Professor Hamlin
Day 3

Agenda

1) Tutoring groups, HW 1
2) Review
3) Negative binary representation
   - two's complement

Review

- Dec → Bin/Hex/other bases
  1. Euclids
  2. Subtraction
- Modular Mathematics

1) 35_{10} → Binary
(your choice of method)

$35 = 17 \cdot 2 + 1$
$17 = 8 \cdot 2 + 1$
$8 = 4 \cdot 2 + 0$
$4 = 2 \cdot 2 + 0$
$2 = 1 \cdot 2 + 0$
$1 = 0 \cdot 2 + 1$

100011_2

2) $(15 + 12) \bmod 5 = ?$

$15 \bmod 5 = 0$
$12 \bmod 5 = 2$

$0 + 2 \bmod 5 = 2$

$27 \bmod 5 = 2$

Quick note on how computers store numbers. Remember Bytes? 8 bits

Computers store numbers using the same # of bits

$7 \to 111_2$

0	0	0	0	0	1	1	1

$14 \to 1110_2$

0	0	0	0	1	1	1	0

1 1 1 1 1 1 1 0

Sometimes 8 bits, 32, 64 etc. Why?

More efficient and easier to find things

This means sometimes numbers are too big for their # of bits e.g w/ 3-bits

1	1	1	+	0	0	1	=	0	0	0

7 + 1 = 8

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

discard

This is like working in mod 8

Negative Representation

In most math negative numbers represented by -17, -456

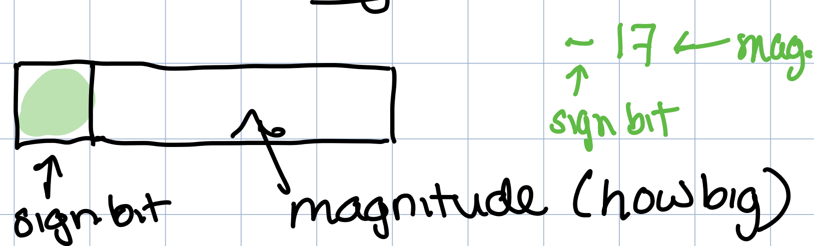
But computers only store 0/1, how to we indicate negative numbers?

Vocab

unsigned number: whole non-negative binary numbers
e.g. 101_2

Starter idea! Just add a sign bit infront of number

1 = negative
0 = positive



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

0	0	0	0	= +0
1	1	0	1	= -5

Ex] Write −13 in 5 bits

1	1	1	0	1

$13 = 6 \cdot 2 + 1$
$3 = 3 \cdot 2 + 0$
$3 = 1 \cdot 2 + 1$
$1 = 0 \cdot 2 + 1$

−5 in 5 bits

1	0	1	0	1

(one's complement)

However just a sign bit has some problems....

1) Two ways of representing zero
(4-bit example)

0 0 0 0 = +0

1 0 0 0 = −0

This means we aren't the most efficient!

2) Sign bit doesn't always add correctly!
(3-bit)

1	1	1

+

0	0	1

= 1

0	0	0

discard

−3 + 1 = ~~−4~~ 0

$$\begin{array}{r} 1\,1\,1 \\ +\,0\,0\,1 \\ \hline 1\,0\,0\,0 \end{array}$$

Two's complement: a better negative rep.

Intition: most significant (biggest) place value is negative, everything else positive

eg Two's complement value in 3 bits

1 1 1

1)

$$\begin{array}{ccc} -2^2 & 2^1 & 2^0 \\ \hline 1 & 1 & 1 \end{array}$$

$$1 \cdot -2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$
$$-4 + 2 + 1 = \boxed{-1}$$

2) $111_2 = 1 \cdot -100_2 + 1 \cdot 10_2 + 1 \cdot 1_2$

$$1 \cdot -4 + 1 \cdot 2 + 1 = \boxed{-1}$$

Exercise:

1) 0 1 0 1 in 4-bit two's complement to decimal

$$\begin{array}{cccc} -2^3 & 2^2 & 2^1 & 2^0 \\ \hline 0 & 1 & 0 & 1 \end{array}$$

$$0 \cdot -2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$
$$4 + 1 = \boxed{5}$$

2) 1 0 1 1 in 4-bit two's to decimal

$$\begin{array}{cccc} -2^3 & 2^2 & 2^1 & 2^0 \\ \hline 1 & 0 & 1 & 1 \end{array}$$

$$1 \cdot -8 + 0 \cdot 4 + 2 \cdot 1 + 1 \cdot 1 = \boxed{-5}$$

What pattern do we see (3-bits)

-4	2	1	
1	0	0	-4
1	0	1	-3
1	1	0	-2
1	1	1	-1
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3

1) only one zero! :)

2) Does the math work?

Yes

Checking math

$-2 + 3 = 1$

$$\begin{array}{r} 1\,1\,0 \\ +\ 0\,1\,1 \\ \hline 1\,\boxed{0\,0\,1} \end{array}$$

$001 \rightarrow 1_{10}$:)

Note: discarding digits isn't always bad: It only counts as overflow if the result is wrong

$$\begin{array}{r} 0\,1\,1 \\ +\ 0\,0\,1 \\ \hline 1\,0\,0 \end{array}$$

$011 + 001 = 100$

$3 \quad 1 = \cancel{4} \ -4$

Overflow - when outcome cannot be rep. w/ # of bits

Connection to Mod

The magnitude is too big for 2 bits – like mod things wrap around

2 bits → mod 4 (2^2)

Careful this is only true for positive!

Exercise: Which of the following values fit into their bits (does it overflow)

1)	0	unsigned 2-bit	Yes 00
2)	-2	unsigned 3-bit	No - unsigned no neg!
3)	0	2 bit two's comp.	Yes 00
4)	-4	3 bit two's comp	Yes
5)	-4	4 bit two's comp	Yes
6)	5	4 bit two's comp	Yes $\boxed{0101}$
7)	10	4 bit two's comp	No too big
8)	-3	4 bit two's comp	Yes

This leads us to ask what values can we represent w/ N-bits

Unsigned

2^3	2^2	2^1	2^0

2^{N-1}	2^{N-2}	…	2^1	2^0
0	1	…	1	0

smallest – $\boxed{0}$

largest – $\underbrace{111\ldots11}_{N} + 1 = 1\underbrace{00\ldots00}_{N}$

$\boxed{2^N - 1}$

Two's complement

-2^{N-1}	2^{N-2}	…	2^1	2^0
		…		

smallest = $1\underbrace{00\ldots00}_{N-1}$

$1\cdot -2^{N-1} + 0\cdot 2^{N-2} + \ldots = \boxed{-2^{N-1}}$

largest = $0\underbrace{11\ldots111}_{N-1}$

$\boxed{2^{N-1} - 1}$

Exercise: With 5 bits can rep. $N=5$

1) unsigned

$0 \rightarrow 2^5 - 1$

$0 \rightarrow 31$

2) two's comp

$-2^{5-1} \rightarrow 2^{5-1} - 1$

$-16 \rightarrow 15$

Exercise: Interpret 11011_2 as

1) unsigned

2) two's complement

1) $1 \cdot 10000_2 + 1 \cdot 1000_2 + 0 \cdot 100_2 + 1 \cdot 10_2 + 1 \cdot 1_2$

$1 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$

$\boxed{27}$

2) $1 \cdot -10000_2 +$ " "

$-16 + 8 + 2 + 1$

$\boxed{-5}$

Two's comp: Dec → Binary

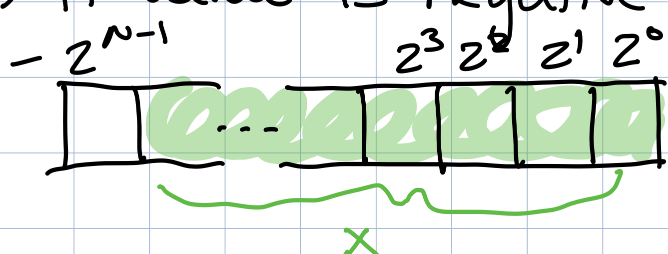
1) First we have to check if it fits in the number of bits

e.g 4-bits Would 13 fit

$2^{4-1} - 1 = 7$ $\qquad 7 \cancel{<} 13$

2) if value is positive - use Euclid or subtraction (remember to pad w/ leading zeros!)

3) if value is negative - solve for x



a) value $= -2^{N-1} + x$

b) just solve for x

c) turn x into binary
d) append leading 1

$$\text{Value} = -2^{N-1} + x$$
$$x = \text{value} + -2^{N-1}$$

$$-2^{N-1} + \overbrace{\qquad\qquad\qquad}^{x} = \text{value}$$

Example: -4 as 4-bit two's comp.

find x:
$$-2^{N-1} + x = -4$$
$$-8 + x = -4$$
$$x = 4$$

x to binary: $4 \to 100_2$

add leading 1 $\boxed{1\,1\,0\,0_2}$

5-bit two's complement
$$-2^{N-1} + x = -4$$
$$-2^{4} + x = -4$$
$$-16 + x = -4$$
$$x = 12$$

$12 = 6 \cdot 2 + 0$
$6 = 3 \cdot 2 + 0$
$3 = 1 \cdot 2 + 1$
$1 = 0 \cdot 2 + 1$

$\boxed{11100_2}$

Flow chart:

Can you rep it in N-bits $\xrightarrow{\text{yes}}$ is it pos or neg
- $\xrightarrow{\text{pos}}$ use euclid or sub method → pad to # of bits
- $\xrightarrow{\text{neg}}$ solve for x → use euclid /sub → pad to # of N-1 bits → add leading 1

Can you rep it in N-bits $\xrightarrow{\text{no}}$ STOP

Exercise: Represent following as 6-bit two's comp

1) -5

$-2^{N-1} = -2^5$ ✓

-32

$-32 + x = -5$

$x = 27$

$27 = 13 \cdot 2 + 1$

$13 = 6 \cdot 2 + 1$

$6 = 3 \cdot 2 + 0$

$3 = 1 \cdot 2 + 1$

$1 = 0 \cdot 2 + 1$

11011_2

$\boxed{111011_2}$

2) 5

$5_{10} \rightarrow 101_2$

$\boxed{000101_2}$

3) 32

$2^{6-1} - 1 = 32 - 1$

31

$\boxed{32 > 31}$