(1) Write verilog code for signal assignment statement $y = 2*x + 3$ with 2-bit input. Show the synthesized logic symbol & gate level diagram. Write structural code in verilog using gate level diagram.

Verilog code:

```
Module sign1 (x, y);
input [1:0] x;
output [3:0] y;
reg [3:0] y;
always @ (x)
begin
    y = 2* x + 3
end
endmodule
```

⇒ The module has one input x of 2-bits and one output of 4-bits.

⇒ The always contains one signal 'x' in the sensitivity list. Also it has one signal assignment statement $y = 2*x + 3$.

⇒ To synthesis the code, we construct a truthtable to find the logic diagram, and use gate-level synthesis.

input(x)		output(y)			
x_1	x_0	y_3	y_2	y_1	y_0
0	0	0	0	1	1
0	1	0	1	0	1
1	0	0	1	1	1
1	1	1	0	0	1

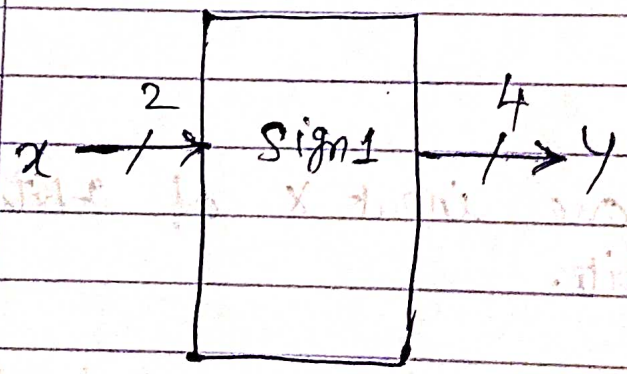
$y_0 = 1$

$y_1 = \overline{x(0)}$

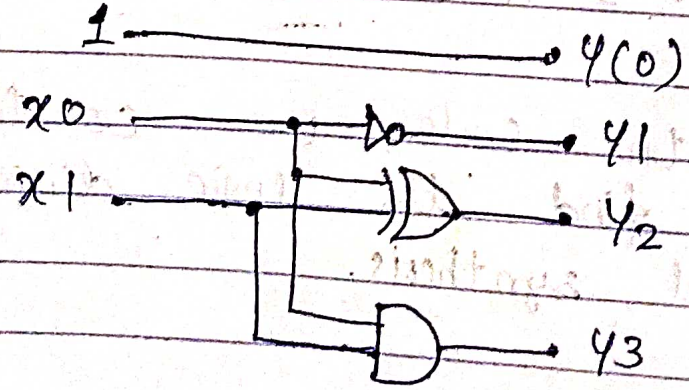
$y_2 = \overline{x_1}\, x_0 + x_1 \overline{x_0} = x_1 \oplus x_0$

$y_3 = x_1 \cdot x_0$

Logic symbol:



Gate level logic diagram:



⇒ To verify our synthesis, we write the structural code for the logic diagram (gate level)

⇒ If the simulation of structural code is same as the simulation wave form of the HDL behavioural code, then synthesis is correct.

The verilog structural code for the logic diagram

```
module sign_struct(x,y);
input [1:0]x;
output [3:0]y;
reg [3:0]y;
always @(x)
begin
  y[0] = 1'b1;
  y[1] = ~x[0];
  y[2] = x[0] ^ x[1];
  y[3] = x[1] &x[0];
end
endmodule
```