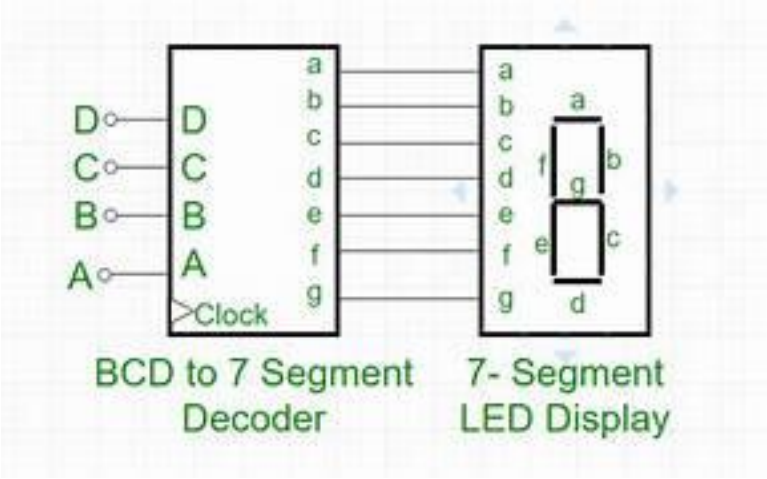
BCD-SEVEN SEGMENT DECODER

In **Binary Coded Decimal (BCD)** encoding scheme each of the decimal numbers(0-9) is represented by its equivalent binary pattern(which is generally of 4-bits). Whereas, **Seven segment** display is an electronic device which consists of seven Light Emitting Diodes (LEDs) arranged in a some definite pattern (common cathode or common anode type), which is used to display Hexadecimal numerals(in this case decimal numbers,as input is BCD i.e., 0-9). Two types of seven segment LED display:

1. **Common Cathode Type:** In this type of display all cathodes of the seven LEDs are connected together to the ground or -Vcc(hence,common cathode) and LED displays digits when some 'HIGH' signal is supplied to the individual anodes.
2. **Common Anode Type:** In this type of display all the anodes of the seven LEDs are connected to battery or +Vcc and LED displays digits when some 'LOW' signal is supplied to the individual cathodes.

But, seven segment display does not work by directly supplying voltage to different segments of LEDs. First, our decimal number is changed to its BCD equivalent signal then BCD to seven segment decoder converts that signals to the form which is fed to seven segment display. This BCD to seven segment decoder has four input lines (A, B, C and D) and 7 output lines (a, b, c, d, e, f and g), this output is given to seven segment LED display which displays the decimal number depending upon inputs.



RTL CODE:

```
module tlsa(input [3:0] data,output reg a,b,c,d,e,f,g);
  always@(*)
    begin
      case(data)
        4'b0000:begin
          a=1'b1;b=1'b1;c=1'b1;d=1'b1;e=1'b1;f=1'b1;g=1'b0;
        end
        4'b0001:begin
          a=1'b0;b=1'b1;c=1'b1;d=1'b0;e=1'b0;f=1'b0;g=1'b0;
        end
        4'b0010:begin
          a=1'b1;b=1'b1;c=1'b0;d=1'b1;e=1'b1;f=1'b0;g=1'b1;
        end
        4'b0011:begin
          a=1'b1;b=1'b1;c=1'b1;d=1'b1;e=1'b0;f=1'b0;g=1'b1;
        end
        4'b0100:begin
          a=1'b0;b=1'b1;c=1'b1;d=1'b0;e=1'b0;f=1'b1;g=1'b1;
```

```
        end
        4'b0101:begin
          a=1'b1;b=1'b0;c=1'b1;d=1'b1;e=1'b0;f=1'b1;g=1'b1;
        end
        4'b0110:begin
          a=1'b1;b=1'b0;c=1'b1;d=1'b1;e=1'b1;f=1'b1;g=1'b1;
        end
        4'b0111:begin
          a=1'b1;b=1'b1;c=1'b1;d=1'b0;e=1'b0;f=1'b0;g=1'b0;
        end
        4'b1000:begin
          a=1'b1;b=1'b1;c=1'b1;d=1'b1;e=1'b1;f=1'b1;g=1'b1;
        end
        4'b1001:begin
          a=1'b1;b=1'b1;c=1'b1;d=1'b0;e=1'b0;f=1'b1;g=1'b1;
        end
      endcase
    end
endmodule
```

TESTBENCH:

```
module test;
  reg [3:0]data;
  wire a,b,c,d,e,f,g;
  tlsa a1(data,a,b,c,d,e,f,g);
  initial begin
    $dumpfile("dump.vcd");
    $dumpvars(1);
```

```
  end
  initial begin
    data=0;
    #5 data=1;
    #5 data=2;
    #5 data=3;
    #5 data=4;
    #5 data=5;
    #5 data=6;
    #5 data=7;
    #5 data=8;
    #5 data=9;
    #5 data=1;
  end
  initial begin
    #60 $finish();
  end
endmodule
```