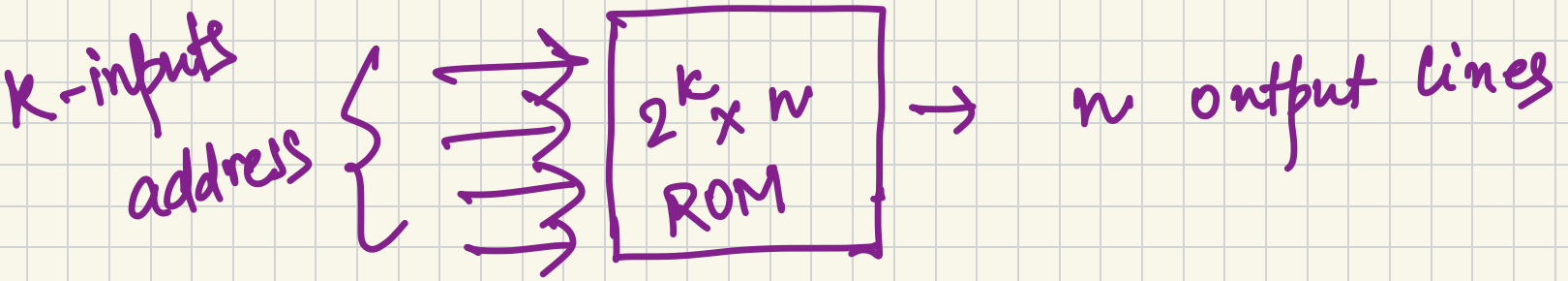
ACOL215

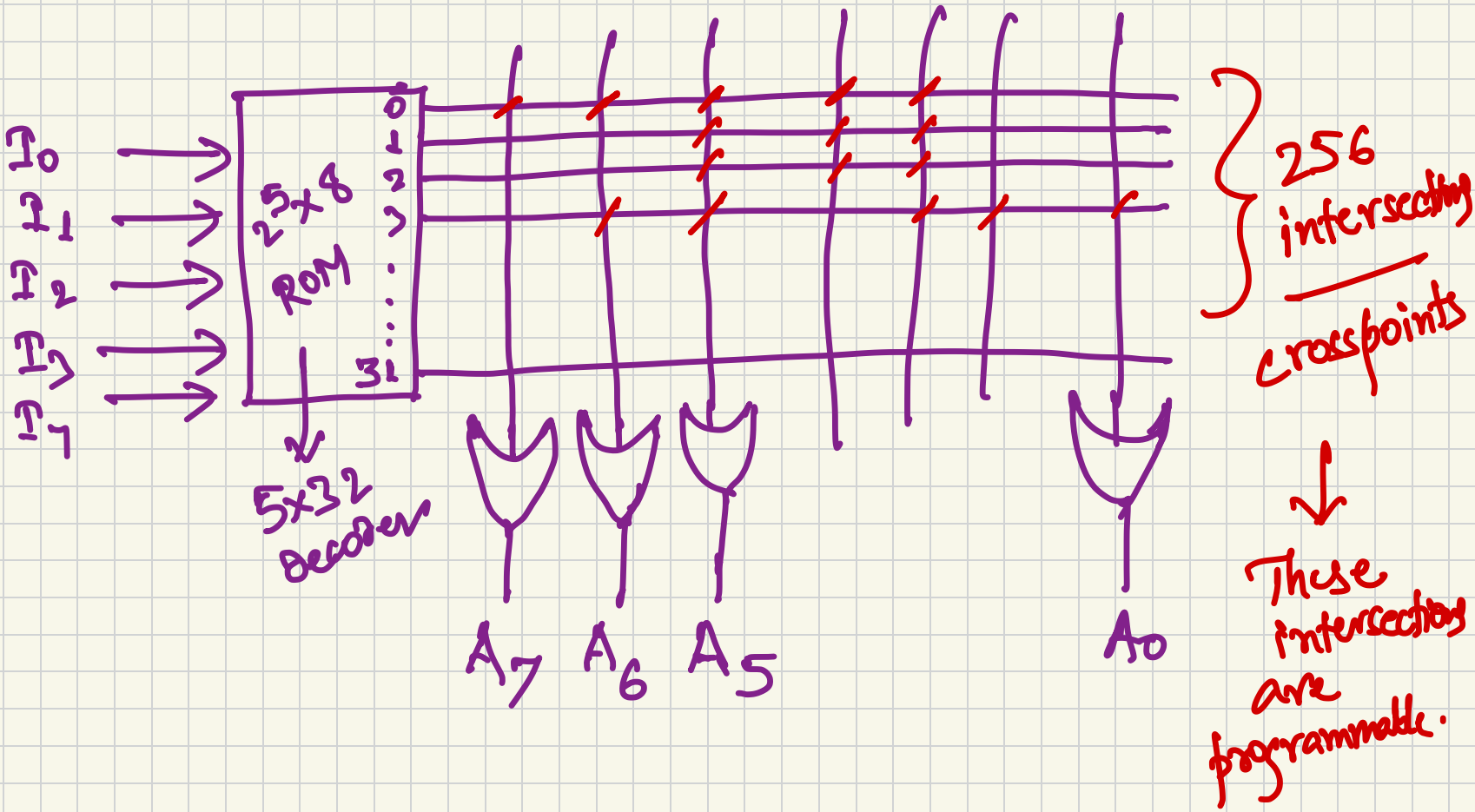
(9th Dec.)

Read Only Memory (ROM)



$2^5 \times 8$ ROM
32×8 ROM

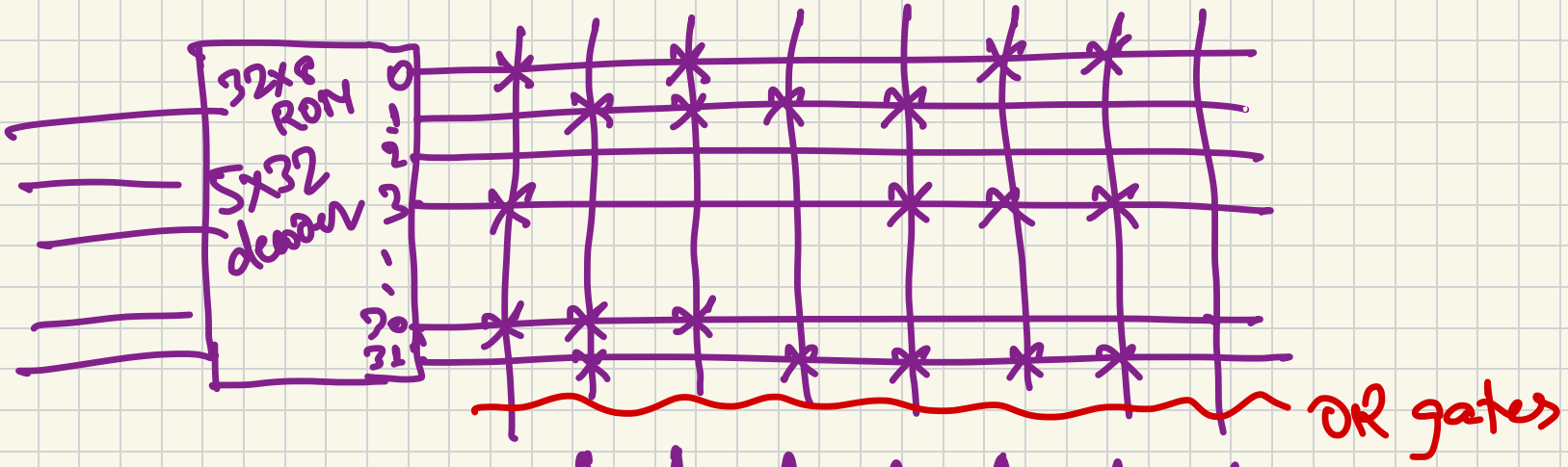
} next slide


I_0
I_1
I_2
I_3
I_4
I_5
I_6
I_7
$2^5 \times 8$
ROM
0
1
2
3
31
5×32
decoder
A_7
A_6
A_5
A_0
256
intersections
crosspoints
These
intersections
are
programmable.

A programmable connection is like a switch that can be altered to be in a closed state (the lines are connected) or an open state (the lines are disconnected)

The internal storage of a ROM can be specified by a truth table

Inputs					Outputs							
I_4	I_3	I_2	I_1	I_0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
0	0	0	0	0	1	0	1	0	0	1	1	0
0	0	0	0	1	0	1	1	1	1	0	0	0
0	0	0	1	0	1	0	0	0	1	1	1	0
0	0	0	1	1								
.	.	.	.	.								
1	1	1	1	0	1	1	1	0	0	0	0	0
1	1	1	1	1	0	1	0	1	1	1	1	0



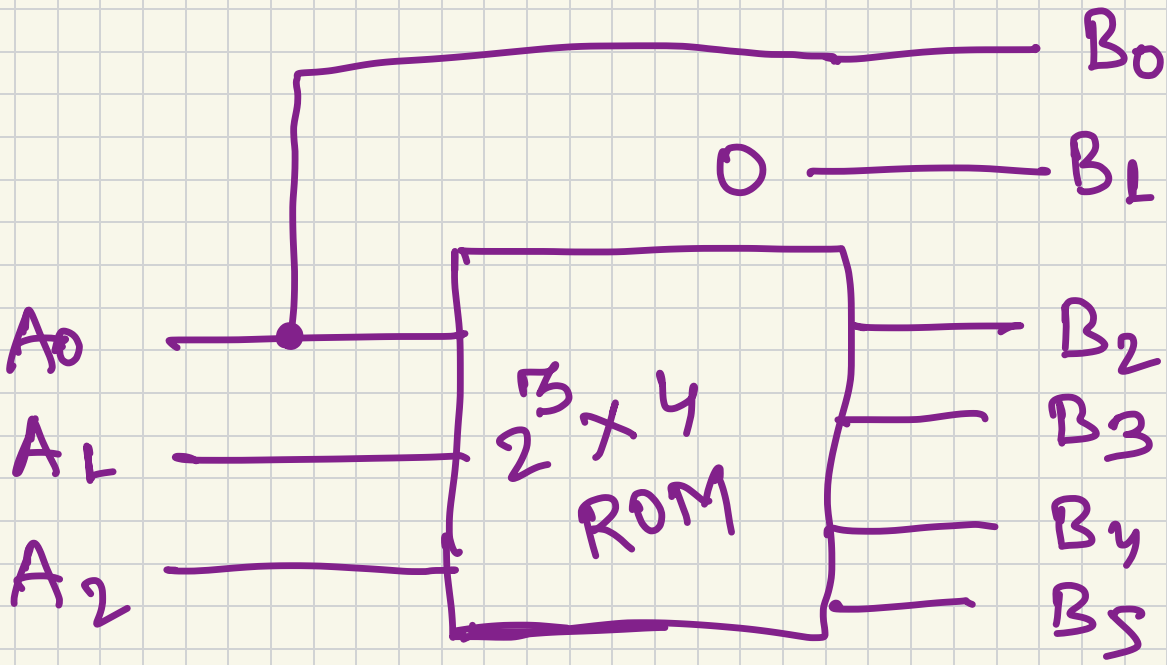
We can also consider each output terminal as a Boolean function expressed as a sum of minterms.

$$A_7(I_4, I_3, I_2, I_1, I_0) = \sum(0, 3, \ldots, 30)$$
$$A_2(\ldots) = \sum(0, 3, \ldots, 31)$$

Exercise Design a combinational circuit using a ROM — the circuit accepts a three bit number and outputs a binary number that is equal to the square of the input.

A_2	A_1	A_0	Square
0	0	0	0
0	0	1	1
0	1	0	4
0	1	1	9
1	0	0	16
1	0	1	25
1	1	0	36
1	1	1	49

B_5	B_4	B_3	B_2	B_1	B_0
0	0	0	0	0	0
0	0	0	0	0	1
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	1	0	0
1	1	0	0	0	1



ROM truth table

$A_2 A_1 A_0$	$D_5 B_4 B_3 B_2$

Mask programming

→ during the fabrication process

PROM Programmable ROM

→ comes with all fuses intact

→ "blow" some of them by applying high voltage

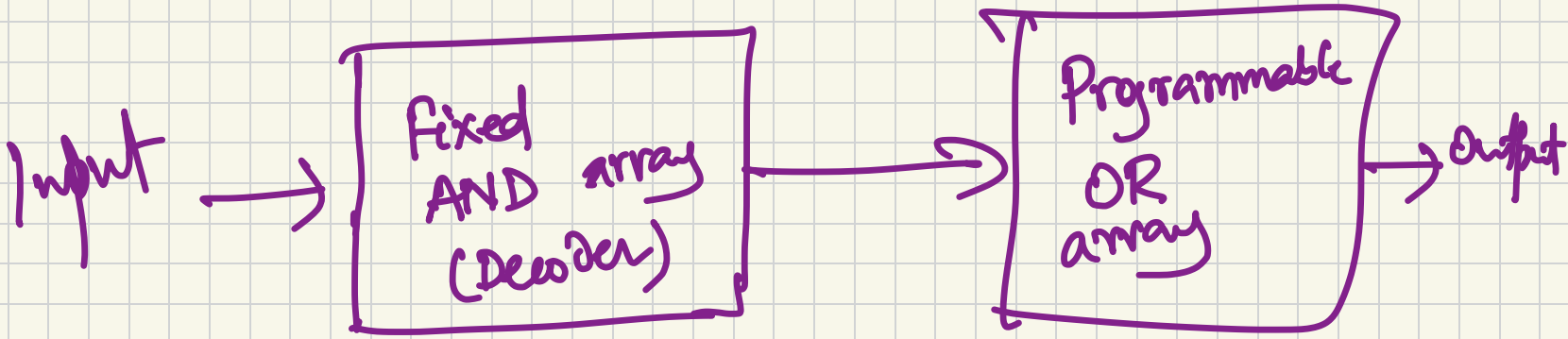
→ once programmed, this is permanent

Erasable PROM

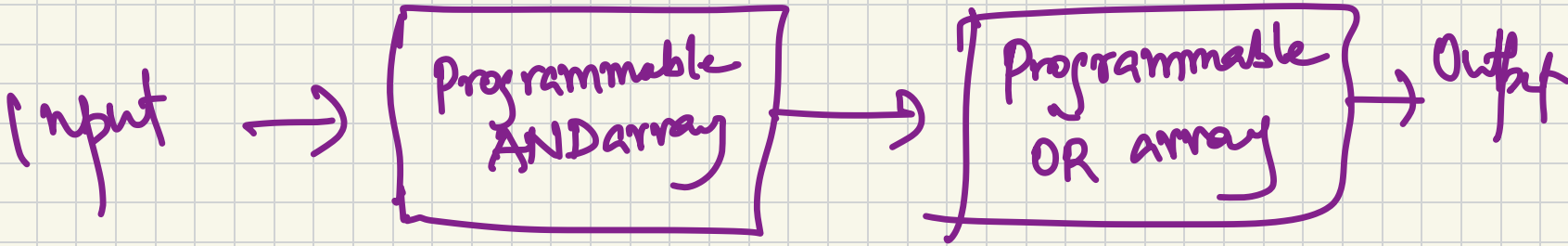
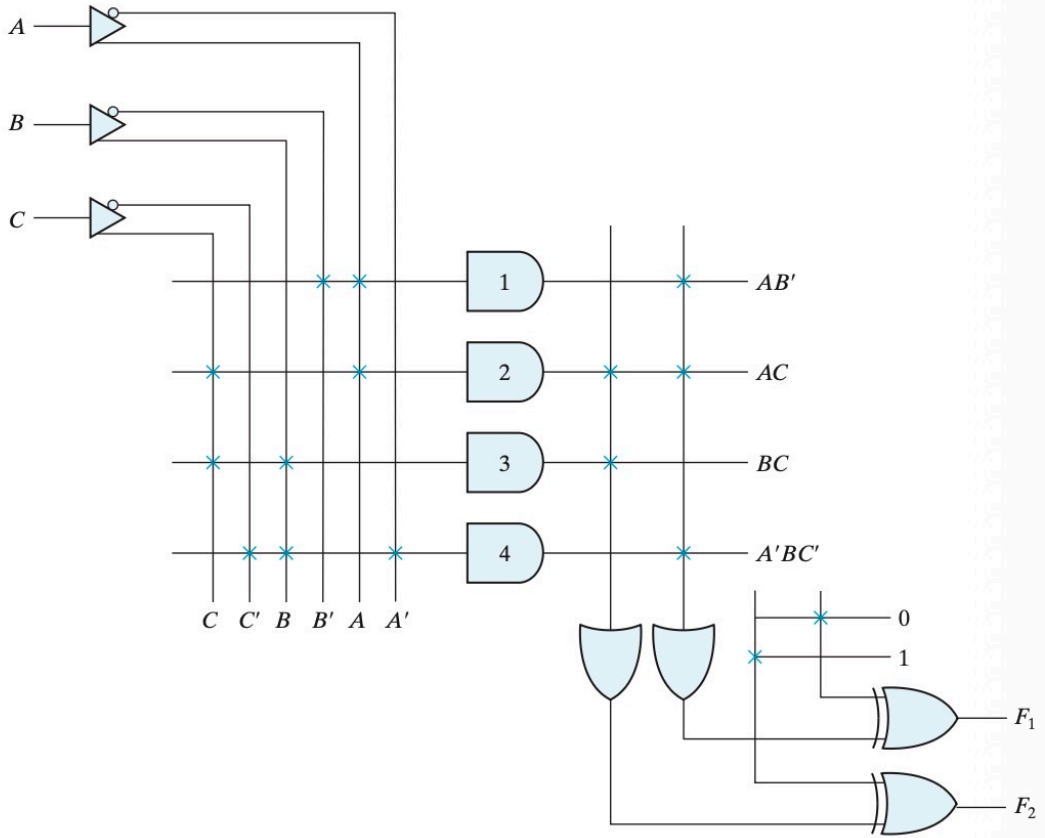
→ can be restructured to the original state by placing it under u/v light

EEPROM

↳ restructuring can be done with electrical signals



(PROM) Programmable ROM



$$F_1 = AB' + AC + A'BC'$$
$$F_2 = (AC + BC)'$$