Question 3

a) Given the 2-cube A = 0x0x, how many other 2-cubes, B, exist such that A * B = 0101?

Answer: ___1 (X1X1)_____

b) Is the following statement true or false: In an n-variable Boolean space, the number of (n-1)-cubes that exist is equal to 2n.

Answer: ___True_____

c) Given the following implicants for a function, f, list the prime implicants.

f = {1xx1x10, 11x0x10, 10x0x10, 111x1x0}

Prime implicants: _____1XXXX10, 111X1X0______

____

11X0X10 * 10X0X10 = 1XX0X10 (second and third implicants)
1XX1X10 * 1XX0X10 = 1XXXX10 (first implicant with previous result)

d) Given

$C^{k+1} = \{0x00, x10x, 11xx, 1000\}$
$G^{k+2} = \{0100, x100, x000, 110x, 1x00\}$

Evaluate the following:

$C^{k+2} = \{C^{k+1} \cup G^{k+2} – (\text{redundant cubes})\}$

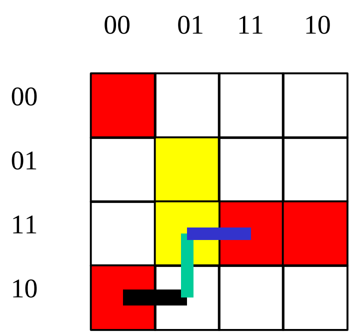
C^{k+2} = ___0X00, X10X, 11XX, X000, 1X00

The redundant cubes are
$0100 \subset 0X00$
$X100 \subset X10X$
$1000 \subset 1X00$
$110X \subset 11XX$

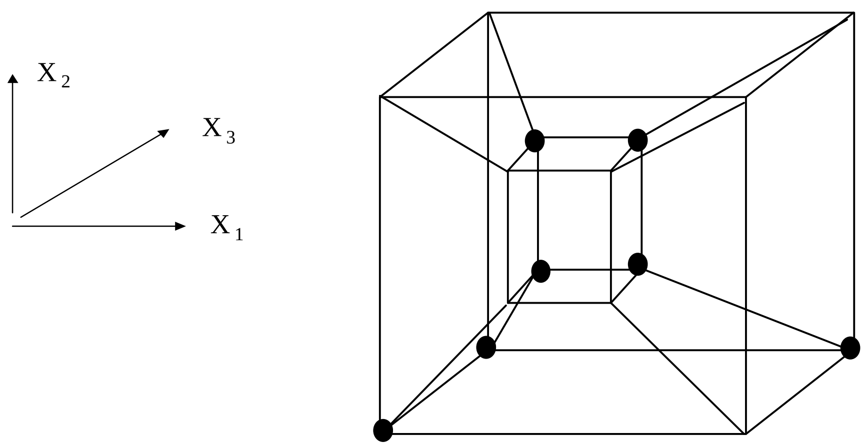
e) The prime implicants for a function f are {00X0, 01X1, 011X, 0X10, X111, 1X11}. By using a K-map, determine the essential prime implicants for the function

Answer:________00X0, 1X11, 01X1

Essential prime implicants are shaded. Non essential are outlined as “bars”.



f) The figure below represents the Boolean space (x_1,x_2,x_3, x_4). Coordinates are defined as in class, as indicated below (x_4 is 1 on outer cube):



A function f is represented by the vertices shown in bold

g) Write an expression in cubical representation that represents a minimal sum-of- products implementation:

Answer: ____X01X, XX10, 00X1________________________________

h) Consider a 7-variable Boolean space. One vertex in this space is 1111000. If you could draw a figure representing the 7-variable space, which other vertices would be connected to by edges to the vertex 1111000?

Answer:_______0111000, 1011000, 1101000, 1110000, 1111100, 1111010, 1111001

Question 4

Figure 1 shows the array implementation of a 4x4 bit array multiplier.

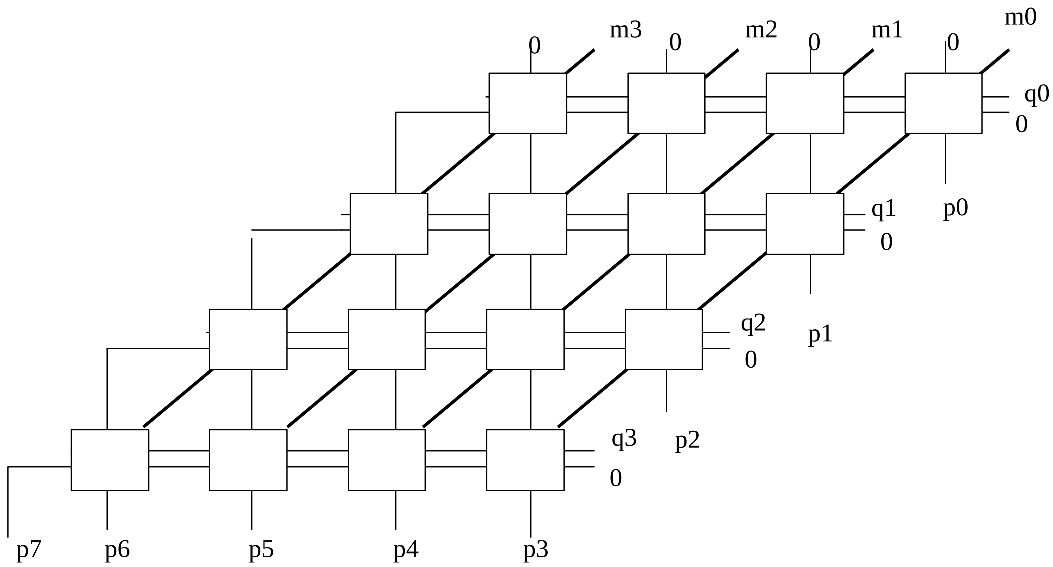
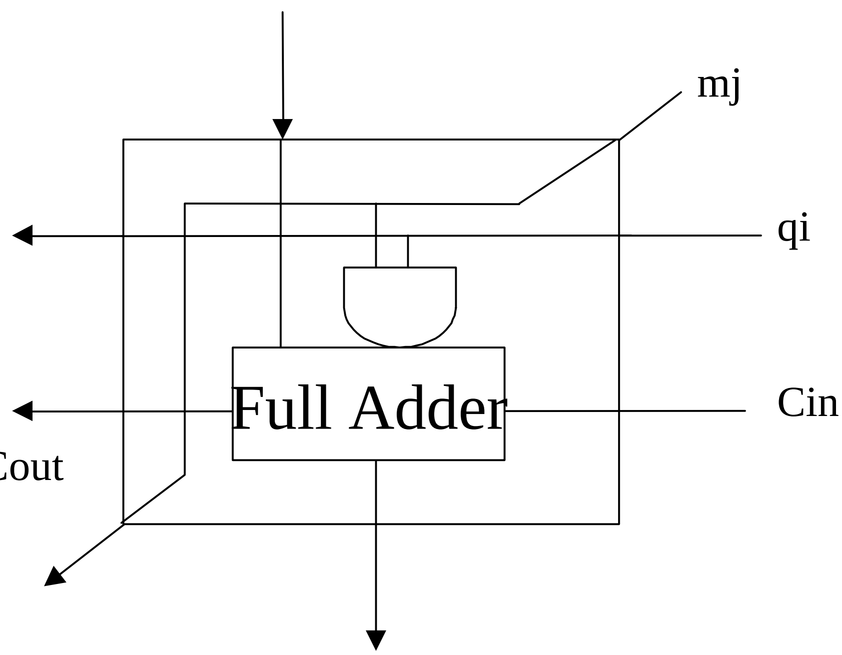


Figure 1 :4x4bit array multiplier

a) Draw the internal part of the multiplier cell.



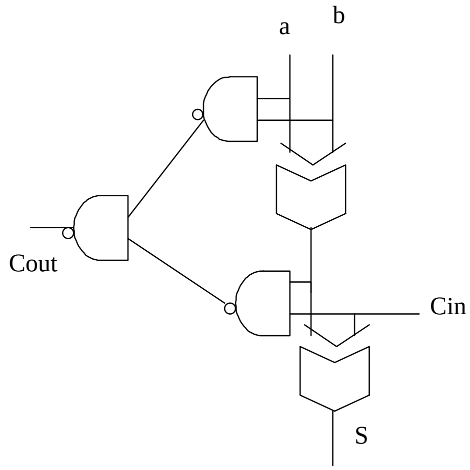
Assume that the different gates have the propagation delays defined below.

NOT:	0.4 ns
Nand2:	0.6 ns (2-input Nand gate)
Xor2	1 ns

Full Adders are implemented according to the following equations

$$s = (a \oplus b) \oplus c_{in}$$
$$c_{out} = ab + c_{in}(a \oplus b)$$

b) Draw the Full Adder by using the minimum number of Xor2, Nand2 and NOTgates.



c) What are the propagation delays for the full adder? (Fill up the table below)

INPUT		OUTPUT	PROPAGATION DELAY
a or b	TO	S	2 x 1 = 2 ns
a or b	TO	C_{out}	2 x 0.6 + 1 = 2.2 ns
C_{in}	TO	S	1 ns
C_{in}	TO	C_{out}	2 x 0.6 = 1.2 ns

d) Compute the worst-case propagation delay (in ns) from input to output for this array multiplier. This forms the *critical path*. Show details of your work and **highlight the critical path** in the figure below.

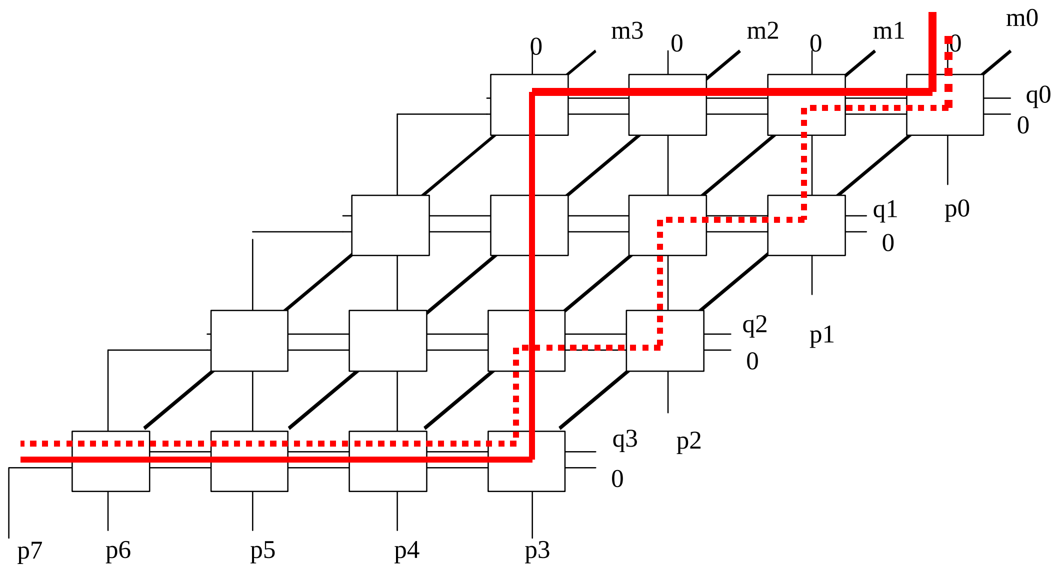
The propagation delay includes the AND gate m0q0 and one path to reach p6 or p7.
The propagation delays include

Horizontal propagation th = 1.2 ns
Horizontal to vertical propagation thv = 1 ns
Vertical propagation tv = 2 ns
Vertical to horizontal propagation tvh= 2.2 ns

As th > thv and tvh > tv, the critical path is for p7 (not p6)
The two "essential" critical paths are

The bold one : tAND + 2 tvh + 5 th + 2 tv + 1 thv = 1 + 4.4 + 6 + 4 + 1 = 16.4 ns
The dash one: tAND + 4 tvh + 3 thv + 3 th = 1 + 8.8 + 3 + 3.6 = 16.4 ns

All other paths are identical to one of the two essential ones.



Worst case propagation delay: 16.4 ns