

B38DF

Computer Architecture and Embedded Systems

Tutorial on Programmable Logic Devices and Memory Address Decoding

Alexander Belyaev

Heriot-Watt University
School of Engineering & Physical Sciences
Electrical, Electronic and Computer Engineering

E-mail: a.belyaev@hw.ac.uk

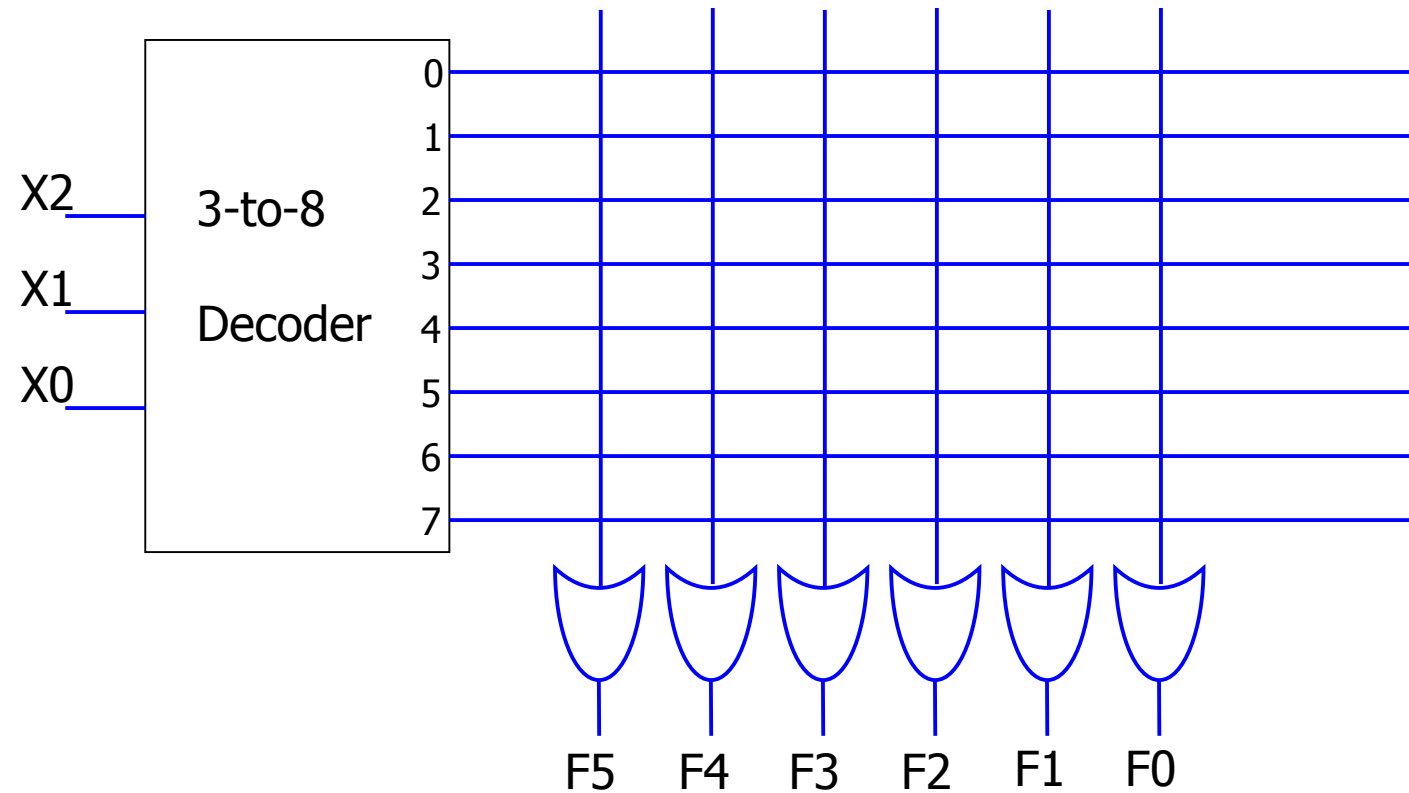
Office: EM2.29

Based on materials prepared by Dr. Mustafa Suphi Erden and Dr. Senthil Muthukumaraswamy

Problem 0. Read Only Memory (ROM)

Use ROM to store the following function.

X	$F(X)=X^2+2$
000	
001	
010	
011	
100	
101	
110	
111	

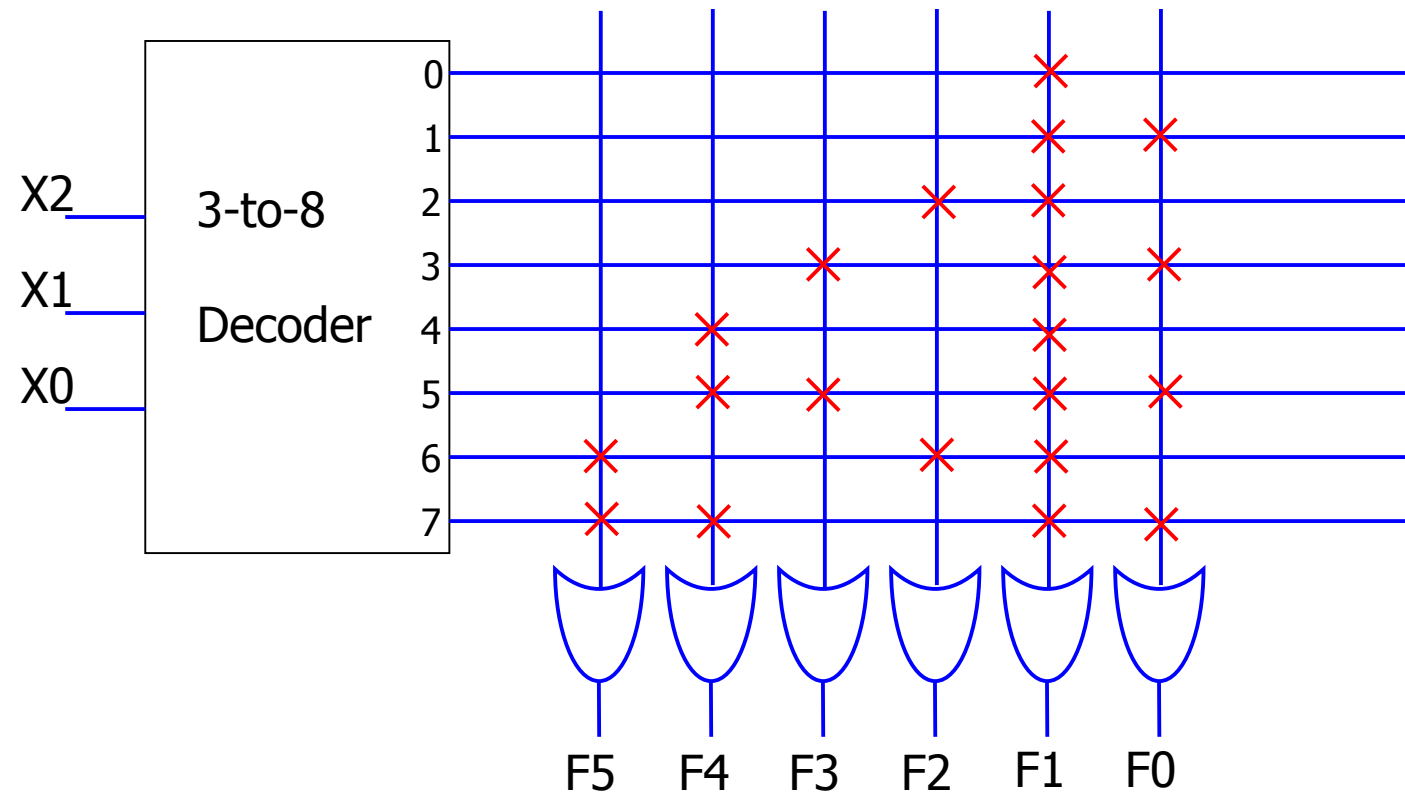


Output contains 6 bits

Solution 0. Read Only Memory (ROM)

Use ROM to store the following function.

X	$F(X)=X^2+2$
000	000010
001	000011
010	000110
011	000110
100	010010
101	011011
110	100110
111	110011

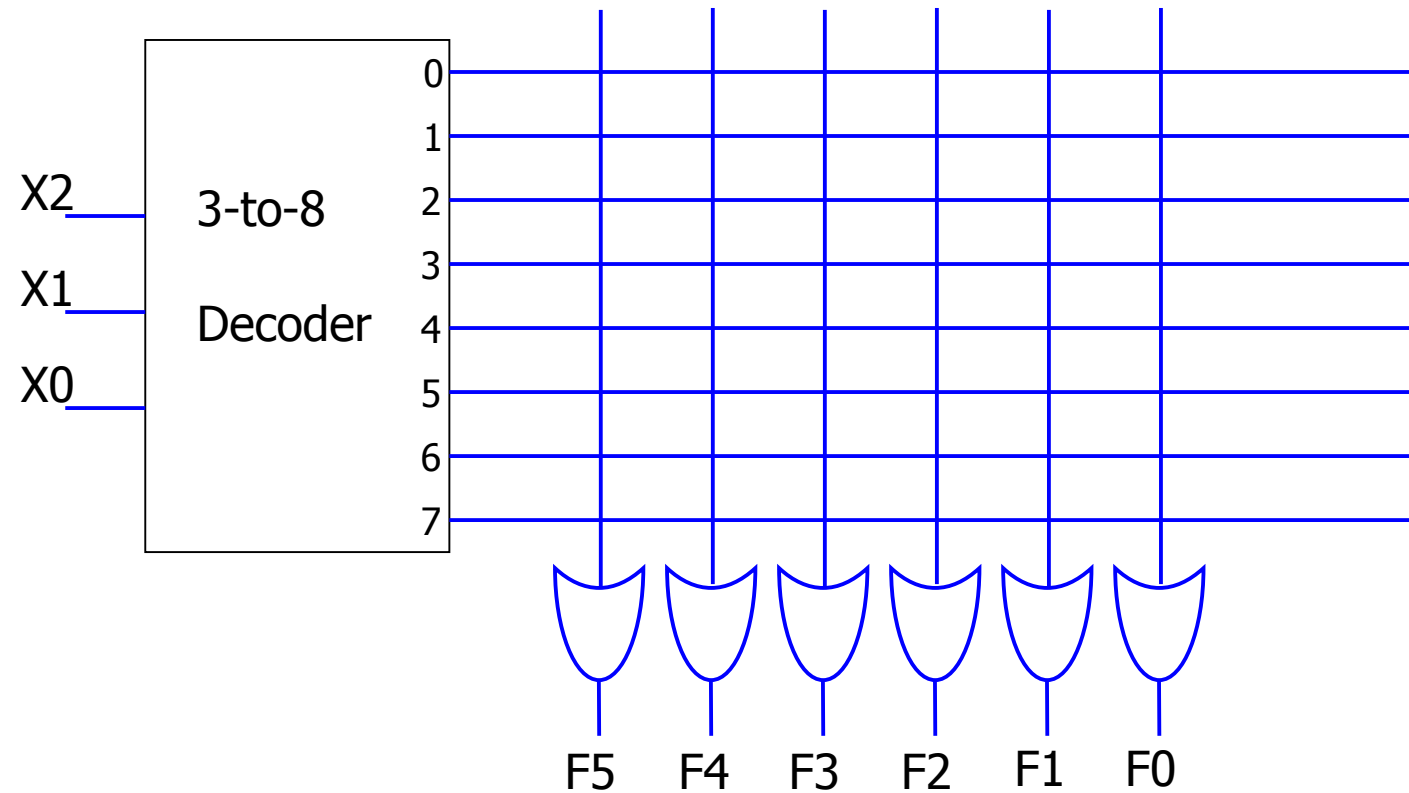


Output contains 6 bits

Problem 1. Read Only Memory (ROM)

Use ROM to store the following function.

X	$F(X)=X^2+3$
000	
001	
010	
011	
100	
101	
110	
111	

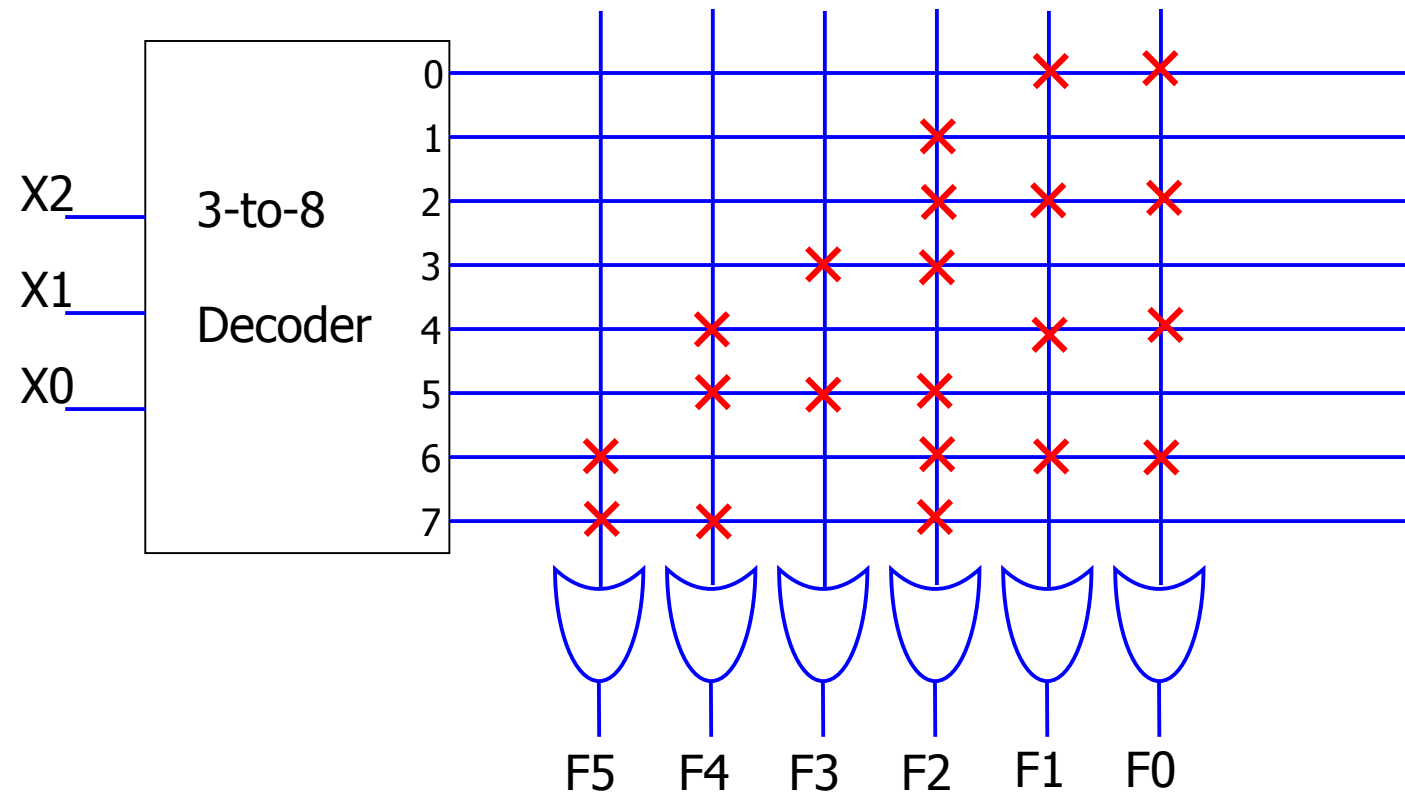


Output contains 6 bits

Solution 1. Read Only Memory (ROM)

Use ROM to store the following function.

X	$F(X)=X^2+3$
000	000011
001	000100
010	000111
011	001100
100	010011
101	011100
110	100111
111	110100



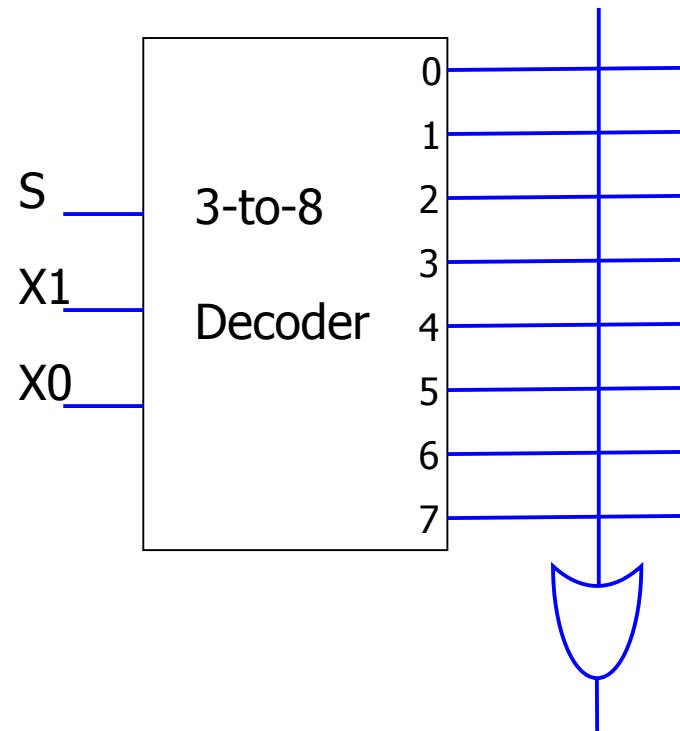
Output contains 6 bits

Problem 2. Read Only Memory (ROM)

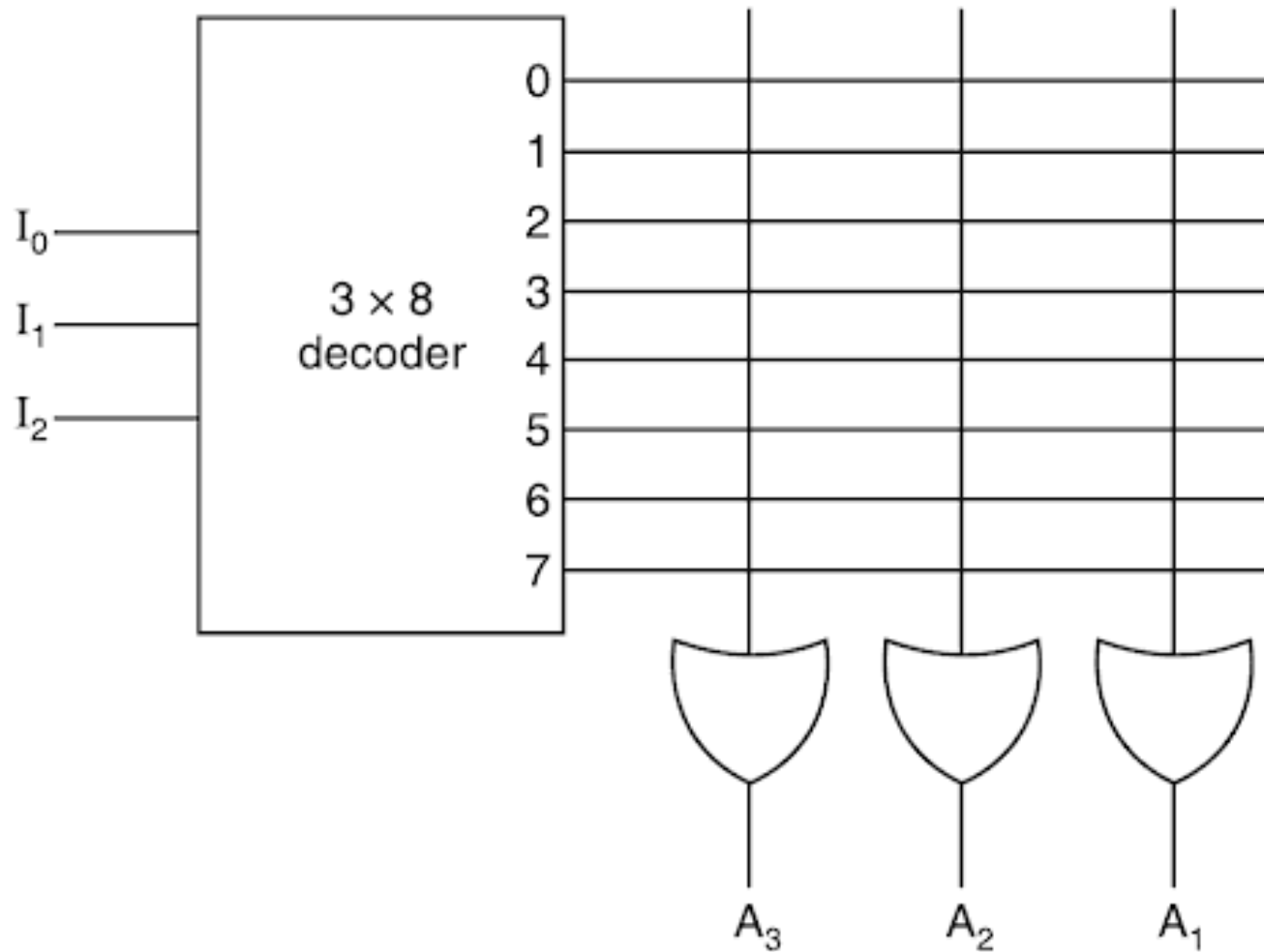
Use ROM to implement a 2-to-1 MUX

What is the size of ROM that is needed to implement a 4-to-1 MUX?

What is the size of ROM that is needed to implement a 8-to-1 MUX?



Problem 3. Read Only Memory (ROM)

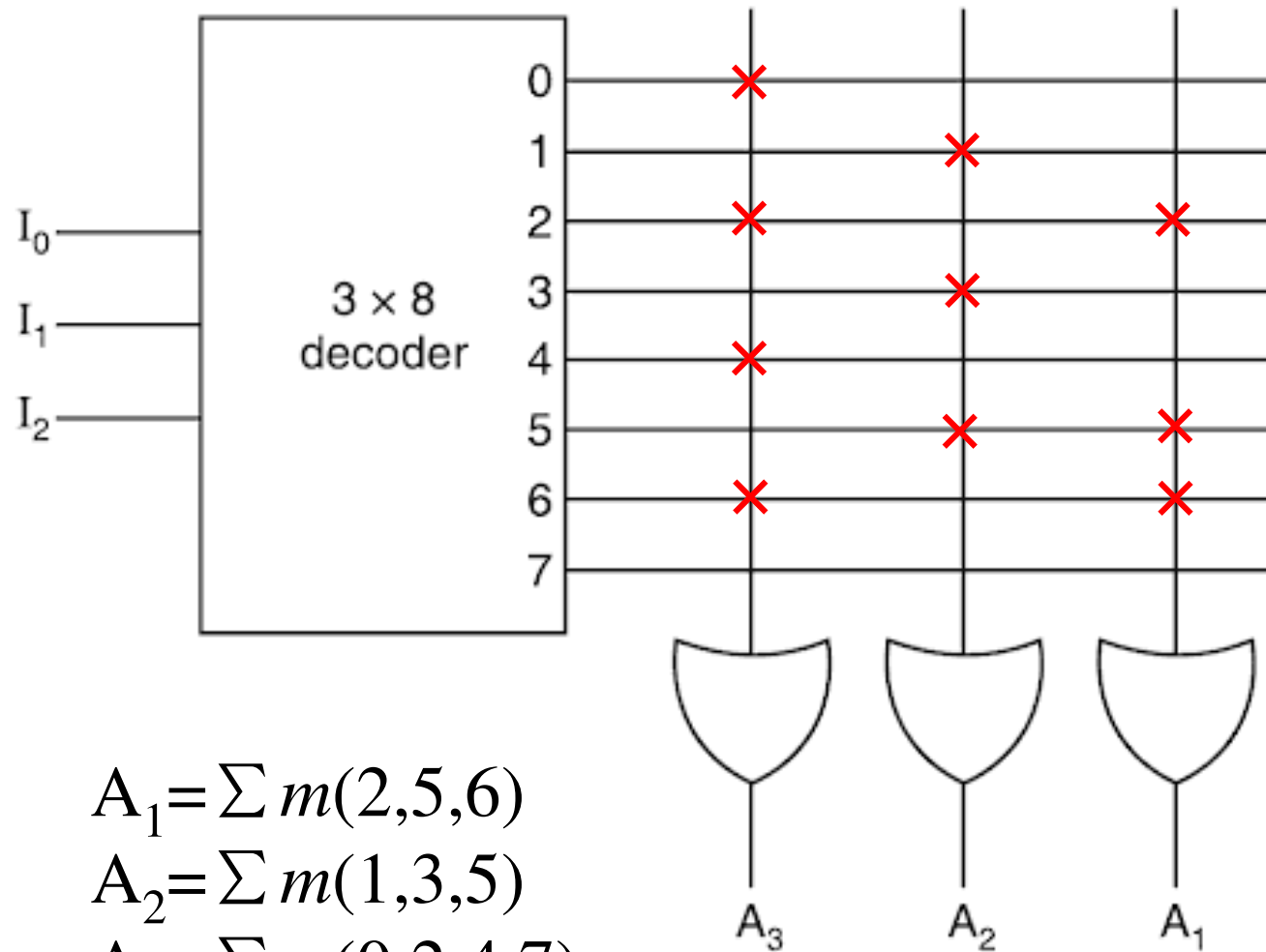


$$A_1 = \sum m(2, 5, 6)$$

$$A_2 = \sum m(1, 3, 5)$$

$$A_3 = \sum m(0, 2, 4, 7)$$

Solution 3. Read Only Memory (ROM)



$$A_1 = \sum m(2, 5, 6)$$

$$A_2 = \sum m(1, 3, 5)$$

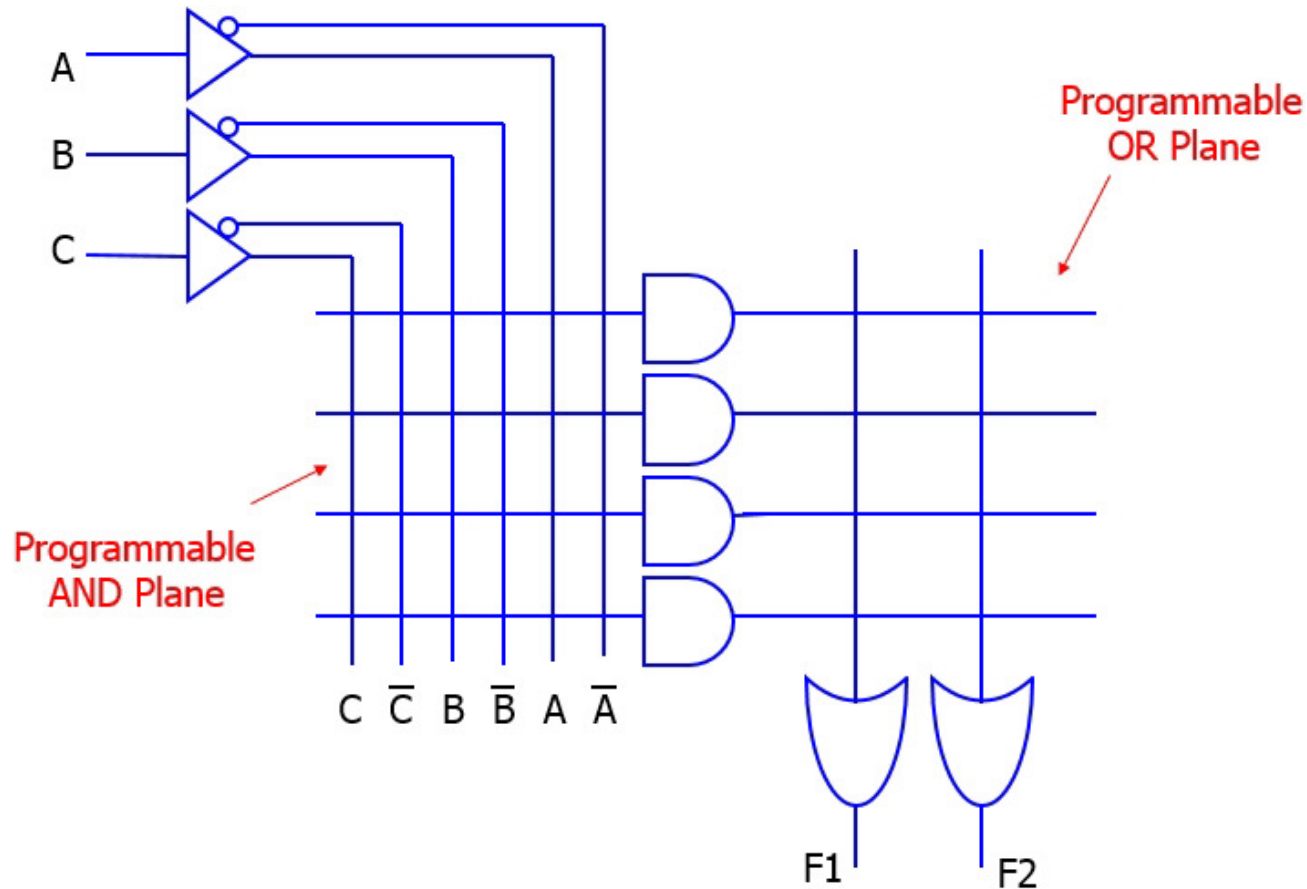
$$A_3 = \sum m(0, 2, 4, 6)$$

	I_2	I_1	I_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Problem 4. Programmable Logic Array (PLA)

$$F_1(A, B, C) = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

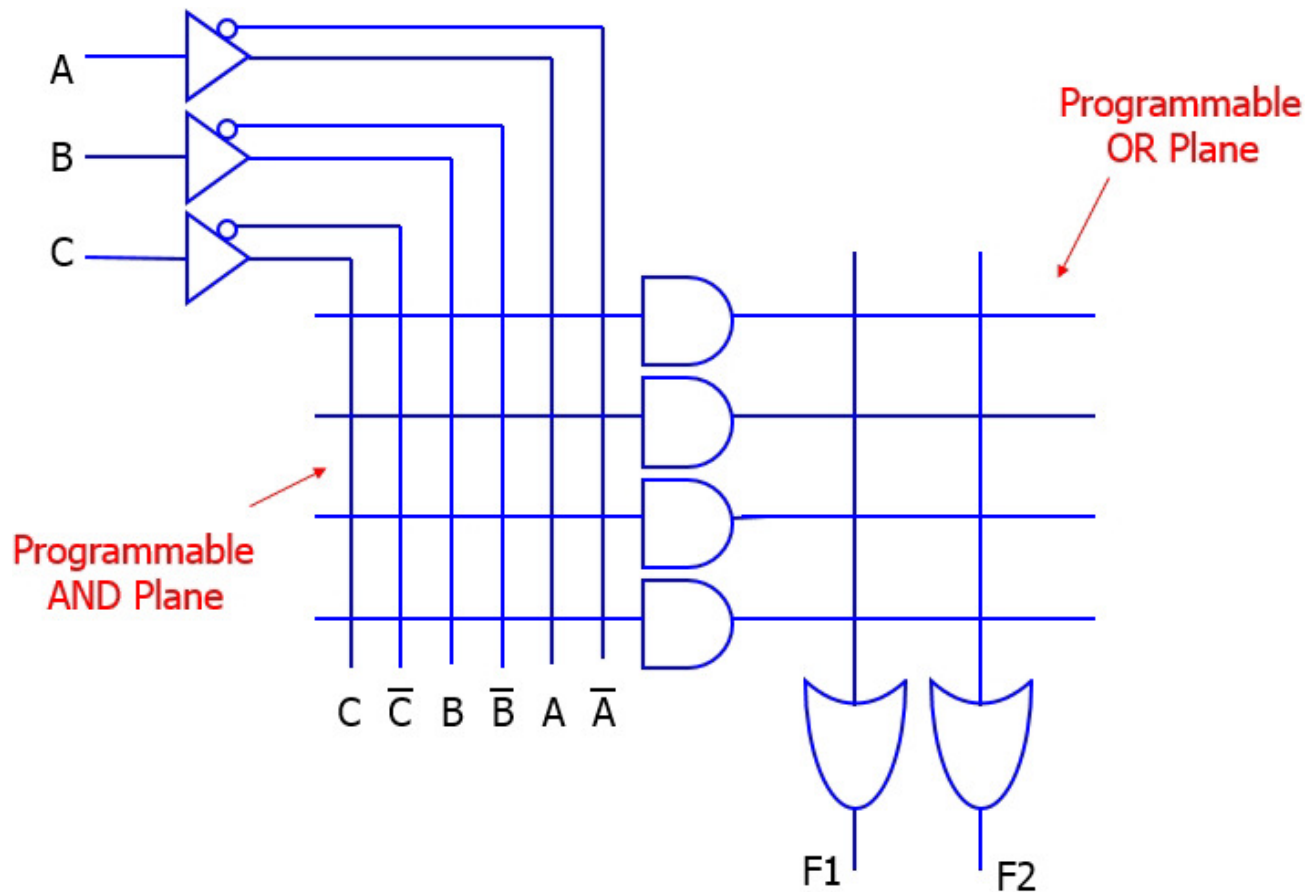
$$F_2(A, B, C) = \overline{A}BC + \overline{A}B\overline{C} + ABC + A\overline{B}\overline{C}$$



Solution 4. Programmable Logic Array (PLA)

$$F_1(A, B, C) = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

$$F_2(A, B, C) = \overline{A}BC + \overline{A}B\overline{C} + ABC + AB\overline{C}$$



		BC			
		00	01	11	10
A	0	1	1		1
	1	1			

$$F_1(A, B, C) = \overline{B}\overline{C} + \overline{A}\overline{B} + \overline{A}\overline{C}$$

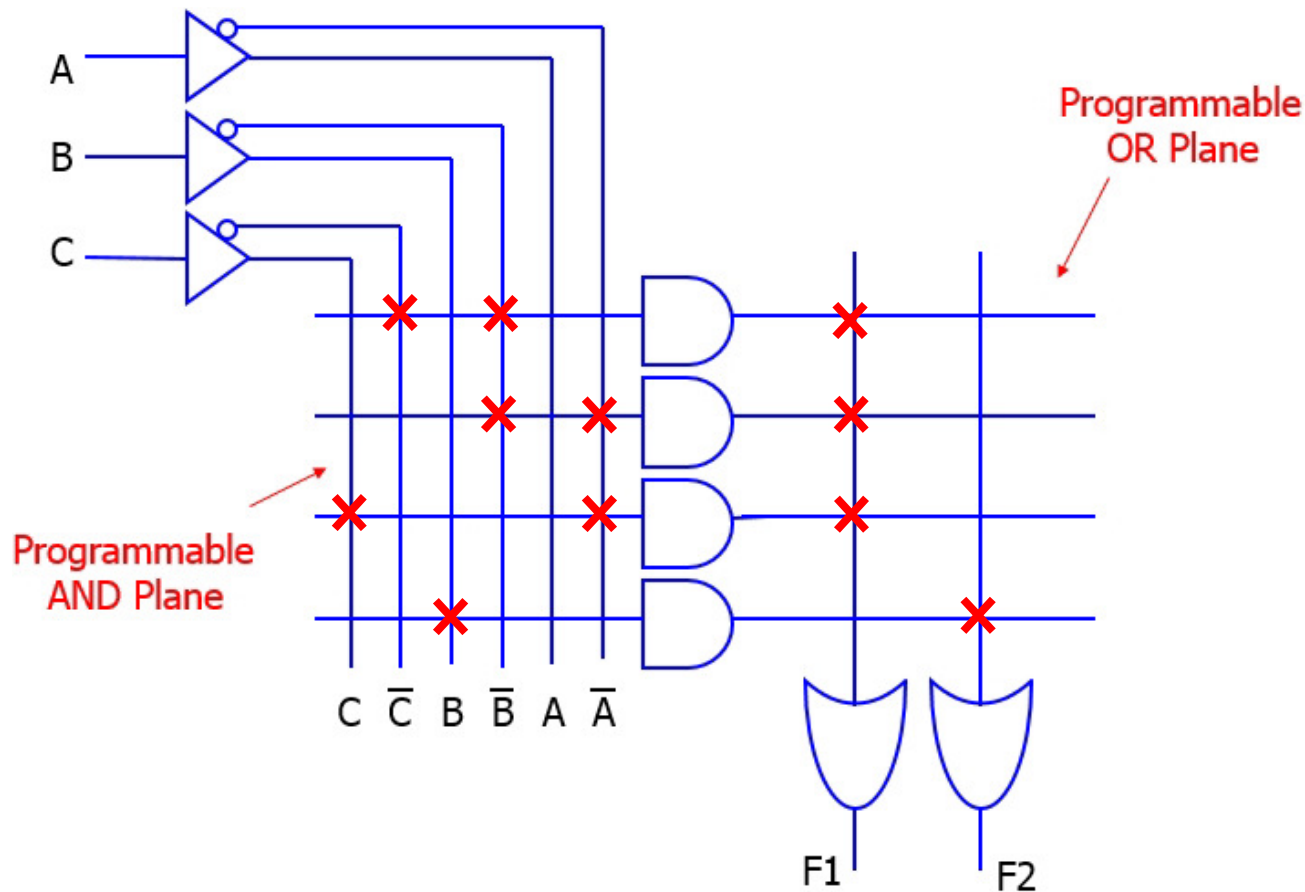
		BC			
		00	01	11	10
A	0			1	1
	1			1	1

$$F_2(A, B, C) = B$$

Solution 4. Programmable Logic Array (PLA)

$$F_1(A, B, C) = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

$$F_2(A, B, C) = \overline{A}BC + \overline{A}B\overline{C} + ABC + ABC\overline{C}$$



		BC			
		00	01	11	10
A	0	1	1		1
	1	1			

$$F_1(A, B, C) = \overline{B}\overline{C} + \overline{A}\overline{B} + \overline{A}\overline{C}$$

		BC			
		00	01	11	10
A	0			1	1
	1			1	1

$$F_2(A, B, C) = B$$

Problem 5. Minimum-row PLA

Find a minimum-row PLA to implement the following two functions

$$f_1(A, B, C, D) = cd + ad + a'bc'd'$$

$$f_2(A, B, C, D) = bc'd' + ac' + ad'$$

Solution 5. Minimum-row PLA

$$f_1(A, B, C, D) = cd + ad + a'bc'd'$$

$$f_2(A, B, C, D) = bc'd' + ac' + ad'$$

AB		00	01	11	10
CD	00	0	1	0	0
	01	0	0	1	1
	11	1	1	1	1
	10	0	0	0	0

f_1

AB		00	01	11	10
CD	00	0	1	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	1

f_2

Product Term	Inputs				Outputs	
	A	B	C	D	f_1	f_2
$A'BC'D'$	0	1	0	0	1	1
$AC'D$	1	-	0	1	1	1
CD	-	-	1	1	1	0
AD'	1	-	-	0	0	1

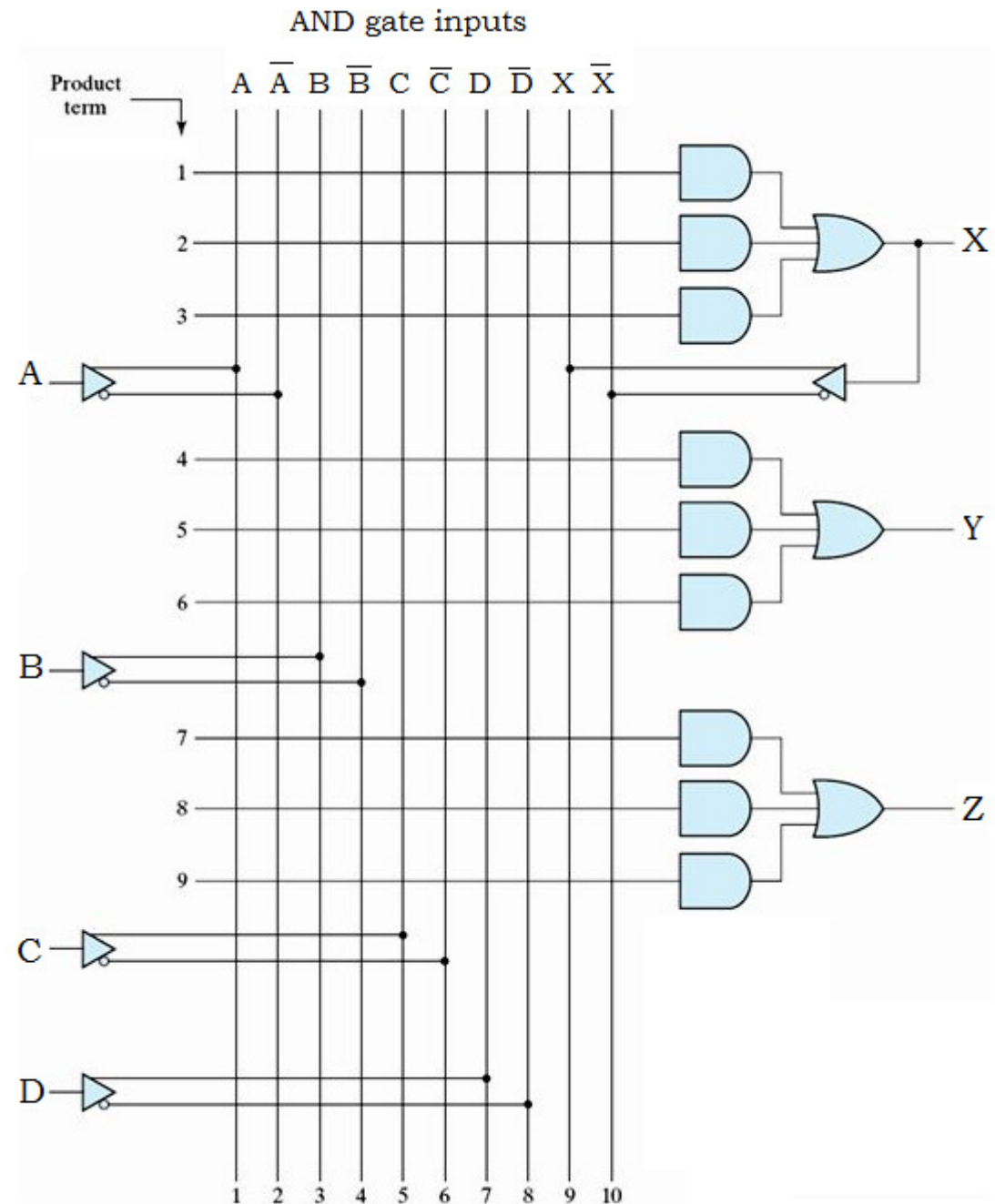
Problem 6. Programmable Array Logic (PAL)

Implement the following
Boolean functions using
PAL

$$X = ABC' + A'B'CD'$$

$$Y = A'B + CD + B'D'$$

$$Z = ABC' + A'B'CD' + AC'D' + A'B'C'D$$



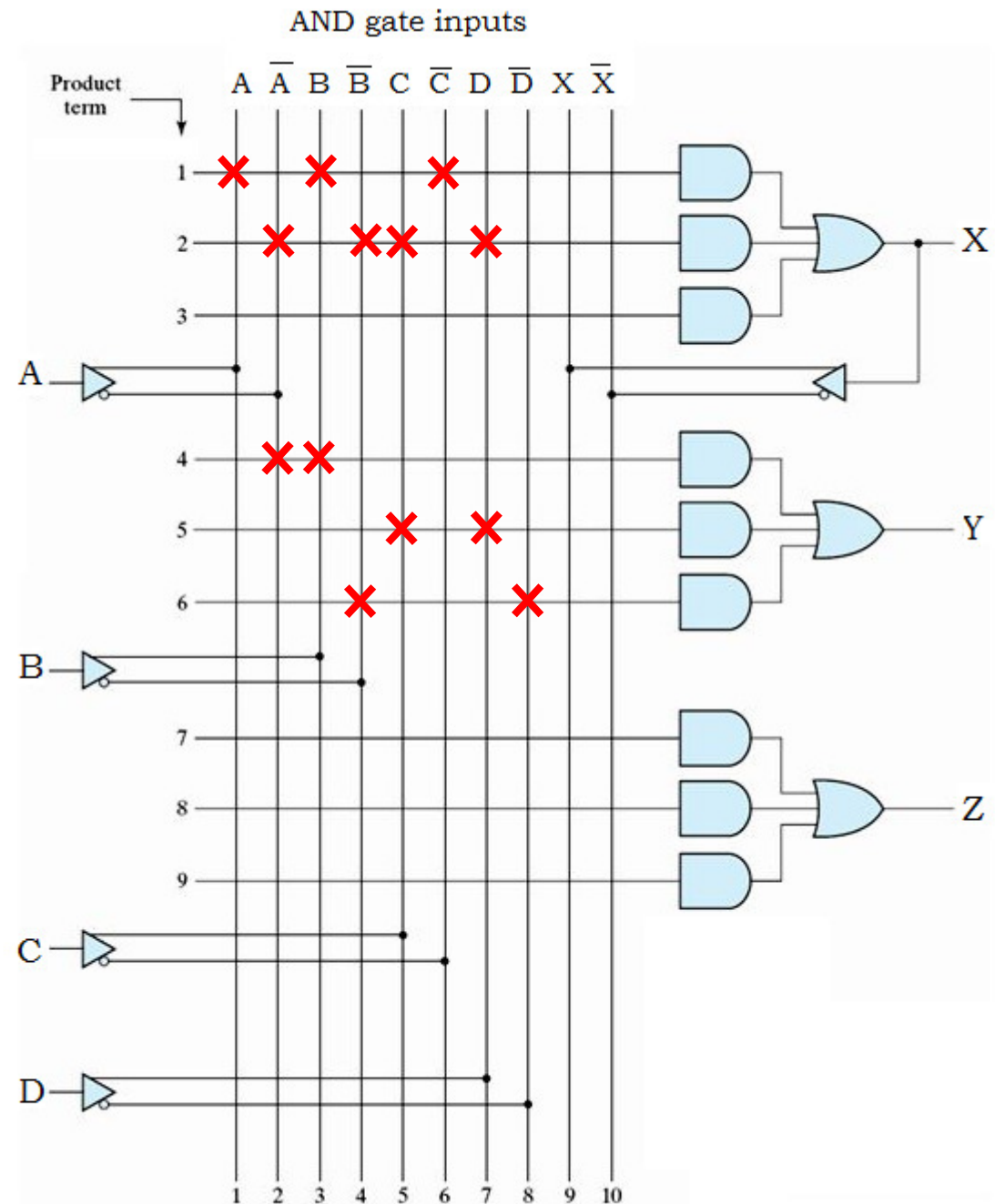
Solution 6. Programmable Array Logic (PAL)

Implement the following
Boolean functions using
PAL

$$X = ABC' + A'B'CD'$$

$$Y = A'B + CD + B'D'$$

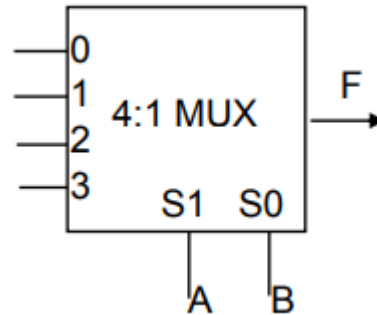
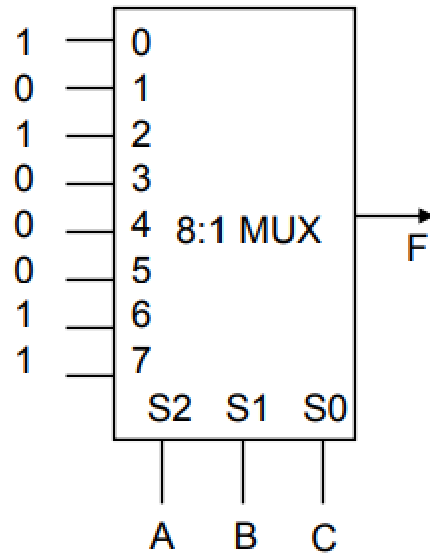
$$Z = ABC' + A'B'CD' + AC'D' + A'B'C'D$$



Problem 7. Multiplexers as general-purpose logic

Use 4-to1 MUX to implement

$$F(A, B, C) = m_0 + m_2 + m_6 + m_7 = A'B'C' + A'BC' + ABC' + ABC$$

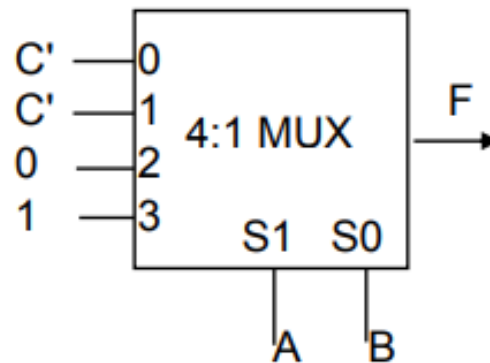


Solution 7. Multiplexers as general-purpose logic

Use 4-to1 MUX to implement

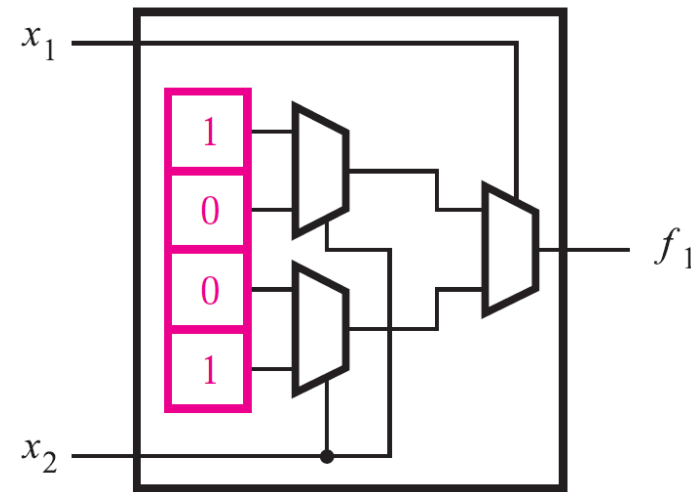
$$F(A, B, C) = m_0 + m_2 + m_6 + m_7 = A'B'C' + A'BC' + ABC' + ABC$$

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



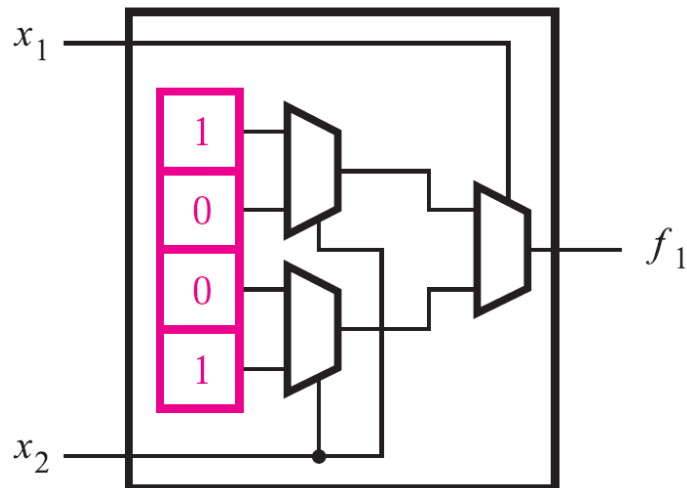
Problem 8. Multiplexers and FPGA

Build the truth table for function $f_1(x_1, x_2)$



Solution 8. Multiplexers and FPGA

Build the truth table for function $f_1(x_1, x_2)$

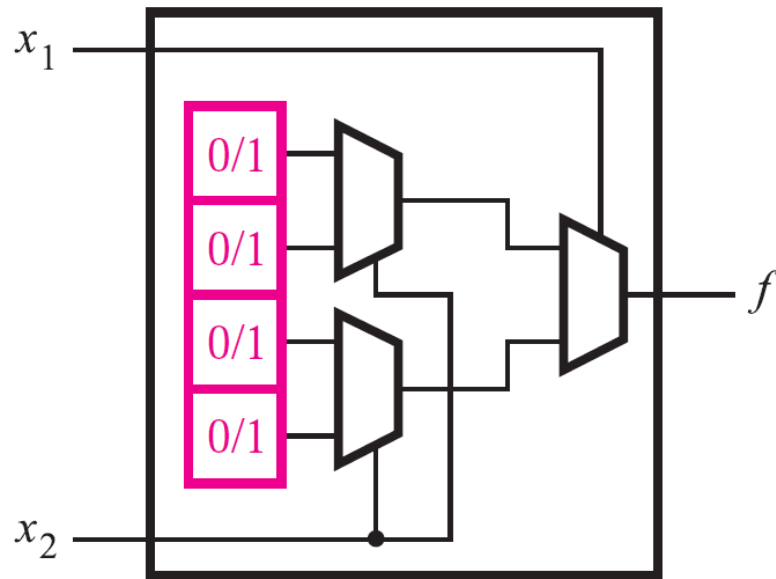


x_1	x_2	f_1
0	0	1
0	1	0
1	0	0
1	1	1

$$f_1(x_1, x_2) = x_1 x_2 + \bar{x}_1 \bar{x}_2$$

Problem 9. Multiplexers and FPGA

Build the LUT tables for $f_1(x_1, x_2) = x_1x_2$ and $f_2(x_1, x_2) = \bar{x}_1x_2$



Solution 9. Multiplexers and FPGA

Build the LUT tables for $f_1(x_1, x_2) = x_1x_2$ and $f_2(x_1, x_2) = \bar{x}_1x_2$

