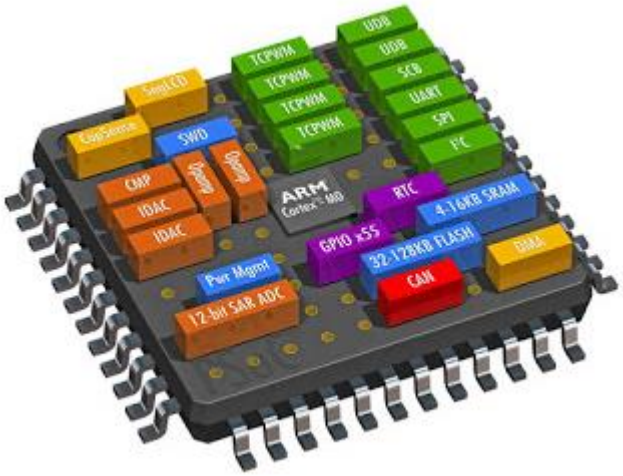


Binary Decoder

For circuit design

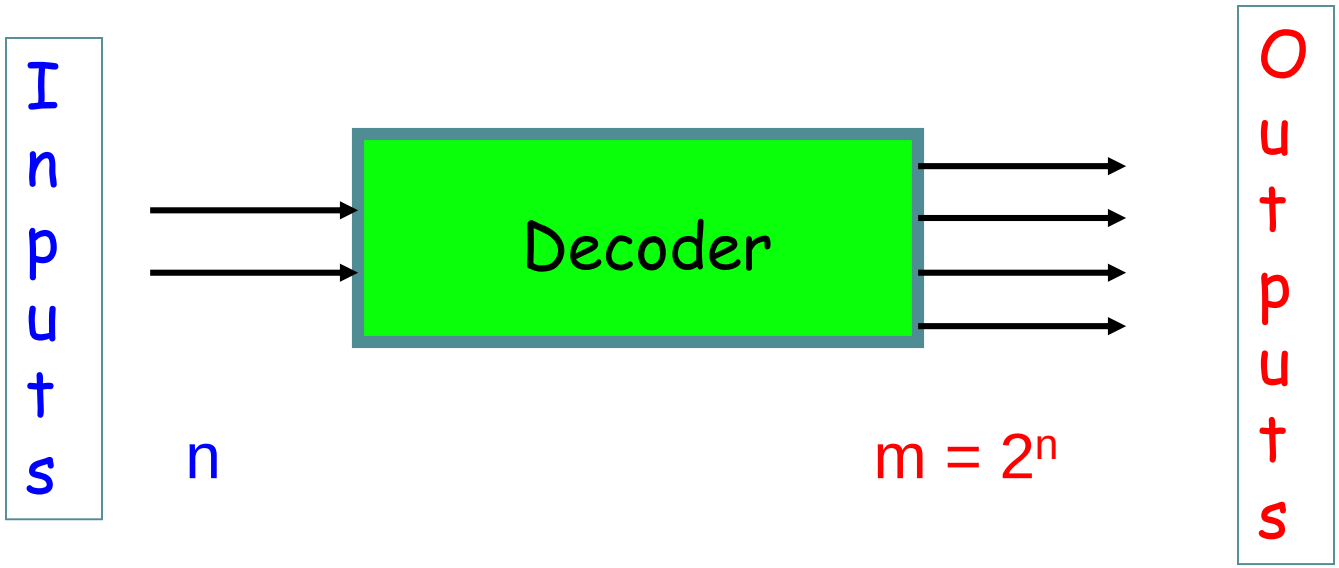
A decoder is a code converter

- A binary decoder converts binary information into a different binary form

Decoder logic circuit

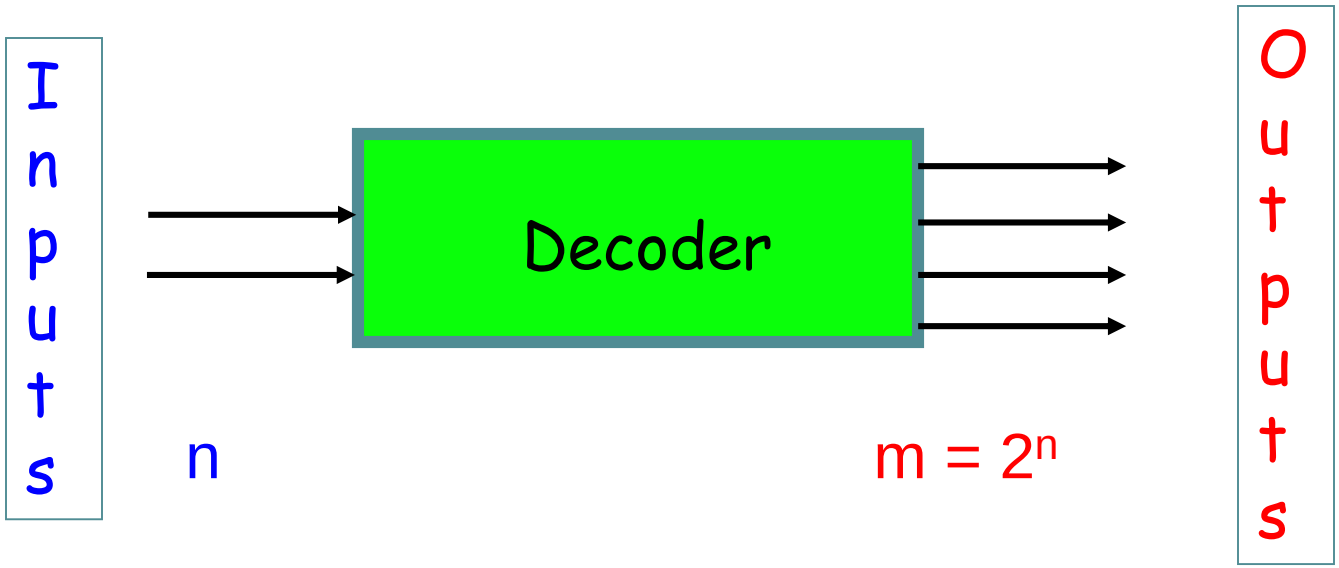
- A decoder is a special logic circuit that converts a n-bit binary input code into $m = 2^n$ output lines…
- … such that each output will be activated for only one possible combination of the inputs.

Decoder: Block diagram



n–m line decoder

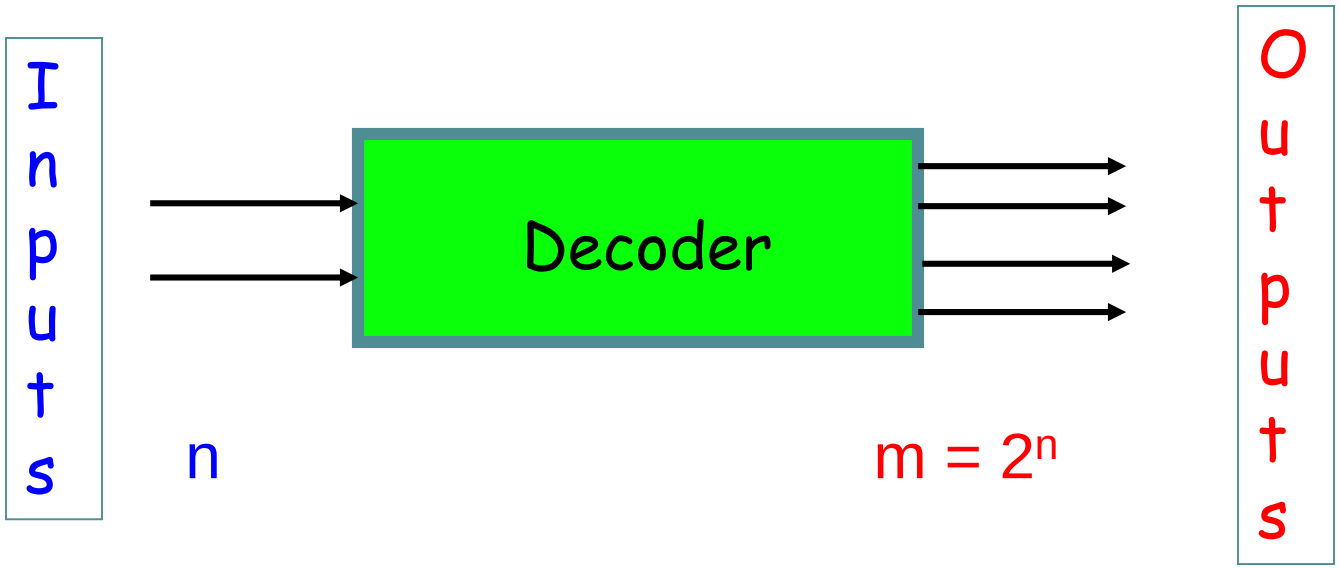
Decoder: Main Characteristic



Only one output is high (1) for each input code

n–m line decoder

Design a 2-4 line Decoder



Only one output is high (1) for each input code

Truth table

A_1	A_0	D_3	D_2	D_1	D_0
0	0				
0	1		?		
1	0				
1	1				

Only one output (diagonal) is high (1) for each input code

Truth table

A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

Only one output (diagonal) is high (1) for each input code

Logic equations

A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

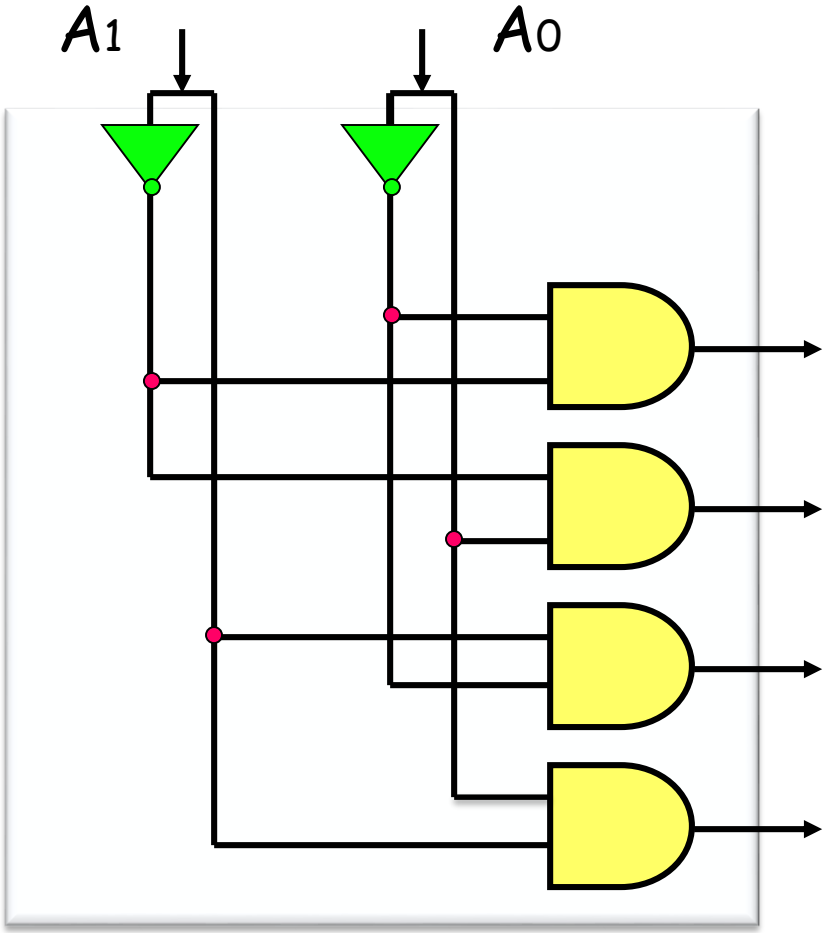
$D_0 = \overline{A_1}\ \overline{A_0}$

$D_1 = \overline{A_1}\ A_0$

$D_2 = A_1\ \overline{A_0}$

$D_3 = A_1\ A_0$

Logic diagram



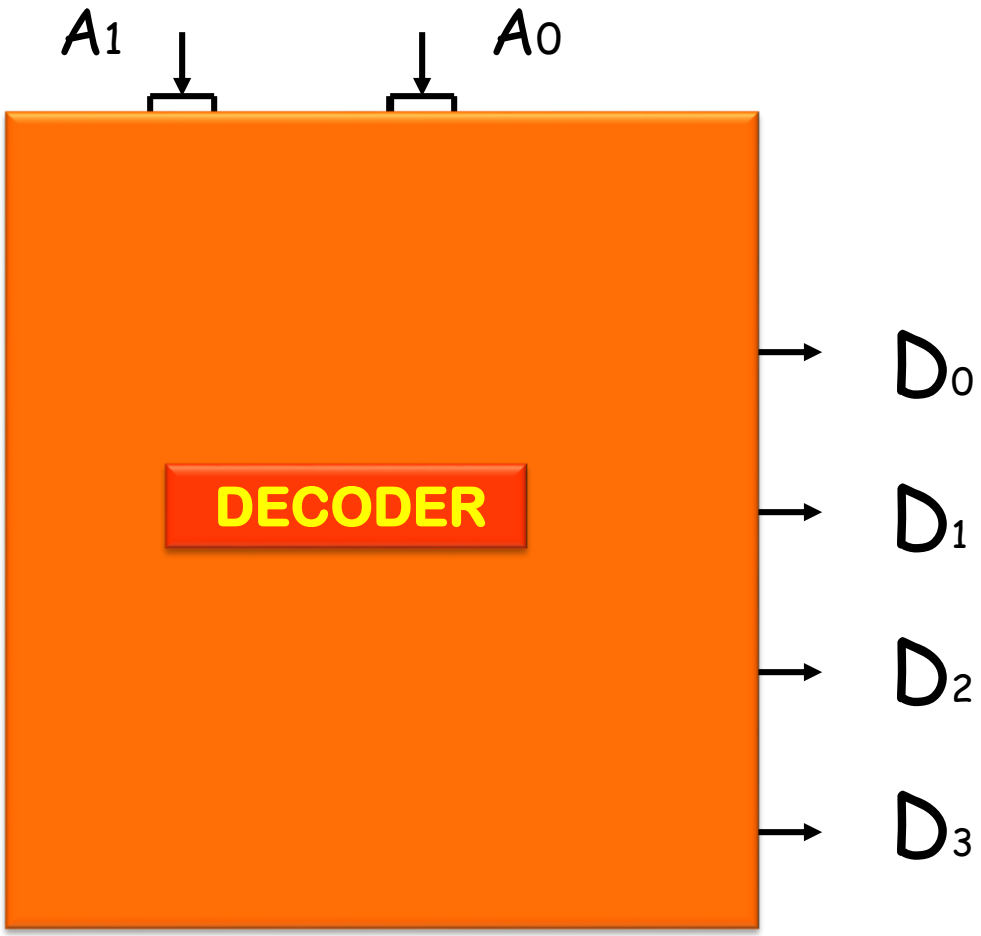
$D_0 = \overline{A_1}\ \overline{A_0}$

$D_1 = \overline{A_1}\ A_0$

$D_2 = A_1\ \overline{A_0}$

$D_3 = A_1\ A_0$

Logic diagram



Design a 3-8 line decoder

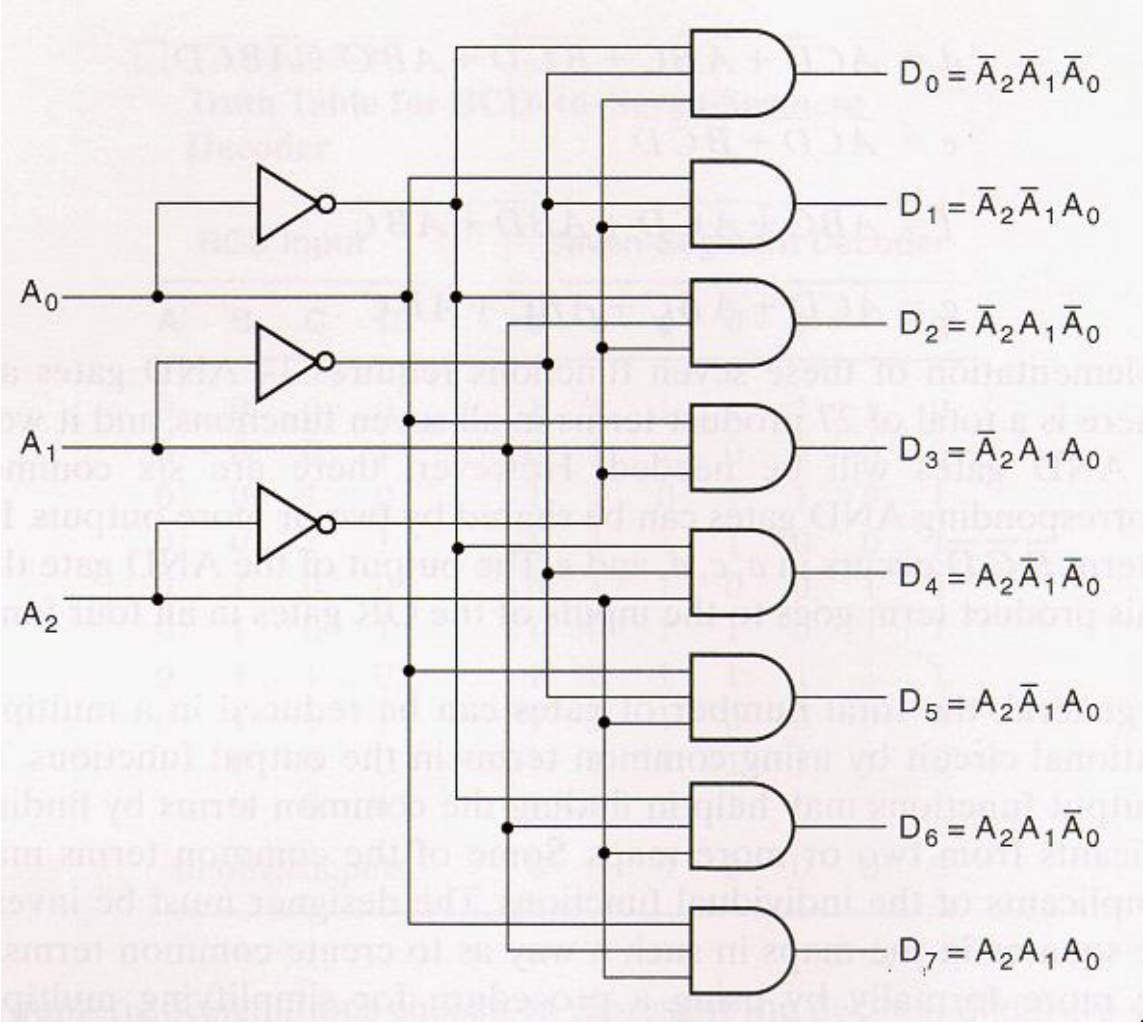
Truth table

Inputs			Outputs							
A_2	A_1	A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	0								
0	0	1								
0	1	0								
0	1	1								
1	0	0								
1	0	1								
1	1	0								
1	1	1								

Truth table

Inputs			Outputs							
A_2	A_1	A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

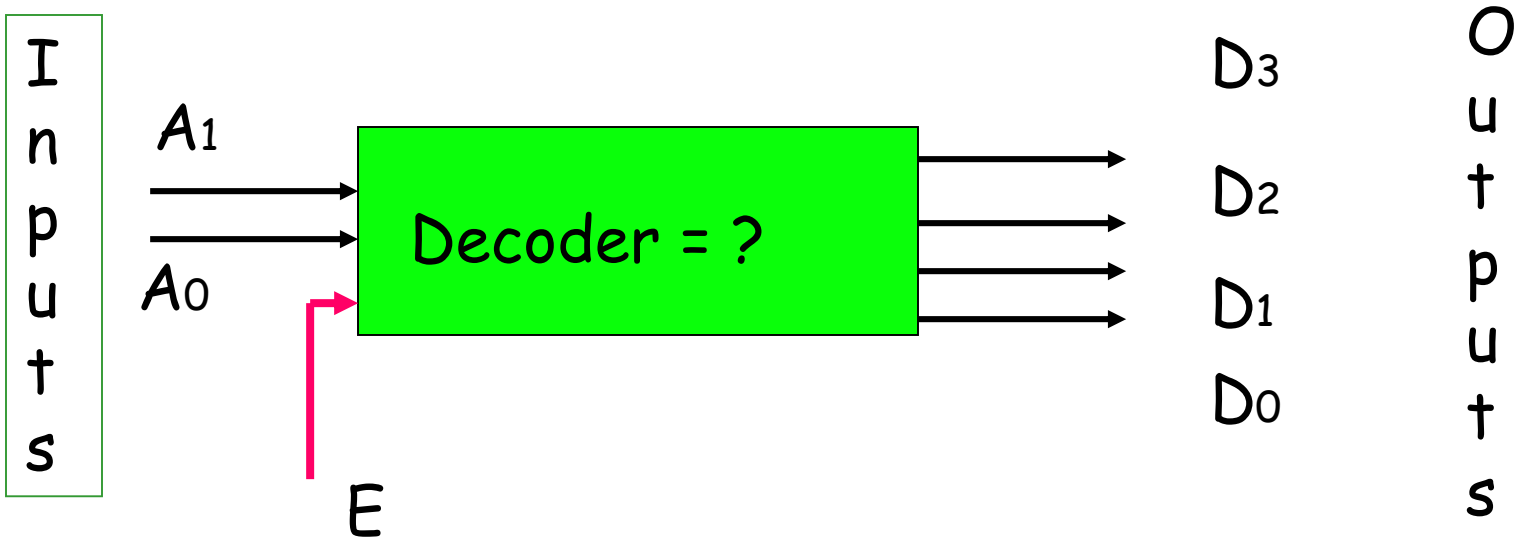
Logic equations and circuit



Sometimes a decoder needs to be disabled …therefore … we need to

- Design a 2-4 line decoder, with enable input

Design a 2-4 line decoder, with enable input



n = 2

$m = 2^n = 4$

Truth table

E	A_1	A_0	D_3	D_2	D_1	D_0
1	0	0				
1	0	1				
1	1	0				
1	1	1				
0	X	X				

Truth table

E	A_1	A_0	D_3	D_2	D_1	D_0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0
0	X	X	0	0	0	0

Logic equations

E	A_1	A_0	D_3	D_2	D_1	D_0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0
0	X	X	0	0	0	0

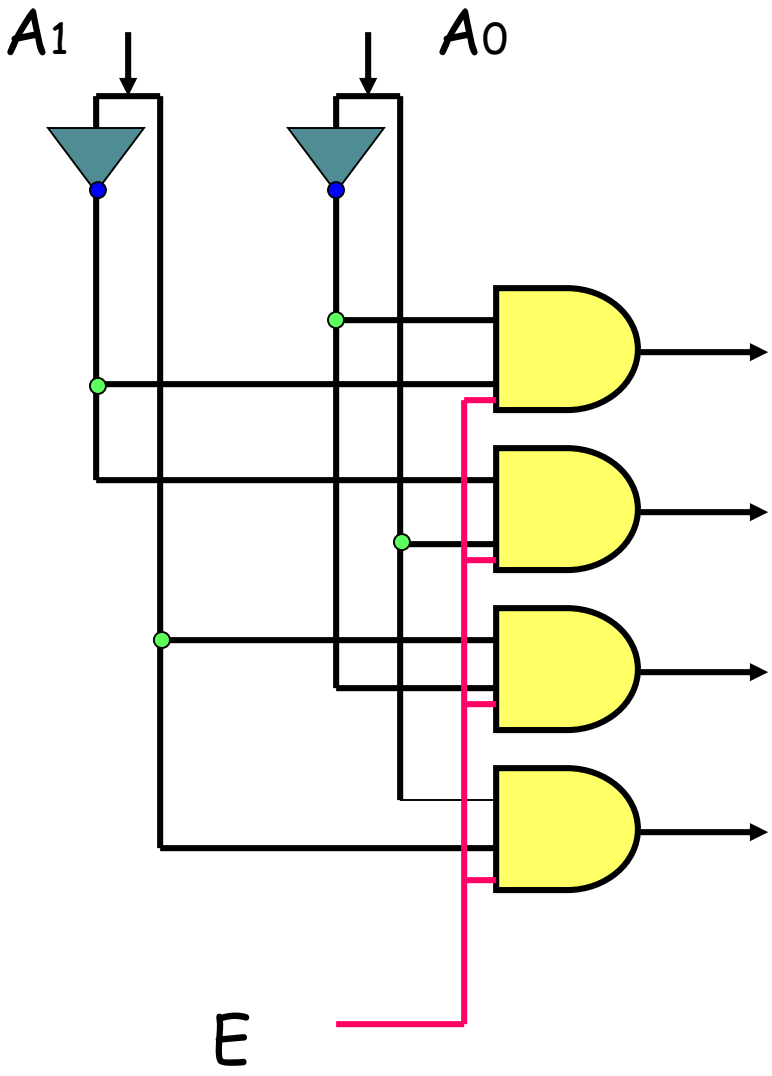
$$D_0 = E\ \overline{A_1}\ \overline{A_0}$$

$$D_1 = E\ \overline{A_1}\ A_0$$

$$D_2 = E\ A_1\ \overline{A_0}$$

$$D_3 = E\ A_1\ A_0$$

Logic equations



$D_0 = \bar{A}_1 \bar{A}_0$

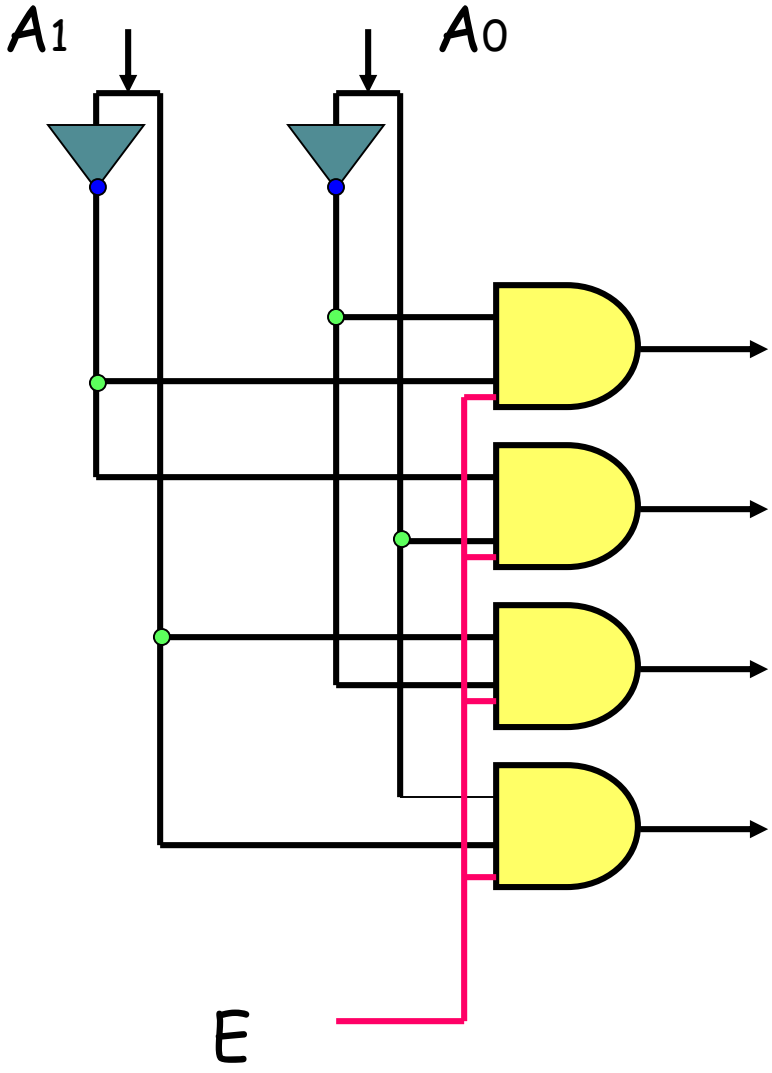
$D_1 = \bar{A}_1 A_0$

$D_2 = A_1 \bar{A}_0$

$D_3 = A_1 A_0$

If E = 1, the decoder is enabled.

If E = 0, the decoder is disabled.



$D_0 = \bar{A}_1 \bar{A}_0$

$D_1 = \bar{A}_1 A_0$

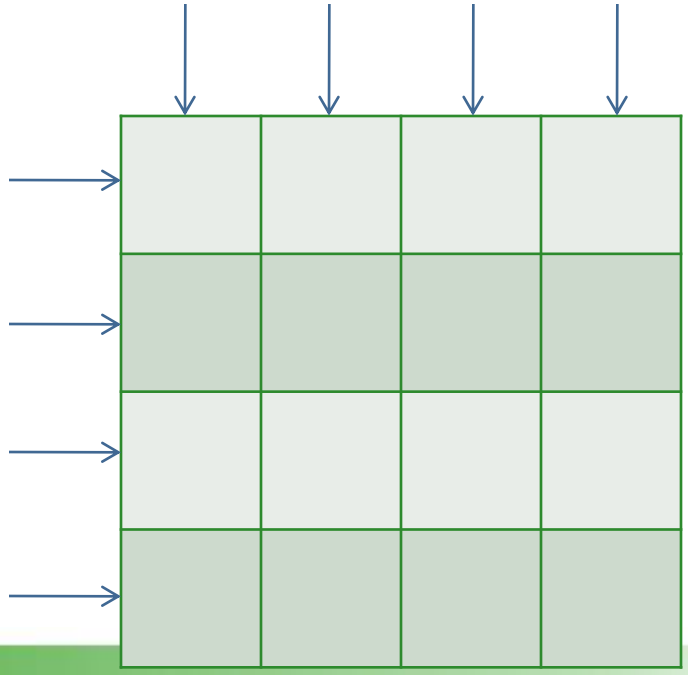
$D_2 = A_1 \bar{A}_0$

$D_3 = A_1 A_0$

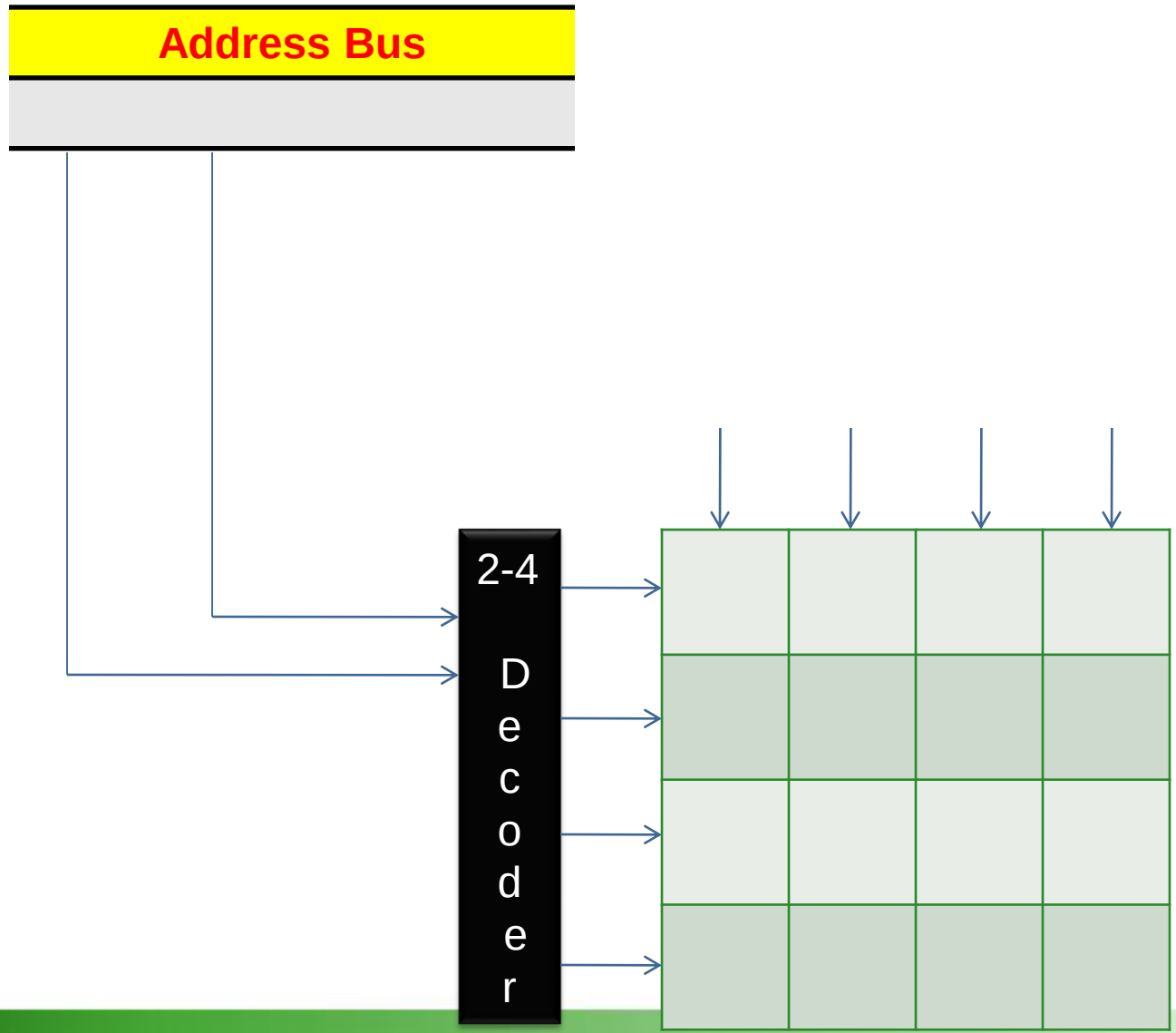
Decoder application

- A decoder can select one of the several output lines
- Are used in memory-systems

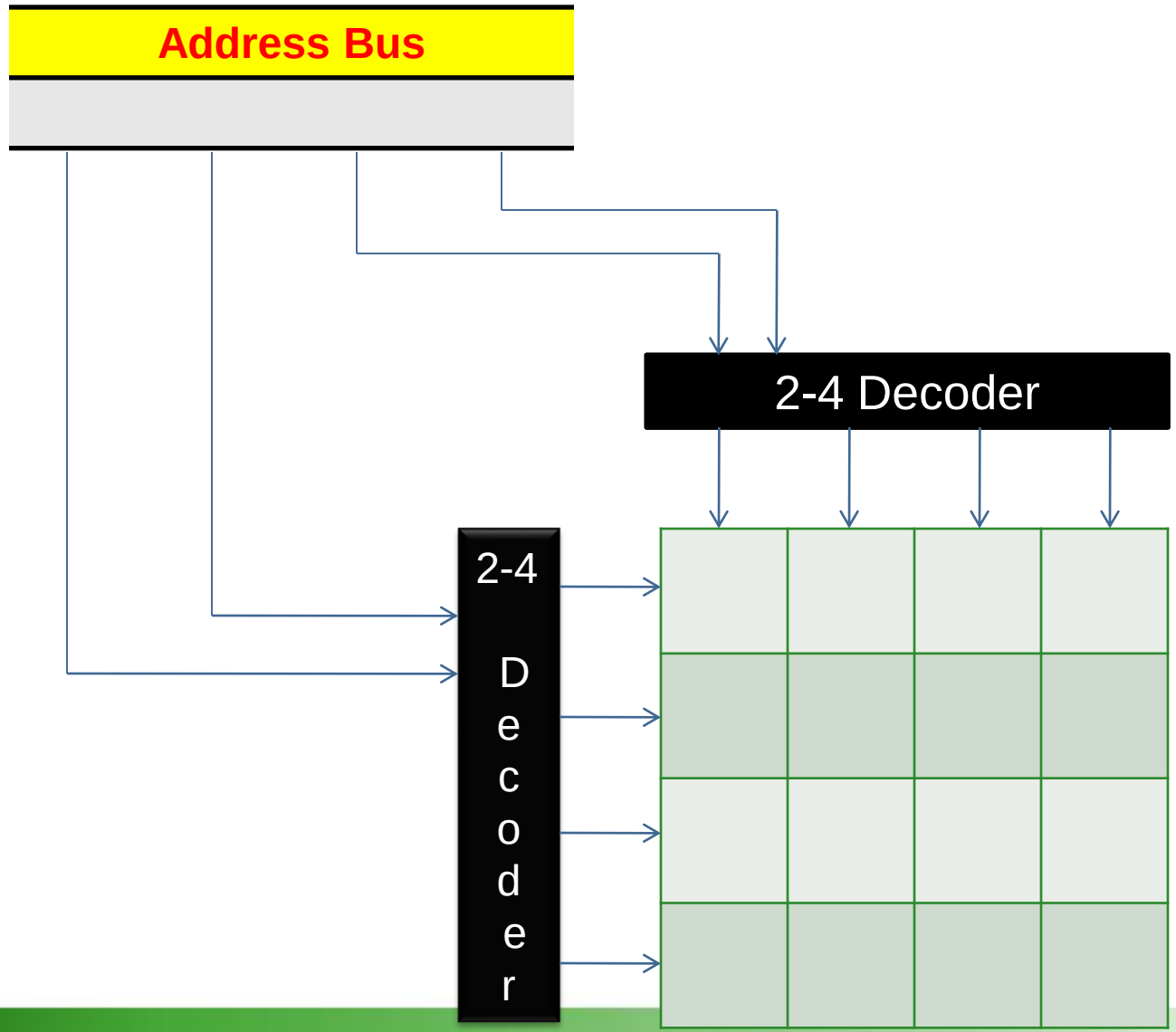
Memory



1D Memory-Decoder



2D Memory-Decoder



Memory … “chip selector”

- Decoders are used to select memory chips
- Suppose that we have eight 1K memory chips and only 2-4 decoder.
  How can we put together the decoders in order to access or select data in the eight (1K) memory chips ?

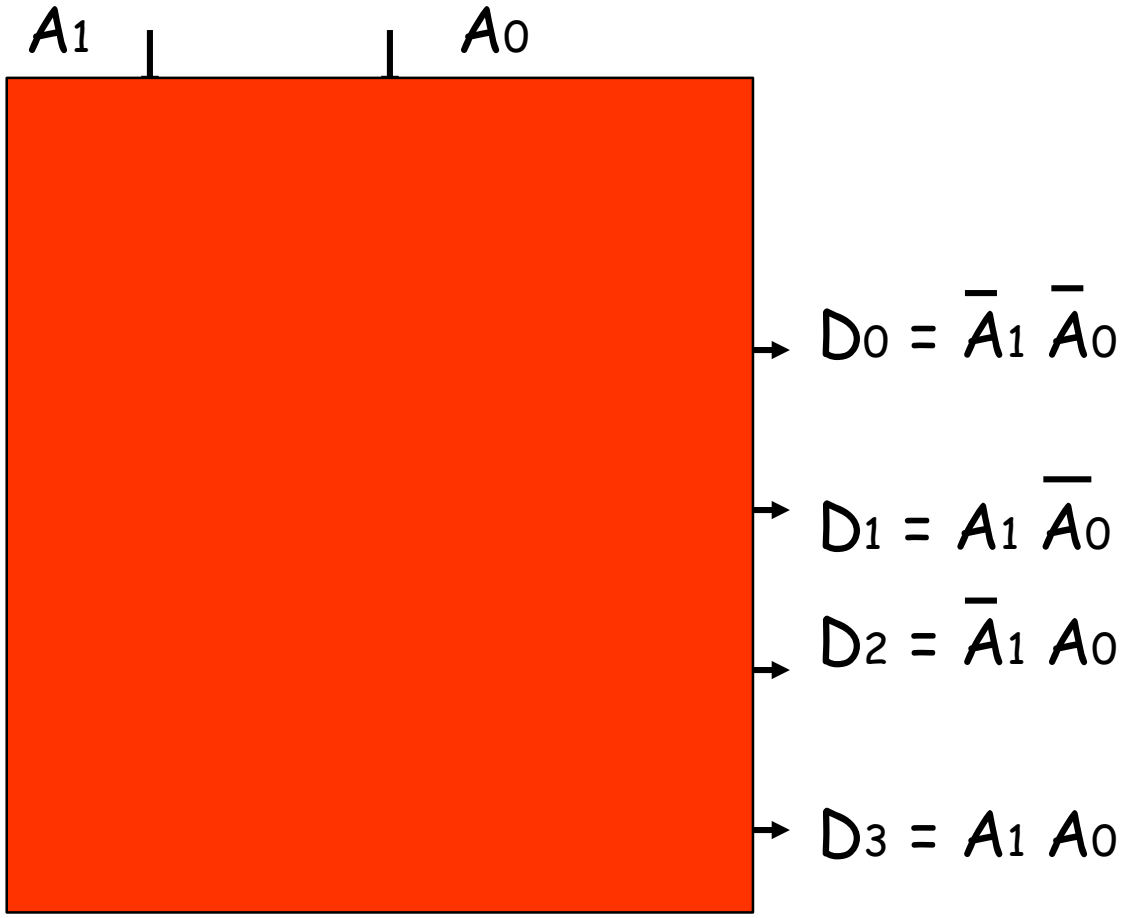
Will be solved in a future lecture (Memory Design)

A decoder can also be used to implement logic functions

Basic idea of PLD's

A decoder can also be used to implement logic functions

Decoder (2-4)



Application

- Design a binary half-adder, using a Decoder and OR Gates.

Truth table

A_1	A_0	C	S
0	0	?	
0	1		
1	0		
1	1		

Truth table

A_1	A_0	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Logic equations

A_1	A_0	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$C = A_0 \, A_1$$

$$S = A_0 \, \overline{A_1} + \overline{A_0} \, A_1$$

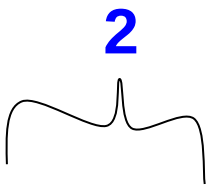
What is the size of the Decoder?

A_1	A_0	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$C = A_0 A_1$$

$$S = A_0 \overline{A_1} + \overline{A_0} A_1$$

Decoder



A_1	A_0	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

4

$$C = A_0 A_1$$

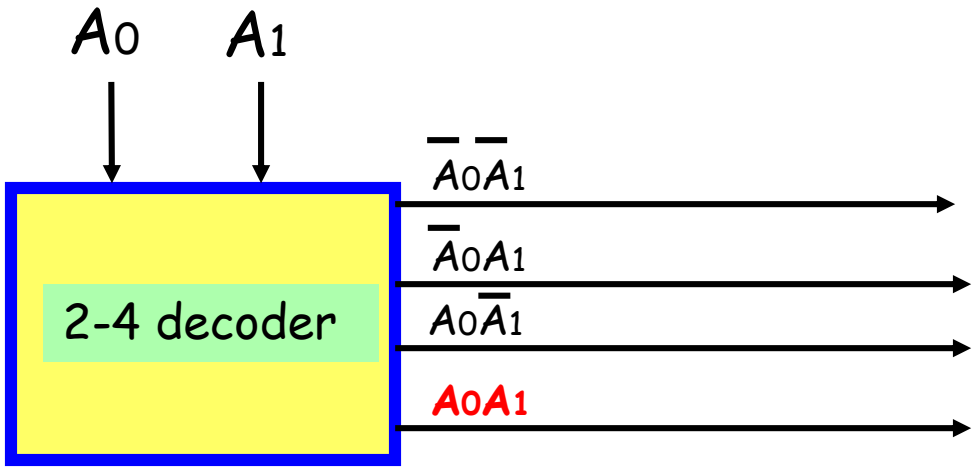
$$S = A_0 \overline{A_1} + \overline{A_0} A_1$$

Size of the decoder = **2** - **4**

Logic Circuit

$$C = A_0 A_1$$

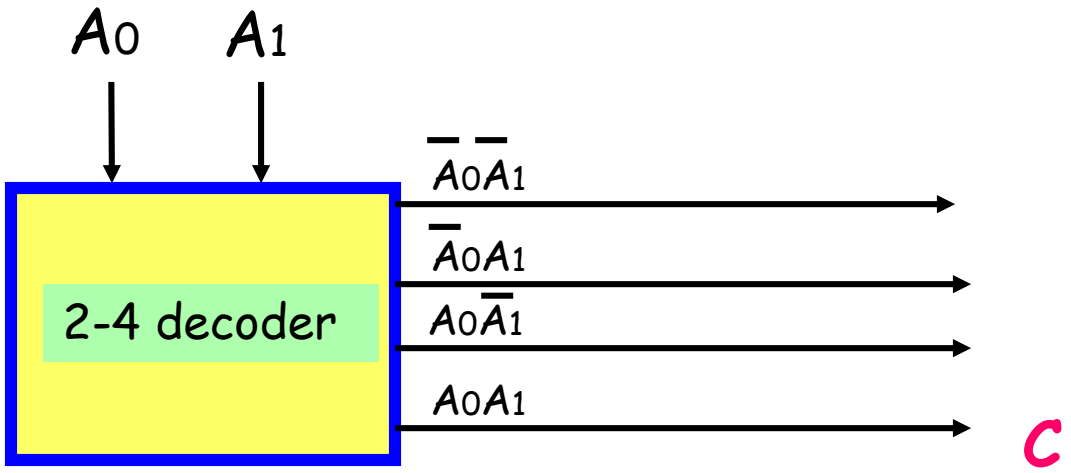
$$S = A_0 \overline{A_1} + \overline{A_0} A_1$$



Logic Circuit: C

✓. $C = A_0 A_1$

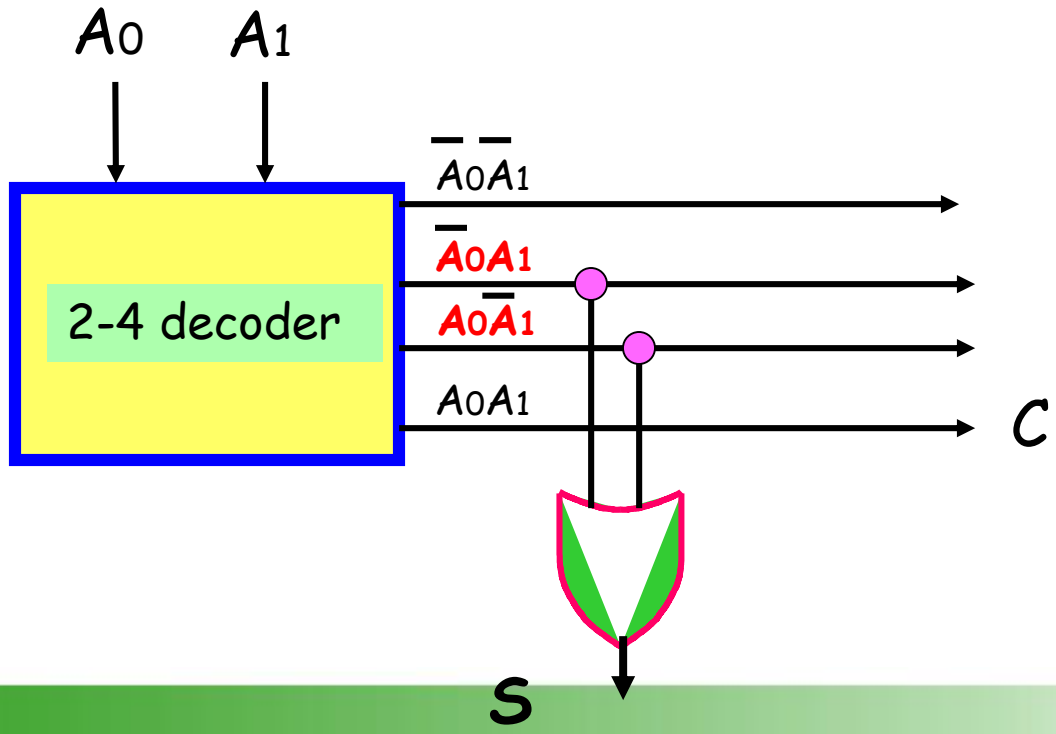
$S = A_0 \overline{A_1} + \overline{A_0} A_1$



Logic Circuit: *S*

$C = A_0 A_1$

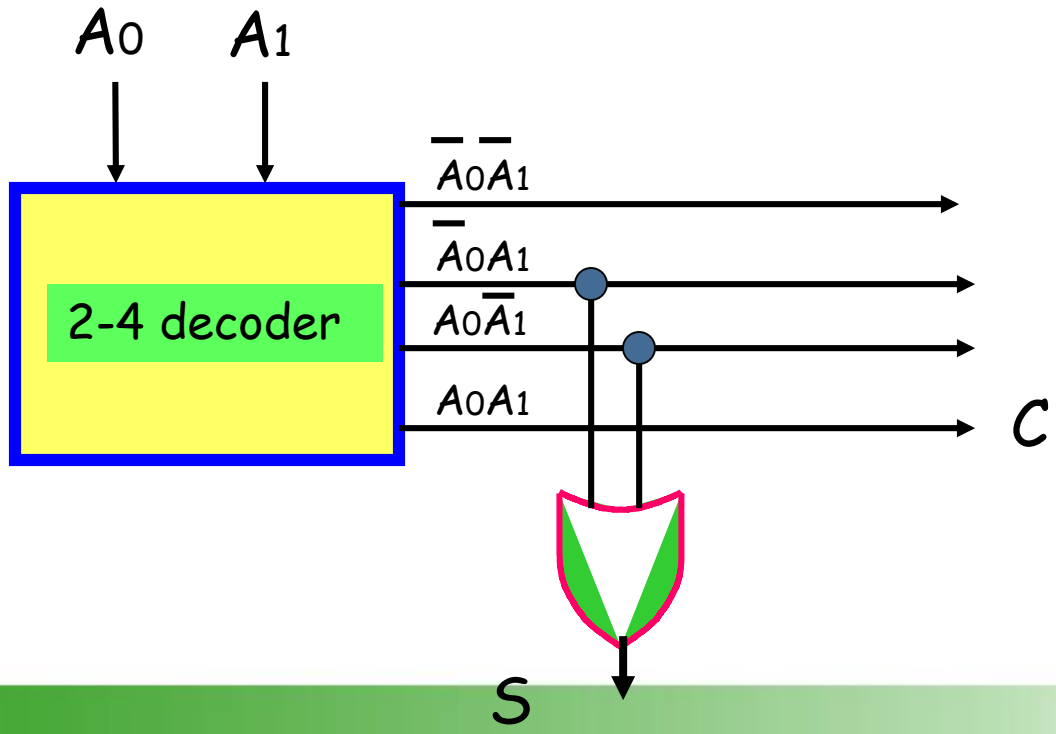
✓. $S = A_0 \overline{A_1} + \overline{A_0} A_1$



Boolean expressions can be implemented with a Decoder and Gates

$$C = A_0 A_1$$

$$S = A_0 \overline{A_1} + \overline{A_0} A_1$$



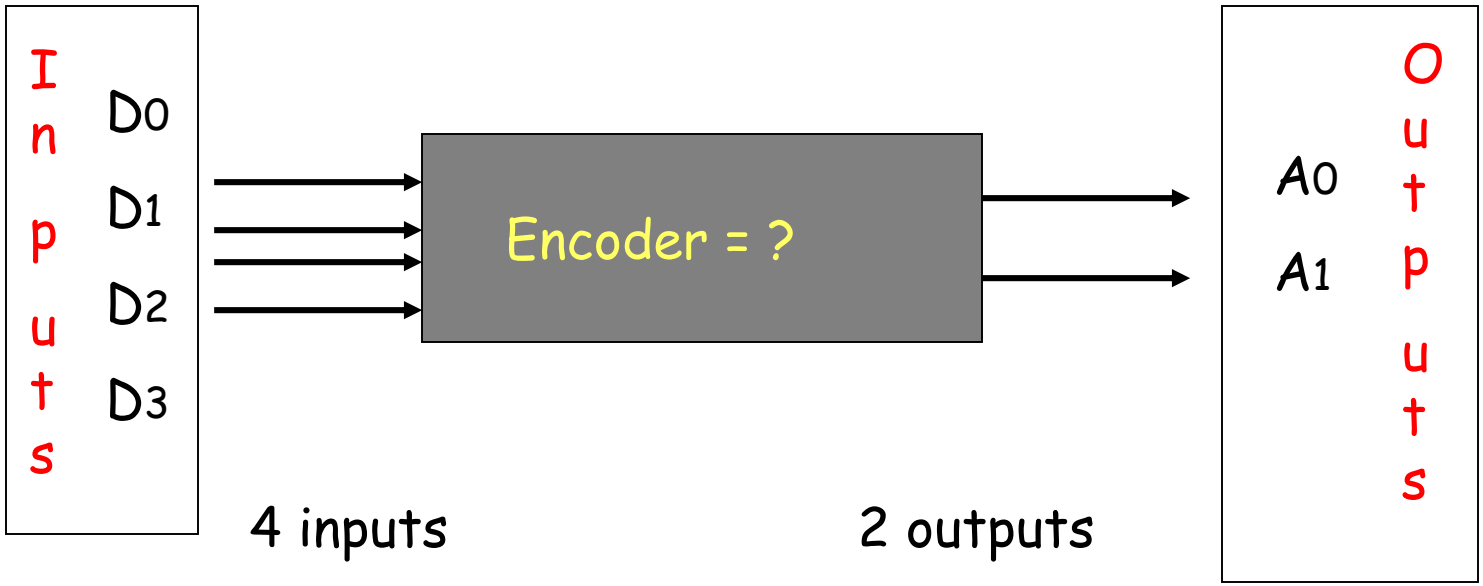
Binary Encoder

For circuit design

Encoder

- An encoder performs the inverse operation of a decoder…
- … encodes binary information into another more compact binary form
- An encoder is a special logic circuit that converts 2^n input lines into n output lines. The input lines are activated for only one possible combination of the output lines.

Design a 4-2 line encoder



The input lines are activated for only one possible combination of the output lines.

Truth Table

D3	D2	D1	D0	A1	A0
		?		?	

The input lines are activated for only one possible combination of the output lines.

Truth Table

D_3	D_2	D_1	D_0	A_1	A_0
0	0	0	1		
0	0	1	0		
0	1	0	0		
1	0	0	0		

The input lines are activated for only one possible combination of the output lines.

Truth Table

D_3	D_2	D_1	D_0	A_1	A_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

The input lines are activated for only one possible combination of the output lines.

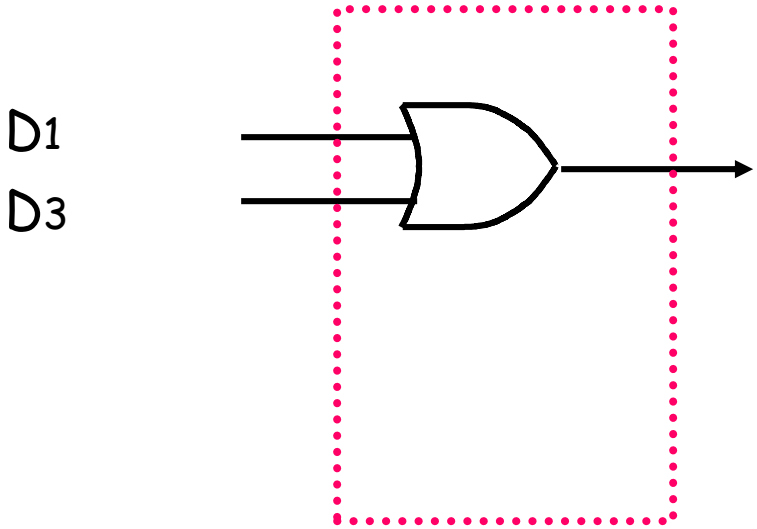
Logic Equations

D_3	D_2	D_1	D_0	A_1	A_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

$A_0 = D_1 + D_3$

$A_1 = D_2 + D_3$

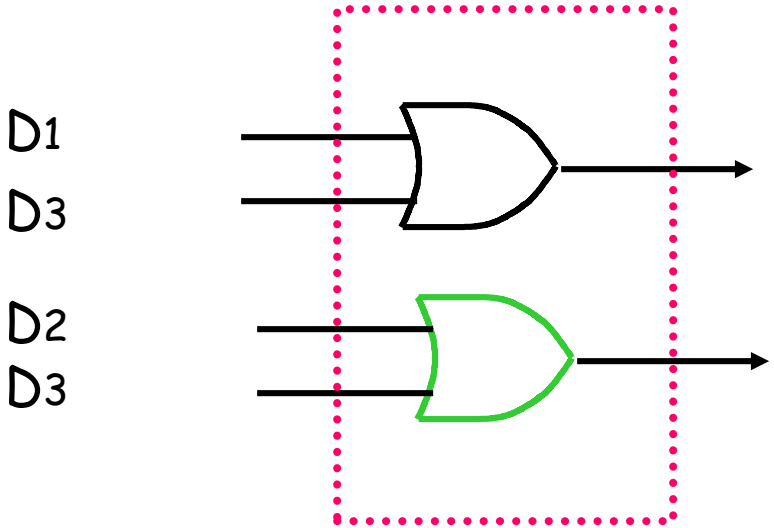
Logic Diagram



$A0 = D1 + D3$

$A1 = D2 + D3$

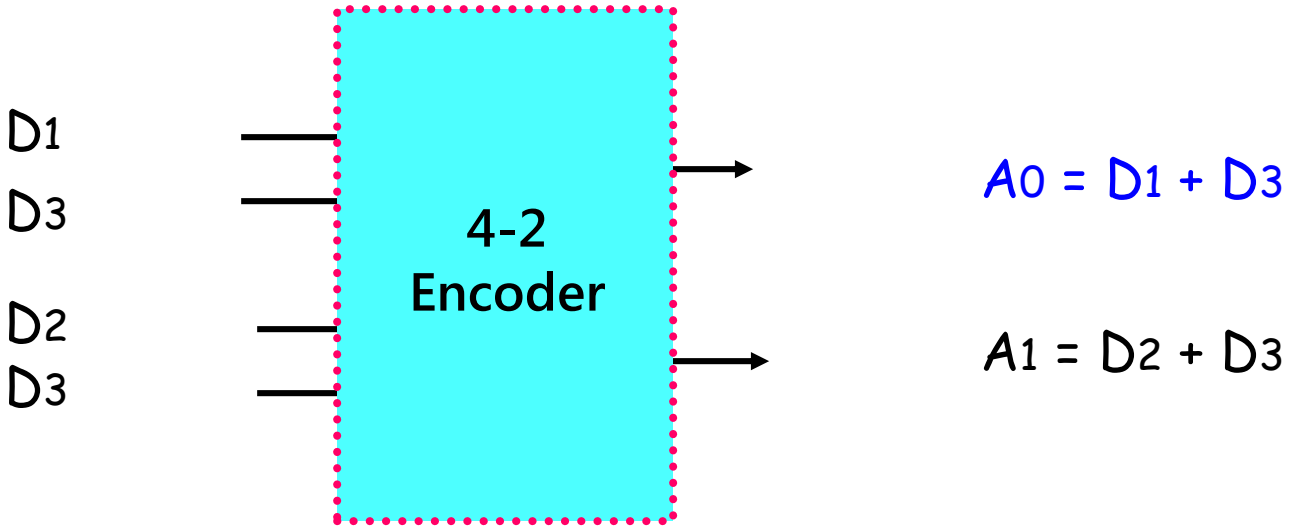
Logic Diagram



$A0 = D1 + D3$

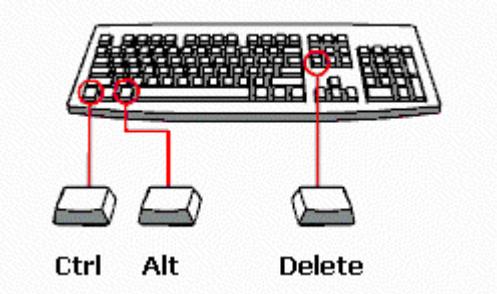
$A1 = D2 + D3$

Logic Circuit



Limitation:

- Only one input can be active at any time. If more than one input is needed to be activated a priority encoder has to be used.

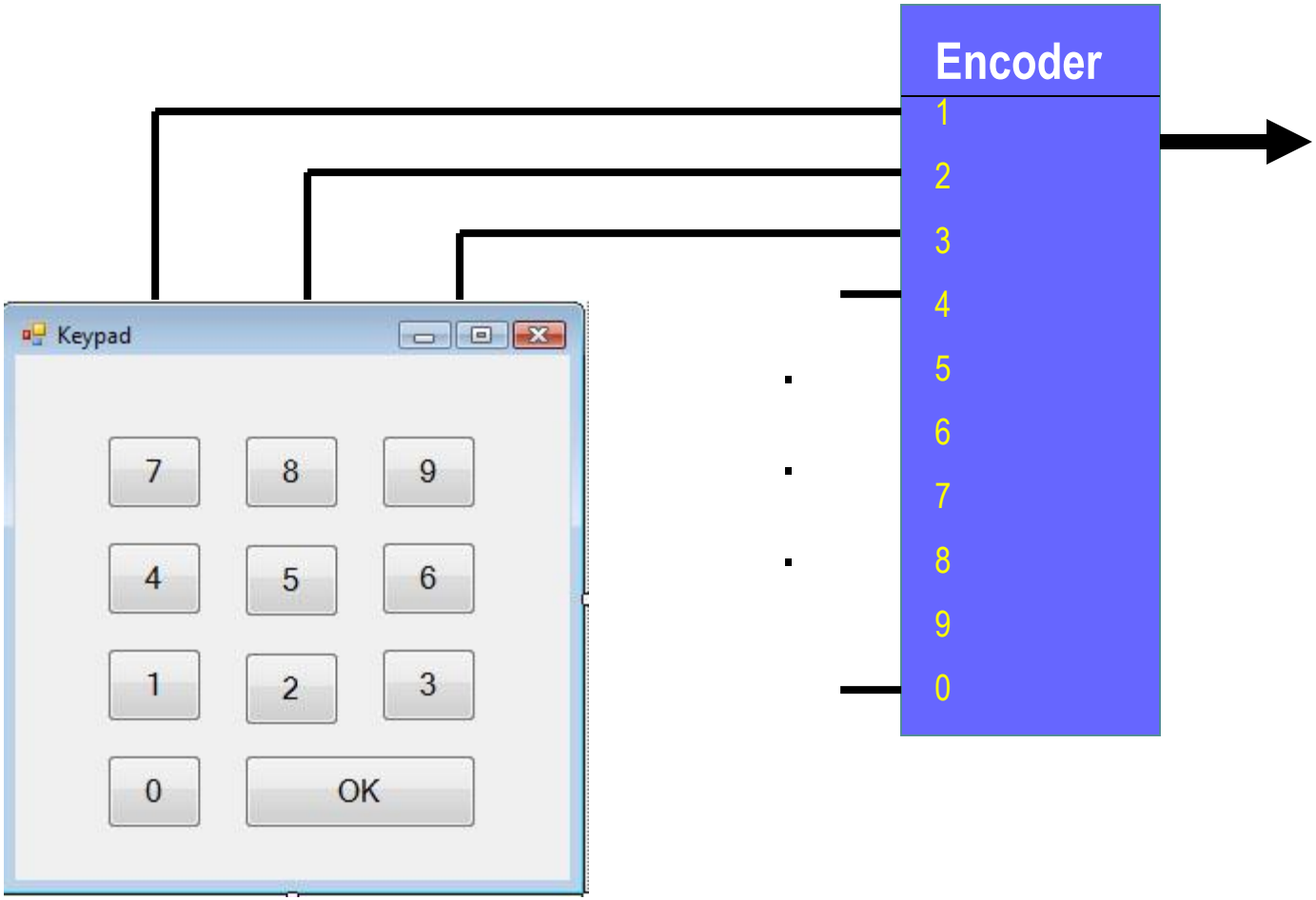


Application: Keyboard

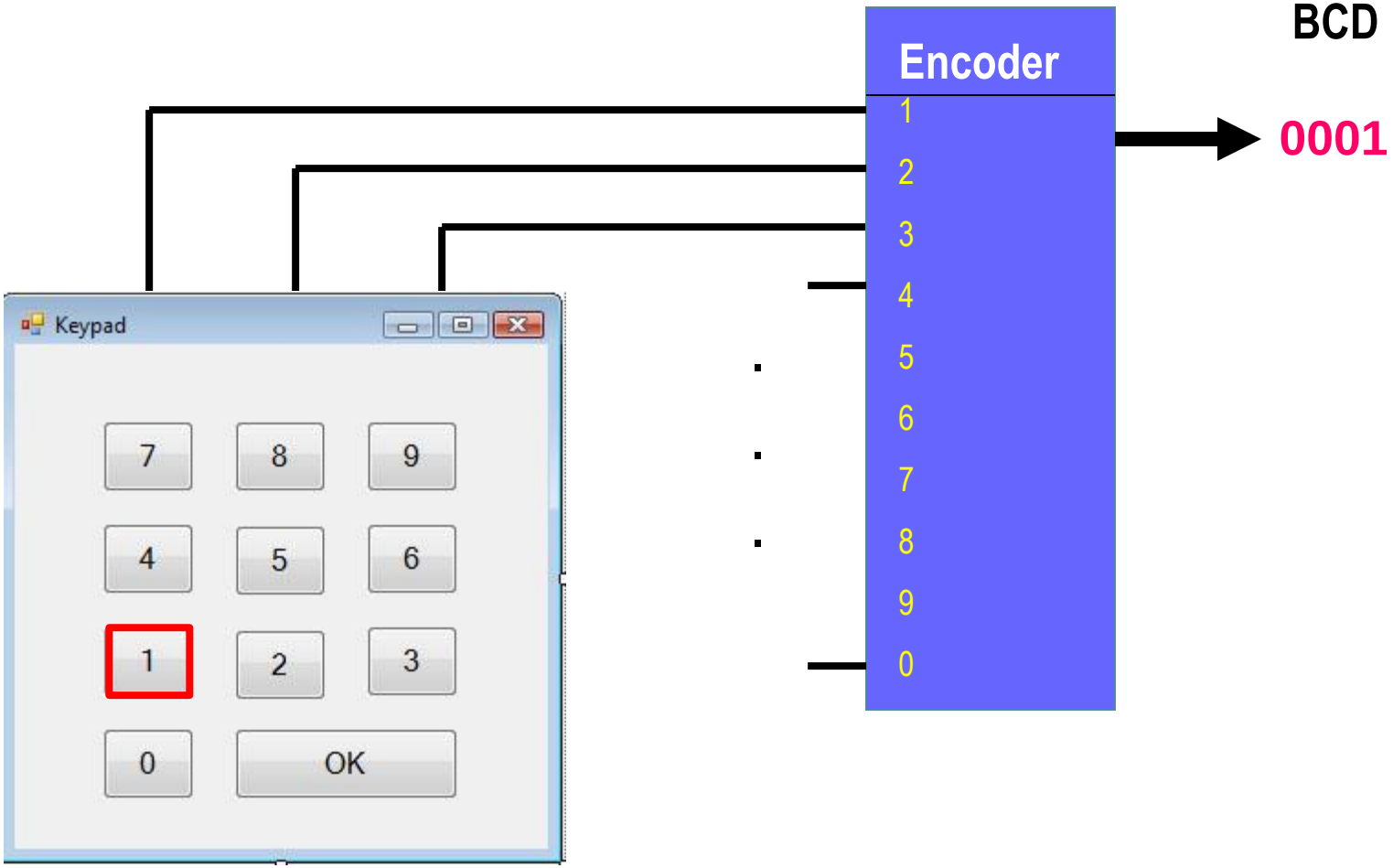
- Design an interface to encode 10 separate input switches and to output the binary value corresponding to each input switch.

Decimal numbers

Application: Keyboard



Application: Keyboard



Decimal number to BCD Encoder

Truth Table: Decimal number to BCD Encoder

	BCD			
0 1 2 3 4 5 6 7 8 9	A	B	C	D
1 0 0 0 0 0 0 0 0 0				
0 1 0 0 0 0 0 0 0 0				
0 0 1 0 0 0 0 0 0 0				
0 0 0 1 0 0 0 0 0 0				
0 0 0 0 1 0 0 0 0 0				
0 0 0 0 0 1 0 0 0 0				
0 0 0 0 0 0 1 0 0 0				
0 0 0 0 0 0 0 1 0 0				
0 0 0 0 0 0 0 0 1 0				
0 0 0 0 0 0 0 0 0 1				

Truth Table: Decimal number to BCD Encoder

	BCD			
0 1 2 3 4 5 6 7 8 9	A	B	C	D
1 0 0 0 0 0 0 0 0 0	0	0	0	0
0 1 0 0 0 0 0 0 0 0	0	0	0	1
0 0 1 0 0 0 0 0 0 0	0	0	1	0
0 0 0 1 0 0 0 0 0 0	0	0	1	1
0 0 0 0 1 0 0 0 0 0	0	1	0	0
0 0 0 0 0 1 0 0 0 0	0	1	0	1
0 0 0 0 0 0 1 0 0 0	0	1	1	0
0 0 0 0 0 0 0 1 0 0	0	1	1	1
0 0 0 0 0 0 0 0 1 0	1	0	0	0
0 0 0 0 0 0 0 0 0 1	1	0	0	1

When an input is activated, the 4-bit output is the binary equivalent of the decimal number

Implementation: Encoder

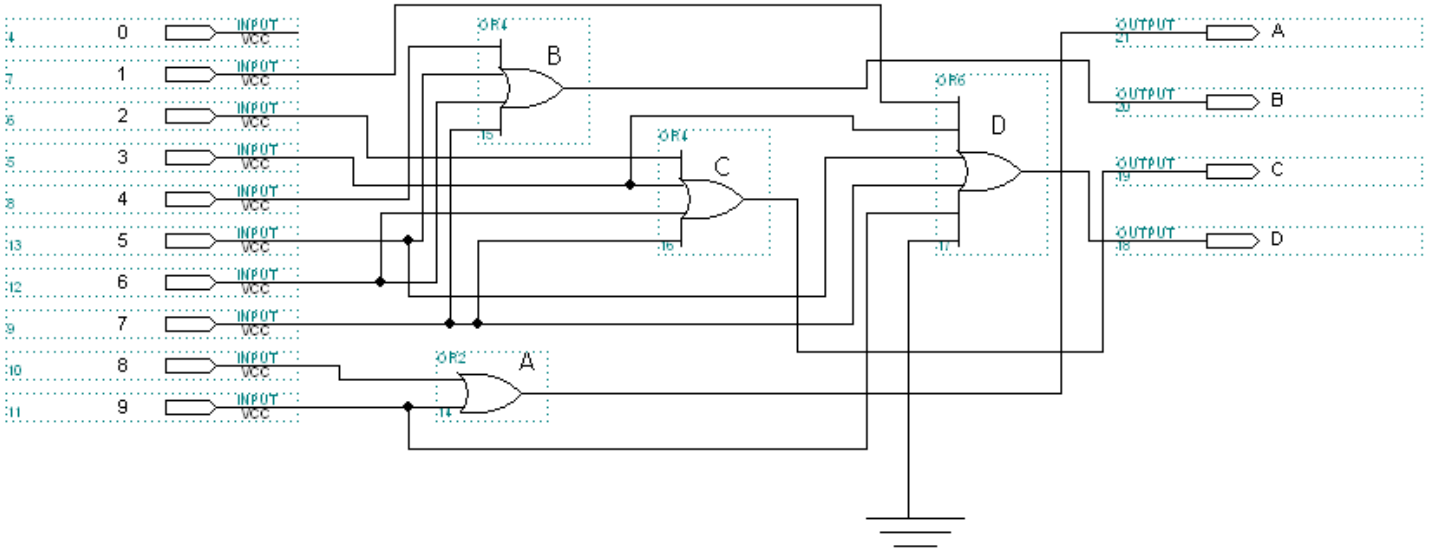
0	1	2	3	4	5	6	7	8	9	A	B	C	D
1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	0	0	0	1	0	0	0	0	1	1	0
0	0	0	0	0	0	0	1	0	0	0	1	1	1
0	0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	1

$A = 8 + 9$

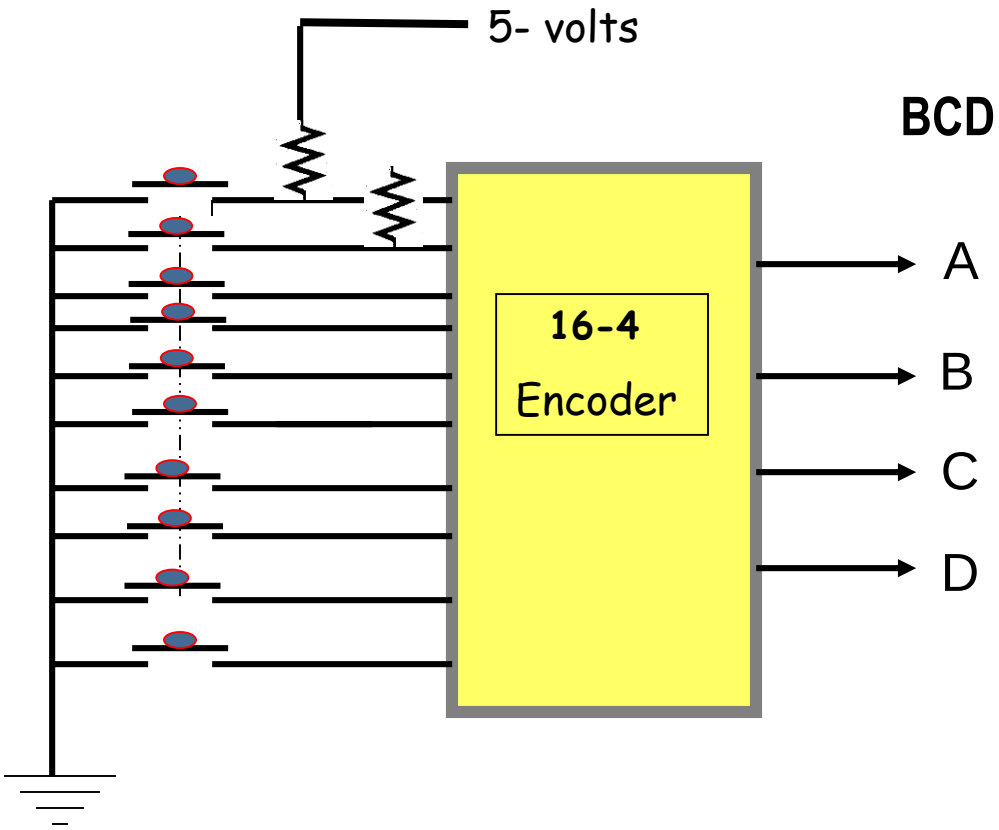
$B = 4 + 5 + 6 + 7$

$C = 2 + 3 + 6 + 7$

$D = 1 + 3 + 5 + 7 + 9$



Logic Circuit

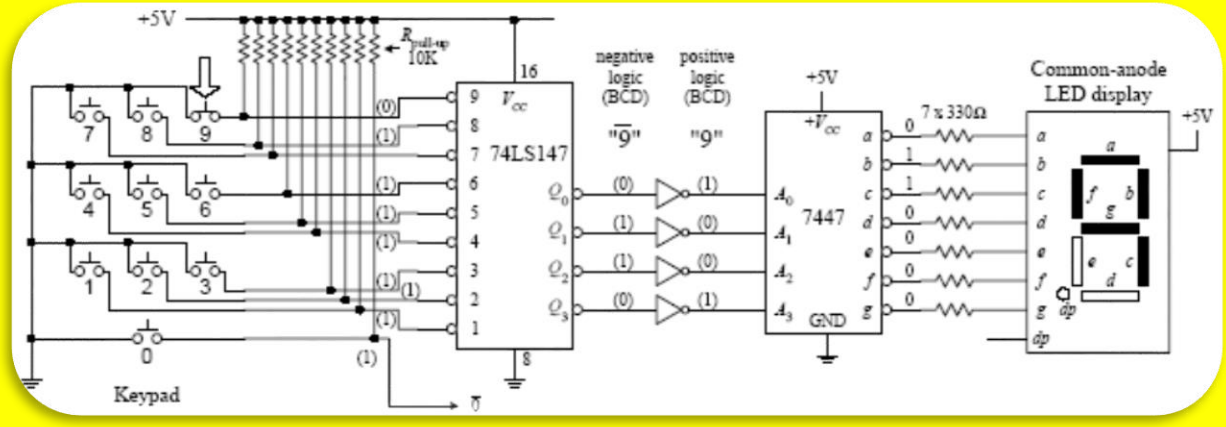


The encoder chip (74LS147)

http://susta.cz/fel/74/pdf/sn_74147.pdf

7-Segment-display application

BCD to 7-segment display



Decimal to BCD encoder