

Computer Organization and Architecture

Prepared by **Dr. Eng. Gamal Selim**

Lecture 04 – Binary Arithmetic Operations



Arithmetic for Computers

Arithmetic for Computers

- Operations on integers
 - Addition and subtraction
 - Multiplication and division
- **ALU: Arithmetic Logic Unit**
 - HW performs addition, subtraction, and usually logical operations such as AND and OR

Addition in Binary

Addition in Binary

- In decimal if you add two digits and the sum is more than the base (10) you add all the 10s in the result to the next higher digit and leave the remainder in the current digit.
- The rule is the same for integer and decimal fractions.
- In binary this rule applies as :
 - $1+0 = 1$
 - $1+1 =$
 - 10 (as no 2 is allowed in binary)

Problem 1

Add the following in binary and check results by doing addition in decimal:

(b) $1111 + 0011$

Solution:

$$\begin{array}{rcl}
 (15)_{10}: & (1111)_2 & \\
 (3)_{10}: & (0011)_2 & + \\
 \hline
 (18)_{10}: & (10010)_2 &
 \end{array}$$

(c) $1011.1101 + 11.1$

Solution:

$$\begin{array}{rcl}
 (11.8125)_{10}: & (1011.1101)_2 & \\
 (3.5)_{10}: & (0011.1000)_2 & + \\
 \hline
 (15.3125)_{10}: & (1111.0101)_2 &
 \end{array}$$

(e) $1111000 + 00010010$

Solution:

$$\begin{array}{rcl}
 (120)_{10}: & (1111000)_2 & \\
 (18)_{10}: & (0010010)_2 & + \\
 \hline
 (138)_{10}: & (10001010)_2 &
 \end{array}$$

Example

Add 10011 and 11101 in binary:

10011 + 11101 ----- 110000

Signed Numbers

Signed number representation in binary

- We have two ways of representing signed numbers:
 1. Sign-magnitude system.
 2. 2's complement system

Sign-magnitude system

- In decimal we have positive and negative numbers like +20 or – 31
- In binary bits are stored in Flip-flops (units that can only store 1 or 0)
- The negative number in signed magnitude binary representation is as follows:
- $-4 = 1100$ (**sign bit** always at the left most bit =1)
- If represented in 8 bits $-4 = 1000\ 0100$
- Positive numbers have the sign bit = 0
- $+4 = 0100$ (**sign bit** always at the left most bit =0)
- If represented in 8 bits $+4 = 0000\ 0100$

2's complement system

- The positive form of a number is the same as the sign magnitude having the MSB as the sign bit with 0 to indicate positive value.
- The negative form of a number is done using the 2's complement of the positive number.
- So how can the 2's complement of a number be deduced ??
- We start by getting the 1's complement form of a number then the 2's complement.

1's complement

- The 1's complement of a number N is the number added to N to make it all 1s ($2^n - 1$).
- The 1's complement of N is $(2^n - 1) - N$.
- Note that $2^n - 1$ is a binary number represented by n 1s.
- For example, with $n = 4$, we have $2^4 = (10000)_2$ and $2^4 - 1 = (1111)_2$.
- Thus the 1's complement of a binary number is obtained by subtracting each digit from 1.

1's complement

- However, the subtraction of a binary digit from 1 causes the bit to change from 0 to 1 or from 1 to 0. Therefore, the 1's complement of a binary number is formed by changing 1s into 0s and 0s into 1s.
- For example, the 1's complement of 1011001 is 0100110 and the 1's complement of 0001111 is 1110000.
- What is the 1's complement of 0 ?

2's complement

- The 2's complement of a number N with n bits is the number added to N to make it 2^n .
- The 2's complement of N is $2^n - N$.
- Note that 2^n is represented by a binary number that consists of a 1 followed by n os.
- For example, with $n = 4$, we have $2^4 = (10000)_2$.
- Hence we have 2 ways to get the 2's complement:
 1. By getting the 1's complement then adding 1 to it $[(2^n - 1 - N) + 1] = 2^n - N$.
 2. By subtracting the number from the 1 followed by n os for example:

the 2's complement of 1100 is

$$\begin{array}{r} 10000 \\ - 1100 \\ \hline 0100 \end{array}$$

2's complement

- Hence, the 2's complement can be formed by leaving all least significant 0's and the first 1 unchanged, and then replacing 1's by 0's and 0's by 1's in all other higher significant bits.
- The 2's complement of 1101100 is 0010100 and is obtained by leaving the two low-order 0's and the first 1 unchanged, and then replacing 1's by 0's and 0's by 1's in the other four most significant bits.
- If the original number N contains a radix point, it should be removed temporarily to form the 2's or 1's complement then restored to the complemented number in the same relative position.

2's complement

- The complement of the complement restores the number to its original value.
- The 2's complement of N is $2^n - N$. The complement of the complement is $2^n - (2^n - N) = N$ giving back the original number.
- Representing binary numbers in 2's complement form enables us to represent negative numbers.

Problem 2

Repeat each of the following signed decimal numbers in 2's-complement system. Use a total of 8 bits including sign bit

(a)+32

Solution:

The representation of 32 in binary is 100000

In 8 bits = **0**0100000

Problem 2

(b) -1

Solution:

- First we obtain the positive number in binary then get the 2's complement of this binary number in 8 bits
- **1 = 00000001**
- **2's complement of 00000001**
- **-1 = 11111111**

Problem 2 cont.

Repeat each of the following signed decimal numbers in 2's-complement system. Use a total of 8 bits including sign bit

(c) -128

Solution:

- First we obtain the positive number in binary then get the 2's complement of this binary number
- $128 = 2^7$
- $128 \rightarrow 1000\underline{0000}$
- 2's complement of $1000\underline{0000}$
- $-128 = 10000000$

Problem 3

Each of the following numbers represents a signed decimal number in the 2's-complement system. determine the decimal value in each case. (Hint: Use negative to convert negative numbers to positive)

(a) 01101

Solution:

The sign bit = 0

Then the number is positive and the value =

$$8+4+1 = 13$$

Problem 3

Each of the following numbers represents a signed decimal number in the 2's-complement system. determine the decimal value in each case. (Hint: Use negative to convert negative numbers to positive)

(b) 01111111

Solution:

The sign bit = 0

Then the number is positive and the value =

$$2^7 - 1 = 128 - 1 = 127$$

Problem 3 cont.

Each of the following numbers represents a signed decimal number in the 2's-complement system. determine the decimal value in each case. (Hint: Use negative to convert negative numbers to positive)

(d) 1000000

Solution:

The sign bit = 1

Then the number is negative in 2's complement form and the value is the 2's complement of the number

The 2's complement of 1000000 = 1000000

Decimal value of 1000000 = $2^6 = -64$

Hence the decimal value of 11111111 = -1

Problem 4

(a) What range of signed decimal values can be represented using 12 bits in 2's complement form including sign bit?

Solution:

When excluding the sign bit we have 11 bits hence the largest decimal positive number = $2^n - 1 = 2^{11} - 1 = 2048 - 1 = 2047$

- In negative the largest value in 12 bits is
- 100000000000
- Because the 2's complement of this value is equal to the same number which is equal to 2^{11} hence the largest negative value is -2048
- The range for signed decimal values can be represented using N bits including sign bit is from $2^{(N-1)} - 1$ to $-2^{(N-1)}$

(b) How many bits would be required to represent decimal numbers from -32,768 to 32,767?

Solution:

we get 2^n representing -32,768 by getting $\log 32,768 / \log 2 = 15$

- Hence in $(15+1) = 16$ bits we can represent -2^{15} to $+2^{15}-1$

Problem 5

Represent each of the following decimal values as 5-bit signed magnitude binary values. Then negate each one.

(b) -12

Solution:

-12 =

11100

The negative of -12 = 01100

Subtraction in Binary

Subtraction in binary

- The direct method of subtraction taught in elementary schools uses the borrow concept.
- In this method we borrow a 1 from a higher significant position when the minuend digit is smaller than the corresponding subtrahend digit.
- In binary this is difficult as bits are stored in Flip-flops and cannot do the borrow operation.
- Hence the subtraction is converted to addition using complements.

Subtraction using 1's complement

- If we have two numbers M, N (M and N have n digits)
- We have the 1's complement of $N = (2^n - 1) - N$
- Hence to get the M-N we have
- $(M + 1's \text{ complement of } N)$
$$= M + ((2^n - 1) - N)$$
$$= M - N + 2^n - 1$$
- That leads to the conclusion that when you need to subtract two numbers you can add the 1's form of the subtrahend to the minuend then discard the carry digit ($2^n = 1$ followed by n digits) and add 1 to get the result you need.

Subtraction using 1's complement

Example:

For $b = 10$ ($M = 35$, $N = 14$, 9s complement of $14 = 85$)

$$\begin{array}{r} 35 + 85 = \\ 1 \quad 20 \end{array}$$

To get the final result discard the 1(MSD) and add 1

$$M - N = 21$$

Example: For $b = 2$ ($M = 100011$, $N = 001110$)

$$\begin{array}{r} 100011 + 110001 = \\ 1 \quad 010100 \end{array}$$

To get the final result discard the 1(MSD) and add 1

$$M - N = (010101)_2 = 21$$

Subtraction using 2's complement

- The subtraction of two n-digit unsigned numbers $M - N$ ($N \neq 0$) in base b can be done as follows:
 1. Add the minuend M to the 2's complement of the subtrahend N . This performs $M + (2^n - N) = M - N + 2^n$.
 2. There are three cases :
 1. If $M > N$, the sum will produce an end carry 2^n which is discarded, and what is left is the result $M - N$.
 2. If $M < N$, the sum does not produce an end carry and is equal to $2^n - (N - M)$, which is the 2's complement of $(N - M)$. To obtain the answer in a familiar form, take the b 's complement of the sum and place a negative sign in front.

Subtraction using 2's complement

3. If N and M are both in b's complement form (i.e. both are considered negative then)

- We add the two numbers to get the result which will be considered negative
- $(2^n - M) + (2^n - N) = 2^n - (M + N) + 2^n$
- The result indicated that there will be a carry bit which will be discarded then the rest of the result is considered the 2's complement of the needed result.

Problem 6

Perform the following operation in 2's-complement system. Use 8-bit (including the sign bit) for each number. Check your results by converting the binary result back to decimal.

(a) Add +19 to -24

$$(24)_{10} = (00011000)_2$$

$$(19)_{10}: (00010011)_2$$

$$(-24)_{10}: (11101000)_2$$

$$(-5)_{10}: 11111011$$

$$(5)_{10}: (00000101)_2$$

+

(b) Add -48 to -80

$$(48)_{10} = (00110000)_2$$

$$(80)_{10} = (01010000)_2$$

$$(-48)_{10}: \overset{1}{(11010000)_2}$$

$$(-80)_{10}: (10110000)_2$$

$$(-128)_{10}: 1\ 10000000$$

$$(128)_{10}: (10000000)_2$$

+

Problem 6 cont.

Perform the following operation in 2's-complement system. Use 8-bit (including the sign bit) for each number. Check your results by converting the binary result back to decimal.

(c) Subtract + 16 from +17

$$17 - 16 = 17 + (-16)$$

$$(16)_{10} = (00010000)_2$$

$$(17)_{10}: \quad (00010001)_2 \quad +$$

$$(-16)_{10}: \quad (11110000)_2$$

$$(1)_{10}: \quad 1 \ 00000001$$

Problem 6 cont.

Perform the following operation in 2's-complement system. Use 8-bit (including the sign bit) for each number. Check your results by converting the binary result back to decimal.

(d) Subtract +21 from -13

$$-13 - (21) = -13 + (-21)$$

$$(13)_{10} = (0000\underline{1101})_2$$

$$(21)_{10} = (0001\underline{0101})_2$$

$$(-13)_{10}: \quad (1111\underline{0011})_2 \quad +$$

$$(-21)_{10}: \quad (1110\underline{1011})_2$$

$$\hline (-34)_{10}: \quad 1 \ 11011110$$

$$(34)_{10}: \quad (0010\underline{0010})_2$$

Problem 6 cont.

Perform the following operation in 2's-complement system. Use 8-bit (including the sign bit) for each number. Check your results by converting the binary result back to decimal.

(g) Subtract -17 from -17
 $-17 - (-17) = -17 + 17 = 17 - 17$

$(17)_{10}:$	$(00010001)_2$	+
$(-17)_{10}:$	$(11101111)_2$	
$(0)_{10}:$	1 00000000	

Multiplication in Binary

Problem 7

Multiply the following pairs of binary numbers and check by doing multiplication in decimal.

(b) 1011×1011

Solution:

$$(11)_{10} : (1011)_2$$

$$(11)_{10} : (1011)_2$$

×

$$\begin{array}{r}
 1011 \\
 1011 \\
 0000 \\
 1011 \\
 \hline
 \end{array}$$

$$(121)_{10} : 111\underline{1001}$$

(d) 11111×101

Solution:

$$(31)_{10} : (11111)_2$$

$$(5)_{10} : (101)_2$$

×

$$\begin{array}{r}
 11111 \\
 00000 \\
 11111 \\
 \hline
 \end{array}$$

$$(155)_{10} : 1001\underline{1011}$$

Problem 8

Another example of binary multiplication with a decimal point is as follows:

Question: 1011.01×110.1

Solution:

$$\begin{array}{r}
 1011.01 \\
 110.1 \\
 \hline
 101101 \\
 000000 \\
 \hline
 0101101 \quad \text{..... First Intermediate sum} \\
 101101 \\
 \hline
 11100001 \quad \text{..... Second Intermediate Sum} \\
 101101 \\
 \hline
 1001001.001 \quad \text{..... Final Sum}
 \end{array}$$

Division in Binary

Problem 9

Perform the following divisions. Check the results by doing division in decimal.

(a) $111111/1001$

Solution

$$\begin{array}{r}
 000111 \text{ (7)}_{10} \\
 (1001)_2 \overline{) (111111)_2} \\
 \underline{1001} \\
 01101 \\
 \underline{1001} \\
 01001 \\
 \underline{1001} \\
 0000
 \end{array}$$

Problem 9 cont.

Perform the following divisions. Check the results by doing division in decimal.

(b) $1110111/111$

Solution

$$\begin{array}{r}
 0010001 \quad (17)_{10} \\
 \hline
 (7)_{10} : (111)_2 \quad (1110111)_2 : (111)_2 \\
 \underline{111} \\
 0000111 \\
 \underline{111} \\
 000
 \end{array}$$



مجموعة جامعة
الجميع الذكية
للتعليم القابضة

MIDOCEAN
UNIVERSITY
BY EVERYONE'S SMART UNIVERSITY



Thank You

