One's-complement

- Negative number: Bitwise complement positive number
  - $0011 \equiv 3_{10}$
  - $1100 \equiv -3_{10}$
- Solves the arithmetic problem

Add		Invert, add, add carry		Invert and add	
4	0100	4	0100	− 4	1011
+ 3	+ 0011	− 3	+ 1100	+ 3	+ 0011
= 7	= 0111	= 1	1 0000	− 1	1110
		add carry:	+1		
			= 0001		

- Remaining problem: Two representations for zero
  - 0 = 0000 and also –0 = 1111

Two's-complement

- Negative number: Bitwise complement **plus one**
  - 0011 ≡ 3_{10}
  - 110**1** ≡ -3_{10}
- Only one zero!
- MSB is the sign digit
  - 0 ≡ positive
  - 1 ≡ negative

Add		Invert and add		Invert and add	
4	0100	4	0100	− 4	1100
+ 3	+ 0011	− 3	+ 1101	+ 3	+ 0011
= 7	= 0111	= 1	1 0001	− 1	1111
		drop carry	= 0001		

Exercise

	6-bit Binary of Magnitude	Sign and Magnitude	One's Complement	Two's Complement
-10	001010	101010	110101	110110
-14	001110	101110	1100001	1100010
-3	000011	100011	111100	111101
-17	010001	110001	101110	101111

Arithmetic’s using 1’s and 2’s complement

One’s-complement (Arithmetic)

Addition of a positive number and a negative number.

Case I: When the positive number has a greater magnitude

In this case the end-around **carry will be generated** and is **added** into the final result.

	Binary	5-bit Binary	
			1 1 1
11	1011	01011	01011
-5	101	00101	+ 11010 (**1’s Complement**)
			00101
			+ 1
			00110

One’s-complement (Arithmetic)

Addition of a positive number and a negative number

Case II: When the negative number is greater.

When the negative numbers is greater **no end-around carry** will be generated. The result of addition will be negative, and the **final result is obtained by taking 1’s complement** of the result.

	Binary	5-bit Binary	
			1
-11	1011	01011	10100 (**1’s Complement**)
5	101	00101	+ 00101
			11001
			00110 (**1’s Complement**)

One’s-complement (Arithmetic)

When the numbers are negative

A end-around **carry will be generated** which will be **added** in sum. 1’s complement of the result will give the magnitude.

	Binary	5-bit Binary	
-10	1010	01010	10101 (**1’s Complement**)
-5	101	00101	+ 11010 (**1’s Complement**)

1

01111

\+ 1

10000

01111 (**1’s Complement**)

Two's-complement (Arithmetic)

Addition of a positive number and a negative number.

Case I: When the positive number has a greater magnitude

In this case the **end-around carry will be generated** and is **discarded**. The final result is the result of addition.

	Binary	5-bit Binary	
			1 1 11
11	1011	01011	01011
-5	101	00101	+ 11011 (**2's Complement**)
			00110

Two's-complement (Arthimetic)

Addition of a positive number and a negative number

Case I: When the negative number is greater.

When the negative numbers is greater **no end-around carry will be generated**. The result of addition will be negative, and the **final result is obtained by taking 2's complement** of the final sum.

	Binary	5-bit Binary	
			1 1
-11	1011	01011	10101 (**2's Complement**)
5	101	00101	+ 00101
			11010
			00110 (**2's Complement**)

Two's-complement (Arthimetic)

When the numbers are negative

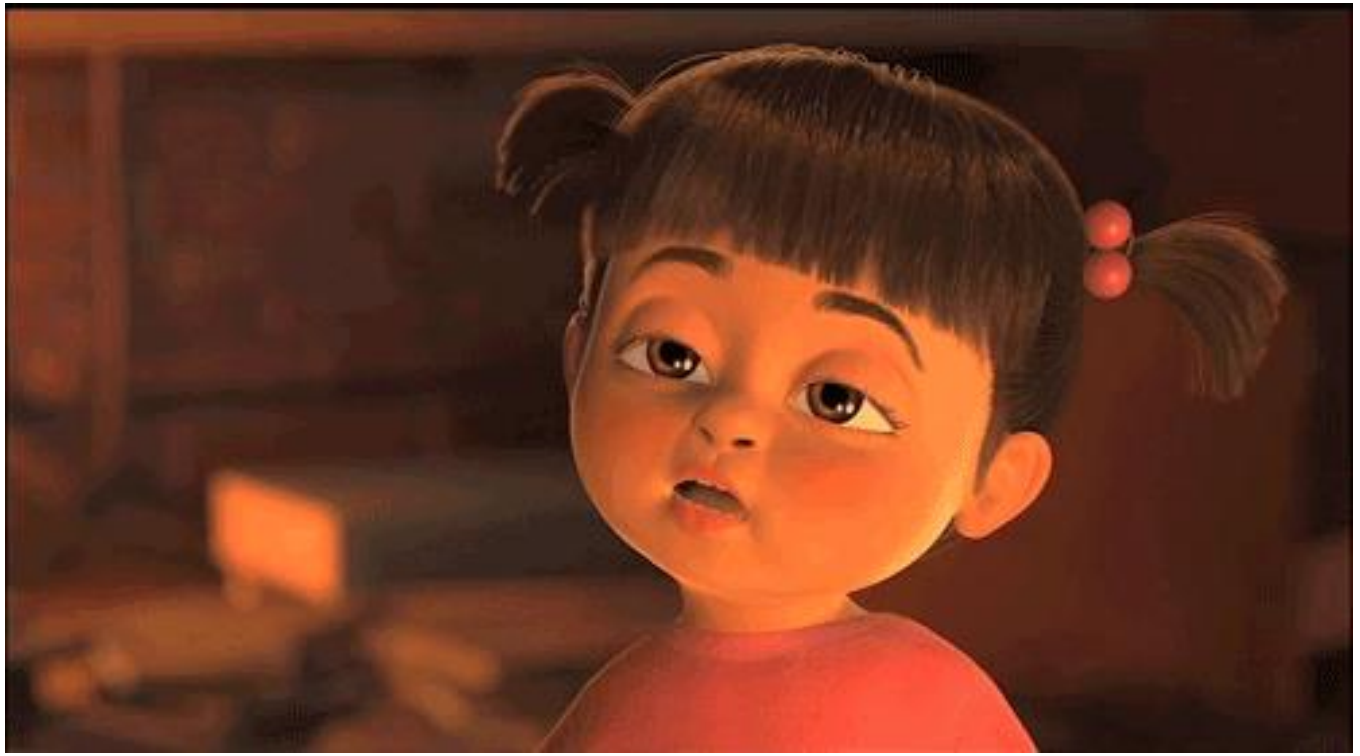
A end-around **carry will be generated** which will be **discarded**. The final result is obtained by taking 2's complement of the sum.

	Binary	6-bit Binary	
-10	1010	01011	10110 (**2's Complement**)
-5	101	00101	+ 11011 (**2's Complement**)
			1 111
			10001
			01111 (**2's Complement**)

In summary

- Can’t infer a representation from a number
  - 11001 is 25 (unsigned)
  - 11001 is –9 (sign magnitude)
  - 11001 is –6 (ones complement)
  - 11001 is –7 (twos complement)
- 1’s complement **-> Add Carry**
- 2’s complement **-> Drop Carry**

Thanks a lot



If you are taking a Nap, **wake up**.....Lecture Over