QUESTION 1

Using truth table or Karnaugh map, derive the Boolean expressions for carry-out and sum for half-adder and full-adder. Provide your answer in the sum of products (SOP) form.

[10 Marks]

[CO1, PO1, C3]

QUESTION 2

Figure 1 shows a 2-bit ripple carry adder circuit where HA is the half adder and FA is the full adder. Implement the circuit using the programmable array logic (PAL) device in Appendix C. Show all your calculation. Attach Appendix B with your answer sheet.

[10 Marks]

[CO1, PO1, C4]

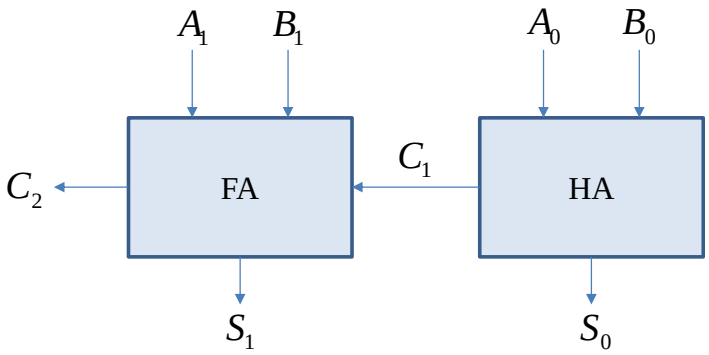


Figure 1

QUESTION 3

Figure 2 shows a combinational logic circuit with FOUR (4) inputs and ONE (1) output implemented using 8-to-1 Multiplexer.

(i) Construt a Truth Table for the logic circuit in Figure 2.

(ii) Express the Boolean representations of the outputs, `F` in terms of inputs `A, B, C` and `D`.

(iii) Write a Verilog module for the logic circuit using Gate-Level Modeling or Logical Operator.

[10 Marks]

[CO3, PO3, C4]

8:1 MUX

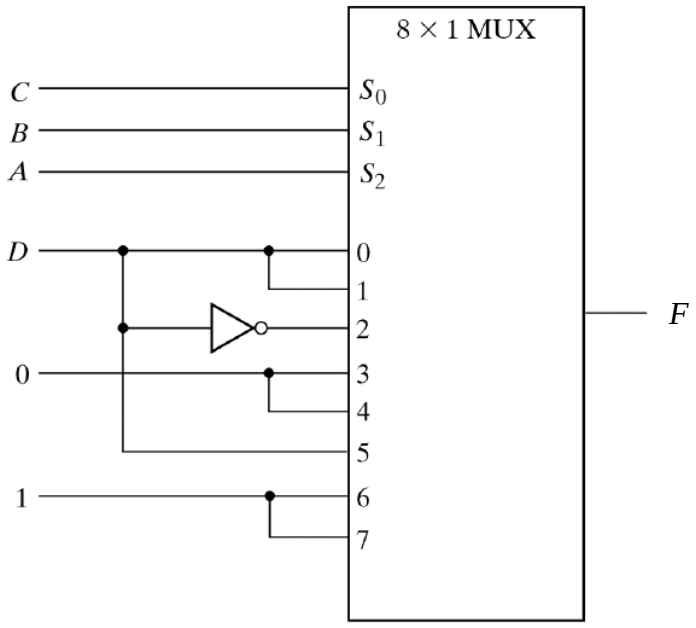


Figure 2

QUESTION 4

Table 1 shows a truth table that describes functions $X(A, B, C)$, $Y(A, B, C)$ and $Z(A, B, C)$ where symbol × denotes don't care. Realize all three functions *X*, *Y* and *Z* using the 3×4×3 PLA provided in Appendix B. (Attach the appendix with your answer sheets.)

[10 Marks]

Table 1

A	*B*	*C*	*X*	*Y*	*Z*
0	0	0	×	0	0
0	0	1	0	1	1
0	1	0	0	0	1
0	1	1	0	1	1
1	0	0	×	0	0
1	0	1	0	0	1
1	1	0	1	×	1
1	1	1	1	1	0

QUESTION 5

Figure 1 shows an incomplete 4-bit carry-lookahead adder in which the circuit for C_4 is missing.

(i) Identify and draw the internal circuit of the Σ block.

(ii) Derive the expression for C_4 and draw the circuit that produces C_4.

[10 Marks]

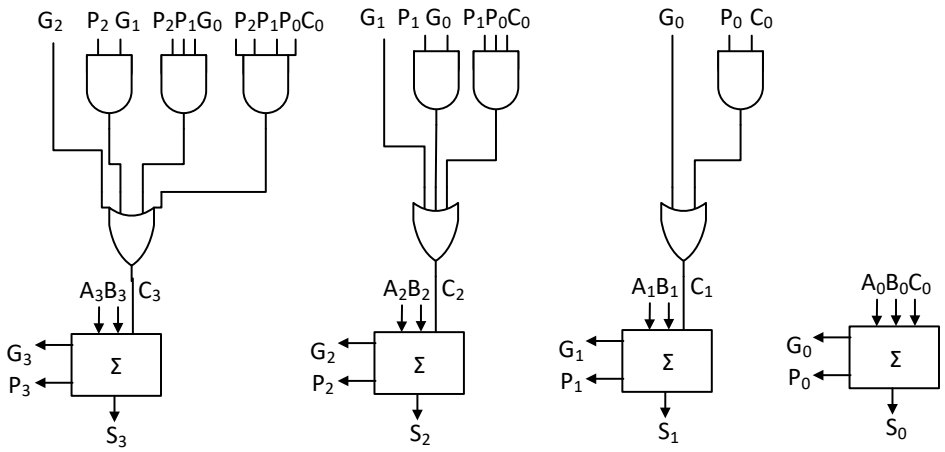


Figure 1

QUESTION 6

Consider the Verilog module in Figure 2.

(i) From the Verilog codes, sketch an equivalent logic circuit.

(ii) Construct the truth table of the equivalent circuit in (i).

(iii) Identify the application of the module.

[10 Marks]

```
module test1(
     input select,
     input[1:0] d,
     output     q);

wire q1, q2, q3, q4, NOTselect;

not(NOTselect, select);

and(q1, NOTselect, d[0]);
and(q2, select, d[1]);

or( q, q1, q2);

endmodule
```

Figure 2

APPENDIX A – List of Formulas

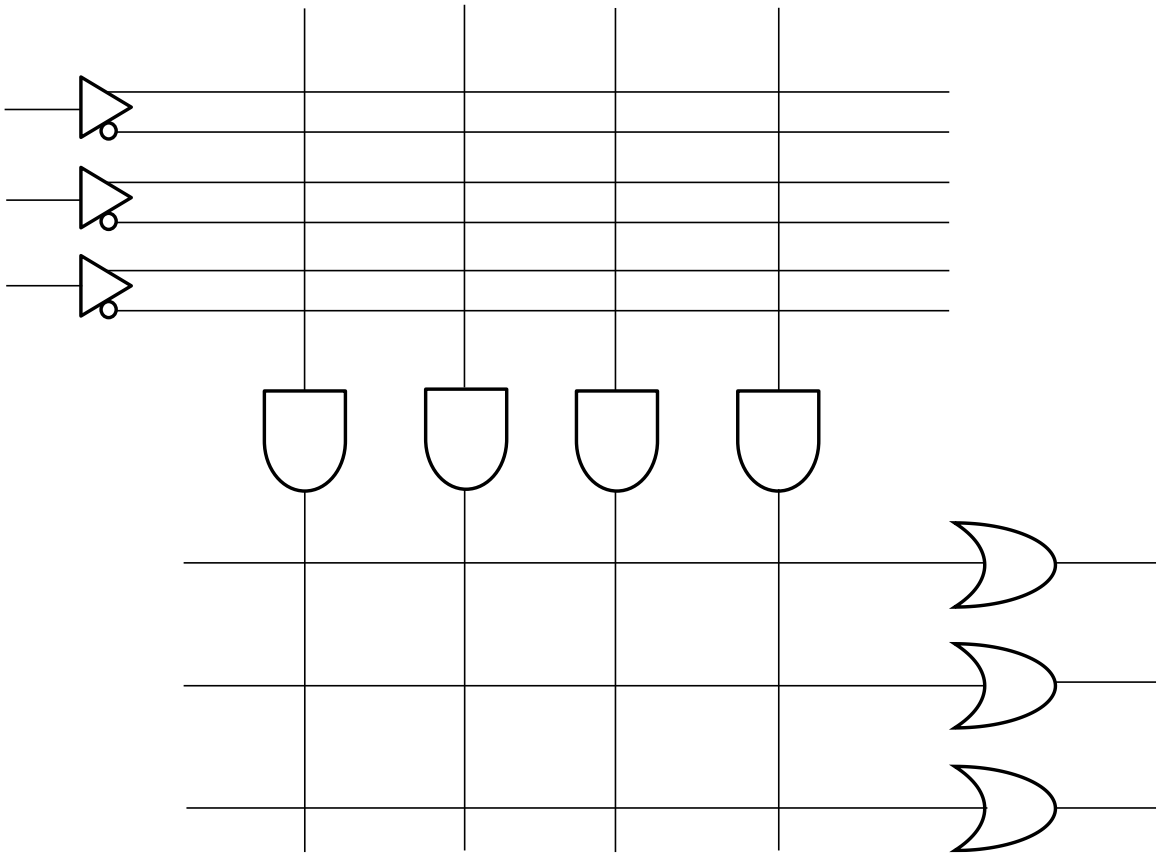
Boolean identities

1. $X + 0 = X$
2. $X + 1 = 1$
3. $X + X = X$
4. $\overline{\overline{X}} = X$
5. $X + \overline{X} = 1$
6. $X + Y = Y + X$
7. $(X + Y) + Z = X + (Y + Z)$
8. $X \cdot (Y + Z) = X \cdot Y + X \cdot Z$
9. $(W + X) \cdot (Y + Z) = WY + XY + WZ + XZ$
10. $X \cdot Y + X \cdot \overline{Y} = X$
11. $X + X \cdot Y = X$
12. $X + \overline{X} \cdot Y = X + Y$
13. $(X + \overline{Y}) \cdot Y = X \cdot Y$
14. $\overline{(X + Y + Z + \cdots)} = \overline{X} \cdot \overline{Y} \cdot \overline{Z} \cdots$
15. $(X + Y) \cdot (\overline{X} + Z) = X \cdot Z + \overline{X} \cdot Y$
16. $X \cdot Y + Y \cdot Z + \overline{X} \cdot Z = X \cdot Y + \overline{X} \cdot Z$

Carry lookahead adder

$$S_i = P_i \oplus C_i \qquad P_i = A_i \oplus B_i$$
$$C_{i+1} = G_i + P_i C_i \qquad G_i = A_i B_i$$

APPENDIX B – Programmable Logic Array (3×4×3 PLA)



QUESTION 7

A seven-segment display is a form of electronic device for displaying decimal numerals that is an alternative to the more complex dot matrix displays. Seven-segment displays are widely used in digital clocks, electronic meters, basic calculators, and other electronic devices that display numerical information.

Figure 1 shows the individual segments of a seven-segment display with common anode (LED lights up when input is "low") configuration. By referring to Appendix B:-

(i) Implement the binary-coded decimal (BCD) to the seven-segment decoder using the 2^4×7 PROM device.

(ii) Provide the same implementation using the 4×9×7 PLA device.

Your solution must include the truth table and sum of minterm expressions. Attach the appendix with your answer sheet.

[12 Marks]

[CO1, PO1, C4]

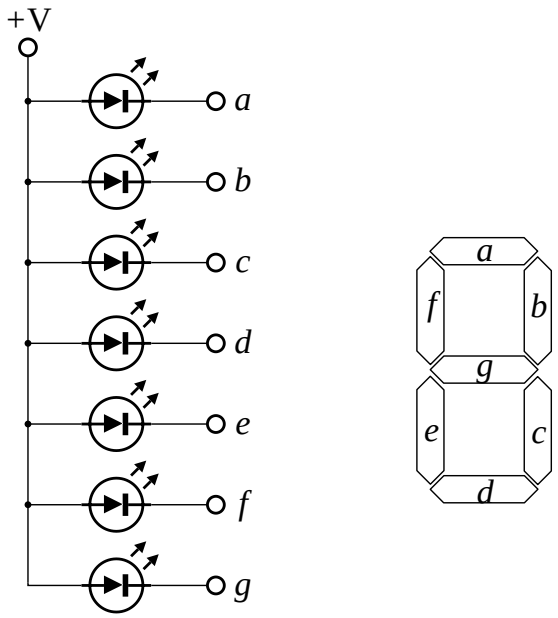


Figure 1

QUESTION 8

Figure 2 shows a logic circuit with 3 inputs, X_2, X_1, X_0 and 3 outputs, $Count_2$, $Count_1$ and $Count_0$.

(i) Construct the truth table for the circuit.

(ii) Express the Boolean representations of the outputs, $Count_2$, $Count_1$ and $Count_0$ in terms of inputs X_2, X_1, and X_0.

[8 Marks]

[CO1, PO1, C3]

x_1	x_0	0

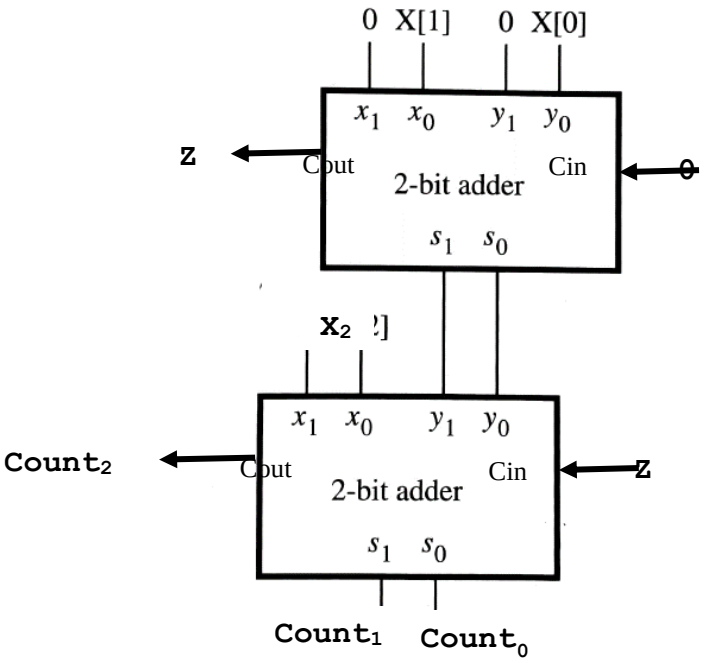


Figure 2

QUESTION 9

Consider the Verilog module in Figure 3.

(i) Using only NAND gates, sketch the corresponding logic circuit for the module. Label all inputs and output.

(ii) Construct the truth table for the logic circuit.

(iii) Sketch an alternative logic circuit for the module other than your answer in (i). Then identify the functionality of the **dunno** module.

[10 Marks]

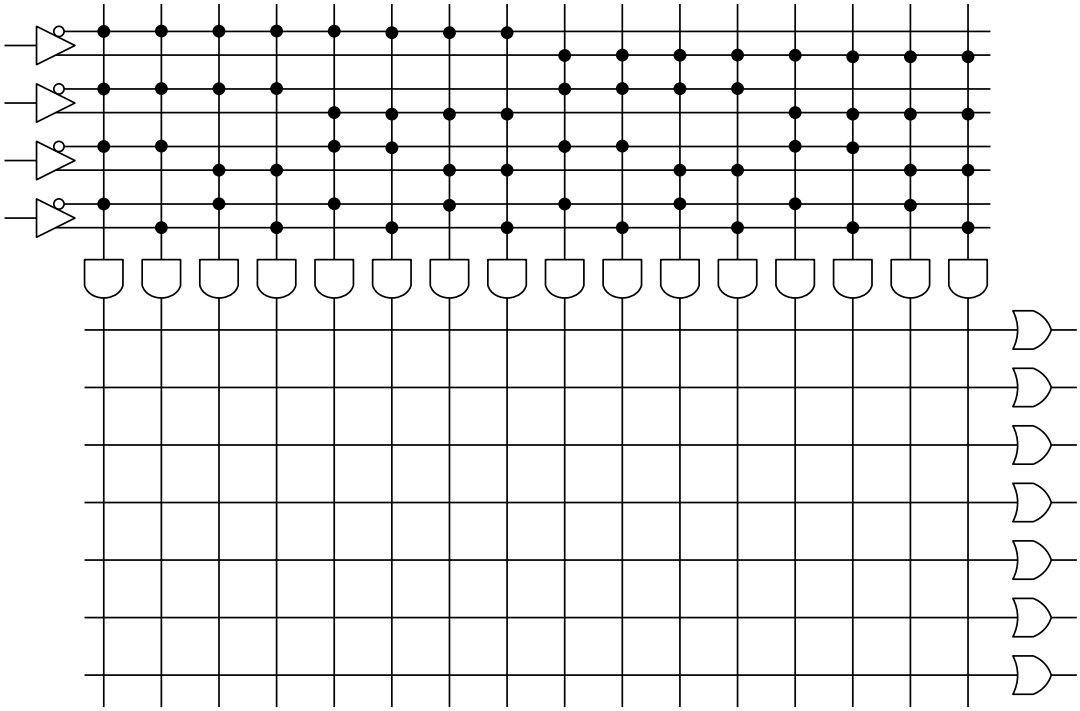
[CO1, PO1, C4]

```
module dunno(
     output    out,
     input     p,q,s);

     wire sbar,p2,q2;

     nand(sbar,s,s);
     nand(p2,p,sbar);
     nand(q2,q,s);
     nand(out,p2,q2);

endmodule
```

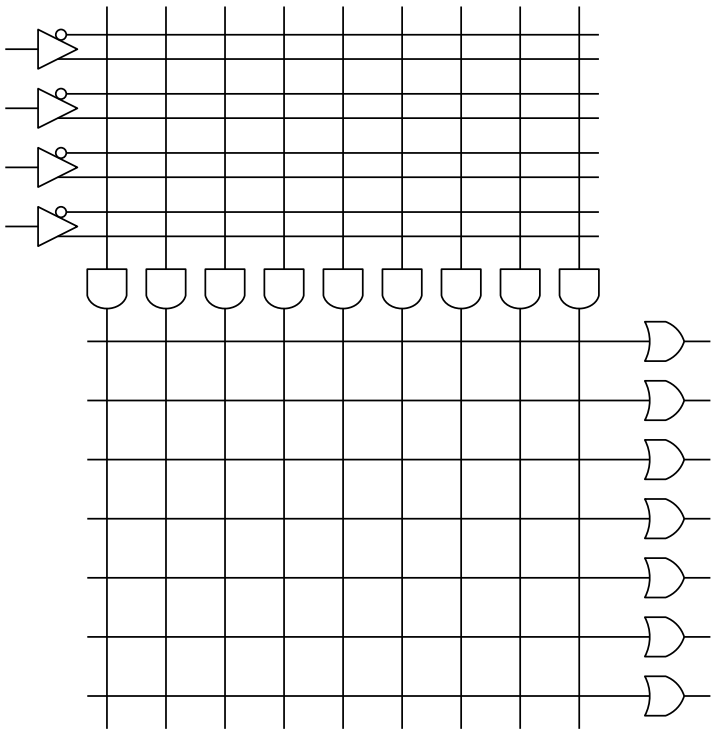
Figure 3

APPENDIX B – Programmable Logic Devices

$2^4 \times 7$ PROM



4×9×7 PLA



QUESTION 10

Design a 3-bit carry lookahead adder using three sigma blocks and the PLA device as given in Appendix B. Show your derivation. Label all connections in the circuit.
(Attach Appendix B with your answer sheets).

[10 Marks]

[CO1, PO1, C4]

QUESTION 11

Design a 4-bit multiplier using appropriate logic gates, half-adders, and full adders. Show your derivation.

[10 Marks]

[CO2, PO3, C4]

QUESTION 12

Consider the Verilog module in Figure 1.

(i) From the Verilog codes, sketch an equivalent logic circuit.

(ii) Construct the truth table corresponds to the circuit in (i).

(iii) Identify the application of the module.

[10 Marks]

[CO3, PO3, C3]

```
module test1(
     input x,y,z,
     output r,s  );

     wire w1,w2,w3;

     xor(w1,x,y);
     and(w2,x,y);
     and(w3,z,w1);
     xor(s,z,w1);
     or(r,w2,w3);

endmodule
```

Figure 1

APPENDIX A – List of Formulas

Boolean identities

1. $X + 0 = X$
2. $X + 1 = 1$
3. $X + X = X$
4. $\overline{\overline{X}} = X$
5. $X + \overline{X} = 1$
6. $X + Y = Y + X$
7. $(X + Y) + Z = X + (Y + Z)$
8. $X \cdot (Y + Z) = X \cdot Y + X \cdot Z$
9. $(W + X) \cdot (Y + Z) = WY + XY + WZ + XZ$
10. $X \cdot Y + X \cdot \overline{Y} = X$
11. $X + X \cdot Y = X$
12. $X + \overline{X} \cdot Y = X + Y$
13. $\left(X + \overline{Y}\right) \cdot Y = X \cdot Y$
14. $\left(\overline{X + Y + Z + \cdots}\right) = \overline{X} \cdot \overline{Y} \cdot \overline{Z} \cdots$
15. $(X + Y) \cdot \left(\overline{X} + Z\right) = X \cdot Z + \overline{X} \cdot Y$
16. $X \cdot Y + Y \cdot Z + \overline{X} \cdot Z = X \cdot Y + \overline{X} \cdot Z$

Carry lookahead adder

$$S_i = P_i \oplus C_i \qquad P_i = A_i \oplus B_i$$

$$C_{i+1} = G_i + P_i C_i \qquad G_i = A_i B_i$$

APPENDIX B –Sigma Blocks and Programmable Logic Array