Computer Arithmetic

Conventional Number Systems

Dae Hyun Kim

EECS
Washington State University

Notations

- X: Value
  - 4
  - 20
  - -20
  - ...

- $(X)_K$: Representation of value X in a certain number system (denoted by K)
  - $4 = (00000100)_{2's}$ = (number 4 in the two's complement)
  - $20 = (00010100)_{SM}$ = (number 20 in the signed-magnitude)
  - $-20 = (10010100)_{SM}$ = (number -20 in the signed-magnitude)
  - $-20 = (11101011)_{1's}$ = (number -20 in the one's complement)
  - $-20 = (11101100)_{2's}$ = (number -20 in the two's complement)

Number Systems

- A **number representation** of length n is an ordered sequence of digits.

$$(x_{n-1}, x_{n-2}, \cdots, x_1, x_0) = x_{n-1}x_{n-2}\cdots x_1x_0$$

- Example

Representation	Value
$(000)_{10}$	0
$(001)_{10}$	1
$(002)_{10}$	2
$(009)_{10}$	9
$(023)_{10}$	23
$(926)_{10}$	-74

Decimal, $n = 3$, two's complement

Representation	Value
$(000)_2$	0
$(001)_2$	1
$(010)_2$	2
$(011)_2$	3
$(101)_2$	5
$(111)_2$	7

Binary, $n = 3$, unsigned

Unsigned Numbers

- Radix-r number
  - Representation

    $(x_{n-1}x_{n-2}\cdots x_1x_0)_r$ where x_i is $0, 1, \cdots, r-1$

  - Numerical value (unsigned)

$$X = x_{n-1}\cdot r^{n-1} + \cdots + x_1\cdot r^1 + x_0\cdot r^0 = \sum_{i=0}^{n-1} x_i\cdot 2^i$$

- Example (Representation → Value)

$$(475)_{10} = 4\cdot 10^2 + 7\cdot 10^1 + 5\cdot 10^0 = 475$$

$$(521)_6 = 5\cdot 6^2 + 2\cdot 6^1 + 1\cdot 6^0 = 193$$

$$(110)_2 = 1\cdot 2^2 + 1\cdot 2^1 + 0\cdot 2^0 = 6$$

Example

$(0011)_2$

$(372)_9$

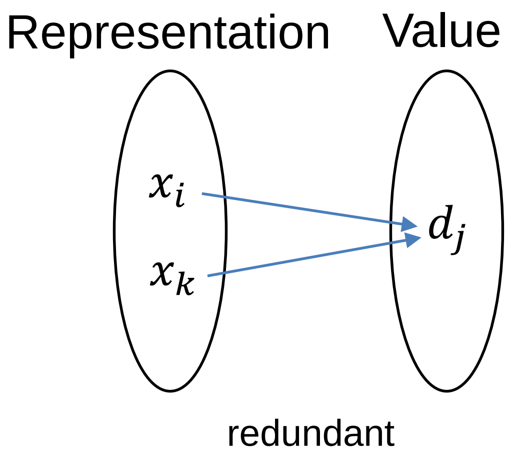
$(530)_5$(wrong!)

Number Systems

- Terminologies
  - Radix-2: Binary numbers
    - $x_i \in \{0, 1\}$
  - Radix-8: Octal numbers
    - $x_i \in \{0, 1, 2, 3, 4, 5, 6, 7\}$
  - Radix-10: Decimal numbers
    - $x_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
  - Radix-16: Hexadecimal numbers
    - $x_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$
      - $A: 10$
      - $B: 11$
      - $C: 12$
      - $D: 13$
      - $E: 14$
      - $F: 15$

Conventional Number Systems

- Non-redundant
  - Every number has a unique representation.
    - Ex: $(28)_{10} \neq (32)_{10}$



- Weighted
  - There are weights for the conversion of a representation into its value.

$$X = x_{n-1} \cdot w_{n-1} + \cdots + x_1 \cdot w_1 + x_0 \cdot w_0 = \sum_{i=0}^{n-1} x_i \cdot w_i$$

- Positional
  - The weight w_i depends only on the position i.

Number Representation – Unsigned

- For a given number X, find its number representation in the destination number system.

- $X \rightarrow (x_{n-1}x_{n-2}\cdots x_1x_0)_d$
  - d: the destination number system

- Example

X	$d = 5, n = 2$
3	$(03)_5$
6	$(11)_5$
9	$(14)_5$
10	cannot represent (# bits should be ≥ 3)

Number Representation – Unsigned

- How-to
  1. Divide X by d, get the quotient Q_1 and the remainder r_0.
  2. Divide Q_1 by d, get the quotient Q_2 and the remainder r_1.
  3. Divide Q_2 by d, get the quotient Q_3 and the remainder r_2.
  4. Repeat until the quotient becomes zero.
  5. Then, $X = (r_{n-1} \cdots r_1 r_0)_d$

- Example
  - $X = 157$, $d = 4$
    - $\frac{157}{4} = 39 \ldots \underline{1}$
    - $\frac{39}{4} = 9 \ldots \underline{3}$
    - $\frac{9}{4} = 2 \ldots \underline{1}$
    - $\frac{2}{4} = 0 \ldots \underline{2}$
    - $157 = (2131)_4$

Number Representation – Unsigned

- Proof

$$X = (x_{n-1} \cdots x_1 x_0)_d$$
$$= x_{n-1} \cdot d^{n-1} + \cdots + x_1 \cdot d^1 + x_0 \cdot d^0$$
$$= \underbrace{(x_{n-1} \cdot d^{n-2} + \cdots + x_2 \cdot d + x_1)}_{\text{Quotient}} \cdot \underline{d} + \underbrace{x_0}_{\text{Remainder}}$$

↓

$$x_{n-1} \cdot d^{n-2} + \cdots + x_2 \cdot d + x_1$$
$$= \underbrace{(x_{n-1} \cdot d^{n-3} + \cdots + x_3 \cdot d + x_2)}_{\text{Quotient}} \cdot \underline{d} + \underbrace{x_1}_{\text{Remainder}}$$

↓

…

Radix Conversion – Unsigned

- $(x_{n-1}x_{n-2}\cdots x_1x_0)_s \rightarrow (y_{m-1}y_{m-2}\cdots y_1y_0)_d$
  - s: the source number system
  - d: the destination number system

- Example

$s = 2, n = 3$	$d = 5, m = 2$
$(000)_2$	$(00)_5$
$(001)_2$	$(01)_5$
$(010)_2$	$(02)_5$
$(011)_2$	$(03)_5$
$(100)_2$	$(04)_5$
$(101)_2$	$(10)_5$
$(110)_2$	$(11)_5$
$(111)_2$	$(12)_5$

Radix Conversion – Unsigned

- For arbitrary number systems
  1. Convert $(x_{n-1}x_{n-2}\cdots x_1x_0)_s$ to its numerical value X.
  2. Convert X to its representation in the destination number system.

- Example
  - $(1101111)_2 \rightarrow (?)_5$
    - $(1101111)_2 = 64 + 32 + 8 + 4 + 2 + 1 = 111$
    - $\frac{111}{5} = 22 \ldots 1$
    - $\frac{22}{5} = 4 \ldots 2$
    - $\frac{4}{5} = 0 \ldots 4$
    - $(1101111)_2 = (421)_5$

Radix Conversion – Unsigned

- Special case
  - When $s = r^a$ and $d = r^b$
- Example
  - $(11010011)_2 \rightarrow (?)_{16}$
    - $s = 2^1$
    - $d = 2^4$
- How-to
  - $(\underbrace{1101}_{D}\underbrace{0011}_{3})_2 = (D3)_{16}$
  - $(7F)_{16} = (\underbrace{0111}_{7}\underbrace{1111}_{F})_2$

Example

$(0011\ 0101\ 1010)_2$
$\downarrow$
$(35A)_{16}$

$(FE8)_{16}$
$\downarrow$
$(1111\ 1110\ 1000)_2$
$\downarrow$
$(111\ 111\ 101\ 000)_2$
$\downarrow$
$(7750)_8$
$\downarrow$
$(11\ 11\ 11\ 10\ 10\ 00)_2$
$\downarrow$
$(333220)_4$

Signed (Negative Numbers)

- How do we represent negative numbers?
  - Use the prefix “–”.

- However, it is actually using an additional symbol “–” (or digit).
  - Unfair

- We should use only digits (or bits) to represent negative numbers.

Negative Numbers

- Example

Representation	Value
$(0000)_{10}$	0
$(0010)_{10}$	10
$(3857)_{10}$	3857
$(9999)_{10}$	-1
$(9998)_{10}$	-2
$(6143)_{10}$	-3857

Decimal, $n = 4$, two's complement

- As you see, the representation doesn't use the "$-$" symbol. It uses only digits ($0, 1, \cdots, 9$) to represent negative numbers.

Negative Numbers in the Binary Number System

- Sign and magnitude (signed-magnitude)
  - The first bit is the sign bit.
    - 0: Positive
    - 1: Negative
  - Value
    - If $X = (0 \cdots)_2$, its value is $x_{n-2} \cdot 2^{n-2} + \cdots + x_1 \cdot 2^1 + x_0 \cdot 2^0$.
    - If $X = (1 \cdots)_2$, its value is $-(x_{n-2} \cdot 2^{n-2} + \cdots + x_1 \cdot 2^1 + x_0 \cdot 2^0)$.

$(x_{n-1} x_{n-2} \cdots x_1 x_0)_2$

Sign: x_{n-1} Magnitude: $x_{n-2} \cdots x_1 x_0$

Representation	Value
$(000)_2$	0
$(001)_2$	1
$(010)_2$	2
$(011)_2$	3
$(100)_2$	0 (negative zero)
$(101)_2$	-1
$(110)_2$	-2
$(111)_2$	-3

Example

$(000)_2 = 0$

$(010)_2 = 2$

$(110)_2 = -(1 \cdot 2^1 + 0 \cdot 2^0) = -2$

Negative Numbers in the Binary Number System

- One's complement (diminished-radix complement)
  - The first bit is the sign bit.
    - 0: Positive
    - 1: Negative
  - Value
    - If $X = (0\cdots)_2$, its value is $x_{n-2} \cdot 2^{n-2} + \cdots + x_1 \cdot 2^1 + x_0 \cdot 2^0$.
    - If $X = (1\cdots)_2$, its value is $-(\overline{x_{n-2}} \cdot 2^{n-2} + \cdots + \overline{x_1} \cdot 2^1 + \overline{x_0} \cdot 2^0)$.

Representation	Value
$(000)_2$	0
$(001)_2$	1
$(010)_2$	2
$(011)_2$	3
$(100)_2$	-3
$(101)_2$	-2
$(110)_2$	-1
$(111)_2$	0 (negative zero)

Example

$(000)_2 = 0$

$(010)_2 = 2$

$(110)_2 = -(0 \cdot 2^1 + 1 \cdot 2^0) = -1$

$(101)_2 = -(1 \cdot 2^1 + 0 \cdot 2^0) = -2$

Negative Numbers in the Binary Number System

- Two's complement (radix complement)
  - The first bit is the sign bit.
    - 0: Positive
    - 1: Negative
  - Value
    - If $X = (0\cdots)_2$, its value is $x_{n-2}\cdot 2^{n-2}+\cdots+x_1\cdot 2^1+x_0\cdot 2^0$.
    - If $X = (1\cdots)_2$, its value is $-(\overline{x_{n-2}}\cdot 2^{n-2}+\cdots+\overline{x_1}\cdot 2^1+\overline{x_0}\cdot 2^0+1)$.

Representation	Value
$(000)_2$	0
$(001)_2$	1
$(010)_2$	2
$(011)_2$	3
$(100)_2$	-4 (exception)
$(101)_2$	-3
$(110)_2$	-2
$(111)_2$	-1

Example

$(000)_2 = 0$

$(010)_2 = 2$

$(110)_2 = -(0\cdot 2^1 + 1\cdot 2^0$

Negative Numbers in the Binary Number System

- Conversion rules
  - Positive $(0x_{n-2}\cdots x_1x_0)_2 \rightarrow$ Negative $(1\cdots)_2$

Sign and Magnitude	One's complement	Two's complement
$(1x_{n-2}\cdots x_1x_0)_2$	$(1\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$	$(1\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$ $+(0\cdots 01)_2$

  - Negative $(1x_{n-2}\cdots x_1x_0)_2 \rightarrow$ Positive $(0\cdots)_2$

Sign and Magnitude	One's complement	Two's complement
$(0x_{n-2}\cdots x_1x_0)_2$	$(0\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$	$(0\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$ $+(0\cdots 01)_2$

  - Positive $\leftrightarrow$ Negative: Given $(x_{n-1}\cdots x_1x_0)_2$

Sign and Magnitude	One's complement	Two's complement
$(\overline{x_{n-1}}x_{n-2}\cdots x_1x_0)_2$	$(\overline{x_{n-1}}\,\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$	$(\overline{x_{n-1}}\,\overline{x_{n-2}}\cdots\overline{x_1}\,\overline{x_0})_2$ $+(0\cdots 01)_2$

Negative Numbers in the One's Complement

- Different interpretation of the one's complement
  - Positive: $X = (0x_{n-2}x_{n-3}\cdots x_1x_0)_2$
  - Negative: $-X = (1\overline{x_{n-2}}\,\overline{x_{n-3}}\cdots\overline{x_1}\,\overline{x_0})_2$

- Observation
  - $(S)_2 = (X)_2 + (-X)_2$ (just a literal addition, not arithmetic)

$$X = (0x_{n-2}x_{n-3}\cdots x_1x_0)_2$$
$$+\ -X = (1\overline{x_{n-2}}\,\overline{x_{n-3}}\cdots\overline{x_1}\,\overline{x_0})_2$$
$$S = (1\ \ 1\ \ \ \ 1\ \cdots\ 1\ \ 1)_2$$

  - $(X)_2 + (-X)_2 = (11\cdots1)_2 = 2^n - 1$ (n: # bits, unsigned)

Example

$$X = (0011)_2 = 3$$
$$-X = (1100)_2 = -3$$

$$(0011)_2$$
$$(1100)_2$$
$$S = (1111)_2 = 15$$
$$= 2^4 - 1$$

Negative Numbers in the One's Complement

- How to obtain $(-X)_2$ from a positive number $X = (x_{n-1} \cdots x_1 x_0)_2$
  - $(\overline{x_{n-1}} \cdots \overline{x_1}\ \overline{x_0})_2$
  - $(2^n - 1 - X)_2$ (treat $2^n - 1 - X$ as an unsigned number)

- How to obtain $(X)_2$ from a negative number $-X = (x_{n-1} \cdots x_1 x_0)_2$
  - $(\overline{x_{n-1}} \cdots \overline{x_1}\ \overline{x_0})_2$
  - $(2^n - 1 - (-X))_2$ (treat $(-X)$ and $2^n - 1 - (-X)$ as unsigned numbers)

Example

$$X = (0011)_2 = 3$$
$$-X = (1100)_2 = -3$$
$$2^4 - 1 - 3 = 12$$
$$12 = (1100)_2$$

$$-X = (1100)_2 = -3$$
$$2^4 - 1 - 12 = 3$$
$$3 = (0011)_2$$

Negative Numbers in the One's Complement

- How can we use the new interpretation?
  - Prove $-(-X) = X$ for a positive number X.
    - Proof 1)

$$X = (x_{n-1} \cdots x_1 x_0)_2$$
$$-X = (\overline{x_{n-1}} \cdots \overline{x_1}\ \overline{x_0})_2$$
$$-(-X) = (\overline{\overline{x_{n-1}}} \cdots \overline{\overline{x_1}}\ \overline{\overline{x_0}})_2 = (x_{n-1} \cdots x_1 x_0)_2 = X$$

    - Proof 2)

$$X = (x_{n-1} \cdots x_1 x_0)_2$$
$-X$ is represented by $2^n - 1 - X$
$-(-X)$ is represented by $2^n - 1 - (2^n - 1 - X)$
$$2^n - 1 - (2^n - 1 - X) = X$$
$$\therefore -(-X) = X$$

Negative Numbers in the Two's Complement

- Different interpretation of the two's complement
  - Positive: $X = (0x_{n-2}x_{n-3}\cdots x_1x_0)_2$
  - Negative: $-X = (1\overline{x_{n-2}}\,\overline{x_{n-3}}\cdots\overline{x_1}\,\overline{x_0} + 0\cdots01)_2$
- Observation
  - $(S)_2 = (X)_2 + (-X)_2$ (just a literal addition, not arithmetic)

$$X = (0x_{n-2}x_{n-3}\cdots x_1x_0)_2$$
$$+\ -X = (1\overline{x_{n-2}}\,\overline{x_{n-3}}\cdots\overline{x_1}\,\overline{x_0} + 1)_2$$

$$S = (1\ \ 1\ \ 1\ \cdots\ 1\ \ 1)_2$$
$$+(0\ \ 0\ \ 0\ \ldots\ 0\ \ 1)_2$$
$$= (100\cdots0)_2$$

  - $(X)_2 + (-X)_2 = (100\cdots0)_2 = 2^n$ (n: # bits, unsigned)

Example

$$X = (0011)_2 = 3$$
$$-X = (1100 + 0001)_2$$
$$= (1101)_2 = -3$$

$$(0011)_2$$
$$(1101)_2$$

$$S = (10000)_2 = 16$$
$$= 2^4$$

Negative Numbers in the Two's Complement

- How to obtain $(-X)_2$ from a positive number $X = (x_{n-1} \cdots x_1 x_0)_2$
  - $(\overline{x_{n-1}} \cdots \overline{x_1}\,\overline{x_0} + 1)_2$
  - $(2^n - X)_2$ (treat $2^n - X$ as an unsigned number)

- How to obtain $(X)_2$ from a negative number $-X = (x_{n-1} \cdots x_1 x_0)_2$
  - $(\overline{x_{n-1}} \cdots \overline{x_1}\,\overline{x_0} + 1)_2$
  - $(2^n - (-X))_2$ (treat $(-X)$ and $2^n - (-X)$ as unsigned numbers)

Example

$$X = (0011)_2 = 3$$
$$-X = (1101)_2 = -3$$
$$2^4 - 3 = 13$$
$$13 = (1101)_2$$

$$-X = (1101)_2 = -3$$
$$2^4 - 13 = 3$$
$$3 = (0011)_2$$

Negative Numbers in the Two's Complement

- How can we use the new interpretation?
  - Prove $-(-X) = X$ for a positive number X.
    - Proof 1)

$$X = (x_{n-1} \cdots x_1 x_0)_2$$
$$-X = (\overline{x_{n-1}} \cdots \overline{x_1}\,\overline{x_0} + 1)_2$$

If $X = 0$, $-X = (11 \cdots 1 + 1)_2 = (00 \cdots 0)_2$

$$-(-X) = (11 \cdots 1 + 1)_2 = (00 \cdots 0)_2 = 0$$

If $X \neq 0$, suppose x_i is the first LSB 1 appearing in $(x_{n-1} \cdots x_1 x_0)_2$.
(i.e., $x_i = 1$ and $x_{i-1} = x_{i-2} = \cdots = x_0 = 0$)
Then, $-X = (\overline{x_{n-1}} \cdots \overline{x_1}\,\overline{x_0} + 1)_2 = (\overline{x_{n-1}} \cdots \overline{x_{i+1}} 100 \cdots 0)_2$
(where the tailing zeros are from x_{i-1} to x_0.)

$$-(-X) = (\overline{\overline{x_{n-1}}} \cdots \overline{\overline{x_{i+1}}} 011 \cdots 1 + 1)_2 = (x_{n-1} \cdots x_{i+1} 100 \cdots 0)_2 = (x_{n-1} \cdots x_1 x_0)_2$$

    - Proof 2)

$$X = (x_{n-1} \cdots x_1 x_0)_2$$

$-X$ is represented by $2^n - X$

$-(-X)$ is represented by $2^n - (2^n - X)$

$$2^n - (2^n - X) = X$$
$$\therefore -(-X) = X$$

Addition (Unsigned)

- The range of the input: $[0, 2^n - 1]$
- The range of the output (sum): $[0, 2^{n+1} - 2]$
  - How many bits do we need to represent the sum?
    - $\underline{2^n - 1} < 2^{n+1} - 2 < \underline{2^{n+1} - 1}$

Max. that can be represented with n bits

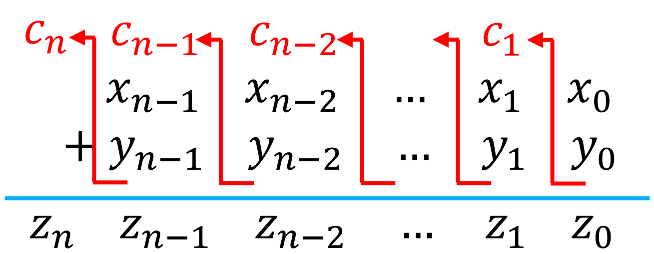
Max. that can be represented with $n+1$ bits

- Sum

$$
\begin{array}{r}
x_{n-1}x_{n-2}\ldots x_1x_0 \\
+\ y_{n-1}y_{n-2}\ldots y_1y_0 \\
\hline
\underline{z_n}z_{n-1}z_{n-2}\ldots z_1z_0
\end{array}
$$

Overflow

  - How do you compute $z_0, z_1, \ldots, z_{n-1}$?



Addition (Unsigned)

- Overflow (the sum cannot be represented with n bits)

$$
\begin{array}{r}
1111\ 1111 \\
+0000\ 0001 \\
\hline
1\ 0000\ 0000
\end{array}
$$

- How can we detect overflows?
  - Use the most-significant carry-out bit

$$
\begin{array}{r}
1111\ 1111 \\
+0000\ 0001 \\
\hline
\mathbf{1}\ 0000\ 0000
\end{array}
$$

  - Compare the MSBs

$$
\begin{array}{r}
\mathbf{1}111\ 1111 \\
+\mathbf{0}000\ 0001 \\
\hline
\mathbf{0}000\ 0000
\end{array}
$$

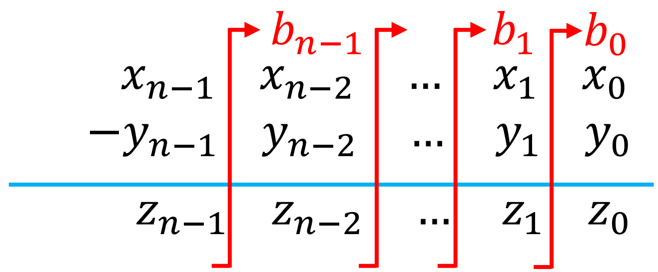
x_{n-1}	0	0 (or 1)	0 (or 1)	1
y_{n-1}	0	1 (or 0)	1 (or 0)	1
z_{n-1}	X	0	1	X
Overflow	No	Yes	No	Yes

Subtraction (Unsigned)

- Subtraction

$$
\begin{array}{r}
x_{n-1}x_{n-2}\ldots x_1x_0 \\
-y_{n-1}y_{n-2}\ldots y_1y_0 \\
\hline
z_{n-1}z_{n-2}\ldots z_1z_0
\end{array}
$$

  - How do you compute $z_0, z_1, \ldots, z_{n-1}$?



- How can we detect underflows?

Addition (S&M)

- Now, there is no difference between addition and subtraction.
  - $A - B = A + (-B)$
- The range of the input: $[-2^{n-1}, 2^{n-1} - 1]$
- The range of the output (sum): $[-2^n, 2^n - 2]$
  - How many bits do we need to represent the sum?
    - Max: $2^n - 2 < \underline{2^n - 1}$

      Max. that can be represented with $n + 1$ bits
    - Min: -2^n (need $n + 1$ bits)
  - This means that there could be an overflow or underflow.

Addition (S&M)

- How to add
  - Positive + positive: Easy (exactly the same as the unsigned numbers).
    - Overflow
  - Negative + negative: Easy (exactly the same as the unsigned numbers)
    - Underflow

x_{n-1}	1	1
y_{n-1}	1	1
z_{n-1}	0	1
Overflow	Yes	No

  - Positive + negative (or negative + positive): Not easy to implement.

Addition (One's Complement)

- How to add
  - Positive + positive: Easy (exactly the same as the unsigned numbers).
    - Overflow
  - Positive + negative (or negative + positive)
    - X (positive), Y (negative)
    - Let $Y = -T$ (where T is positive.)
    - $Z = \underbrace{X + Y}_{\text{Value}} = \underbrace{X + (-T)}_{\text{Value}} = \underbrace{(X)_2 + (\bar{T})_2}_{\text{Representation}} = \underbrace{(X)_2 + (2^n - T - 1)_2}_{\text{Representation}} = \underbrace{2^n + (X - T - 1)}_{\text{Rearrangement in unsigned}}$
    - If $X > |Y|$ (i.e., $X > T$), then $Z = 2^n + S$ where $S = X - T - 1 \geq 0$.

$$
\begin{array}{r}
x_{n-1}x_{n-2}\,\dots\,x_1x_0 \\
+y_{n-1}y_{n-2}\,\dots\,y_1y_0 \\
\hline
\underbrace{1}_{2^n}\ \underbrace{\boxed{s_{n-1}s_{n-2}\,\dots\,s_1s_0}}_{X-T-1}
\end{array}
$$

    - To obtain $X - T$ $(= X + Y)$, we should add 1 to S (this is a correction step).

Addition (One's Complement)

- Positive + negative (or negative + positive)
  - X (positive), Y (negative)
  - Let $Y = -T$ (where T is positive.)
  - $Z = \underbrace{X + Y}_{\text{Value}} = \underbrace{X + (-T)}_{\text{Value}} = \underbrace{(X)_2 + (\bar{T})_2}_{\text{Representation}} = \underbrace{(X)_2 + (2^n - T - 1)_2}_{\text{Representation}} = \underbrace{2^n + (X - T - 1)}_{\text{Rearrangement in unsigned}}$
  - If $X < |Y|$ (i.e., $X < T$), then $Z = 2^n - 1 - S$ where $S = T - X > 0$. $Z = \bar{S}$, which is the representation for $-S$ in one's complement. $-S = (X - T) = X + Y$, so we obtained the correct result (no correction step is needed).

- Negative + negative
  - X (negative), Y (negative)
  - Let $X = -A$ and $Y = -B$ (so A and B are positive)
  - $Z = X + Y = (-A) + (-B) = (\bar{A})_2 + (\bar{B})_2 = (2^n - A - 1)_2 + (2^n - B - 1)_2 = \underline{\{2^n - (A + B) + 1\}} + \{\underline{2^n - 1}\}$

$\overline{A + B} = -(A + B) = X + Y$ should be removed (by adding 1 to the result, i.e., need a correction step).

Addition (One's Complement)

- Summary (add X and Y)

	$X, Y > 0$	$X > 0, Y < 0\ (X > \lvert Y \rvert)$	$X > 0, Y < 0\ (X < \lvert Y \rvert)$	$X, Y < 0$
How to add	$(X)_2 + (Y)_2$	$(X)_2 + (Y)_2 + 1$	$(X)_2 + (Y)_2$	$(X)_2 + (Y)_2 + 1$
Overflow/underflow	Overflow	No	No	Underflow

- Example

$$\begin{array}{rr} 4 & 0100 \\ +\,2 & +0010 \\ \hline =6 & 0110 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ -\,3 & +1100 \\ & +0001 \\ \hline =1 & 0001 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ -\,7 & +1000 \\ \hline =-3 & 1100 \end{array}$$

$$\begin{array}{rr} -4 & 1011 \\ -3 & +1100 \\ & +0001 \\ \hline =-7 & 1000 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ +\;6 & +0110 \\ \hline =-2 & \underline{1}010 \end{array}$$

overflow

$$\begin{array}{rr} 7 & 0111 \\ -\;2 & +1101 \\ & +0001 \\ \hline =5 & 0101 \end{array}$$

$$\begin{array}{rr} 2 & 0010 \\ -4 & +1011 \\ \hline =-2 & 1101 \end{array}$$

$$\begin{array}{rr} -5 & 1010 \\ -3 & +1100 \\ & +0001 \\ \hline =7 & \underline{0}111 \end{array}$$

underflow

Addition (Two's Complement)

- How to add
  - Positive + positive: Easy (exactly the same as the unsigned numbers).
    - Overflow
  - Positive + negative (or negative + positive)
    - X (positive), Y (negative)
    - Let $Y = -T$ (where T is positive.)
    - $Z = \underbrace{X + Y}_{\text{Value}} = \underbrace{X + (-T)}_{\text{Value}} = \underbrace{(X)_2 + (\bar{T} + 1)_2}_{\text{Representation}} = \underbrace{(X)_2 + (2^n - T)_2}_{\text{Representation}} = \underbrace{2^n + (X - T)}_{\text{Rearrangement in unsigned}}$

      Value | Value | Representation | Representation | Rearrangement in unsigned
    - If $X > |Y|$ (i.e., $X > T$), then $Z = 2^n + S$ where $S = X - T > 0$.

$$
\begin{array}{r}
x_{n-1}x_{n-2} \dots x_1x_0 \\
+y_{n-1}y_{n-2} \dots y_1y_0 \\
\hline
\underbrace{1}_{2^n}\ \underbrace{\boxed{s_{n-1}s_{n-2} \dots s_1s_0}}_{X - T}
\end{array}
$$

    - To obtain $X - T$ $(= X + Y)$, just drop 1 in the (n+1)-th position.

Addition (Two's Complement)

- Positive + negative (or negative + positive)
  - X (positive), Y (negative)
  - Let $Y = -T$ (where T is positive.)
  - $Z = \underbrace{X + Y}_{\text{Value}} = \underbrace{X + (-T)}_{\text{Value}} = \underbrace{(X)_2 + (\bar{T} + 1)_2}_{\text{Representation}} = \underbrace{(X)_2 + (2^n - T)_2}_{\text{Representation}} = \underbrace{2^n + (X - T)}_{\text{Rearrangement in unsigned}}$
  - If $X < |Y|$ (i.e., $X < T$), then $Z = 2^n - S$ where $S = T - X > 0$. $Z = \bar{S} + 1$, which is the representation for $-S$ in two's complement. $-S = (X - T) = X + Y$, so we obtained the correct result (no correction step is needed).

- Negative + negative
  - X (negative), Y (negative)
  - Let $X = -A$ and $Y = -B$ (so A and B are positive)
  - $Z = X + Y = (-A) + (-B) = (\bar{A} + 1)_2 + (\bar{B} + 1)_2 = (2^n - A)_2 + (2^n - B)_2 = \underline{\{2^n - (A + B)\}} + \underline{\{2^n\}} = 2^n + \{\overline{(A + B)} + 1\}$

$\overline{A + B} + 1 = -(A + B) = X + Y$ should be removed (just by dropping 1 in the (n+1)-th position).

Addition (Two's Complement)

- Summary (add X and Y)

	$X, Y > 0$	$X > 0, Y < 0\ (X > \lvert Y \rvert)$	$X > 0, Y < 0\ (X < \lvert Y \rvert)$	$X, Y < 0$
How to add	$(X)_2 + (Y)_2$			
Overflow/underflow	Overflow	No	No	Underflow

- Example

$$\begin{array}{rr} 4 & 0100 \\ +\,2 & +0010 \\ \hline =6 & 0110 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ -\,3 & +1101 \\ \hline =1 & 0001 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ -\,7 & +1001 \\ \hline =-3 & 1101 \end{array}$$

$$\begin{array}{rr} -4 & 1100 \\ -3 & +1101 \\ \hline =-7 & 1001 \end{array}$$

$$\begin{array}{rr} 4 & 0100 \\ +\ \ 6 & +0110 \\ \hline =-2 & 1010 \end{array}$$

overflow

$$\begin{array}{rr} 7 & 0111 \\ -2 & +1110 \\ \hline =5 & 0101 \end{array}$$

$$\begin{array}{rr} 2 & 0010 \\ -4 & +1100 \\ \hline =-2 & 1110 \end{array}$$

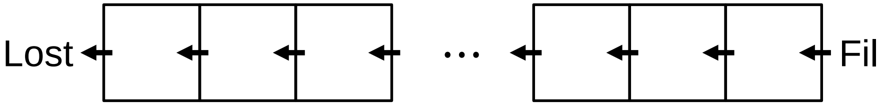
$$\begin{array}{rr} -5 & 1011 \\ -3 & +1101 \\ \hline =-8 & 1000 \end{array}$$

$$\begin{array}{rr} -5 & 1011 \\ -4 & +1100 \\ \hline =7 & 0111 \end{array}$$

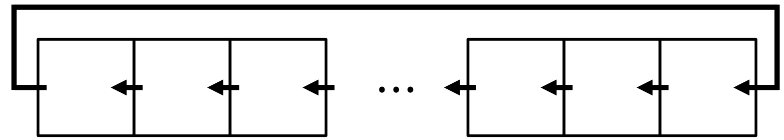
underflow

Shifting in the Binary Number System

- Shift vs. Rotate

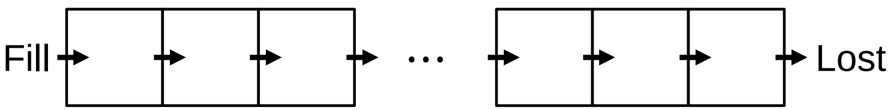


Shift operation (left)

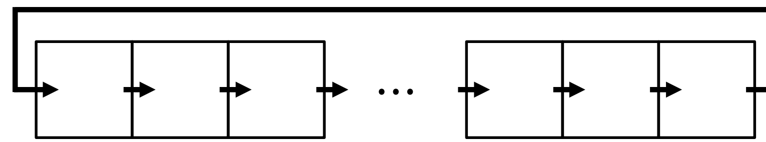


Rotate operation (left)

- Left vs. Right

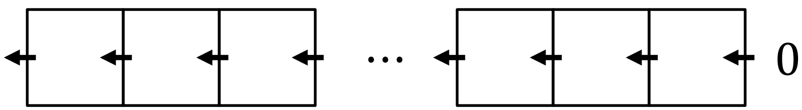


Shift operation (right)

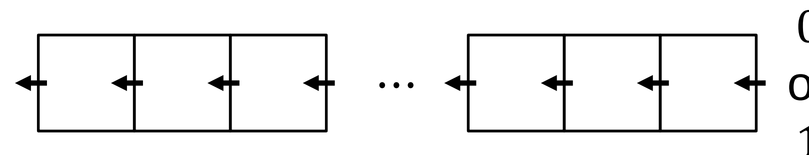


Rotate operation (right)

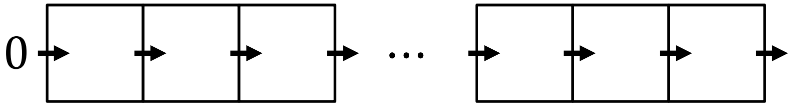
- Logical vs. Arithmetic



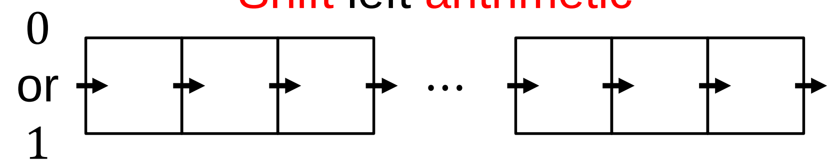
Shift left logical



Shift left arithmetic



Shift right logical



Shift right arithmetic

Shifting in the Binary Number System

- Shift left logical (by 1 bit, filled with 0's)
  - $X = (x_{n-1} \cdots x_2 x_1 x_0)_2$
  - $Y = (x_{n-2} \cdots x_1 x_0 0)_2$
  - Unsigned
    - $X = x_{n-1} \cdot 2^{n-1} + \cdots + x_2 \cdot 2^2 + x_1 \cdot 2^1 + x_0 \cdot 2^0$
    - $Y = x_{n-2} \cdot 2^{n-1} + \cdots + x_1 \cdot 2^2 + x_0 \cdot 2^1 + 0 \cdot 2^0$
      $= 2 \cdot (x_{n-2} \cdot 2^{n-2} + \cdots + x_1 \cdot 2^1 + x_0 \cdot 2^0)$
    - $Y = 2 \cdot (X - x_{n-1} \cdot 2^{n-1}) = 2X - x_{n-1} \cdot 2^n$
  - Sign-and-magnitude
    - If $x_{n-1}x_{n-2} = 00$ $(X > 0)$, $Y = 2X$
    - If $x_{n-1}x_{n-2} = 01$ $(X > 0)$, $Y = -(2X - 2^{n-1})$
    - If $x_{n-1}x_{n-2} = 10$ $(X < 0)$, $Y = -2X$
    - If $x_{n-1}x_{n-2} = 11$ $(X < 0)$, $Y = 2X + 2^{n-1}$

Example ($n = 5$)

1) $X = (00111)_2 = 7$
   $Y = (01110)_2 = 14 = 2 \cdot 7$
2) $X = (01011)_2 = 11$
   $Y = (10110)_2 = -6$
   $= -(2 \cdot 11 - 2^4)$
3) $X = (10110)_2 = -6$
   $Y = (01100)_2 = 12 = -2 \cdot 6$
4) $X = (11011)_2 = -11$
   $Y = (10110)_2 = -6$
   $= 2 \cdot (-11) + 2^4$
   $X = (11101)_2 = -13$
   $Y = (11010)_2 = -10$
   $= 2 \cdot (-13) + 2^4$

Shifting in the Binary Number System

- Shift left logical (by 1 bit, filled with 0's): $\times$ 2 in general
  - $X = (x_{n-1} \cdots x_2 x_1 x_0)_2$
  - $Y = (x_{n-2} \cdots x_1 x_0 0)_2$
  - 1's complement
    - If $x_{n-1}x_{n-2} = 00$ $(X > 0)$, $Y = 2X$
    - If $x_{n-1}x_{n-2} = 01$ $(X > 0)$, $Y = 2X + 1 - 2^n$
    - If $x_{n-1}x_{n-2} = 10$ $(X < 0)$, $Y = 2X - 2 + 2^n$
    - If $x_{n-1}x_{n-2} = 11$ $(X < 0)$, $Y = 2X - 1$
  - 2's complement
    - If $x_{n-1}x_{n-2} = 00$ $(X > 0)$, $Y = 2X$
    - If $x_{n-1}x_{n-2} = 01$ $(X > 0)$, $Y = 2X - 2^n = 2X \pmod{2^n}$
    - If $x_{n-1}x_{n-2} = 10$ $(X < 0)$, $Y = 2X + 2^n = 2X \pmod{2^n}$
    - If $x_{n-1}x_{n-2} = 11$ $(X < 0)$, $Y = 2X$

Example ($n = 5$) for 1's complement

2) $X = (01011)_2 = 11$
$Y = (10110)_2 = -9$
$= 2 \cdot 11 + 1 - 2^5$

3) $X = (10110)_2 = -9$
$Y = (01100)_2 = 12$
$= 2 \cdot (-9) - 2 + 2^5$

4) $X = (11011)_2 = -4$
$Y = (10110)_2 = -9$
$= 2 \cdot (-4) - 1$
$X = (11101)_2 = -2$
$Y = (11010)_2 = -5$
$= 2 \cdot (-2) - 1$

Shifting in the Binary Number System

- Shift right logical (by 1 bit, filled with 0's)
  - $X = (x_{n-1} \cdots x_2 x_1 x_0)_2$
  - $Y = (0x_{n-1} \cdots x_2 x_1)_2$
  - Unsigned
    - $X = x_{n-1} \cdot 2^{n-1} + \cdots + x_2 \cdot 2^2 + x_1 \cdot 2^1 + x_0 \cdot 2^0$
    - $Y = x_{n-1} \cdot 2^{n-2} + \cdots + x_3 \cdot 2^2 + x_2 \cdot 2^1 + x_1 \cdot 2^0$
      - If $x_0 = 0$, $Y = \frac{X}{2}$
      - If $x_0 = 1$, $Y = \frac{X-1}{2}$
      - Merge: $Y = \left\lfloor \frac{X}{2} \right\rfloor$ or $Y = \left\lfloor \frac{X-x_0}{2} \right\rfloor$
  - Sign-and-magnitude
    - If $x_{n-1} = 0$ $(X > 0)$, $Y = \left\lfloor \frac{X}{2} \right\rfloor$
    - If $x_{n-1}x_{n-2} = 10$ $(X < 0)$, $Y = x^{n-2} - \left\lfloor \frac{X}{2} \right\rfloor$
    - If $x_{n-1}x_{n-2} = 11$ $(X < 0)$, $Y = 2X + 2^{n-1}$

Example ($n = 5$)

1) $X = (00111)_2 = 7$
   $Y = (01110)_2 = 14 = 2 \cdot 7$
2) $X = (01011)_2 = 11$
   $Y = (10110)_2 = -6$
   $= -(2 \cdot 11 - 2^4)$
3) $X = (10110)_2 = -6$
   $Y = (01100)_2 = 12 = -2 \cdot 6$
4) $X = (11011)_2 = -11$
   $Y = (10110)_2 = -6$
   $= 2 \cdot (-11) + 2^4$
   $X = (11101)_2 = -13$
   $Y = (11010)_2 = -10$
   $= 2 \cdot (-13) + 2^4$

Extension

- Input: n-bit data X
- Output: $(n+k)$-bit data Y ($k > 0$) with $|Y| = |X|$
- Usage
  - int a = 8; (suppose “int” is a 32-bit data)
  - long b = -25; (suppose “long” is a 64-bit data)
  - long s = a + b (we should extent “a” to 64-bit)

- Sign & Magnitude
  - $X = x_{n-1}x_{n-2} \dots x_1x_0$
  - If x_{n-1} is 0 (i.e., X is positive), $Y = \underbrace{00 \dots 0}_{k \text{ bits}}x_{n-1}x_{n-2} \dots x_1x_0$
  - If x_{n-1} is 1 (i.e., X is negative), $Y = x_{n-1}\underbrace{0 \dots 00}_{k \text{ bits}}x_{n-2} \dots x_1x_0$
  - $Y = x_{n-1}\underbrace{0 \dots 00}_{k \text{ bits}}x_{n-2} \dots x_1x_0$

Extension

- One's complement
  - $X = x_{n-1}x_{n-2} \ldots x_1x_0$
  - If x_{n-1} is 0 (i.e., X is positive), $Y = \underbrace{00 \ldots 0}_{k \text{ bits}}x_{n-1}x_{n-2} \ldots x_1x_0$
  - If x_{n-1} is 1 (i.e., X is negative)
    - $|X| = \overline{x_{n-1}} \ldots \overline{x_1x_0}$
    - $|Y| = \overline{y_{(n+k-1)}} \ldots \overline{y_1y_0} = 0 \ldots 0\overline{x_{n-1}} \ldots \overline{x_1x_0} \rightarrow y_{(n+k-1)} = 1, \ldots, y_n = 1, y_{n-1} = x_{n-1}, \ldots, y_0 = x_0$
    - Thus, $Y = \underbrace{11 \ldots 1}_{k \text{ bits}}x_{n-1} \ldots x_1x_0$
  - $Y = \underbrace{x_{n-1}x_{n-1} \ldots x_{n-1}}_{k \text{ bits}}x_{n-1} \ldots x_1x_0$

Extension

- Two's complement
  - $X = x_{n-1}x_{n-2} \dots x_1x_0$
  - If x_{n-1} is 0 (i.e., X is positive), $Y = \underbrace{00 \dots 0}_{k \text{ bits}} x_{n-1}x_{n-2} \dots x_1x_0$
  - If x_{n-1} is 1 (i.e., X is negative)
    - $|X| = \overline{x_{n-1}} \dots \overline{x_1x_0} + 1$
    - $|Y| = \overline{y_{(n+k-1)}} \dots \overline{y_1y_0} + 1 = 0 \dots 0\overline{x_{n-1}} \dots \overline{x_1x_0} + 1 \rightarrow y_{(n+k-1)} = 1, \dots, y_n = 1, y_{n-1} = x_{n-1}, \dots, y_0 = x_0$
    - Thus, $Y = \underbrace{11 \dots 1}_{k \text{ bits}} x_{n-1} \dots x_1x_0$
  - $Y = \underbrace{x_{n-1}x_{n-1} \dots x_{n-1}}_{k \text{ bits}} x_{n-1} \dots x_1x_0$

References

- Israel Koren, “Computer Arithmetic Algorithms,”