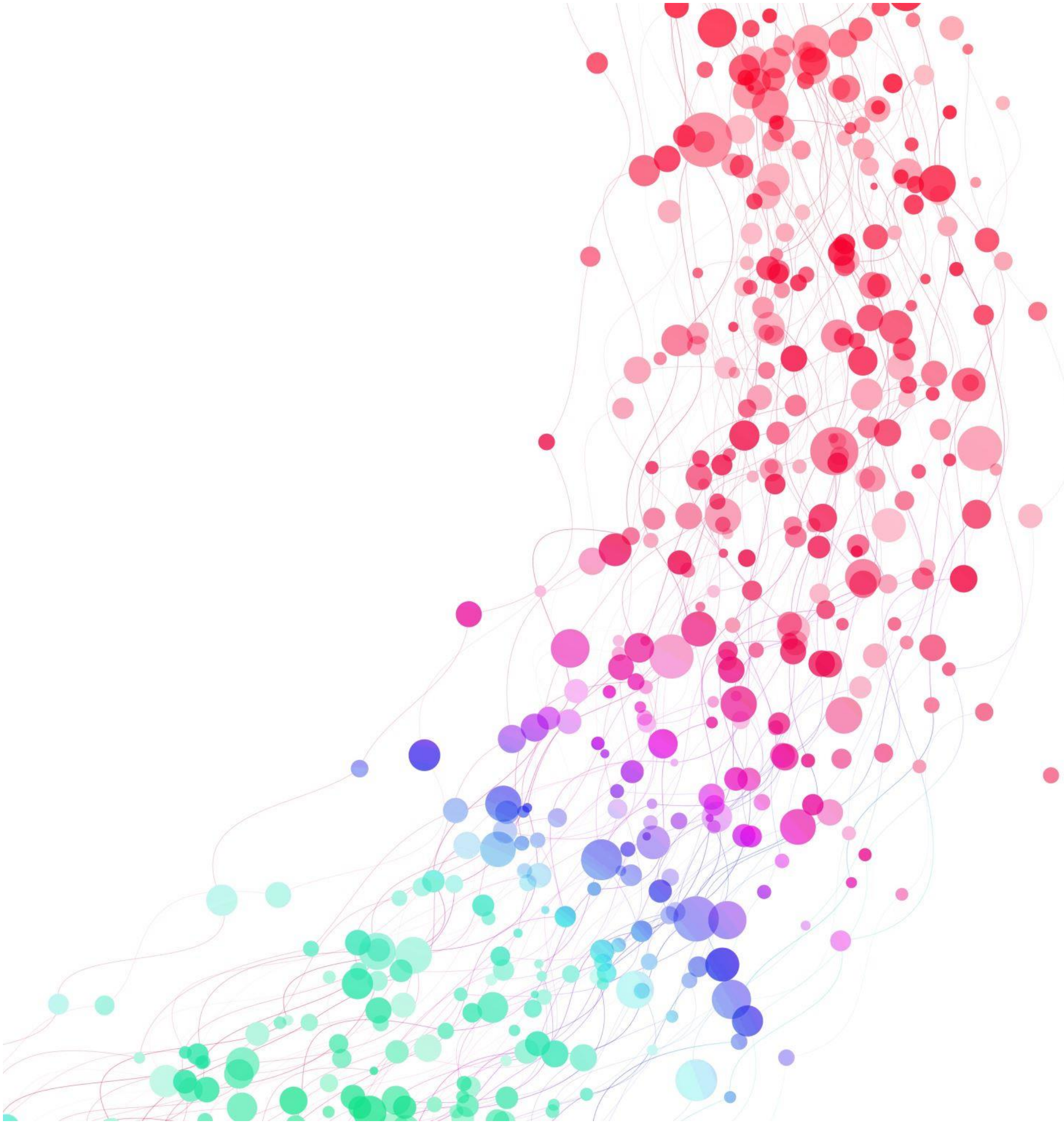
Fundamentals of Electronics

ECE 101

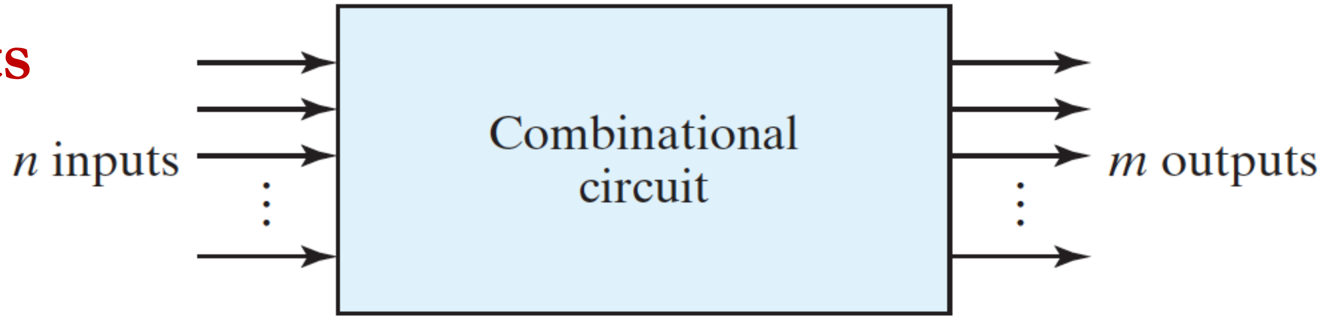


Combinational and Sequential Circuits

- **Combinational circuit:** Logic gates whose outputs at any time are determined from only the present combination of inputs
  - A combinational circuit performs an operation that can be specified logically by a set of Boolean functions.

- **Sequential circuit:** Contain storage elements in addition to logic gates.
  - Their outputs are a function of the inputs and the state of the storage elements.
  - The state of the storage elements is a function of previous inputs
  - The outputs of a sequential circuit depend not only on present values of inputs, but also on past inputs

Combinational Circuits



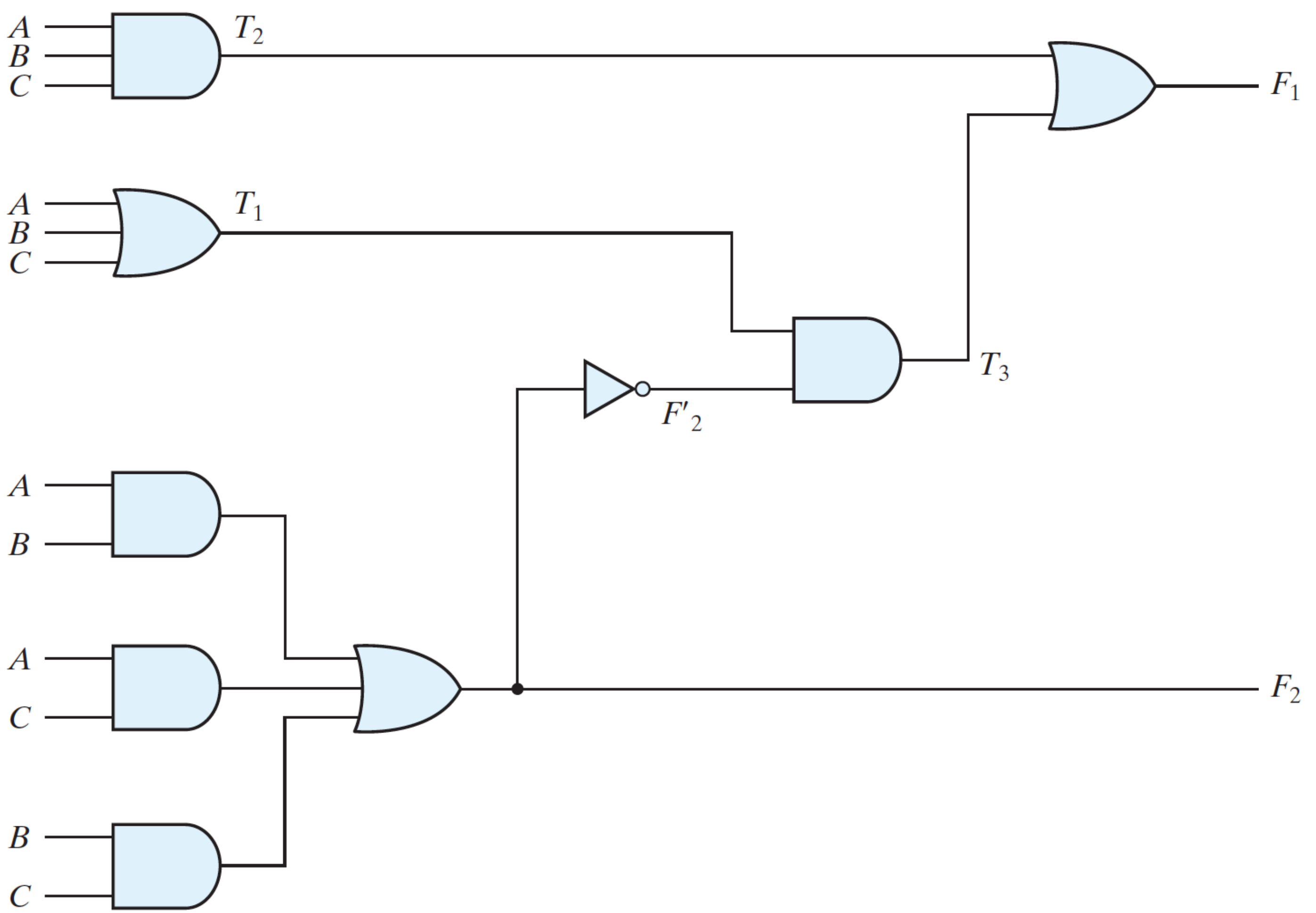
- A combinational circuit can be described by **m** Boolean functions
  - One for each output variable.
- Each output function is expressed in terms of the **n** input variables.
- **The aim here is:**
  - Analyze the behavior of a given logic circuit
  - Synthesize a circuit that will have a given behavior

- There are several combinational circuits that are employed extensively in the design of digital systems. These circuits are available in integrated circuits and are classified as standard components.

Combinational Circuits

- **Examples are:** Adders, subtractors, comparators, decoders, encoders, and multiplexers.
- These components are available in integrated circuits as medium-scale integration (MSI) circuits.
- **The analysis of a combinational circuit requires that we determine the function that the circuit implements.**
- This task starts with a given logic diagram and culminates with a set of Boolean functions, a truth table, or, possibly, an explanation of the circuit operation.
- The diagram of a combinational circuit has logic gates with no feedback paths or memory elements .
- A feedback path is a connection from the output of one gate to the input of a second gate whose output forms part of the input to the first gate.
  - Feedback paths in a digital circuit define a sequential circuit and must be analyzed by special methods

Analysis



$$
\begin{aligned}
F_1 &= T_3 + T_2 = F'_2 T_1 + ABC = (AB + AC + BC)'(A + B + C) + ABC \\
&= (A' + B')(A' + C')(B' + C')(A + B + C) + ABC \\
&= (A' + B'C')(AB' + AC' + BC' + B'C) + ABC \\
&= A'BC' + A'B'C + AB'C' + ABC
\end{aligned}
$$

Binary Adder and Subtractor

Half Adder

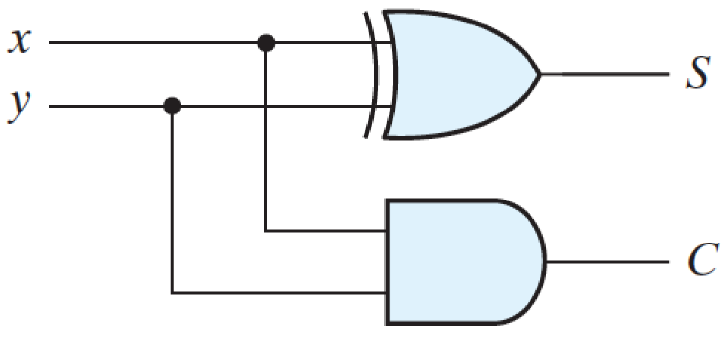
Needs two binary inputs and two binary outputs

A combinational circuit that performs the addition of two bits is called a half adder .

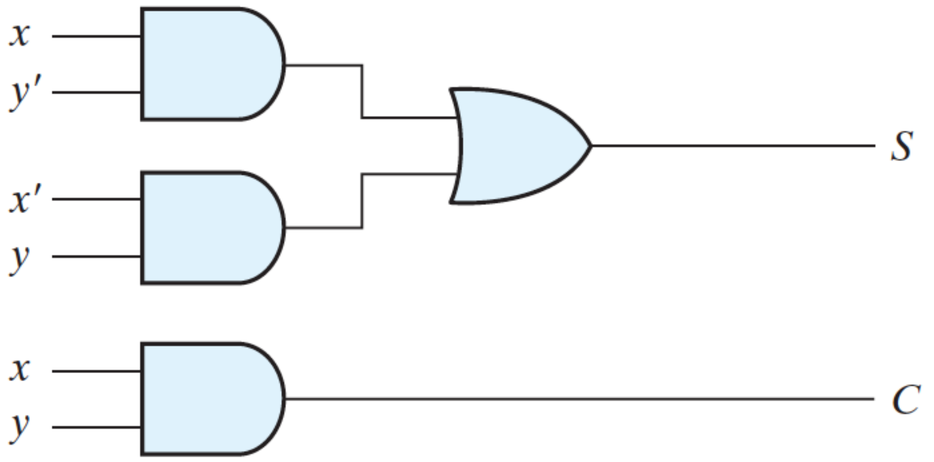
One that performs the addition of three bits (two significant bits and a previous carry) is a full adder .

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$S = x'y + xy'$$
$$C = xy$$



(b) $S = x \oplus y$
$C = xy$

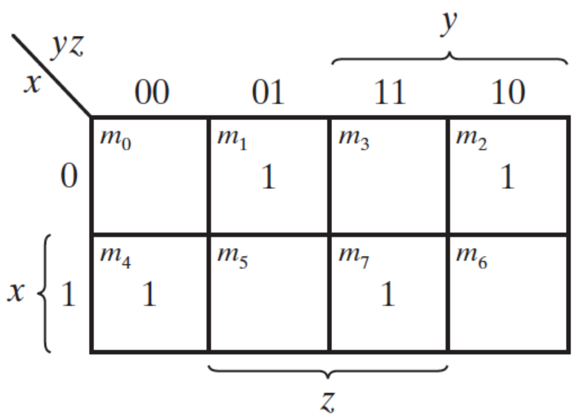


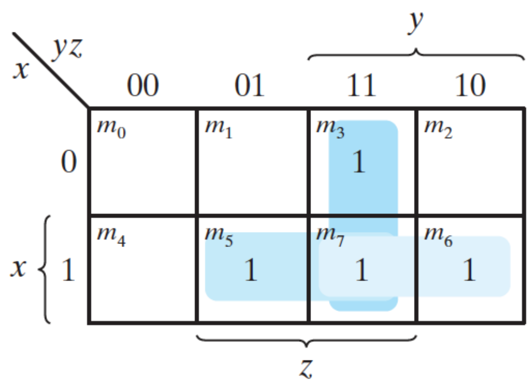
(a) $S = xy' + x'y$
$C = xy$

Full Adder

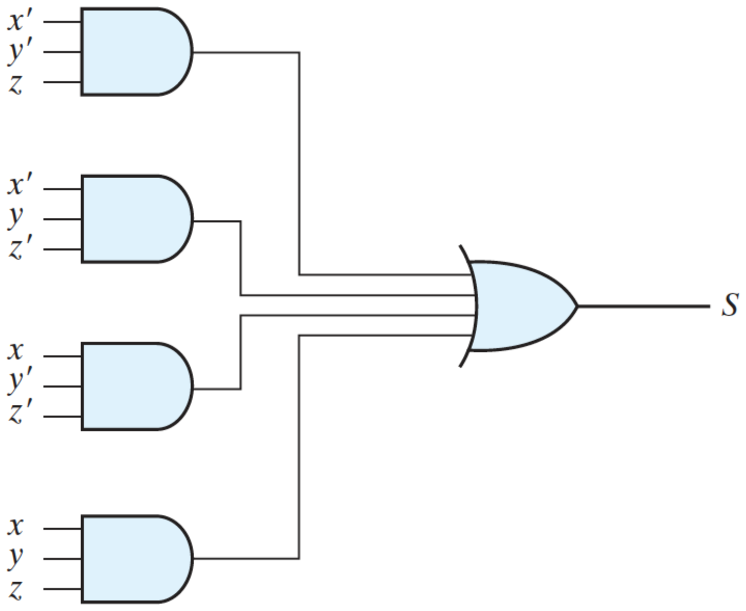
$$A = 1011$$
$$B = 0011$$

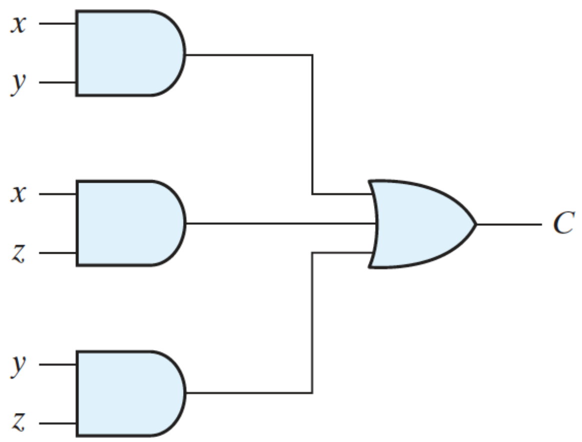
x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



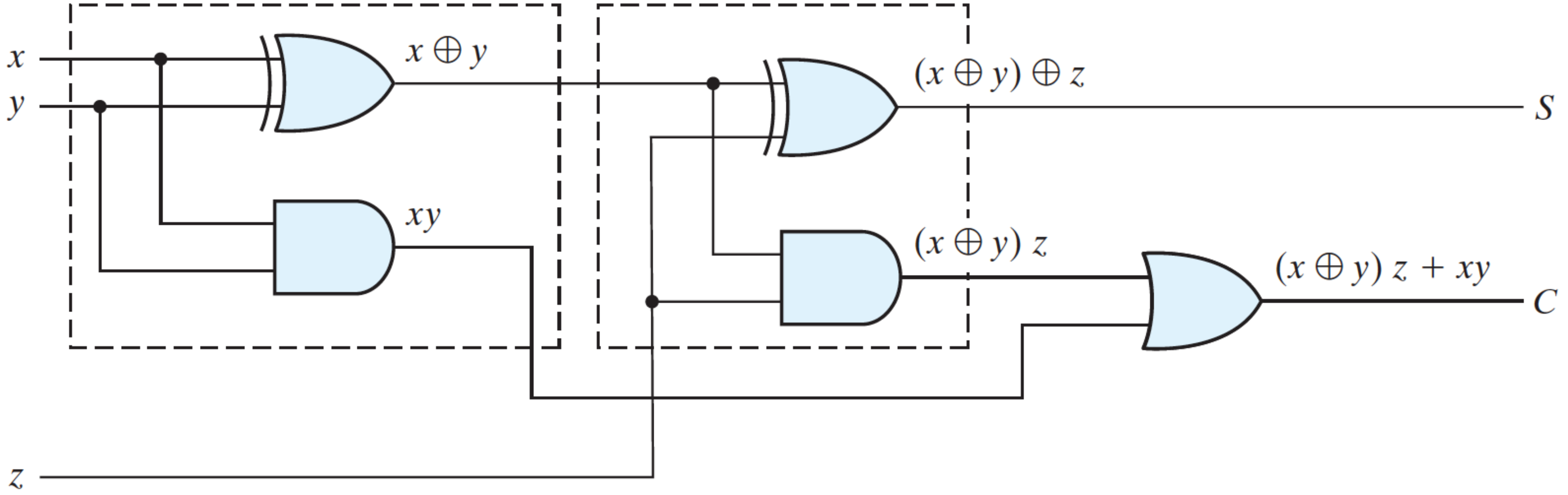


$$S = x'y'z + x'yz' + xy'z' + xyz$$
$$C = xy + xz + yz$$





How do you implement a full adder with 2 half adders?



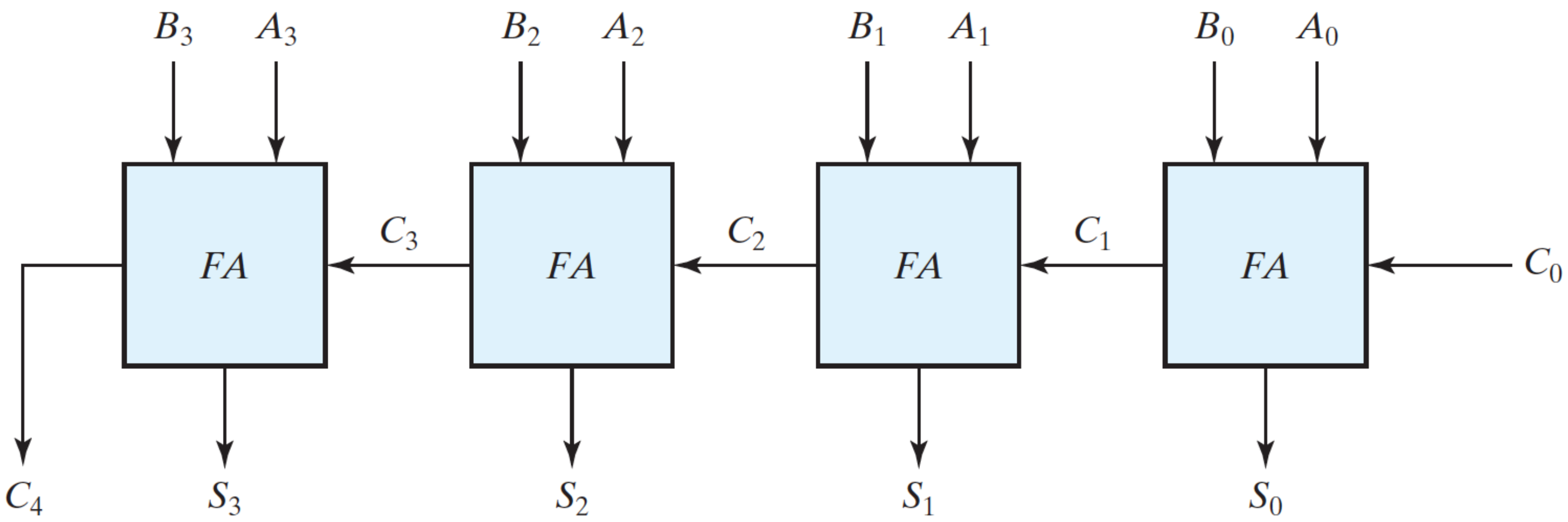
$$\begin{aligned} S &= z \oplus (x \oplus y) \\ &= z'(xy' + x'y) + z(xy' + x'y)' \\ &= z'(xy' + x'y) + z(xy + x'y') \\ &= xy'z' + x'yz' + xyz + x'y'z \end{aligned}$$

$$C = z(xy' + x'y) + xy = xy'z + x'yz + xy$$

Binary Adder

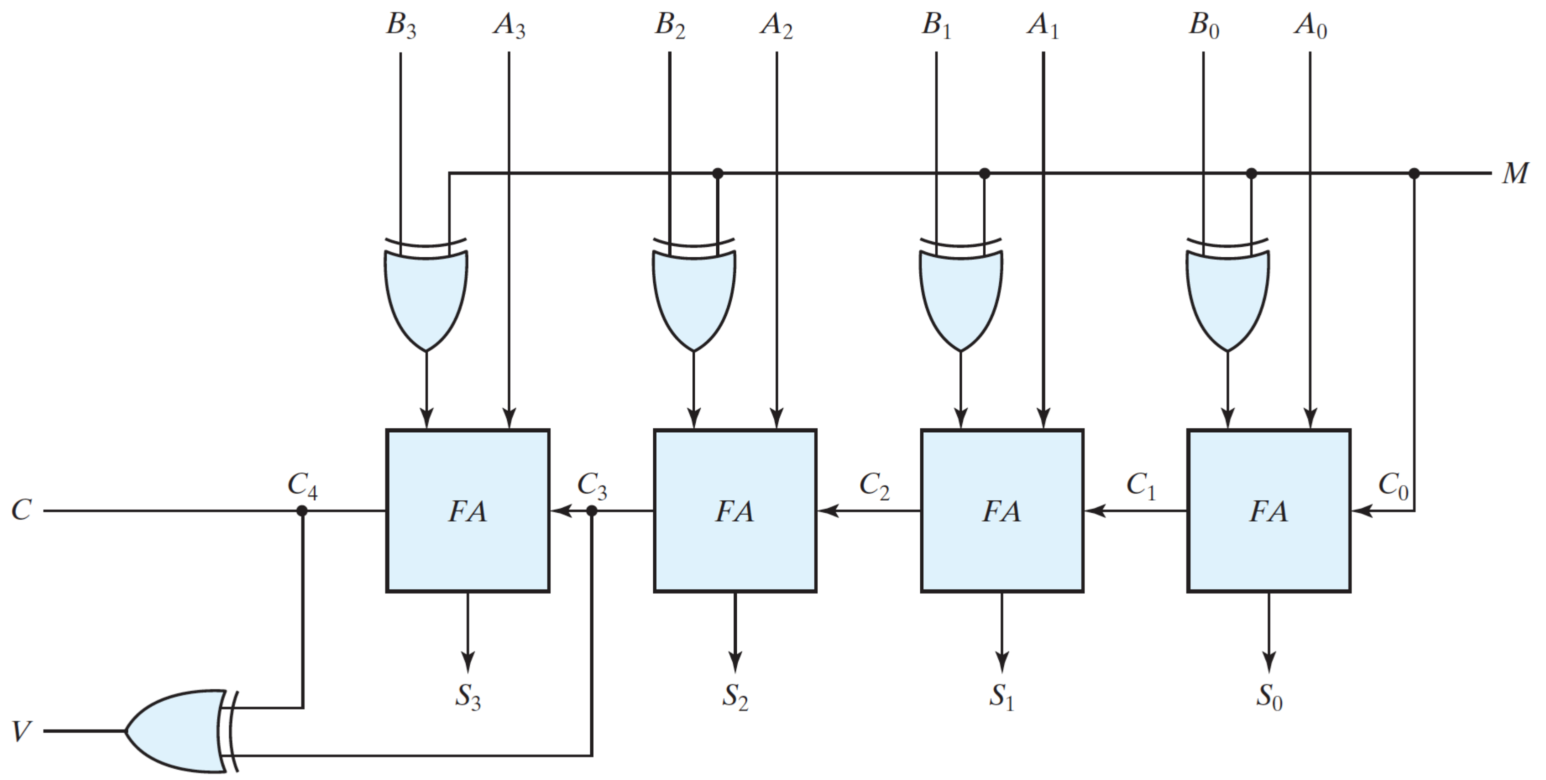
Four bit adder

$$A = 1011$$
$$B = 0011$$
$$S = 1110$$



Binary Subtractor /Adder

- The subtraction A - B can be done by taking the 2's complement of B and adding it to A
- The 2's complement can be obtained by taking the 1's complement and adding 1 to the least significant pair of bits



When M = 0, the circuit is an adder, and when M = 1, the circuit becomes a subtractor.

Decoders

- A binary code of n bits is capable of representing up to 2^n distinct elements of coded information.

- A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines.

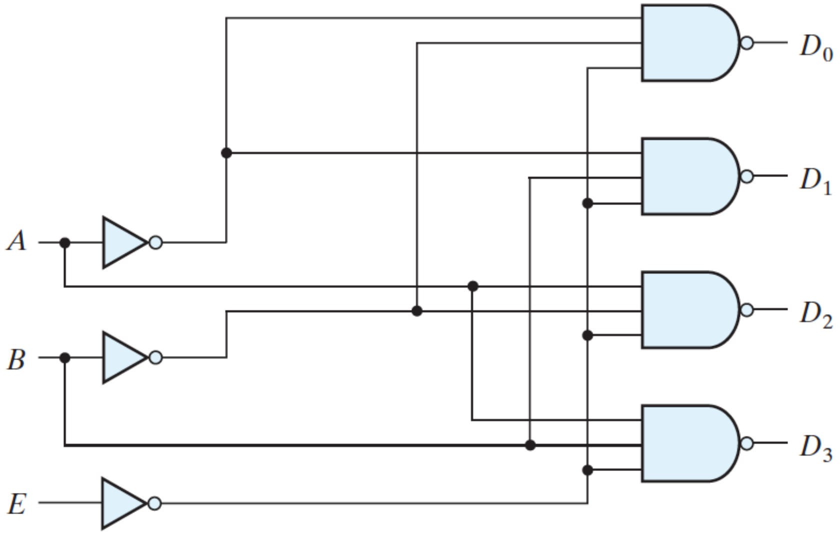
- If the n -bit coded information has unused combinations, the decoder may have fewer than 2n outputs.

Inputs			Outputs							
x	***y***	***z***	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

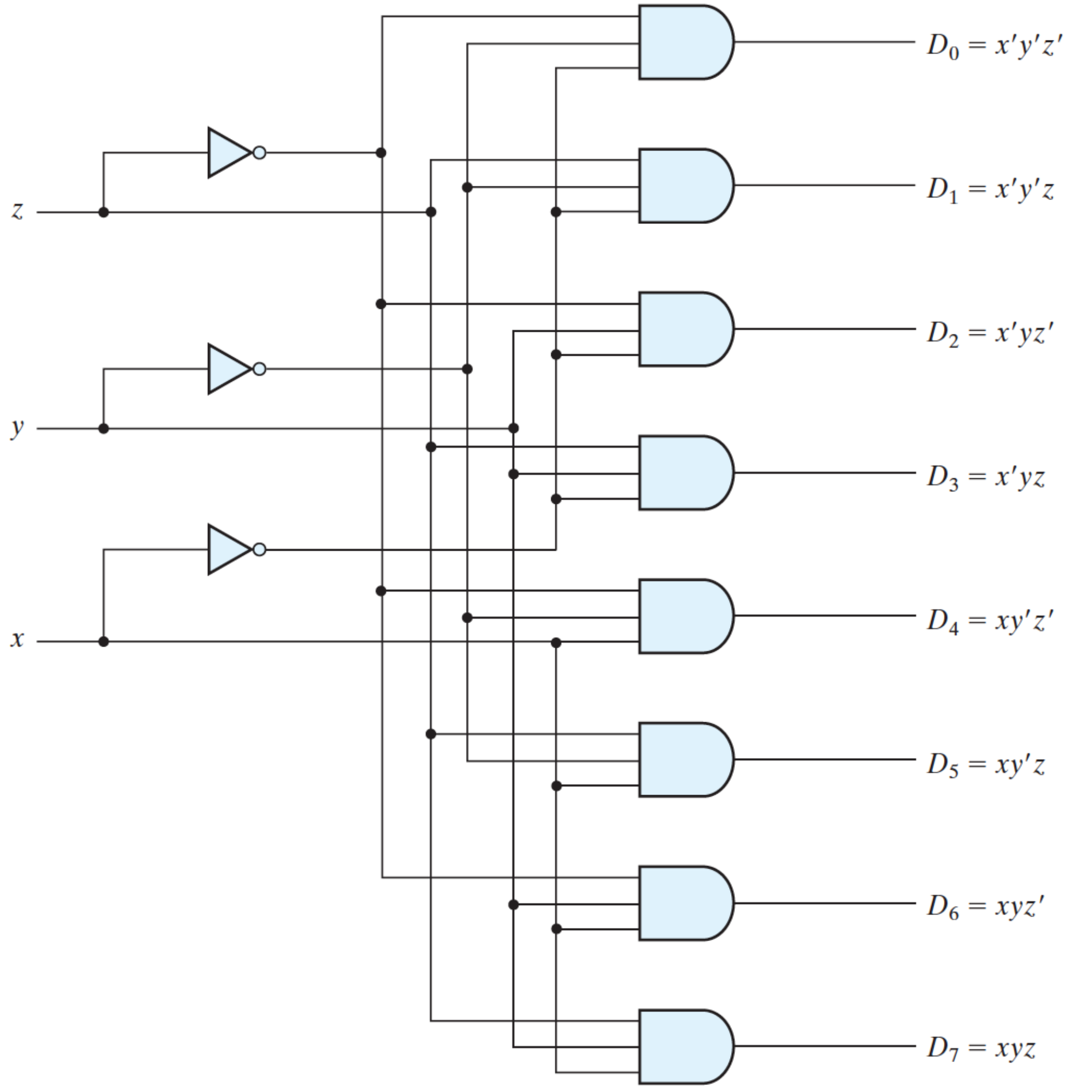
3-to-8 Decoder

Decoders

Inputs			Outputs							
x	***y***	***z***	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1



2-to-4 decoder with Enable



3-to-8 Decoder

Encoders

Performs the inverse operation of a decoder.

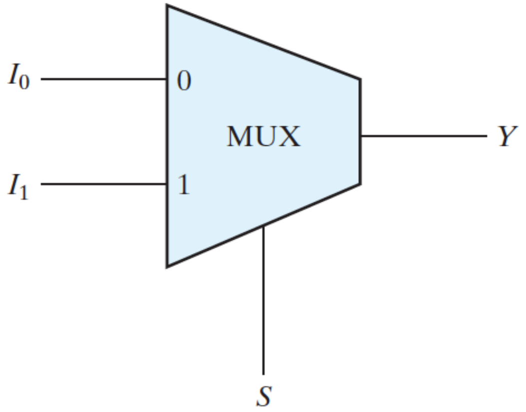
Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$$z = D_1 + D_3 + D_5 + D_7$$
$$y = D_2 + D_3 + D_6 + D_7$$
$$x = D_4 + D_5 + D_6 + D_7$$

Octal-to-Binary Encoder

Multiplexers

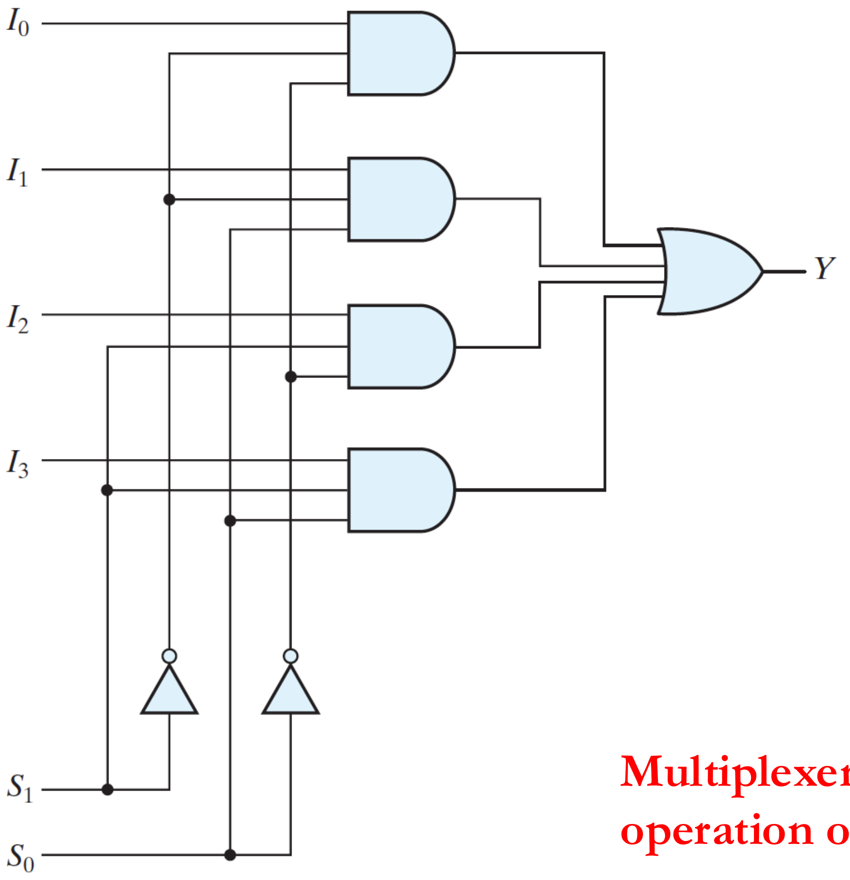
A multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line

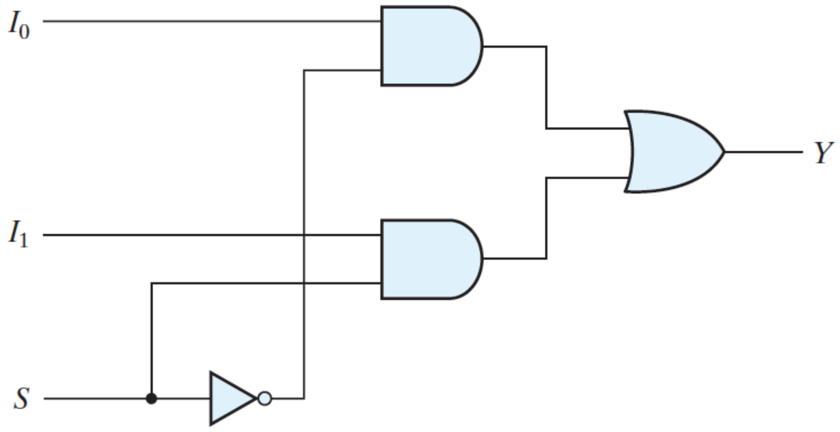


2-to-1 Mux

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

4-to-1 Mux





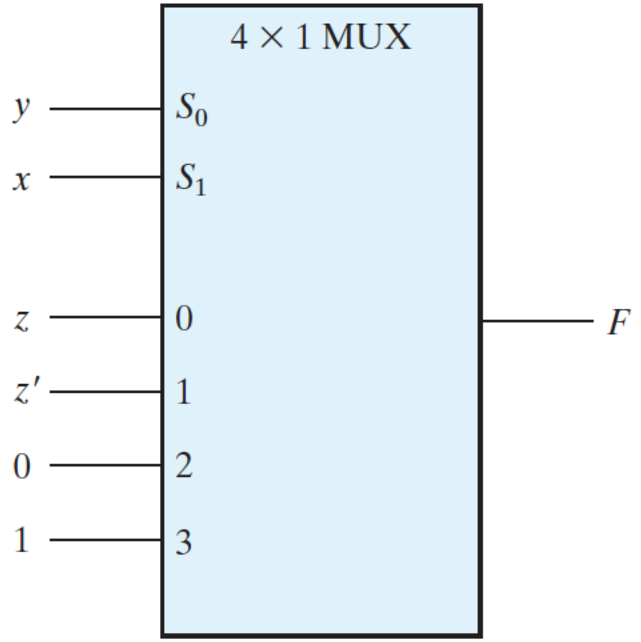
Multiplexers may have an <u>Enable input</u> to control the operation of the unit.

Boolean Function Implementation using MUX

Example:

$$F(x, y, z) = \Sigma(1, 2, 6, 7)$$

x	y	z	F	
0	0	0	0	$F = z$
0	0	1	1	
0	1	0	1	$F = z'$
0	1	1	0	
1	0	0	0	$F = 0$
1	0	1	0	
1	1	0	1	$F = 1$
1	1	1	1	

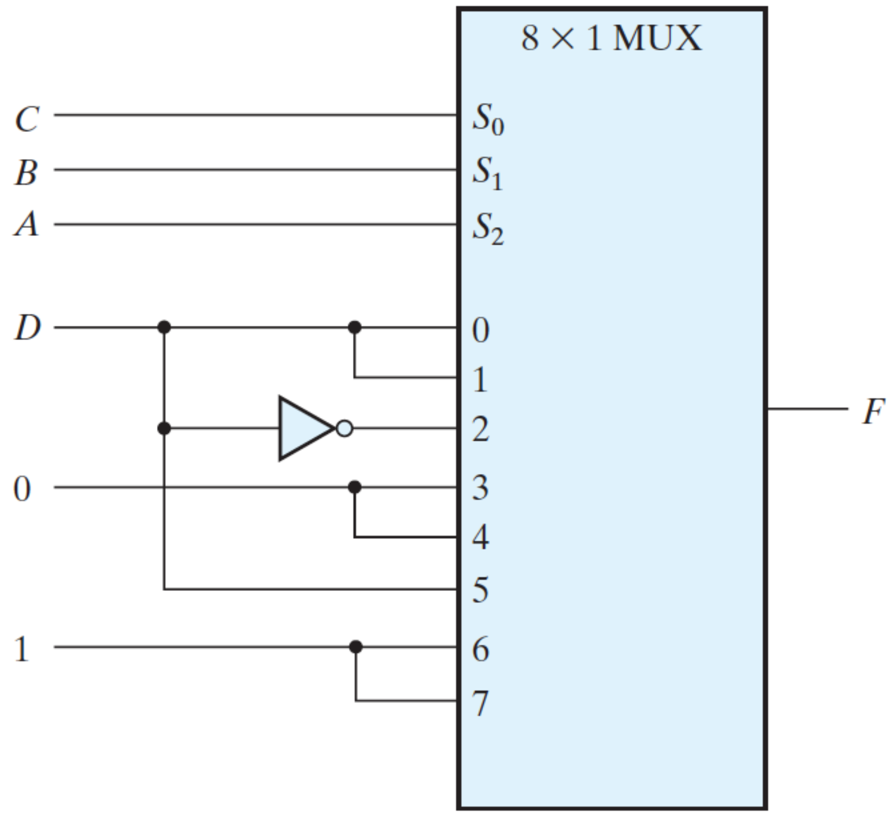


Example:

$$F(A, B, C, D) = \Sigma(1, 3, 4, 11, 12, 13, 14, 15)$$

$$F(A, B, C, D) = \Sigma(1, 3, 4, 11, 12, 13, 14, 15)$$

A	B	C	D	F	
0	0	0	0	0	$F = D$
0	0	0	1	1	
0	0	1	0	0	$F = D$
0	0	1	1	1	
0	1	0	0	1	$F = D'$
0	1	0	1	0	
0	1	1	0	0	$F = 0$
0	1	1	1	0	
1	0	0	0	0	$F = 0$
1	0	0	1	0	
1	0	1	0	0	$F = D$
1	0	1	1	1	
1	1	0	0	1	$F = 1$
1	1	0	1	1	
1	1	1	0	1	$F = 1$
1	1	1	1	1	



Thank you