# DAY5

# 4 BIT RIPPLE CARRY ADDER

# EXPLANATION:

Assume you want to add two operands A and B

where A= A3 A2 A1 A0

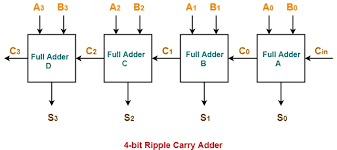
B=B3 B2 B1 B0

For example: A= 1 0 1 1 + B= 1 1 0 1

--------------- A+B= 1 1 0 0 0

C out S3 S2 S1 S0

Ripple carry adder is a combinational circuit that can perform addition operation of two n-bit binary numbers. It accepts two n-bit binary numbers as an inputs, performs addition of both the binary numbers and generated it's addiiton as an output. You'll find easy to understand this topic if you know full adder how the full adder circuit works. If you do not know how full adder works, let us understand full Adder illustrating it as a black box



# RTL CODE:

module FA(input a, b, Cin, output sum, Cout);

assign sum = a ^ b ^ Cin;

assign Cout = (a & b) | (b & Cin) | (Cin & a);

endmodule

module Ripple\_CarryAdder(input [3:0] a, b, Cin, output [3:0] sum, Cout);

wire [2:0] w;

FA f1(a[0], b[0], Cin, sum[0], w[0]);

FA f2(a[1], b[1], w[0], sum[1], w[1]);

FA f3(a[2], b[2], w[1], sum[2], w[2]);

FA f4(a[3], b[3], w[2], sum[3], Cout);

Endmodule

# TEST BENCH:

module testbench;

reg [3:0] a = 4'b0000, b = 4'b0000;

reg Cin = 1'b0;

wire [3:0] sum;

wire Cout;

Ripple\_CarryAdder RCA2(a, b, Cin, sum, Cout);

initial

begin

#10 a = 4'b0001; b = 4'b1000; Cin = 1'b0;

#10 a = 4'b0011; b = 4'b0110; Cin = 1'b1;

end

initial

begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

initial

#60 $finish();

Endmodule

