EE 280
Introduction to Digital Logic Design

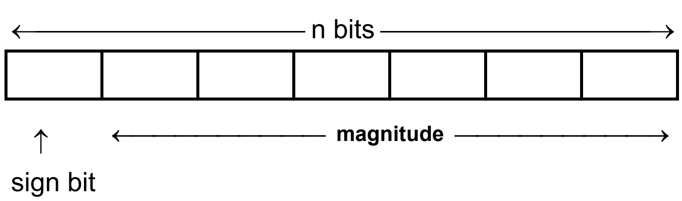
Lecture 3.
Complements: Two’s and One’s

Number Representation

- Read Chapter 1.5, 2.1, and 2.2 for next class

Negative Numbers
How do we represent negative numbers in a word length of n bits?

I. Sign and Magnitude Representation



Number Representation

e.g., **0 1 1 0 ≡ +6** **0 0 0 0 0 1 1 0 ≡ +6**

This representation is not popular - design of logic networks to do arithmetic is awkward.

Number Representation

II. 2's Complement Representation

+ve nos.: **sign and magnitude**
-ve nos.: **-N is represented by N*, the 2's Complement**

where N* =

e.g., **n = 4**

$2^n = 16_{10} = 10000$

+6 = 0110

-6 = 10000 - 0110 =

-1 =

Number Representation

III. 1's Complement Representation

+ve nos.: sign and magnitude
-ve nos.: -N represented by Ñ, the 1's complement

where Ñ =

e.g., n = 4

$2^n - 1 = 16 - 1 = 15 = 10000 - 1$

-6 = 1111 - 0110

-1 =

Number Representation

An alternate method to obtain the 1's complement: complement N bit by bit.

6 = 0 1 1 0 1 = 0 0 0 1
-6 = -1 =

Note that for the 2's complement:

$$N^* = 2^n - N = (2^n - 1 - N) + 1$$

i.e., to form 2's complement, add 1 to 1's complement.

Number Representation

Equivalently: Complement each bit to the left of the least significant 1.

6 = 0 1 1 0

2's Complement of 6   0110

-6 =

Number Representation

Thus, given a negative number in { 2's / 1's } complement representation,

we can obtain the magnitude of that number by taking the { 2's / 1's }

complement of { N* / Ñ }.

e.g.,   if N* = 1 0 0 1 1 0   (2's complement)

N =

or Ñ = 1 0 0 1 0 1   (1's complement)

N =

Addition of 2's Complement Numbers

Adding two n-bit numbers: add numbers as if both were positive (although one or both may be negative).

Six possible cases can occur:

1. Two positive numbers, sum < 2^{n-1}

+ 6	0 1 1 0
<u>+ 1</u>	<u>0 0 0 1</u>

Correct answer.

Addition of 2's Complement Numbers

2. Two positive numbers, sum ≥ 2^{n-1}

+ 6	0 1 1 0
<u>+ 2</u>	<u>0 0 1 0</u>

Overflow! - too big a number!

- Largest number for n = 4 is

- How do we know when overflow occurs?

The 1 in the MSB position indicates a negative number, after adding two +ve numbers.

Addition of 2's Complement Numbers

3. Positive and negative numbers (-ve number has greatest magnitude).

+ 1	0 0 0 1	
<u>- 6</u>	<u>1 0 1 0</u>	
		Correct answer. How do we know?

4. Positive and negative numbers (+ve number has greatest magnitude).

+ 6	0 1 1 0	
<u>- 2</u>	<u>1 1 1 0</u>	
+ 4		Correct answer. Ignore carry from sign bit. Not an overflow.

Addition of 2's Complement Numbers

5. Two negative numbers, $|sum| \leq 2^{n-1}$

- 4	1 1 0 0	
<u>- 2</u>	<u>1 1 1 0</u>	
- 6		Correct answer. Ignore carry from sign bit. Not an overflow.

6. Two negative numbers, $|sum| > 2^{n-1}$

- 6	1 0 1 0	
<u>- 3</u>	<u>1 1 0 1</u>	
- 9		Wrong answer because of overflow: -9 is too large to be represented in a 4 bit number (including sign).

Addition of 1's Complement Numbers

1. Similar to 2's complement addition.

   Two positive numbers cases identical to cases 1 and 2 above.

2. Positive and negative number (-ve number largest magnitude).

2	0 0 1 0	
- 6	______	
- 4		Correct answer.

Addition of 1's Complement Numbers

3. Positive and negative number (+ve number largest magnitude).

- 2		
+ 6	0 1 1 0	
+ 4		Correct answer? No, add carry to LSB. No overflow.

4. Two negative numbers, $|sum| < 2^{n-1}$

- 4	1 0 1 1	
- 2	1 1 0 1	
- 6		Correct answer with end around carry; no overflow.

Addition of 1's Complement Numbers

5. Two negative numbers, $|sum| \geq 2^{n-1}$

- 6	1 0 0 1
- 4	1 0 1 1
-10	

Wrong answer; overflow!