Transmission Complexity**: Simplifies transmitter design by minimizing input lines.

3. **High-Speed Data Splitting**: Ideal for applications requiring rapid data distribution.
4. **Scalability**: Can handle more outputs by increasing control lines.
5. **Versatility**: Widely used in TV broadcasting, communication networks, and more.

Disadvantages of Demultiplexers (DEMUX)

1. **Control Complexity**: Additional circuits are needed for output selection.
2. **Limited Outputs**: The number of outputs depends on available control lines.
3. **Propagation Delay**: Routing input to outputs may introduce delays in larger systems.
4. **Signal Degradation**: Signal strength may reduce when split into multiple outputs.
5. **Higher Power Consumption**: Power usage increases with additional outputs.
6. **Noise Susceptibility**: Splitting signals can lead to interference

Multiplexer(MUX)	Demultiplexer(DEMUX)
Multiplexer processes the digital information from various sources into a single source.	Demultiplexer receives digital information from a single source and converts it into several sources
It is known as Data Selector	It is known as Data Distributor
Multiplexer is a digital switch	Demultiplexer is a digital circuit
It follows combinational logic type	It also follows combinational logic type
It has 2n input data lines	It has single input line
It has a single output data line.	It has 2n output data lines

Multiplexer(MUX)	Demultiplexer(DEMUX)
Efficiently uses bandwidth by combining many signals into a single line for transmission.	Divides a single signal into several parts, so bandwidth is less efficiently used.
Needs control lines to select which input signal to send to the output.	Needs control lines to determine which output line should receive the input signal.
It works on many to one operational principle	It works on one to many operational principle.
May consume more power due to the need for control logic and multiple input connections.	Typically uses less power, especially when splitting a single signal to multiple destinations.
In time division Multiplexing multiplexer is used at the transmitter end.	In time division Multiplexing, demultiplexer is used at the receiver end.