CSE 20
DISCRETE MATH

Fall 2020

http://cseweb.ucsd.edu/classes/fa20/cse20-a/

Today's learning goals

- Represent negative integers in multiple ways
- Perform arithmetic operations on integers using multiple representations
- Relate algorithms for integer operations to bitwise boolean operations
- List the truth tables and meanings for conjunction, disjunction, exclusive or.

In pre-class reading, you saw

- Section 4.2 definition of One's, Two's complement (p. 256).
- Section 1.2 definition of logic gates and circuits (pp. 20-21).

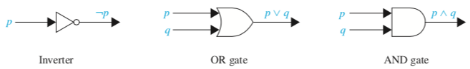


FIGURE 1 **Basic logic gates.**

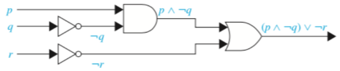


FIGURE 2 **A combinatorial circuit.**

Complicated digital circuits can be constructed from three basic circuits, called **gates**, shown in Figure 1. The **inverter**, or **NOT gate**, takes an input bit p, and produces as output $\neg p$. The **OR gate** takes two input signals p and q, each a bit, and produces as output the signal $p \vee q$. Finally, the **AND gate** takes two input signals p and q, each a bit, and produces as output the signal $p \wedge q$. We use combinations of these three basic gates to build more complicated circuits, such as that shown in Figure 2.

Given a circuit built from the basic logic gates and the inputs to the circuit, we determine the output by tracing through the circuit, as Example 9 shows.

Recap: representing positive integers

base b expansion of n	**base b fixed-width w expansion of n**
For b an integer greater than 1 and n a positive integer, the **base b expansion of n** is $(a_{k-1} \cdots a_1 a_0)_b$ where k is a positive integer, $a_0, a_1, \ldots, a_{k-1}$ are nonnegative integers less than b, $a_{k-1} \neq 0$, and $n = a_{k-1}b^{k-1} + \cdots + a_1 b + a_0$	For b an integer greater than 1, w a positive integer, and n a nonnegative integer with $n < b^w$, the **base b fixed-width w expansion of n** is $(a_{w-1} \cdots a_1 a_0)_{b,w}$ where $a_0, a_1, \ldots, a_{w-1}$ are nonnegative integers less than b and $n = a_{w-1}b^{w-1} + \cdots + a_1 b + a_0$

Signed representations

- Sign-magnitude

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2} \cdots a_0]_{s,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$	$[1a_{w-2} \cdots a_0]_{s,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$

- 2s complement

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2} \cdots a_0]_{2c,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$	$[1a_{w-2} \cdots a_0]_{2c,w}$, where $2^{w-1} - n = (a_{w-2} \cdots a_0)_{2,w-1}$

Signed representations

- Sign-magnitude 17 = [$]_{s,7}$

-17 = [$]_{s,7}$

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2}\cdots a_0]_{s,w}$, where $n = (a_{w-2}\cdots a_0)_{2,w-1}$	$[1a_{w-2}\cdots a_0]_{s,w}$, where $n = (a_{w-2}\cdots a_0)_{2,w-1}$

- 2s complement 17 = [$]_{2c,7}$

-17 = [$]_{2c,7}$

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2}\cdots a_0]_{2c,w}$, where $n = (a_{w-2}\cdots a_0)_{2,w-1}$	$[1a_{w-2}\cdots a_0]_{2c,w}$, where $2^{w-1} - n = (a_{w-2}\cdots a_0)_{2,w-1}$

Tips for calculating 2s complement

To represent –n in 2s complement with width w

- Calculate 2^{w-1}-n, convert to binary fixed-width w-1, pad with leading 1

- Express –n as a sum of powers of 2, where leftmost (2^{w-1}) is negative weight

- Convert n to fixed-width w binary, flip bits, add 1 (ignore overflow)

Bonus: use definitions to explain why all these approaches work!

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2} \cdots a_0]_{s,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$	$[1a_{w-2} \cdots a_0]_{s,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$

Sign-magnitude

To represent a positive integer n	To represent a negative integer $-n$
$[0a_{w-2} \cdots a_0]_{2c,w}$, where $n = (a_{w-2} \cdots a_0)_{2,w-1}$	$[1a_{w-2} \cdots a_0]_{2c,w}$, where $2^{w-1} - n = (a_{w-2} \cdots a_0)_{2,w-1}$

2s complement

*These definitions don't address zero. Which of the following would be *consistent*?*

The number 0 can be written as

A. $[0000]_{s,5}$
B. $[10000]_{s,5}$
C. $[000]_{2c,3}$
D. $[1111]_{2c,4}$
E. $[1000]_{2c,4}$

Arithmetic

$$\begin{array}{r} (1\ 1\ 0\ 1\ 0\ 0)_{2,6} \\ +(0\ 0\ 0\ 1\ 0\ 1)_{2,6} \\ \hline \end{array}$$

(Unsigned) base b fixed-width addition

- Option 1: Convert to decimal, add, convert back (if possible)
- Option 2: adding one bit at time, using the usual column-by-column and carry arithmetic, and dropping the carry from the leftmost column so the result is the same width as the summands

$$\begin{array}{r} (1\ 1\ 0\ 1\ 0\ 0)_{2,6} \\ +(0\ 0\ 0\ 1\ 0\ 1)_{2,6} \\ \hline \end{array}$$

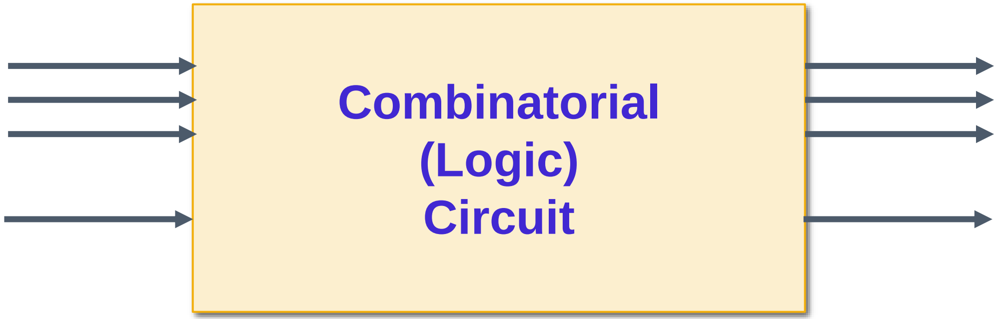
$$\begin{array}{r} [1\ 1\ 0\ 1\ 0\ 0]_{s,6} \\ +[0\ 0\ 0\ 1\ 0\ 1]_{s,6} \\ \hline \end{array}$$

$$\begin{array}{r} [1\ 1\ 0\ 1\ 0\ 0]_{2c,6} \\ +[0\ 0\ 0\ 1\ 0\ 1]_{2c,6} \\ \hline \end{array}$$

Arithmetic in hardware

Other models are possible

Inputs e.g. coefficients in fixed-width binary representation



Outputs e.g. coefficients in fixed-width binary representation

Values flow left to right: possible values on a wire are 0 (low) or 1 (high)
Circuit elements: wires, gates
Gates may share input; outputs of gate can become inputs to other gates

Digital circuits and gates

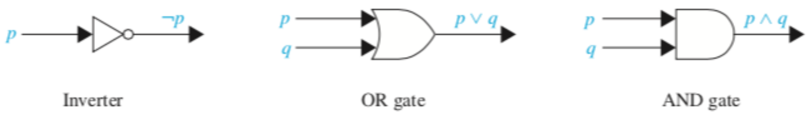


FIGURE 1 **Basic logic gates.**

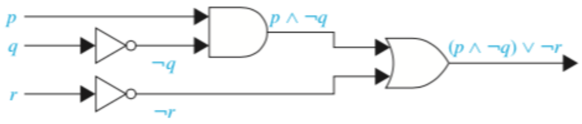
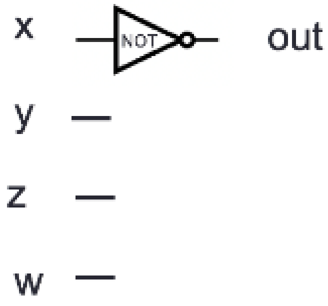


FIGURE 2 **A combinatorial circuit.**

Complicated digital circuits can be constructed from three basic circuits, called **gates**, shown in Figure 1. The **inverter**, or **NOT gate**, takes an input bit p, and produces as output $\neg p$. The **OR gate** takes two input signals p and q, each a bit, and produces as output the signal $p \vee q$. Finally, the **AND gate** takes two input signals p and q, each a bit, and produces as output the signal $p \wedge q$. We use combinations of these three basic gates to build more complicated circuits, such as that shown in Figure 2.

Given a circuit built from the basic logic gates and the inputs to the circuit, we determine the output by tracing through the circuit, as Example 9 shows.

Example: logical circuit

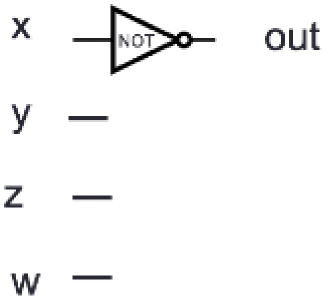


Output when $x = 1, y = 0, z = 0, w = 1$ is ______

Output when $x = 1, y = 1, z = 1, w = 1$ is ______

Output when $x = 0, y = 0, z = 0, w = 1$ is ______

Example: logical circuit



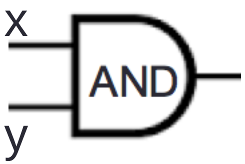
Which of the following is true about all possible input values x,y,z,w? "The output out is set to 1 exactly when

A. x is 0, and is set to 0 otherwise"
B. $(xyzw)_2$ is less than 8, and is set to 0 otherwise"
C. $(wzyx)_{2,4}$ is an even integer, and is set to 0 otherwise"
D. All of the above
E. None of the above

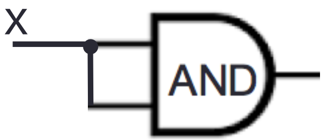
Other gates

Input		Output
x	y	x AND y
1	1	1
1	0	0
0	1	0
0	0	0

Input		Output
x	y	x XOR y
1	1	0
1	0	1
0	1	1
0	0	0



Circuits



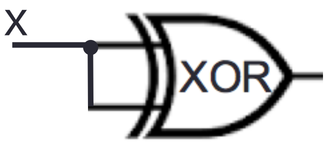
Input	Output
x	
1	
0	

Input		Output
x	y	x AND y
1	1	1
1	0	0
0	1	0
0	0	0

Fill in the input-output table for this one gate logic circuit (reading top to bottom):

A. 1-0
B. 1-1
C. 0-1
D. 0-0
E. I don't know

Circuits



Input	Output
x	
1	
0	

Input		Output
x	y	x XOR y
1	1	0
1	0	1
0	1	1
0	0	0

Fill in the input-output table for this one gate logic circuit (reading top to bottom):

A. 1-0
B. 1-1
C. 0-1
D. 0-0
E. I don’t know

For next time

- Read website carefully

http://cseweb.ucsd.edu/classes/fa20/cse20-a/

- Next pre-class reading:
  - Section 1.1 definition of truth table and Tables 1-5 (pp. 4-6).