CSE20 Lecture 3
Number Systems

2. Binary Numbers
3. Gray Code
4. Negative Numbers

CK Cheng, UC San Diego

Outlines

1. Goal of the Negative Number Systems
2. Definition
    1. Sign Magnitude Rep.
    2. 1’s Complement Rep.
    3. 2.s Complement Rep.
3. Arithmetic Operations

4.1 Goal of Negative Number System: iClicker

The goal of negative number system is

- A. to maximize the range of the numbers
- B. to simplify the hardware implementation
- C. to improve human interface
- D. All of the above.

4.1 Goal of negative number systems

- Signed system: Simple. Just flip the sign bit
    - 0 = positive
    - 1 = negative
- One’s complement: Replace subtraction with addition
    - Easy to derive (Just flip every bit)
- Two’s complement: Replace subtraction with addition
    - Addition of one’s complement and one produces the two’s complement.

4.2 Definitions: Given a positive integer x, we represent -x

- 1's complement:
  - Formula: $2^n - 1 - x$
    - i.e. n=4, $2^4 - 1 - x = 15 - x$
    - In binary: $(1\ 1\ 1\ 1) - (b_3\ b_2\ b_1\ b_0)$
    - Just flip all the bits.
- 2's complement:
  - Formula: $2^n - x$
    - i.e. n=4, $2^4 - x = 16 - x$
    - In binary: $(1\ 0\ 0\ 0\ 0) - (0\ \ b_3\ b_2\ b_1\ b_0)$
    - Just flip all the bits and add 1.

4.2 Definitions: 4-bit example, id vs. value

Signed: $b_3=1$, 1's: 15-x, 2's: 16-x

id	$b_3b_2b_1b_0$	Signed	1's	2's
.	….	.	.	.
7	0111	7	7	7
8	1000	-0	-7	-8
9	1001	-1	-6	-7
10	1010	-2	-5	-6
11	1011	-3	-4	-5
12	1100	-4	-3	-4
13	1101	-5	-2	-3
14	1110	-6	-1	-2
15	1111	-7	-0	-1

15-x=id or id+x=15.
16-x=id or id+x=16.

$b_{n-1}=1$ for negative numbers

4.2 Definitions: 4-bit example, value vs. $b_3b_2b_1b_0$

-x	**signed**	**1's**	**2's**
-0	1000	1111	0000
-1	1001	1110	1111
-2	1010	1101	1110
-3	1011	1100	1101
-4	1100	1011	1100
-5	1101	1010	1011
-6	1110	1001	1010
-7	1111	1000	1001
-8	NA	NA	1000

4.2 Definition: Example

Given bit width n=5 for x= 6 $(00110)_2$, we represent –x.

- Signed number: $(b_4b_3b_2b_1b_0)_2=(10110)_2$
- 1's complement: 2^5-1-x=32-1-6=25

  $(b_4b_3b_2b_1b_0)_2=(11001)_2$
- 2's complement: 2^5-x=32-6=26

  $(b_4b_3b_2b_1b_0)_2=(11010)_2$

4.2 Definition: iClicker

Given bit width n=5 for x= 11 $(01011)_2$, we represent –x in 1's complement as

- A. $(10100)_2$
- B. $(10101)_2$
- C. $(11010)_2$
- D. None of the above.

4.2 Definitions: Examples

Given n-bits, what is the range of my numbers in each system?

- 3 bits:
  - Signed: -3 , 3
  - 1's: -3 , 3
  - 2's: -4 , 3
- 5 bits:
  - Signed: -15, 15
  - 1's: -15, 15
  - 2's: -16, 15
- 6 bits
  - Signed: -31, 31
  - 1's: -31, 31
  - 2's: -32, 31
- Given 8 bits
  - Signed: -127, 127
  - 1's: -127, 127
  - 2's: -128, 127

Formula for calculating the range →

Signed & 1's: $-(2^{n-1} - 1)$, $(2^{n-1} - 1)$

2's: -2^{n-1} , $(2^{n-1} - 1)$

4.3 Arithmetic Operation

- Conversion
- Addition and subtraction
- Inverse conversion
- Overflow

4.3 Arithmetic Operations: Conversion

Derivation of 1's Complement

Theorem 1: For 1's complement, given a positive number $(x_{n-1} , x_{n-2} , \dots , x_0)_2$, the negative number is

$(\bar{x}_{n-1}, \bar{x}_{n-2}, \dots, \bar{x}_0)_2$ where $\bar{x} = 1 - x$

Proof:

(i). 2^n-1 in binary is an n bit vector $(1,1, \dots, 1)_2$

(ii). 2^n-1-x in binary is $(1,1, \dots,1)_2 - (x_{n-1},x_{n-2}, \dots, x_0)_2$. The result is

$$(\bar{x}_{n-1}, \bar{x}_{n-2}, \dots, \bar{x}_0)_2$$

4.3 Arithmetic Operations: Conversion

Derivation of 2's Complement

Theorem 2: For 2's complement, given a positive integer x, the negative number is the sum of its 1's complement and 1.

Proof: $2^n - x = 2^n - 1 - x + 1$. From theorem 1, we have

$$(\bar{x}_{n-1}, \bar{x}_{n-2}, \ldots, \bar{x}_0)_2 + (0, 0, \ldots, 1)_2$$

4.3 Arithmetic Operations: Conversion

Ex: n=5, x = 9 $(01001)_2$

1's complement: 2^5-1-x= 32-1-9=22

$=(10110)_2$

2's complement: 2^5-x=32-9=23

$=(10111)_2$

Ex: n=5, x = 13 $(01101)_2$

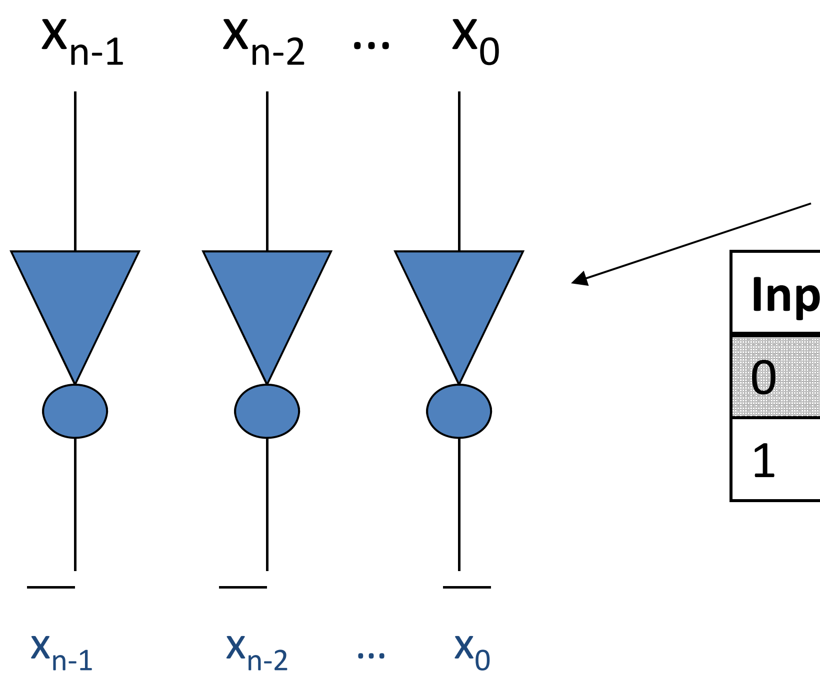
1's complement: 2^5-1-x= 32-1-13=18

$=(10010)_2$

2's complement: 2^5-x=32-13=18

$=(10011)_2$

4.3 Conversion: One's Complement Hardware:



Input	**Output**
0	1
1	0

4.3 Addition and Subtraction:

Given two positive integers x & y,

we replace subtraction with complement conversion.

Suppose the sum is valid in the form of the complement.

Then we don’t need subtraction in hardware implementation.

4.3 Addition and Subtraction: 2's Comp.

Arithmetic	Addition in 2's comp.	Solution in 2's comp.
$x + y$	$x + y$	$x+y$
$x - y$	$x + (2^n-y)$	$2^n+(x-y)$ $(x<y)$ $x-y$ $(x>=y)$
$-x + y$	$(2^n-x) + y$	$2^n+(-x+y)$ $(x>y)$ $-x+y$ $(x<=y)$
$-x - y$	$(2^n-x) + (2^n-y)$	2^n-x-y

Note the similarity of the last two columns

4.3 Addition and Subtraction: 2's Comp.

Input: two positive integers x & y,

1. We represent the operands in two's complement.
2. We sum up the two operands and ignore bit n.
3. The result is the solution in two's complement.

Arithmetic	Addition in 2's comp.	Solution in 2's comp.
$x - y$	$x + (2^n - y) = 2^n + x - y$ If $x<y$, $b_n=0$ Else, $b_n=1$	$2^n+(x-y)$ $(x<y)$ $x-y$ $(x>=y)$
$-x + y$	$(2^n - x) + y = 2^n - x + y$ If $x>y$, $b_n=0$ Else, $b_n=1$	$2^n+(-x+y)$ $(x>y)$ $-x+y$ $(x<=y)$
$-x-y$	$2^n + (2^n - x - y)$ $b_n=1$	$2^n - x - y$

4.3 Arithmetic Operations: 2's comp.

Example: 4 – 3 = 1

In 2's complement, we represent -3 as $(1101)_2$

```
  0100 (4)
+ 1101 (13=16-3)
 10001 (17=16+1)
```

Formula: $x+(2^n-y)=4+(16-3)=16+1$

We discard the extra 1 at the left which is from 2's complement of -3. Note that bit b_{n-1} is 0. Thus, the result is positive.

4.3 Arithmetic Operations: 2’s complement

Example: -4 +3 = -1

In 2’s comp., we represent -4 as $(1100)_2$

```
  1100 (12=16-4)
+ 0011 (3)
  ----
  1111 → -1 in 2’s comp.
```

Formula: $(2^n - x) + y = 16 - 4 + 3 = 16 - 1 = 15$ (-1 in 2’s comp.)

Note that $b_{n-1} = 1$. Thus, the solution is negative.

4.3 Arithmetic Operations: 2's complement

Example: -4 -3 = -7

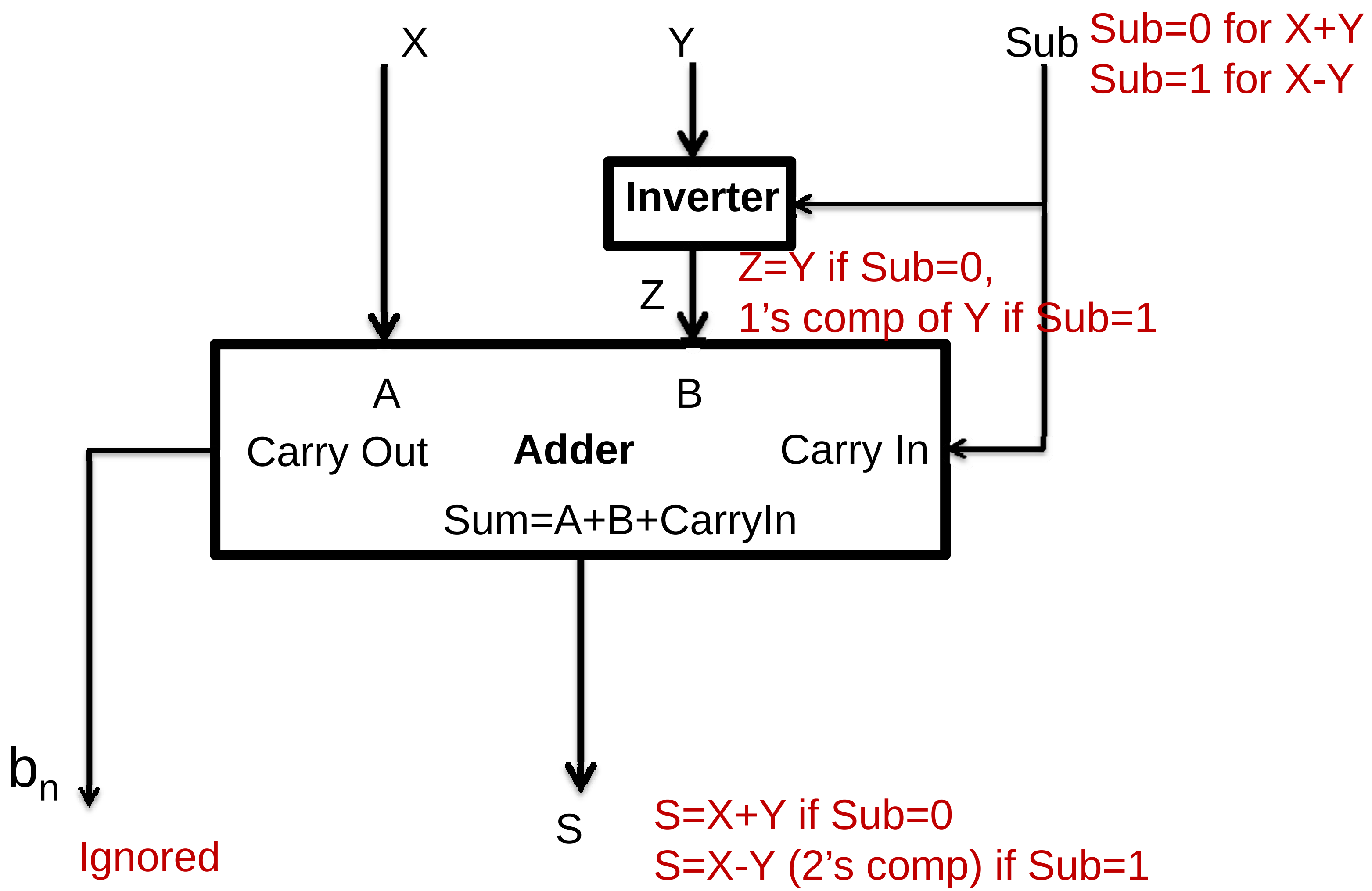
```
   1100 (12=16-4)
+  1101 (13=16-3)
  11001 → 25=16+16-7
```

Formula: $(2^n\text{-}x)+(2^n\text{-}y)$=16-4+16-3=16+16-7

After we delete b_n, the result is 16-7

Note that b_{n-1}=1. Thus, the solution is negative.

4.3 Flow of 2's Complement



4.3 Addition and Subtraction: 1's Comp.

Arith.	Addition in1's	Sol. in1's comp.
x + y	x + y	x+y
x - y	x + (2^n-1-y)	2^n-1+(x-y) (x<=y) x-y (x>y)
-x + y	(2^n-1-x) + y	2^n-1+(-x+y) (x>=y) –x+y (x<y)
-x - y	(2^n-1-x) + (2^n -1-y)	2^n-1-x-y

4.3 Addition and Subtraction: 1's Comp.

Input: two positive integers x & y,

1. We represent the operands in one's complement.
2. We sum up the two operands.
3. We delete 2^n-1 if b_n=1.
4. The result is the solution in one's complement.

Arith.	Addition in1's	Result in 1's comp.
x - y	x+(2^n -1- y)=2^n-1+(x-y). b_n=1 if x>y	2^n-1+(x-y) (x<=y) x-y (x>y)
-x + y	(2^n -1-x)+y=2^n-1+(-x+y). b_n=1 if x<y	2^n-1+(-x+y) (x>=y) –x+y (x<y)
-x - y	(2^n -1-x) + (2^n -1-y) =2^n-1+2^n-1-x-y. b_n=1	2^n-1-x-y

4.3 Addition and Subtraction: 1's Comp.

Example: 4 – 3 = 1

In 1's complement, we represent -3 as $(1100)_2$

```
  0100 (4)
+ 1100 (12=15-3 )
 10000 (16=15+1 )
  0001(after deleting 2^n-1)
```

Formula: $x+(2^n-1-y)=4+(15-3)=15+1$

We discard bit b_n (-2^n) and add one at the first bit (+1), i.e. deduct 2^n-1.

4.3 Addition and Subtraction: 1's Comp.

Example: -4 +3 = -1

In 1's complement, we represent -4 as $(1011)_2$

```
  1011 (11=15-4 )
+ 0011 (3)
  ----
  1110 (14=15-1 )
```

Formula: $(2^n-1-x)+y=15-4+3=14$

Note that $b_{n-1}=1$. Thus, the solution is a negative number.

4.3 Addition and Subtraction: 1's Comp.

Example: – 4 – 3 = – 7

In 1's complement, we represent -4 as $(1011)_2$

-3 as $(1100)_2$

  1011 (11=15-4)

+ 1100 (12=15-3)

 1,0111 (23=15+15-7)

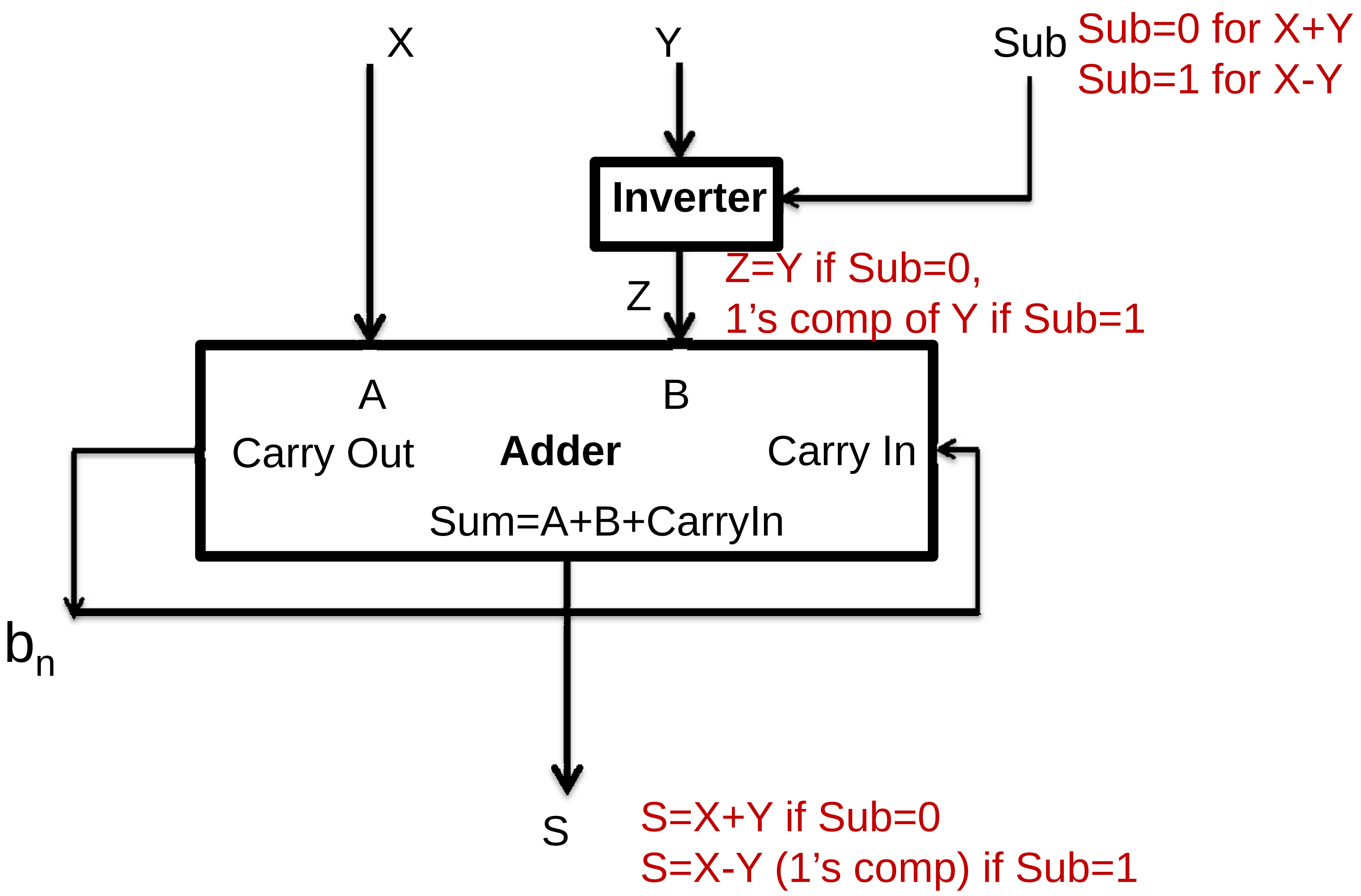
So now take b_n=1 and remove it from the 5th spot and add it to the remainder

    0111

+   1

   1000 (8=15-7)

4.3 Flow of 1's Complement



4.3 Inverse Conversion

1's Compliment:

Let $f(x) = 2^n - 1 - x$

Theorem: $f(f(x)) = x$

Proof: $f(f(x))$

$= f(2^n - 1 - x)$

$= 2^n - 1 - (2^n - 1 - x)$

$= x$

2's Compliment:

Let $g(x) = 2^n - x$

Theorem: $g(g(x)) = x$

Proof: $g(g(x))$

$= g(2^n - x)$

$= 2^n - (2^n - x)$

$= x$

4.4 Overflow

Overflow occurs when the result lies beyond the range of the number system

Examples

Overflow Flag formula

4.4 Overflow: Examples (2's Comp.)

2' Comp: n=4, range -8 to 7

Bit	4	3	2	1	0	
C_a	0	0	1	0	0	
X		0	0	1	0	2
Y		0	0	1	1	3
X+Y	0	0	1	0	1	5

Bit	4	3	2	1	0	
C_a	0	1	1	0	0	
X		0	0	1	1	3
Y		0	1	1	0	6
X+Y	0	1	0	0	1	9

Bit	4	3	2	1	0	
C_a	1	1	0	0	0	
X		1	1	1	0	-2
Y		1	1	0	1	-3
X+Y	1	1	0	1	1	-5

Bit	4	3	2	1	0	
C_a	1	0	0	0	0	
X		1	1	0	1	-3
Y		1	0	1	0	-6
X+Y	1	0	1	1	1	-9

4.4 Overflow: iCliker

For 2's complement, overflow occurs when

The following condition is true.

A. Both of C_n and C_{n-1} are one
B. Both of C_n and C_{n-1} are zero
C. Either of C_n and C_{n-1} is one but not both
D. None of the above.

4.4 Overflow Condition

Exercise:

1. State and prove the condition of the overflow of 1’s complement number system.
2. State and prove the condition of the overflow of 2’s complement number system.