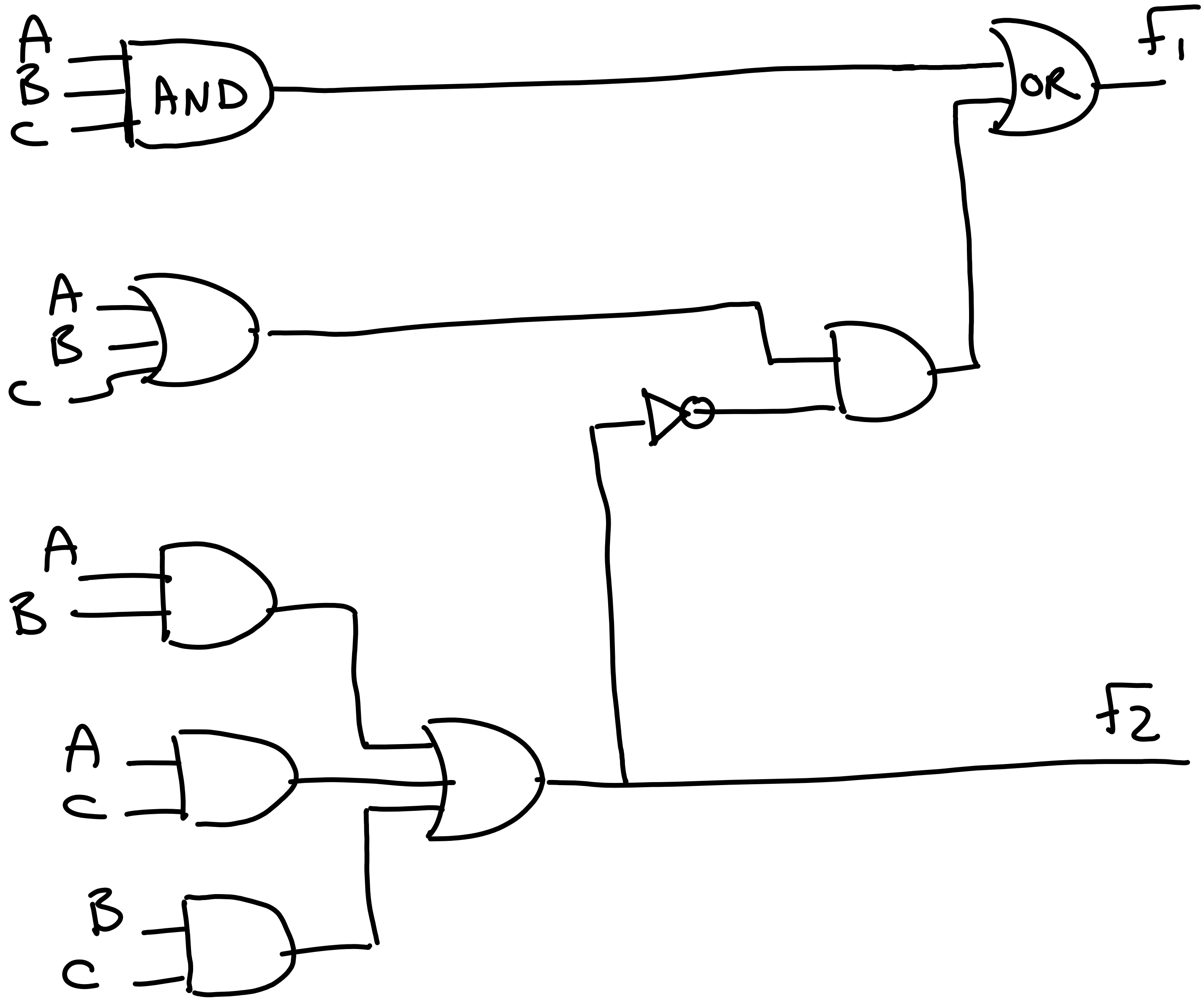
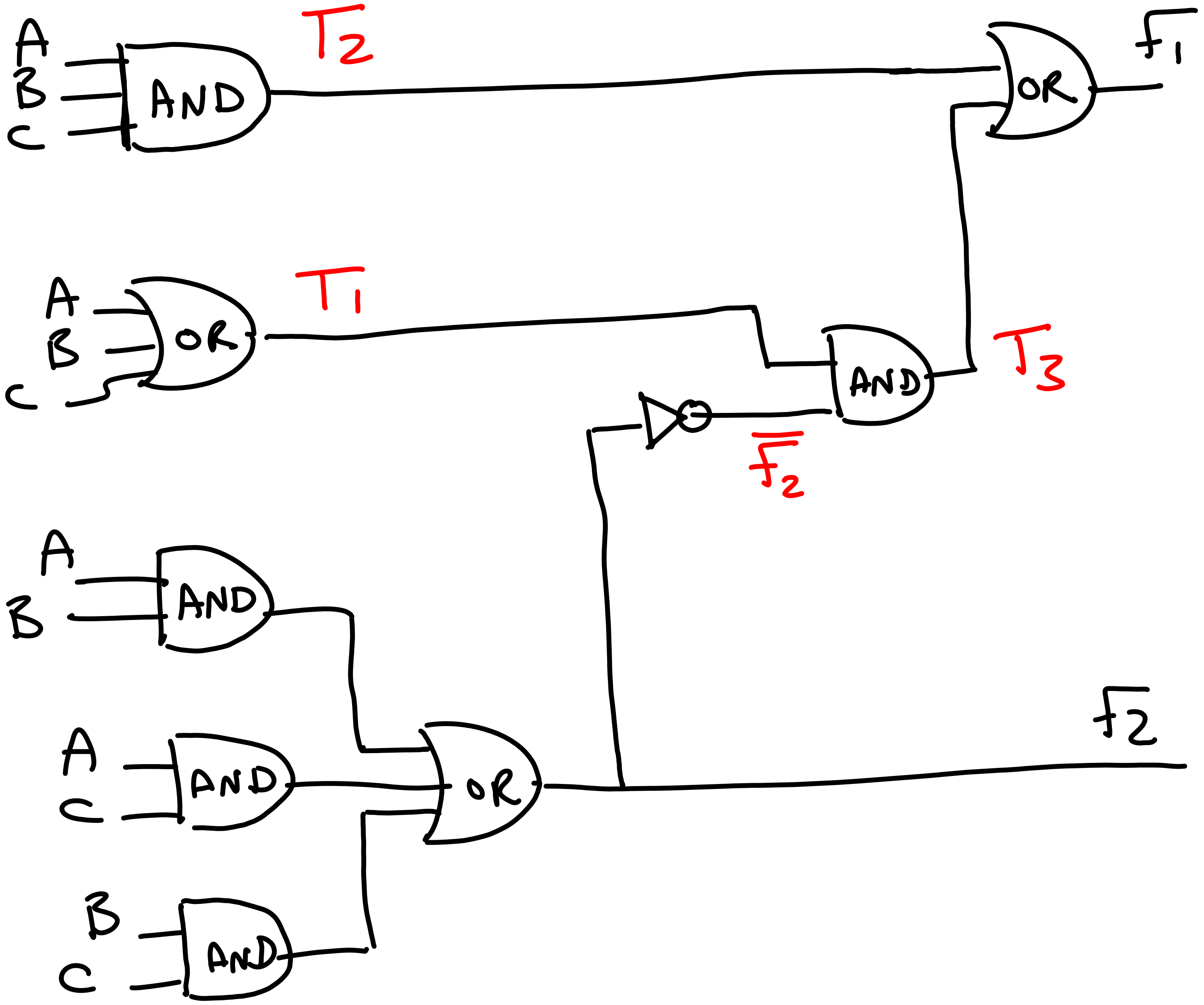
Medium Scale Integrated Circuit MSI

Ex

what are the output function F_1, F_2?





A	B	C	T_2	T_1	F_2	$\overline{f_2}$	T_3	F_1
0	0	0	0	0	0	1	0	0
0	0	1	0	1	0	1	1	1
0	1	0	0	1	0	1	1	1
0	1	1	0	1	1	0	0	0
1	0	0	0	1	0	1	1	1
1	0	1	0	1	1	0	0	0
1	1	0	0	1	1	0	0	0
1	1	1	1	1	1	0	0	1

F_2

A \ BC	00	01	11	10
0			1	
1		1	1	1

$$F_2 = AC + AB + BC$$

A \ BC	00	01	11	10
0		1		1
1	1		1	

$$F_1 = A \oplus B \oplus C$$

Ex Design a logic circuit that convert a BCD digit into Excess-3

A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0				
1	0	1	1				
1	1	0	0				
1	1	0	1				
1	1	1	0				
1	1	1	1				

Do Not Care

$$W = \sum(5,6,7,8,9)$$

$$X = \sum(1,2,3,4,9)$$

$$Y = \sum(0,3,4,7,8)$$

$$Z = \sum(0,2,4,6,8) \Rightarrow \text{by inspection}$$

$$Z = \overline{D}$$

AB \ CD	00	01	11	10
00				
01		1	1	1
11	X	X	X	X
10	1	1	X	X

$$W = A + BD + BC$$

AB \ CD	00	01	11	10
00		1	1	1
01	1			
11	X	X	X	X
10		1	X	X

$$X = B\overline{C}\overline{D} + \overline{B}D + \overline{B}C$$

AB \ CD	00	01	11	10
00	1		1	
01	1		1	
11	X	X	X	X
10	1		X	X

$$Y = \overline{C}\overline{D} + CD$$

Prob

Sketch the logic diagram of the Circuit above.

Ex

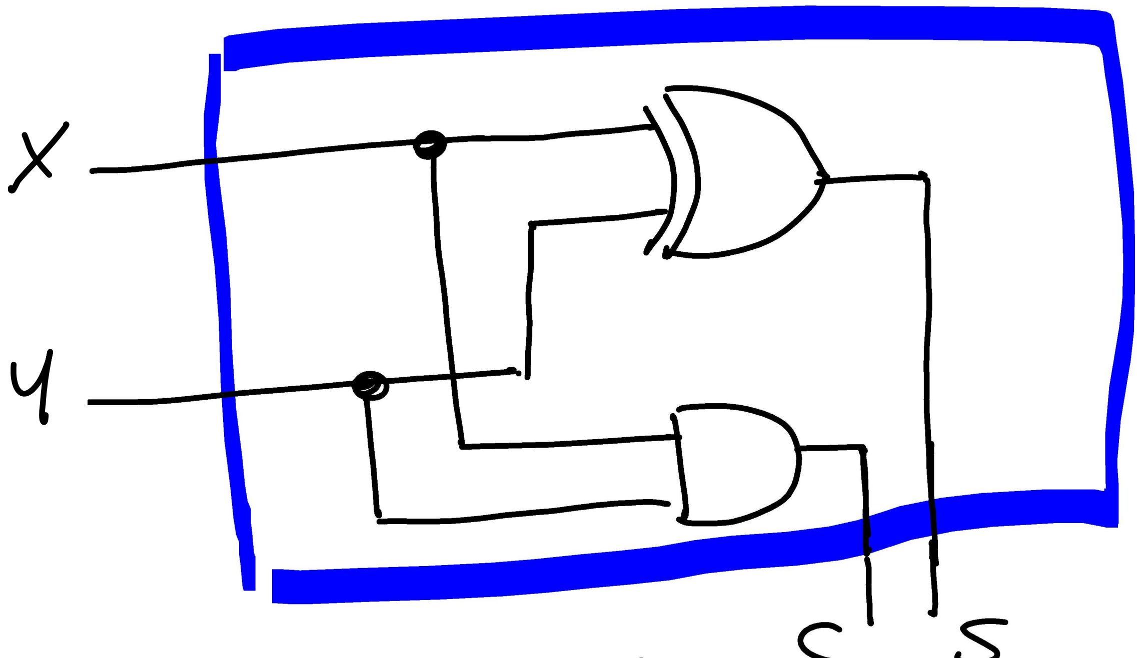
Design a logic circuit that can add two 1-bit number.

X	Y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

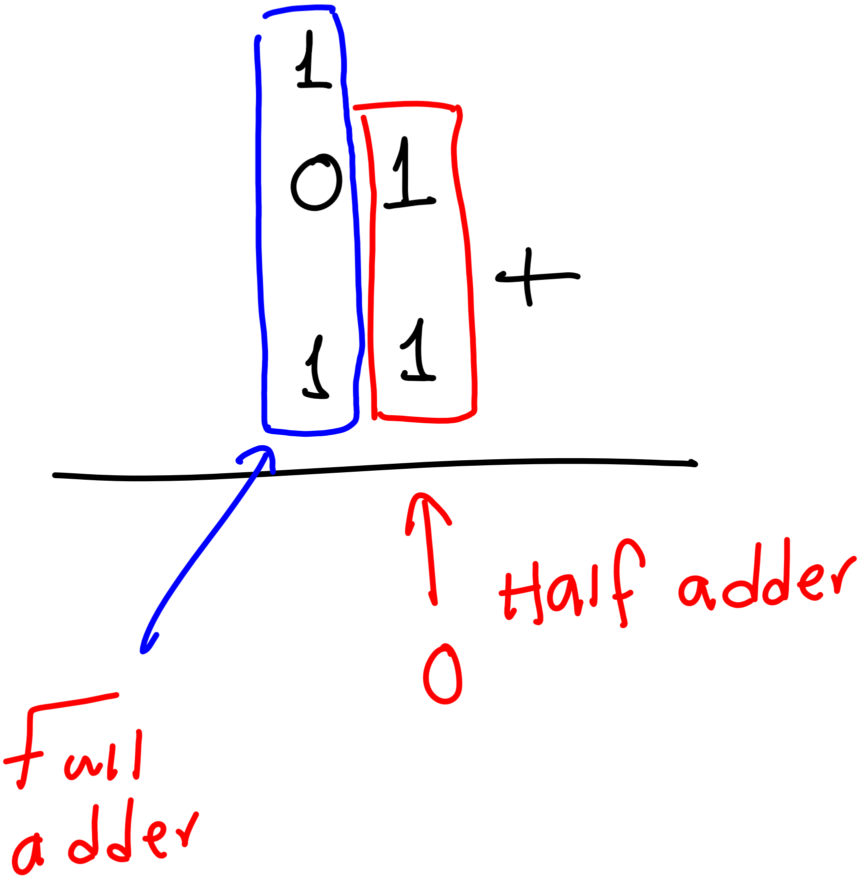
$\Rightarrow$ you do not need to go to karnaugh

$$S = \bar{X}Y + X\bar{Y}$$

$$C = XY$$



Half adder



Ex

Design a digital logic circuit that can add two 1-bit number in addition to carry in

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

$\Rightarrow$

$$C = \sum(3,5,6,7)$$

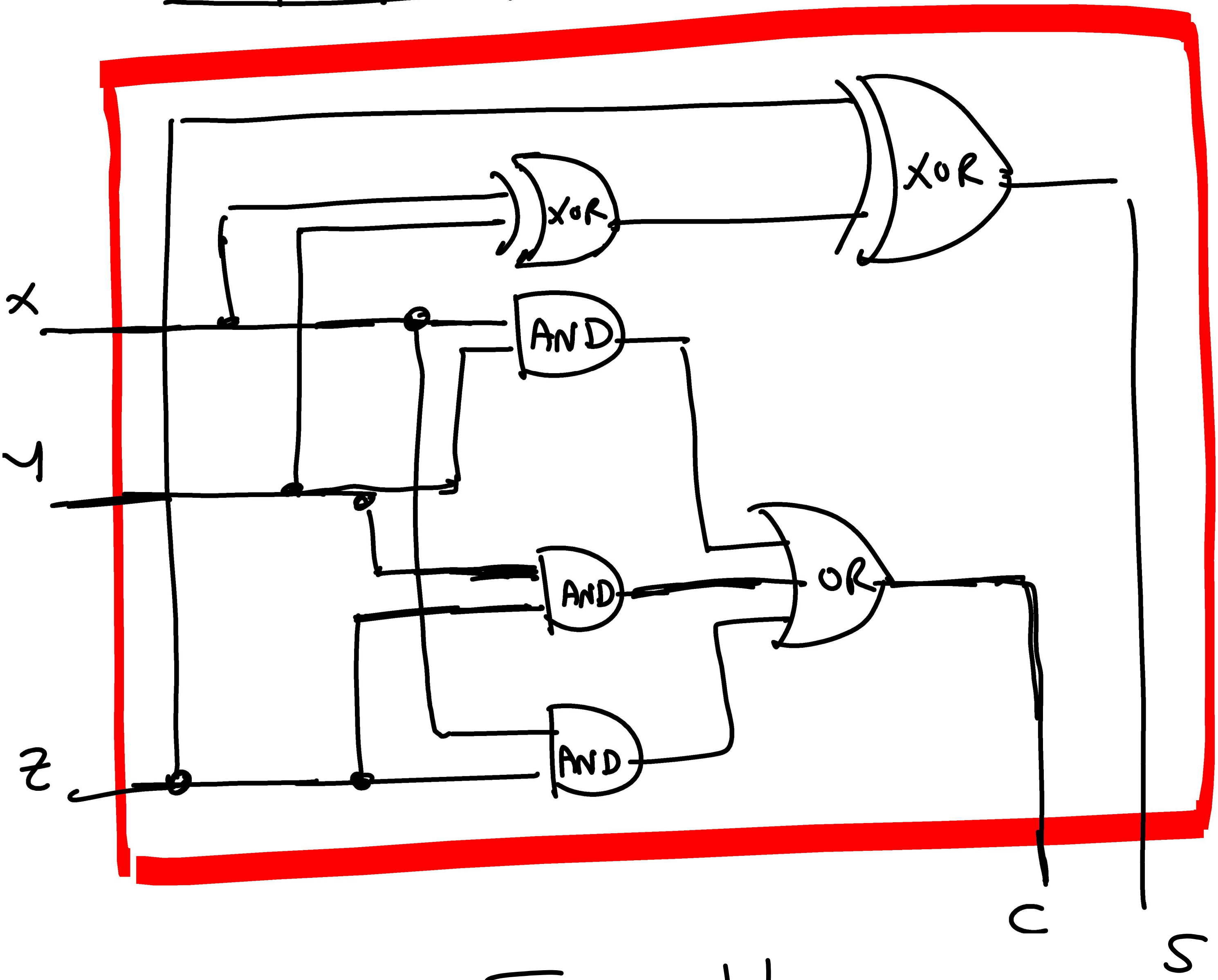
$$S = \sum(1,2,4,7)$$

x \ yz	00	01	11	10
0			1	
1		1	1	1

$C = XY + XZ + YZ$

x \ yz	00	01	11	10
0		1		1
1	1		1	

$S = X \oplus Y \oplus Z$



full adder

Note

$$\therefore \quad C = XY + XZ + YZ$$

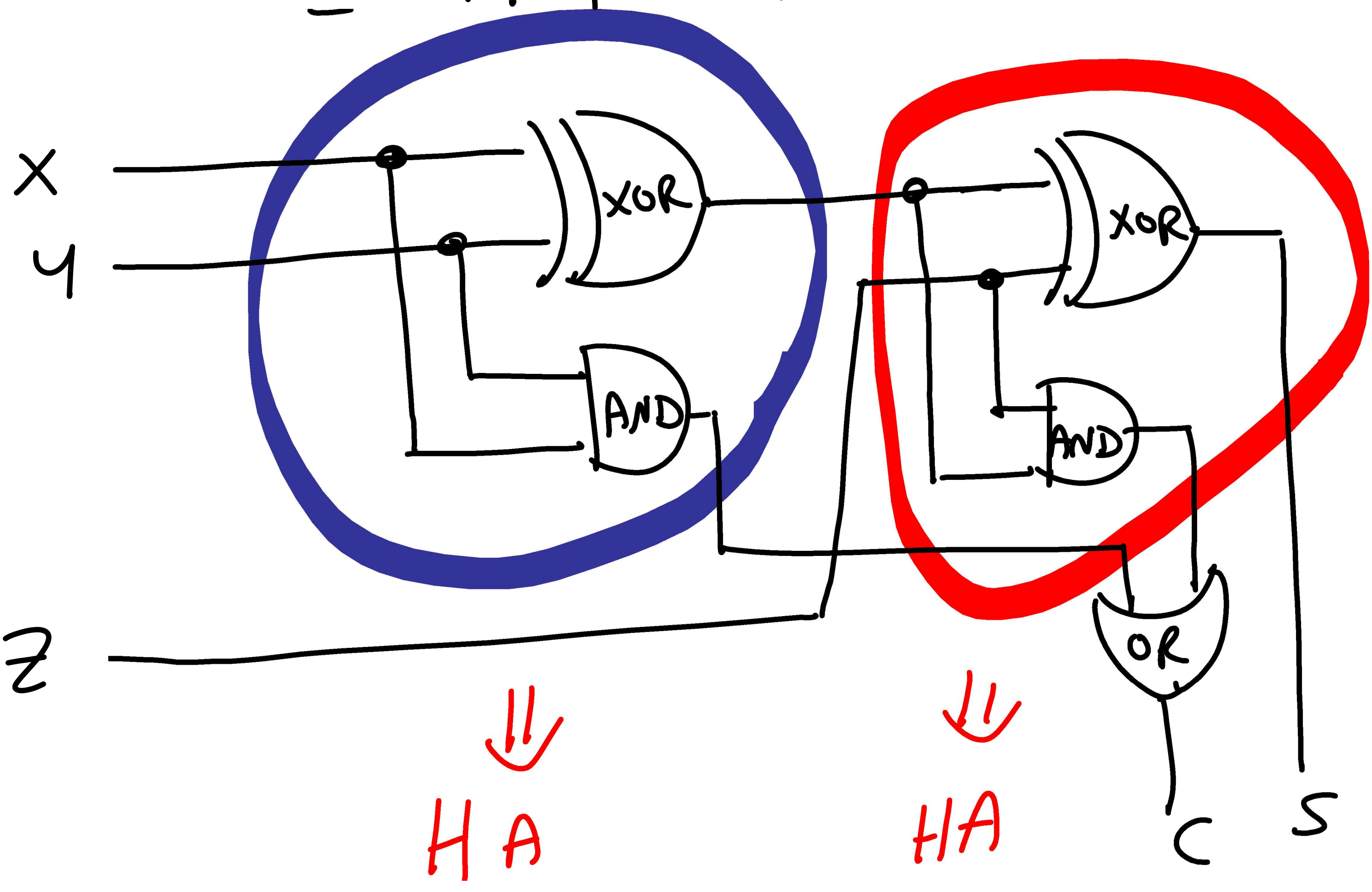
$$= XY + Z(X + Y)$$

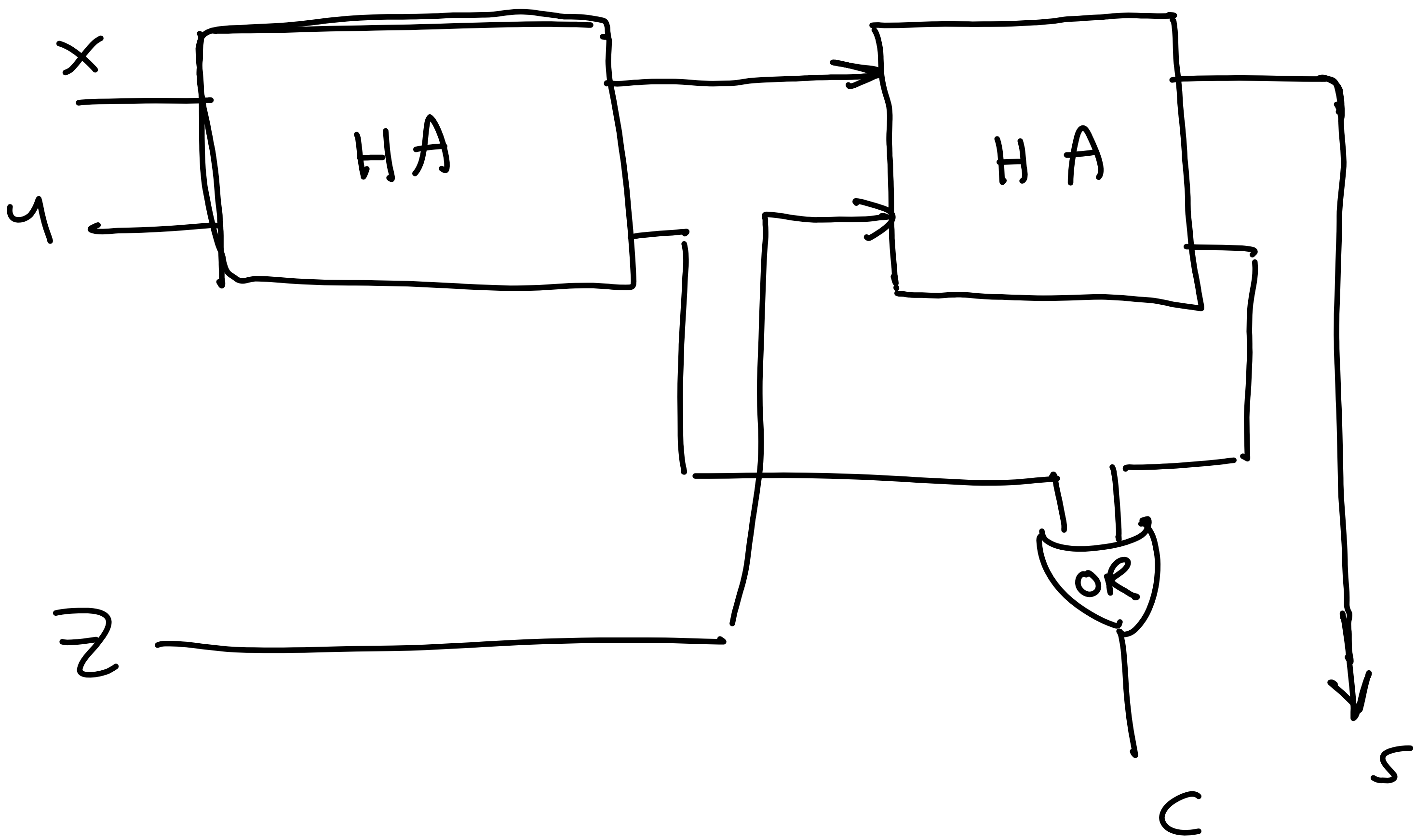
$$= XY + Z(X(Y+\bar{Y}) + (X+\bar{X})Y)$$

$$= XY + Z(XY + X\bar{Y} + XY + \bar{X}Y)$$

$$= XY + Z(XY + X \oplus Y)$$

$$= XY + Z(X \oplus Y)$$





x
y
HA
HA
OR
z
C
S

Ex Design a logic circuit that can add two 2-bit number

A_1	A_0	B_1	B_0	C_2	C_1	C_0

why 3 bits in output

Ex

Design a logic circuit that can add two 4-bit number

A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	C_4	C_3	C_2	C_1	C_0

$\Downarrow$

+1 carry

9 inputs

5 output

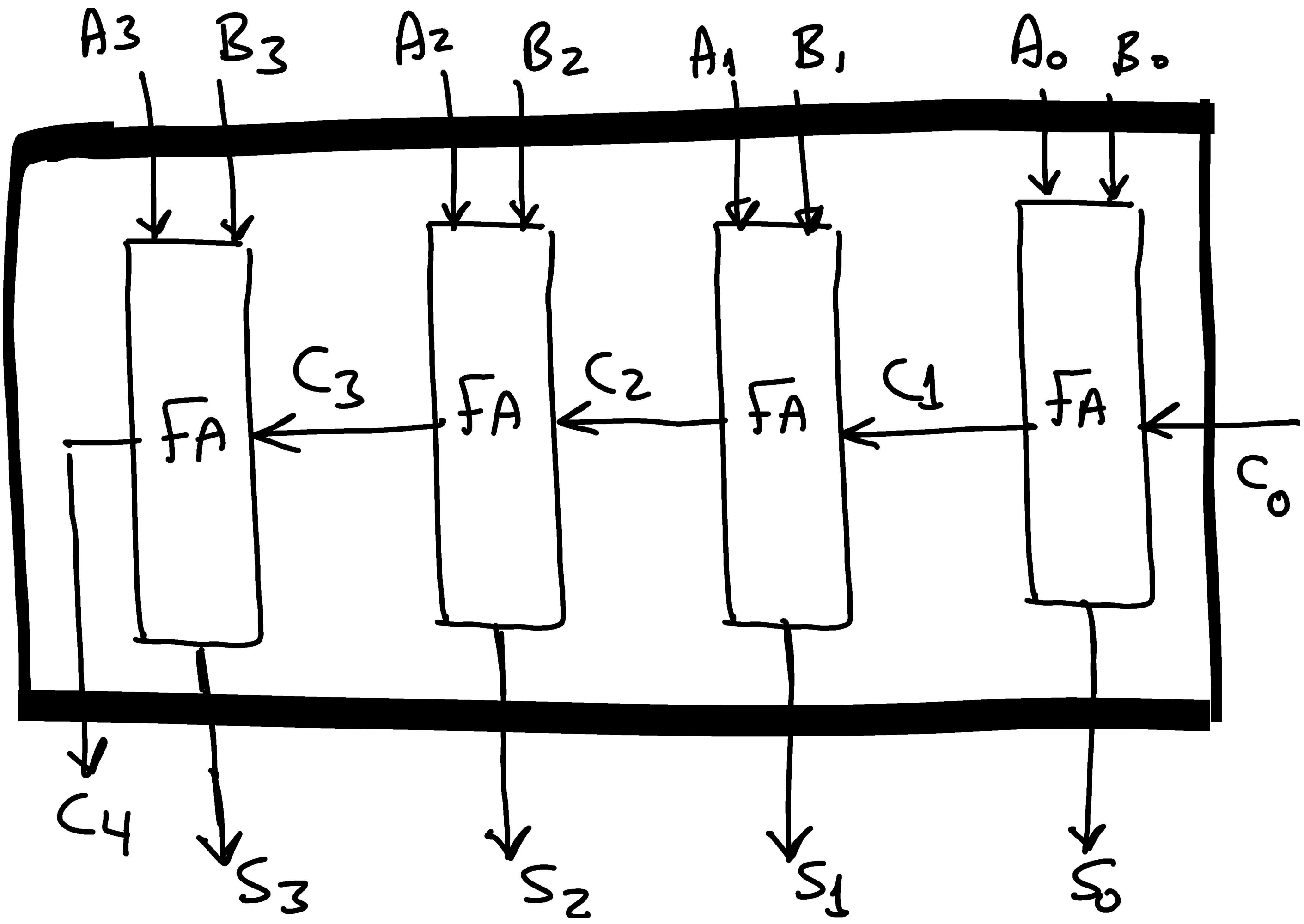
2^9 entries

Hierarchical Design

- We divide our design into smaller functional block
- Connect functional units to produce the big functionality
- How can we build an n-bit adder from full_adders?
    - one adder for each bit (n total)
    - Connect carry to next adder's input
    - Sequence of sums and final carry are results

Note

$$
\begin{array}{ccccc}
 & 1 & 1 & 1 & 0 \\
 & 1 & 1 & 0 & 1 \\
+ & 0 & 0 & 1 & 1 \\
\hline
1 & 0 & 0 & 0 & 0
\end{array}
$$



Note

- the addition of two binary numbers in parallel implies that all the bits of the augend and addend are available for computation at the same time

- As in any combinational circuit, the signal must propagate through the gates before the correct output sum is available in the output terminals

- the total propagation time is equal to the propagation delay of a typical gate times the number of gate levels in the circuit

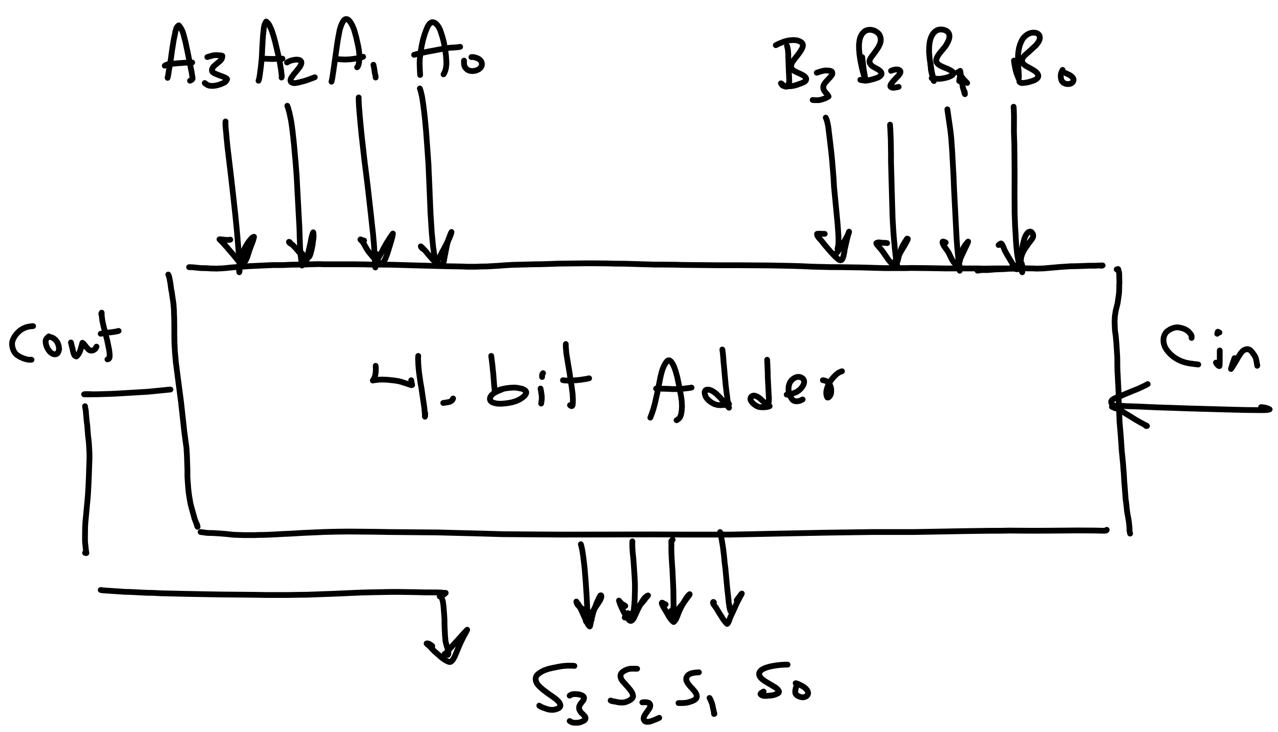
- the longest propagation delay time in an adder is the time it takes the carry to propagate through the full-adders

Report

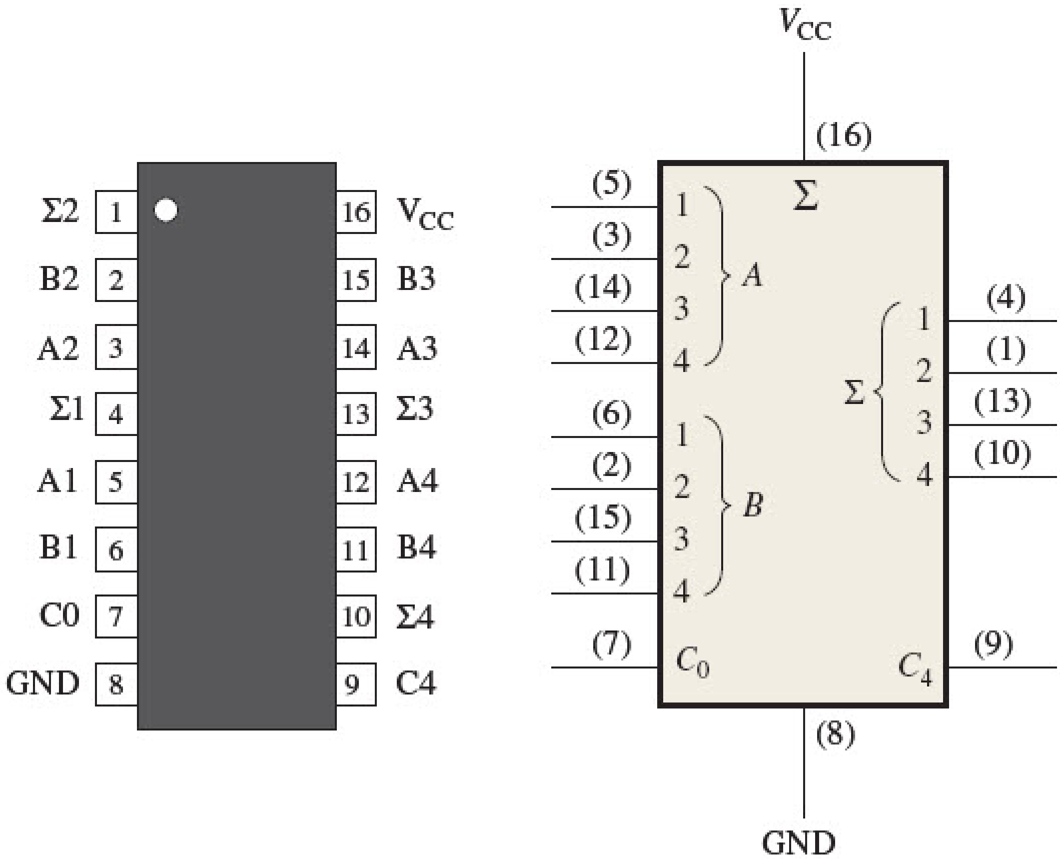
- What is propagation delay time?
- How can we know AND Gate propagation time?
- How can we speed up addition?
  - ↳ **Note** carry lookahead
- Carry lookahead Generator
- carry lookahead adder

Info Note

- Addition is performed by processors on two numbers at a time, called operands
- The source operand is a number that is to added to an existing number called destination operand, which is held in an ALU register, such as the accumulator
- the sum of the two numbers is then stored back in the accumulator



Practical Session



(a) Pin diagram (b) Logic symbol

The 74HC283/74LS283 4-bit parallel adder.

Report

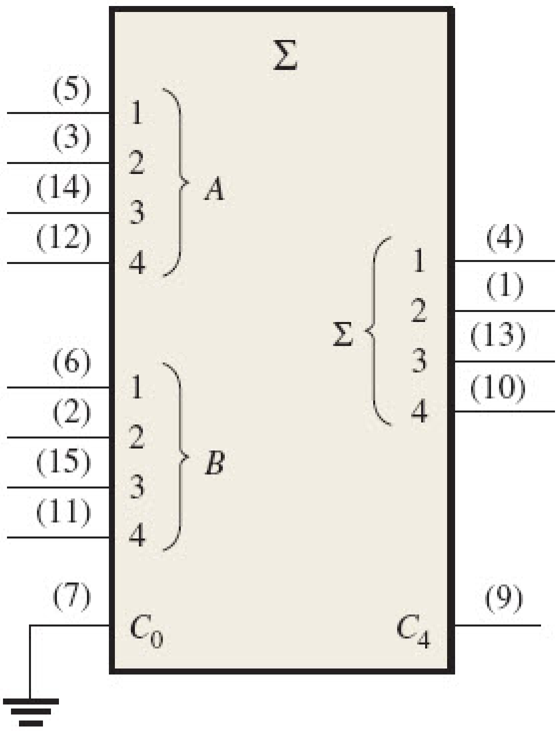
- MultiSim
- proteus
- Printed circuit board
- Go to electronic shop

Note

the 4-bit parallel adder can be expanded to handle more bits by cascading

Report

show how two 74HC283 can be connected to form an 8-bit parallel adder



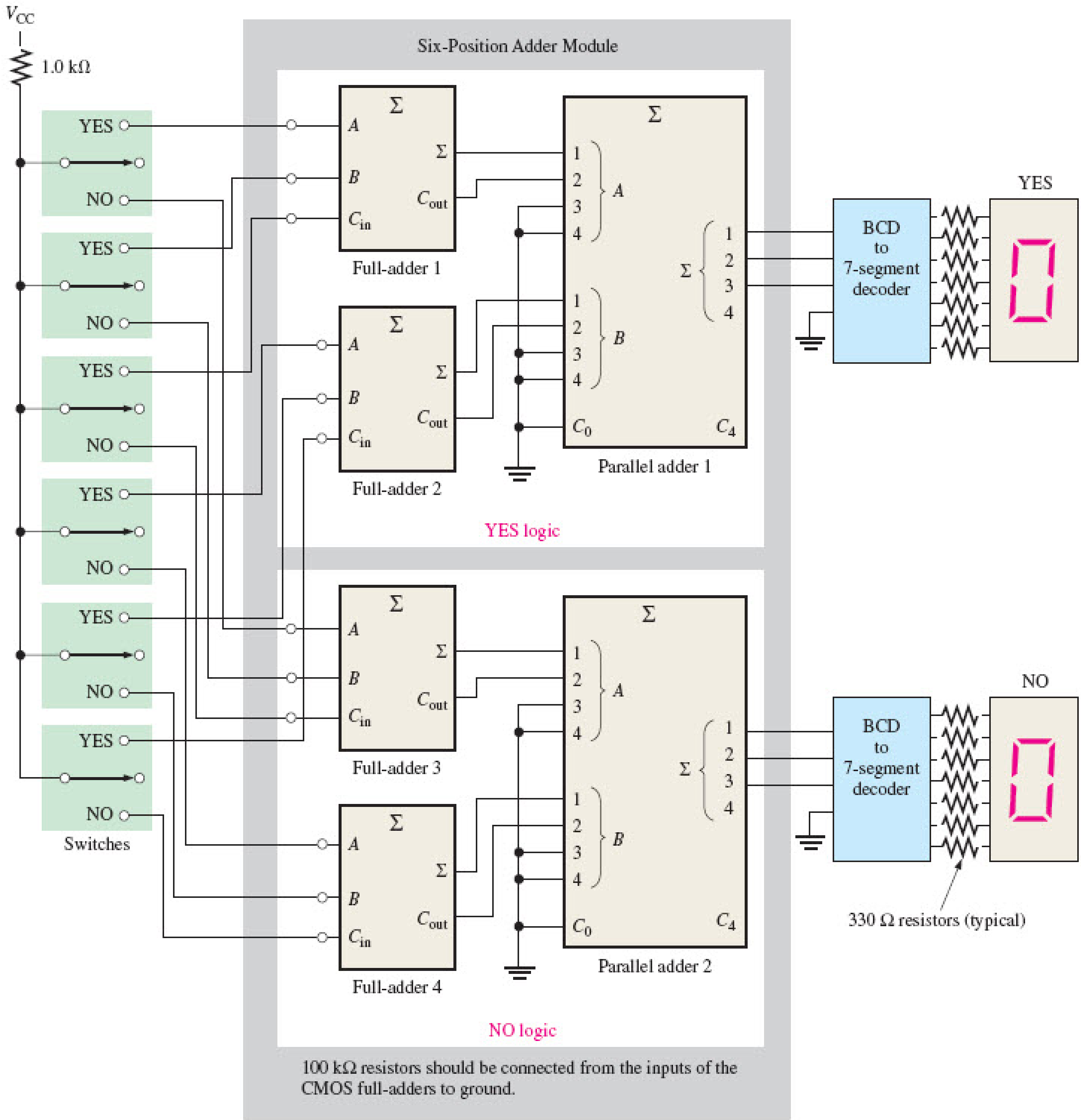
Report

use 74HC283 adders to implement a 12-bit parallel adder.

Voting System

An example of full-adder and parallel adder application is a simple voting system that can be used to simultaneously provide the number of "yes" votes and the number of "no" votes.

this type of system can be used where a group of people are assembled and there is a need for immediately deteimining opinions (for or against), making decisions, or voting on certain issues or other matters



V_{CC}
1.0 kΩ
YES
NO
Switches
Six-Position Adder Module
Σ
A
B
C_{in}
C_{out}
Full-adder 1
Full-adder 2
Full-adder 3
Full-adder 4
Parallel adder 1
Parallel adder 2
C_0
C_4
YES logic
NO logic
BCD to 7-segment decoder
330 Ω resistors (typical)
100 kΩ resistors should be connected from the inputs of the CMOS full-adders to ground.

Next lecture ToC

- Subtractors
- Multipliers
- Comparators

using 4-bit

excess-3 Converter

o ..