IS 1202 Computer Systems

Combinational Logic Circuits

Digital Logic Circuits

There are two types of digital logic circuits

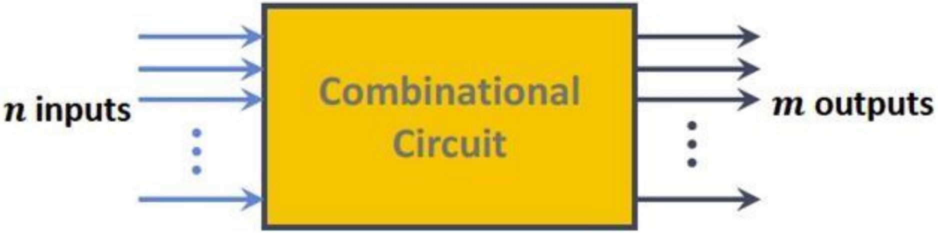
1. Combinational Logic circuits
2. Sequential Logic circuits

Combinational vs Sequential

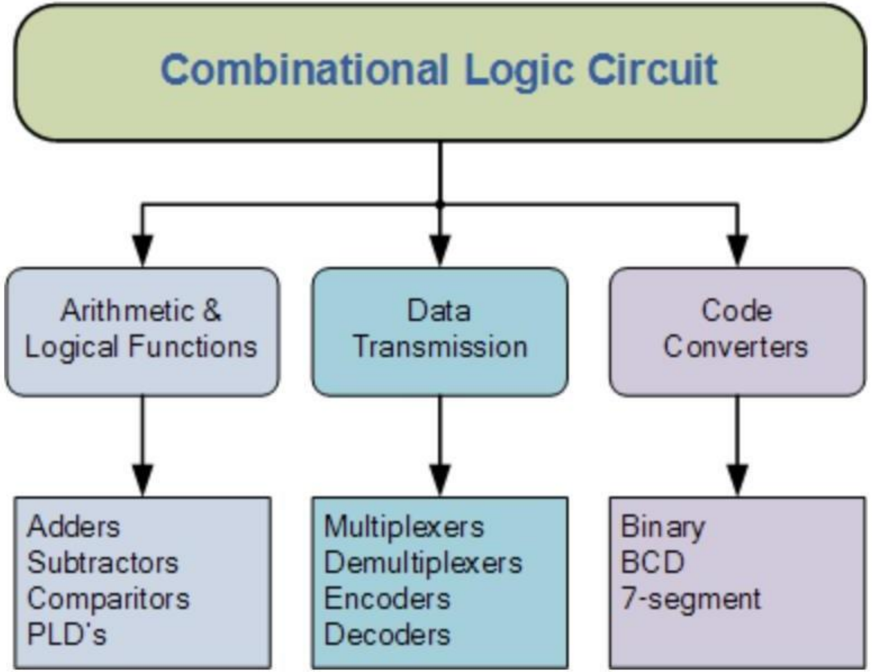
Combinational Circuits	Sequential Circuits
The output of a combinational circuit depends only on its current inputs.	Consider not only its present inputs, but also the previous outputs that are stored in storage elements.
Purely built upon logic gates	Stores previous outputs.
No feedback paths or memory elements	

Combinational Circuit

- 2^n possible input patterns
- Circuits can be represented using
  - Truth table
  - Boolean function
  - Logic gates
- Examples:
  - half adders, full adders, multiplexers, demultiplexers, encoders and decoder



Classification of combinational circuits



Combinational Logic Circuit

Arithmetic & Logical Functions

Data Transmission

Code Converters

Adders
Subtractors
Comparitors
PLD's

Multiplexers
Demultiplexers
Encoders
Decoders

Binary
BCD
7-segment

Binary Addition

Half Adder

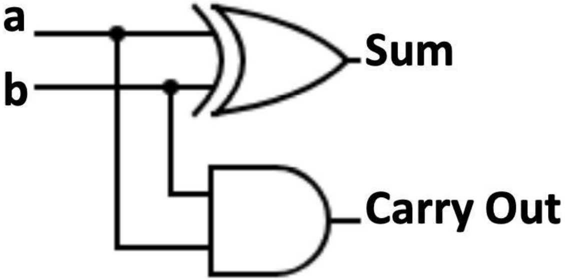
- a combinational circuit that performs the **addition of two bits**

Full Adder

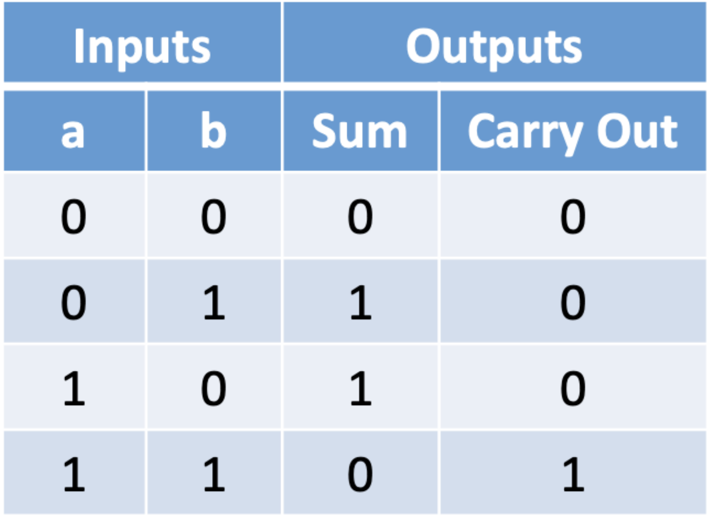
- a combinational circuit that performs the **addition of three bits** (two significant bits and a previous carry)

Half Adder

- Used to add two bits.
- Has two inputs and two outputs, with SUM and CARRY



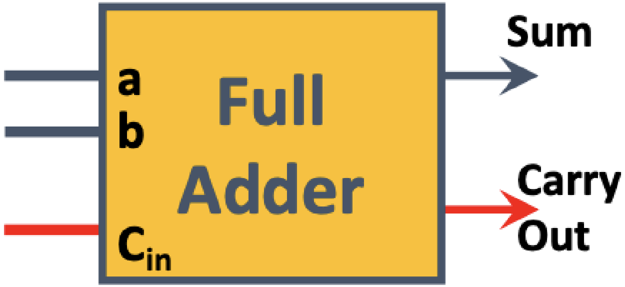
Sum $= A \oplus B$
Carry out $= A \,.\, B$



Inputs		Outputs	
a	b	Sum	Carry Out
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full Adder

- Adds two binary numbers along with carrying input
- Three inputs and two outputs
  - INPUT: Two bits to be added
  - INPUT: A bit represents Carry in
  - OUTPUT: A bit to hold Sum of all inputs
  - OUTPUT: A bit to hold Carry out



Full Adder

Inputs			Outputs	
a	b	C_{in}	Sum	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\text{Sum} = a'b'C_{in} + a'bC'_{in} + ab'C'_{in} + abC_{in}$$

$$C_{out} = ab + bC_{in} + aC_{in}$$

OR

$$\text{Sum} = a \oplus b \oplus C_{in}$$

$$C_{out} = a.b + C_{in}(a \oplus b)$$

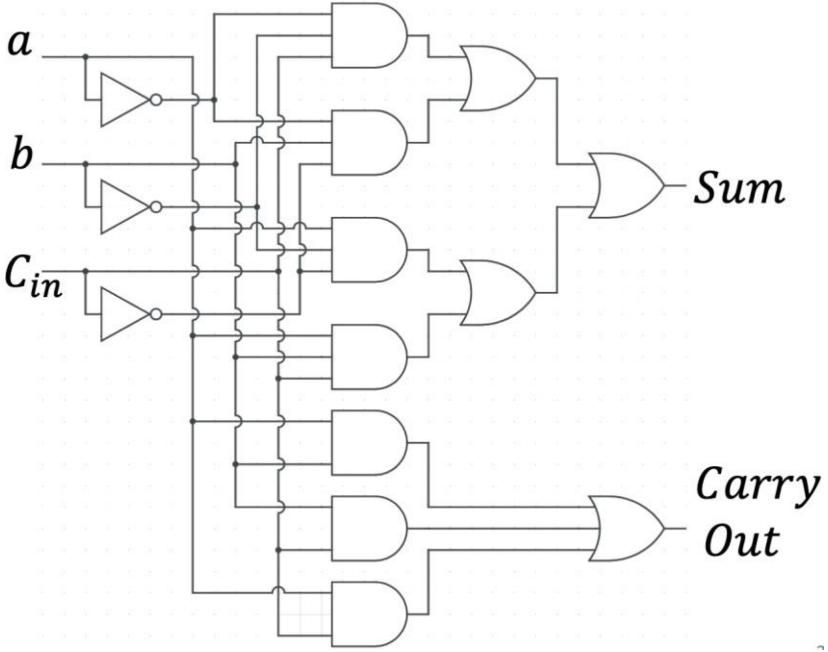
Full Adder

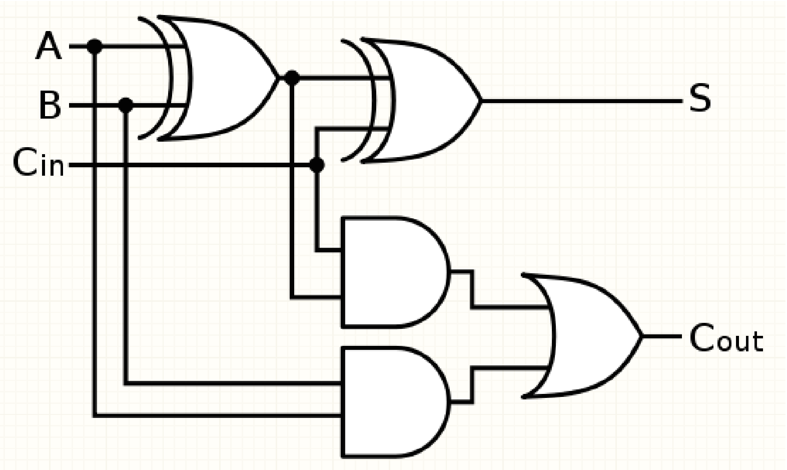
$\text{Sum} = a'b'C_{in} + a'bC'_{in} + ab'C'_{in} + abC_{in}$

$C_{out} = ab + bC_{in} + aC_{in}$

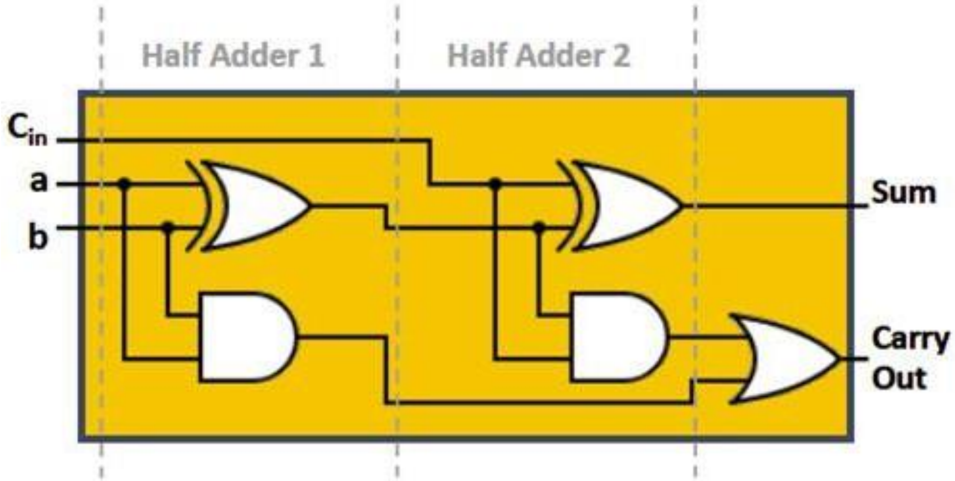
$\text{Sum} = a \oplus b \oplus C_{in}$

$C_{out} = a.b + C_{in}(a \oplus b)$



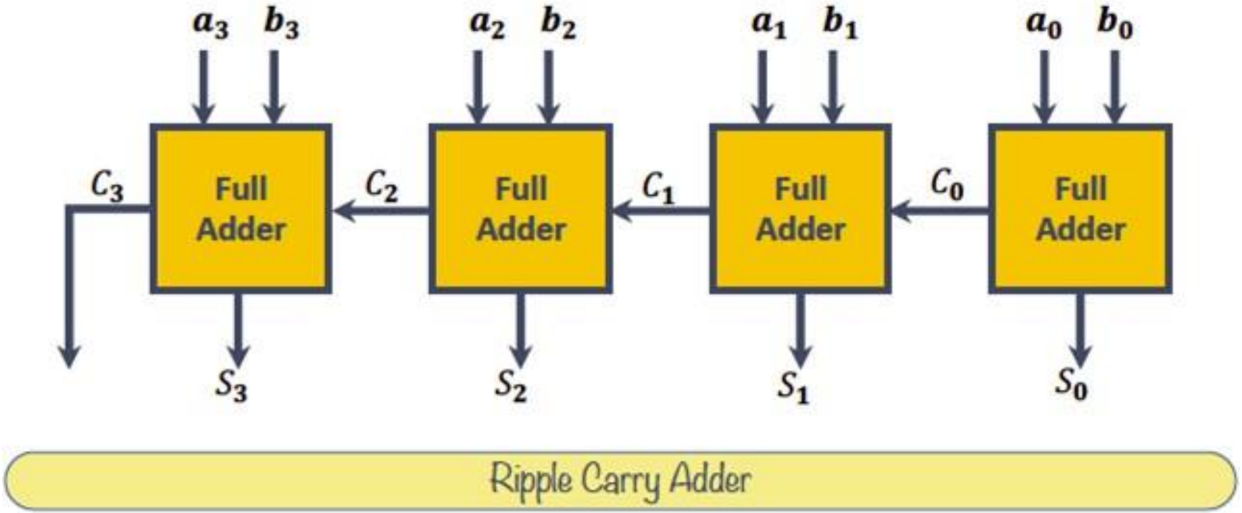


Full Adder with half adders



4-bit adder

Number 1	$a_3\ a_2\ a_1\ a_0$
Number 2	$b_3\ b_2\ b_1\ b_0$

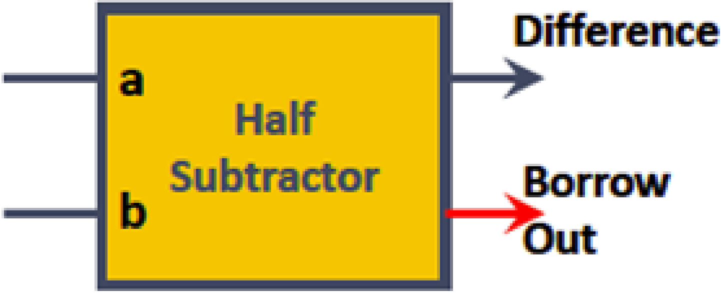


Binary Subtraction

- Half Subtractor
- Full Subtractor

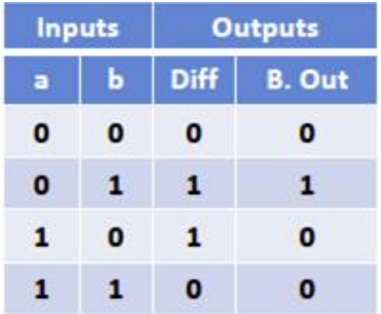
Half Subtractor

- INPUTS: Two single bit numbers (two bits)
- OUTPUTS: Difference and Borrow out
- Does not consider Borrowing in.

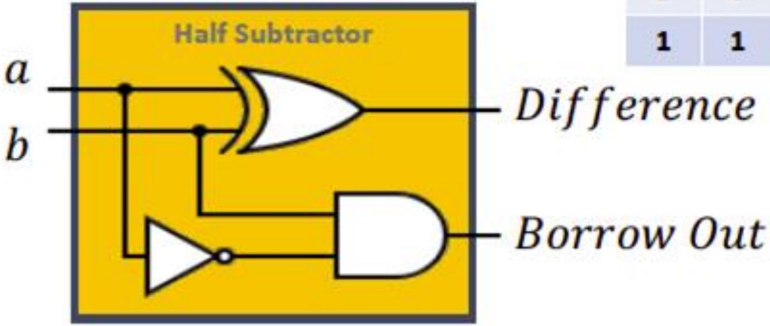


Half Subtractor

$$\text{Difference} = A \oplus B$$
$$\text{Borrow out} = \overline{A} \, . \, B$$



Inputs		Outputs	
a	b	Diff	B. Out
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



Full subtractor

Performs subtraction of two bits, one is minuend and other is subtrahend, taking into account borrow of the previous adjacent lower minuend bit.

Has three inputs and two outputs.

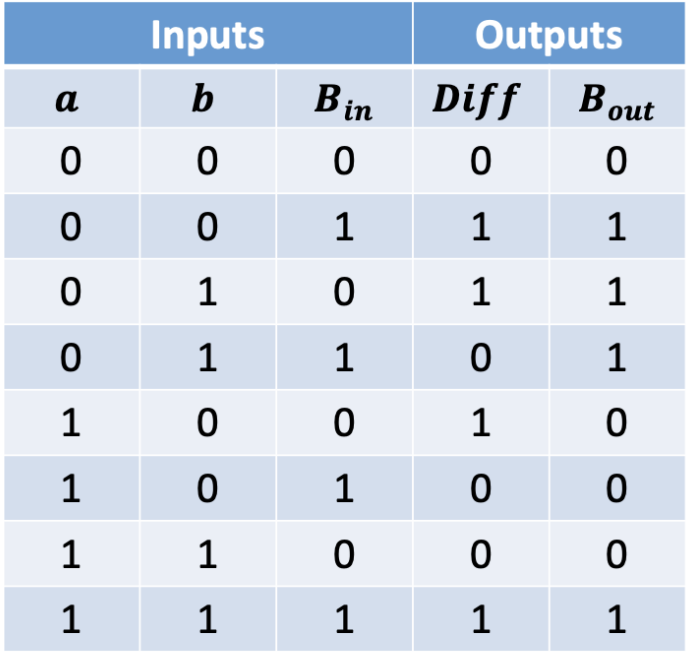
A (minuend)
B (subtrahend)
B_{in} (Borrow in / previous borrow)

Inputs

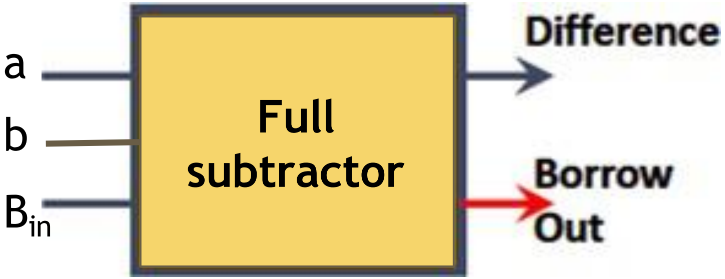
D (Difference)
B_{out} (Borrow out)

Outputs

Full subtractor

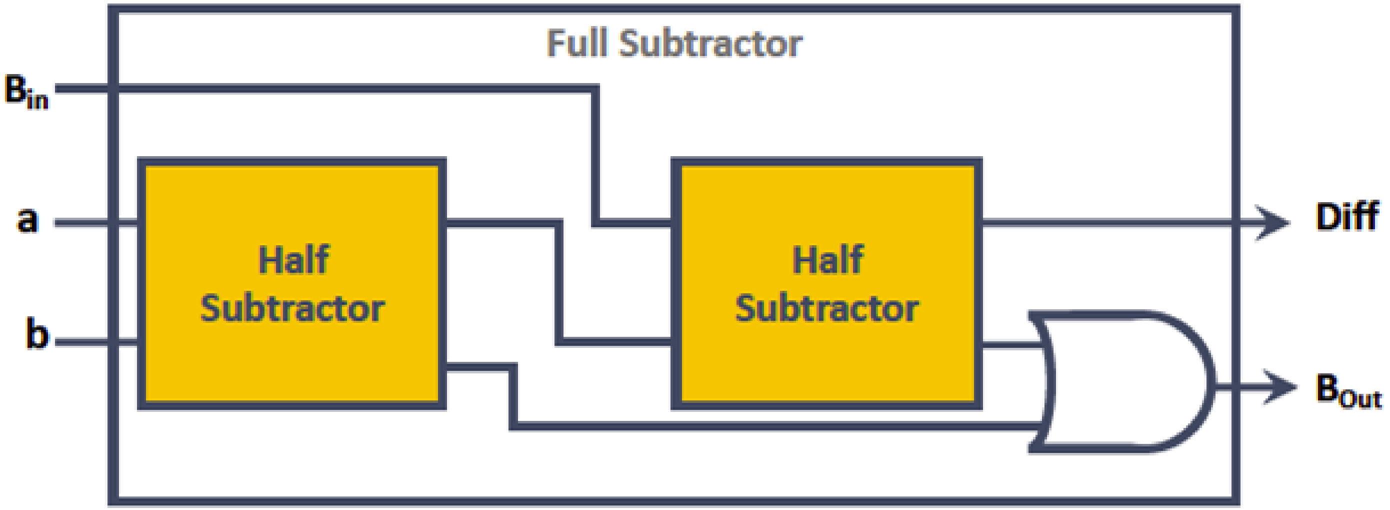


Inputs			Outputs	
a	b	B_{in}	$Diff$	B_{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



$$\text{Difference} = A \oplus B \oplus B_{in}$$
$$\text{Borrow out} = \overline{A}\,B + (\overline{A} + B)B_{in}$$

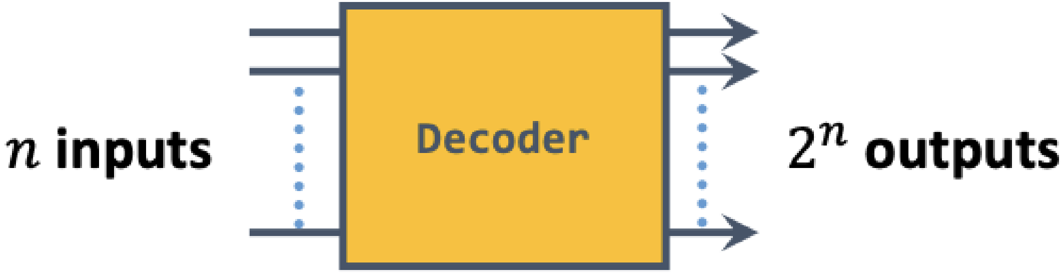
Full subtractor - Alternative approach



Decoders

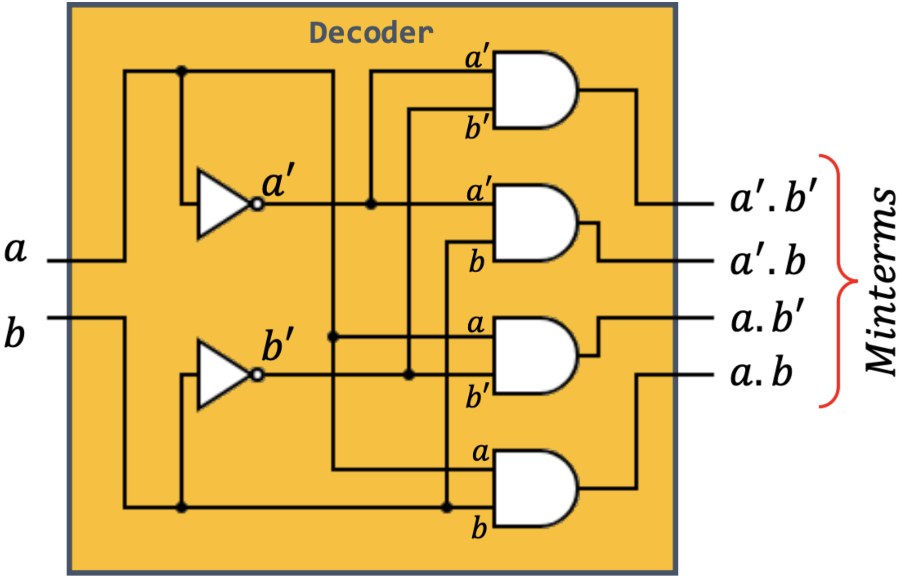
A multi-input and multi-output logic circuit which decodes n inputs into 2^n possible outputs.

Given an input code, only one output channel must be activated.



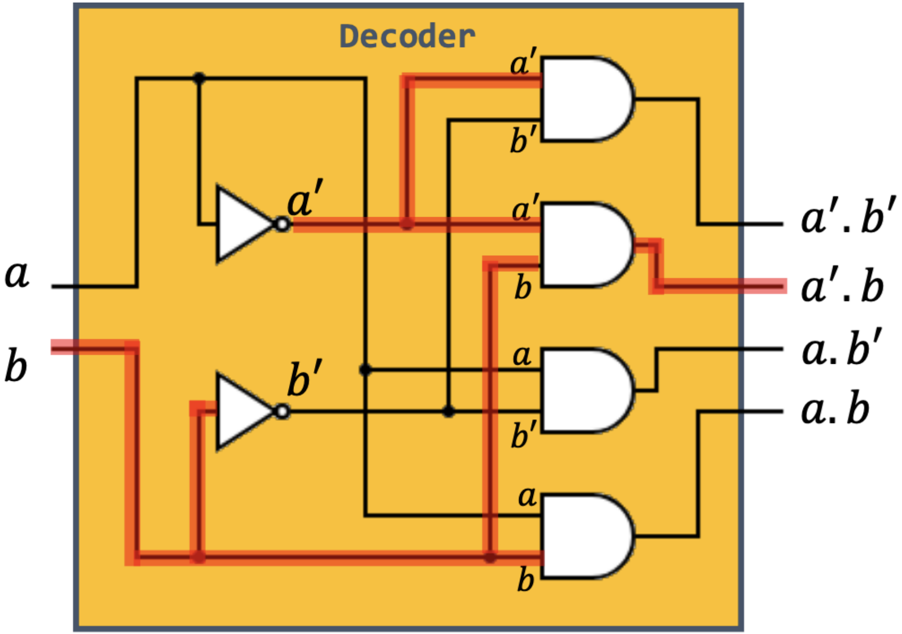
2×4 Decoder

a	b	Y_0	Y_1	Y_2	Y_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



2×4 Decoder

$a = 0$
$b = 1$



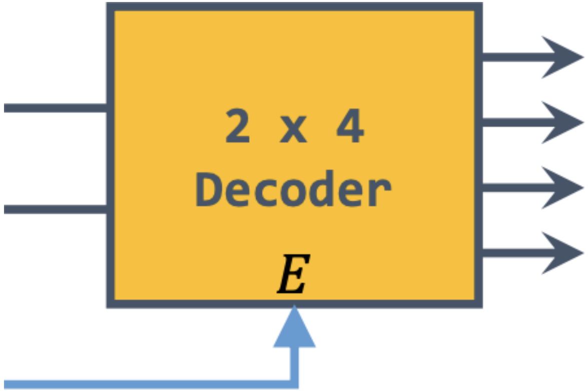
Boolean state 1 *channels are highliged in RED*

Decoder with Enable Input

A standard decoder typically has an additional input called Enable.

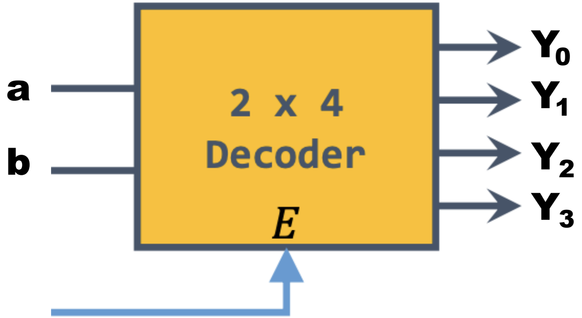
A single controller E input to enable/ disable

- If E = 1 then the decoder will operate normal
- If E = 0 then the decoder output will be inactive



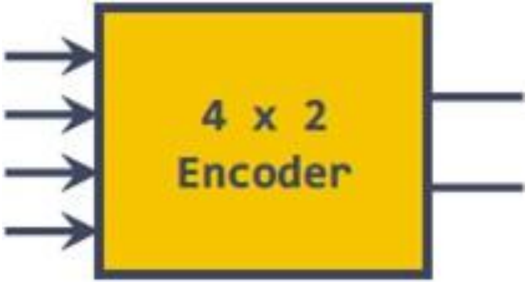
Decoder with Enable Input

E	a	b	Y_0	Y_1	Y_2	Y_3
0	x	x	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

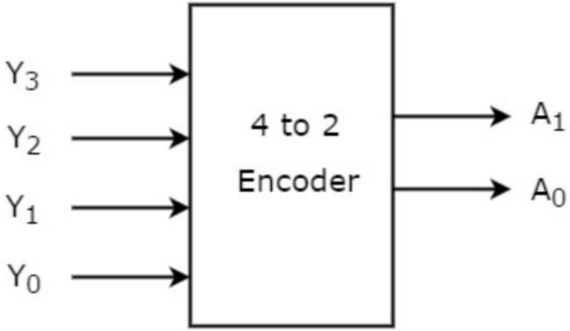


Encoder

- Performs inverse operation of a decoder.
- Encoder can have maximum of 2^n of inputs and n output channels.



4 x 2 Encoder



Y_0	Y_1	Y_2	Y_3	A_1	A_0
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

$$A_1 = Y_3 + Y_2$$

$$A_0 = Y_3 + Y_1$$

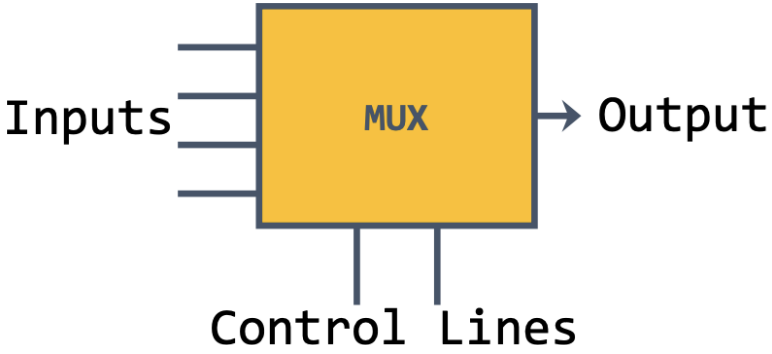
Multiplexers

- One of many input channels is directed to the single output chanel.

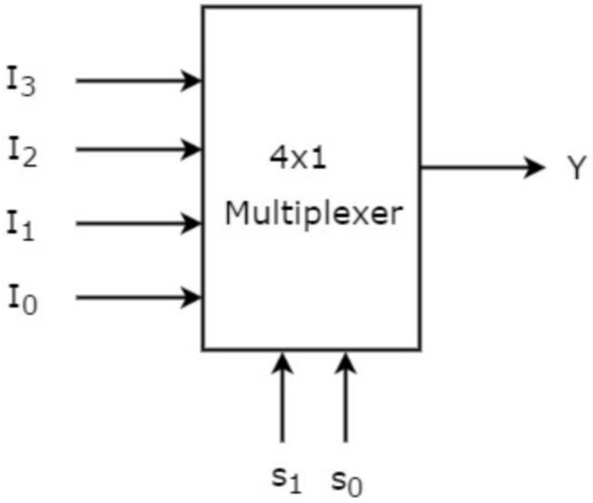
  Several input channels

  Single output channel

  Several control lines to direct an input channel to the output channel
- Also known as a data selector



4 : 1 MUX



s_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$$Y = S_1{}'S_0{}'I_0 + S_1{}'S_0I_1 + S_1S_0{}'I_2 + S_1S_0I_3$$