

2nd Day

Decoder

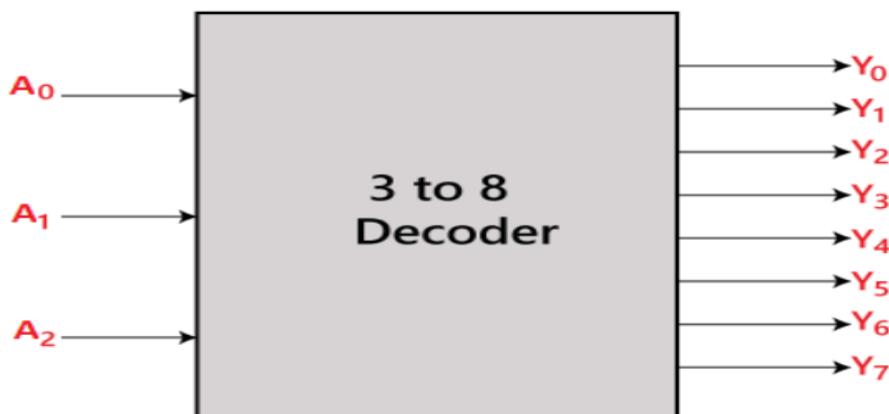
In digital electronics, a combinational logic circuit that converts an N-bit binary input code into M output channels in such a way that only one output channel is activated for each one of the possible combinations of inputs is known as a **decoder**.

- ✓ A decoder can implement any Boolean function by OR-ing minterms.
- ✓ It is a powerful combinational circuit, used in multiplexers, arithmetic circuits, and memory

Applications

📌 Where is it Used?

- Memory Address Decoding** – Selecting specific memory locations in RAM
- Instruction Decoding** – CPUs use decoders to understand machine instructions
- Data Routing** – Determines which signal path



A ₂	A ₁	A ₀	Y ₇	Y ₆	Y ₅	Y ₄	Y ₃	Y ₂	Y ₁	Y ₀
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Design

```
module decoder_3_8(input [2:0]a, output reg [7:0]y);

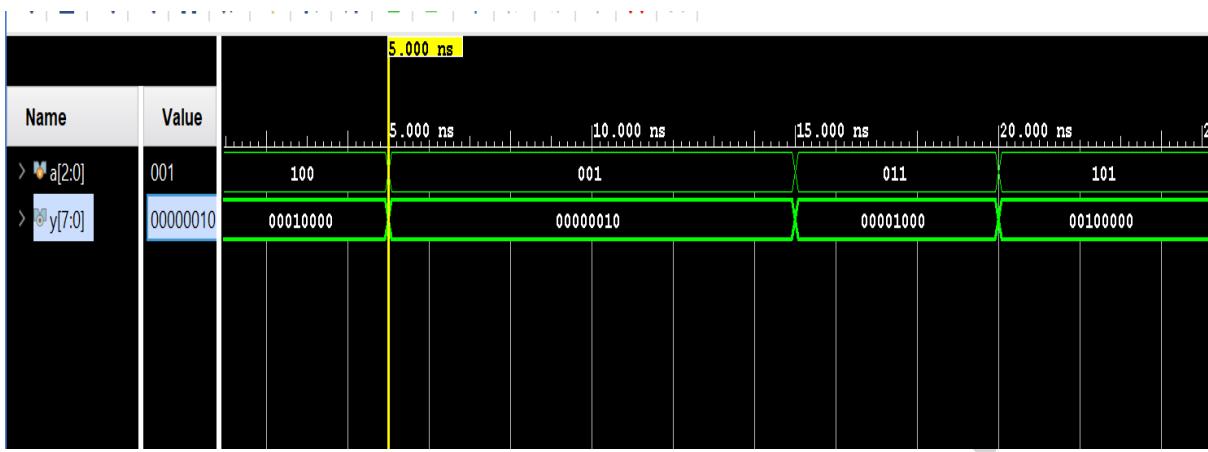
    always @(*)
        begin
            case(a)
                3'b000:y=8'b00000001;
                3'b001:y=8'b00000010;
                3'b010:y=8'b00000100;
                3'b011:y=8'b00001000;
                3'b100:y=8'b00010000;
                3'b101:y=8'b00100000;
                3'b110:y=8'b01000000;
                3'b111:y=8'b10000000;
                default:
                    begin
                        y=8'bxxxxxxxxx;
                    end
            endcase
        end
    endmodule
```

Testbench

```
module tb_decoder( );
reg [2:0]a;
wire[7:0]y;
decoder_3_8 dut(.a(a),.y(y));
initial
begin
$monitor("a=%b,y=%b",a,y);
repeat(7)
begin
a=$random();
#5;
end
end
endmodule
```

Simulation

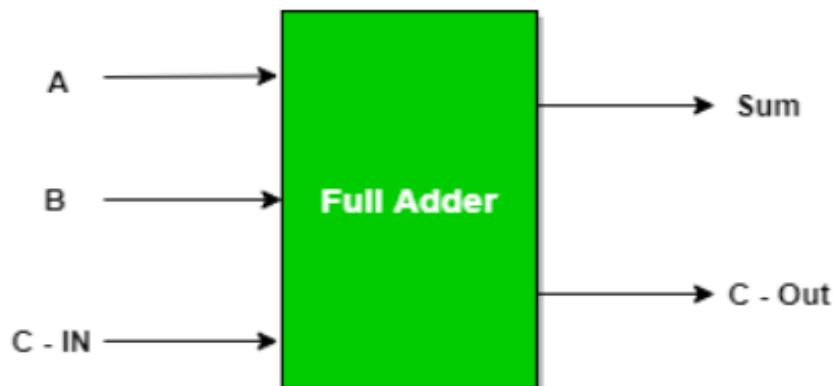
```
# }
# run 1000ns
a=100,y=00010000
a=001,y=00000010
a=011,y=00001000
a=101,y=00100000
INFO: [USF-XSim-96] XSim completed. Design snapshot 'tb_decoder_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:03 ; elapsed = 00:00:10 . Memory (MB): peak = 2387.555 ; gain = 0.000
```



ManojKurishetli

Full Adder

Full Adder is the adder that adds three inputs and produces two outputs. 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.



Inputs			Outputs	
A	B	C - IN	Sum	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

$$\text{Sum} = m(1, 2, 4, 7)$$

$$\text{Cout} = m(3, 5, 6, 7)$$

Full Adder Using 3*8 Decoder

Design

```
module fa_using_dec(input a,b,cin,output sum,cout);
wire [7:0]m;
decoder_3_8 m1(.a({a,b,cin}),.y(m));
assign sum= m[1] | m[2] | m[4] | m[7];
assign cout=m[3] | m[5] | m[6] | m[7];
endmodule
```

Testbench

```
module fa_using_dec_tb();
reg a,b,cin;
wire sum,cout;
fa_using_dec dut(a,b,cin,sum,cout);
initial
begin
$monitor("a=%b b=%b cin=%b sum=%b cout=%b",a,b,cin,sum,cout);
repeat(7)
begin
{a,b,cin}=$random;
#5;
end
end
endmodule
```

Simulation

```
#      send_msg_to_Add_Wave-1 WARNING "NO top level signals found. Simulator will start without a wave window. If you want
#  }
# }
# run 1000ns
a=1 b=0 cin=0 sum=1 cout=0
a=0 b=0 cin=1 sum=1 cout=0
a=0 b=1 cin=1 sum=0 cout=1
a=1 b=0 cin=1 sum=0 cout=1
INFO: [USF-XSim-96] XSim completed. Design snapshot 'fa_using_dec_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:03 ; elapsed = 00:00:13 . Memory (MB): peak = 2387.555 ; gain = 0.000
```

