COE 202, Term 151

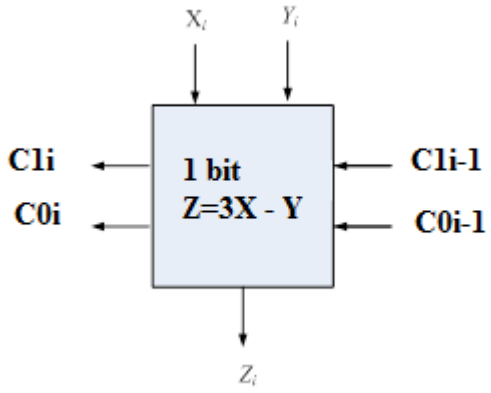
Digital Logic Design

Assignment# 2 Solution

Due date: Sunday, Oct. 31

Q.1. It is required to design an iterative combinational circuit that computes the equation Z=3*X-Y, where X and Y are n-bit unsigned numbers.

(i) Determine the number of inputs and outputs needed for your 1-bit cell.



The meaning of the values of $C1_{i-1}$ and $C0_{i-1}$ is given in the table below:

$C1_{i-1}$	$C0_{i-1}$	Meaning
0	0	There is no carry or borrow
0	1	There is a carry of 1
1	0	There is a carry of 2
1	1	There is a borrow of 1

(ii) Derive the truth table of your 1-bit cell.

$C1_{i-1}$	$C0_{i-1}$	X_i	Y_i	$C1_i$	$C0_i$	Z_i
0	0	0	0	0	0	0
0	0	0	1	1	1	1
0	0	1	0	0	1	1
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	0	0	0
0	1	1	0	1	0	0
0	1	1	1	0	1	1
1	0	0	0	0	1	0
1	0	0	1	0	0	1
1	0	1	0	1	0	1
1	0	1	1	1	0	0
1	1	0	0	1	1	1
1	1	0	1	1	1	0
1	1	1	0	0	1	0
1	1	1	1	0	0	1

(iii) Derive minimized equations for your 1-bit using K-Map method.

	00	01	11	10
00	0 0	1 1	0 3	1 2
01	1 4	0 5	1 7	0 6
11	1 12	0 13	1 15	0 14
10	0 8	1 9	0 11	1 10

Zi = C0i-1 Xi'Yi' + C0i-1 Xi Yi + C0i-1' Xi'Yi + C0i-1' Xi Yi'

= C0i-1(Xi'Yi' + Xi Yi)+ C0i-1' (Xi' Yi + Xi Yi')

= C0i-1(Xi ⊕ Yi)' + C0i-1' (Xi ⊕ Yi)

= C0i-1 ⊕ Xi ⊕ Yi

	00	01	11	10
00	0 0	1 1	1 3	1 2
01	0 4	0 5	1 7	0 6
11	1 12	1 13	0 15	1 14
10	1 8	0 9	0 11	0 10

C0i = C1i-1' C0i-1' Xi + C1i-1' C0i-1' Yi + C1i-1 C0i-1 Xi' + C1i-1 C0i-1 Yi' + C1i-1 Xi' Yi' + C1i-1' Xi Yi

= C1i-1' C0i-1' (Xi + Yi) + C1i-1 C0i-1 (Xi' +Yi')+ C1i-1 Xi' Yi' + C1i-1' Xi Yi

	00	01	11	10
00	0 0	1 1	0 3	0 2
01	0 4	0 5	0 7	1 6
11	1 12	1 13	0 15	0
10	0 8	0 9	1 11	1 10

C1i = C1i-1 C0i-1 Xi' + C1i-1 C0i-1' Xi + C1i-1' C0i-1' Xi' Yi + C1i-1' C0i-1 Xi Yi'
= C1i-1 (C0i-1 ⊕ Xi) + C1i-1' (C0i-1' Xi' Yi + C0i-1 Xi Yi')

(iv) Write a Verilog model for modeling your 1-bit cell by using an assign statement for each output.

```
module OneBit3X_Y (input C1i_1, C0i_1, Xi, Yi, output C1i,
C0i, Zi);

assign Zi =  C0i_1 ^ Xi  ^ Yi ;

assign C0i  =  ~C1i_1 & ~C0i_1 & (Xi | Yi ) | C1i_1 & C0i_1 &
(~ Xi | ~Yi ) | C1i_1 & ~Xi &  ~Yi | ~C1i_1 &  Xi & Yi;

assign C1i =    C1i_1 & (C0i_1 ^ Xi) | ~C1i_1 & (~C0i_1 &  ~Xi
& Yi | C0i_1 & Xi & ~Yi);

endmodule
```

(v) Write a Verilog model for modeling a 4-bit circuit based on the 1-bit model you have.

```
module FourBit3X_Y (input [3:0] X, Y, output [3:0] Z, output
C1_out, C0_out);

wire [3:0] C1, C0;

assign C1_out = C1[3];
```

```
assign C0_out = C0[3];

OneBit3X_Y M0 (1'b0, 1'b0, X[0], Y[0], C1[0], C0[0], Z[0]);
OneBit3X_Y M1 (C1[0], C0[0], X[1], Y[1], C1[1], C0[1], Z[1]);
OneBit3X_Y M2 (C1[1], C0[1], X[2], Y[2], C1[2], C0[2], Z[2]);
OneBit3X_Y M3 (C1[2], C0[2], X[3], Y[3], C1[3], C0[3], Z[3]);

endmodule
```

(vi) Write a Verilog test bench to test the correctness of your design for the following input values: {X=1,Y=1}, {X=3, Y=5}, {X=5, Y=1}, {X=4, Y=15}, and {X=15, Y=0}.

```
module FourBit3X_Ytest();


reg [3:0] X, Y;
wire [3:0] Z;
wire C1_out, C0_out;

FourBit3X_Y M1 (X, Y, Z, C1_out, C0_out);

initial begin
    X=4'd1; Y=4'd1;
    #10 X=4'd3; Y=4'd5;
    #10 X=4'd5; Y=4'd1;
    #10 X=4'd4; Y=4'd15;
    #10 X=4'd15; Y=4'd0;
  end
endmodule
```