





Bits



OBJECTIVES



After reading this chapter, the reader should be able to:

- Apply arithmetic operations on bits when the integer is represented in two's complement.
- Apply logical operations on bits.
- Understand the applications of logical operations using masks.
- Understand the shift operations on numbers and how a number can be multiplied or divided by powers of two using shift operations.





Operations on bits











Arithmetic operations

- Arithmetic operations involve:
 - Adding (+)
 - Subtracting (--)
 - Multiplying (X)
 - Dividing (/)
 - And so on...





Addition in two's complement





Table 4.1Adding bits







Rule of Adding Integers in Two's Complement

Add 2 bits and propagate the carry to the next column. If there is a final carry after the leftmost column addition, discard (捨棄) it





Add two numbers in two's complement representation: $(+17) + (+22) \square (+39)$

Solution

Carry 1

 0
 0
 0
 1
 0
 0
 0
 1
 +

 0
 0
 0
 1
 0
 1
 1
 0
 +

Result 0 0 1 0 0 1 1 1 □ 39





Add two numbers in two's complement representation: $(+24) + (-17) \square (+7)$

Solution

Carry 1 1 1 1 1 0 0 0 1 1 0 0 0 + 1 1 1 0 1 1 1

Result 0 0 0 0 0 1 1 1 □ +7





Add two numbers in two's complement representation: $(-35) + (+20) \square (-15)$

Solution

Carry 1 1 1

 1
 1
 0
 1
 1
 1
 0
 1
 +

 0
 0
 0
 1
 0
 1
 0
 0
 +

Result 1 1 1 1 0 0 0 1 -15





Add two numbers in two's complement representation: $(+127) + (+3) \square (+130)$

Solution

Carry 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 + 0 0 0 0 0 0 0 1 1

 Result
 1
 0
 0
 0
 1
 0
 -126 (Error)

 An overflow has occurred.
 An overflow has occurred.







Range of numbers in two's complement representation





Two's complement numbers visualization





©Brooks/Cole, 2003





When you do arithmetic operations on numbers in a computer, remember that each number and the result should be in the range defined by the bit allocation.







Arithmetic operations on floating-point numbers



- Addition and subtraction for floating-point numbers are one process. (p. 54)
 - Check the sign. (a, b)
 - Move the decimal points to make the exponents the same.
 - Add or subtract the mantissas (底數).
 - Normalize the result before storing in memory.
 - Check for any overflow.



Addition

Example 6

Solution











Logical operations



- A logical operation can accept 1 or 2 bits to create only 1 bit.
 - Unary operation (Figure 4.3)
 - Binary operation (Figure 4.3)







©Brooks/Cole, 2003



Logical operations







Truth tables

			AND		
10 - 10 - 10 - 10			Х	У	x AND y
NOT			0	0	0
X	NOTx		0	1	0
0	1		1	0	0
1	0		1	1	
					N
OR	A		XOR		
OR x	у	x OR y	XOR x	у	x XORy
OR x 0	У 0	x OR y 0	XOR x	у 0	x XORy 0
OR x 0 0	у 0 1	x OR y 0 1	XOR x 0 0	у 0 1	x XORy 0 1
OR x 0 0 1	y 0 1 0	x OR y 0 1 1	XOR	y 0 1 0	x XORy 0 1 1



Unary operator -- NOT operator





©Brooks/Cole, 2003



NOT operator

Example 7

Use the NOT operator on the bit pattern 10011000

Solution

 Target
 10011000
 NOT

Result

01100111



Binary operator--AND operator



AND operator *Example 8*



Use the AND operator on bit patterns 10011000 and 00110101.

Solution

 Target
 10011000
 AND

 00110101

Result

00010000





Inherent (本質的) rule of the AND operator





Binary operator--OR operator





©Brooks/Cole, 2003

OR operator *Example 9*



Use the OR operator on bit patterns 10011000 and 00110101

10111101

Solution

 Target
 10011000
 OR

 00110101

Result





Inherent rule of the OR operator





Binary operator--XOR operator





©Brooks/Cole, 2003

XOR operator *Example 10*



Use the XOR operator on bit patterns 10011000 and 00110101.

Solution

Target 10011000 XOR 00110101



10101101





Inherent rule of the XOR operator





Applications







Example of unsetting specific bits







Use a mask to unset (clear) the 5 leftmost bits of a pattern. Test the mask with the pattern 10100110.

Solution

The mask is 00000111.

 Target
 10100110
 AND

 Mask
 00000111

Result

00000110





Imagine a power plant (水力發電廠) that pumps water (供水) to a city using eight pumps (抽水機). The state of the pumps (on or off) can be represented by an 8-bit pattern. For example, the pattern 11000111 shows that pumps 1 to 3 (from the right), 7 and 8 are on while pumps 4, 5, and 6 are off. Now assume pump 7 shuts down. How can a mask show this situation?

Solution on the next slide.





Solution

Use the mask 10111111 to AND with the target pattern. The only 0 bit (bit 7) in the mask turns off the seventh bit in the target.

 Target
 11000111
 AND

 Mask
 1011111

Result

10000111





Example of setting specific bits







Use a mask to set the 5 leftmost bits of a pattern. Test the mask with the pattern 10100110.

1111110

Solution

The mask is **11111**000.

 Target
 10100110
 OR

 Mask
 1111000



Brooks/Cole Thomson Learning_™



Using the power plant example, how can you use a mask to to show that pump 6 is now turned on?

Solution

Use the mask 0010000.

Target 10000111 OR Mask 001000

Result

10100111





Example of flipping (跳動的) specific bits







Use a mask to flip the 5 leftmost bits of a pattern. Test the mask with the pattern 10100110.

Solution

Target Mask

10100110 XOR 1111000

01011110

Result









Shift operations



Left shift

Right shift







Show how you can divide or multiply a number by 2 using shift operations.

Solution

If a bit pattern represents an **unsigned number**, a **right-shift** operation **divides** the number by two. The pattern 00111011 represents **59**. When you shift the number to the right, you get 00011101, which is **29**. If you shift the original number to the **left**, you get 01110110, which is **118**.



Use a combination of logical and shift operations to find the value (0 or 1) of the fourth bit (from the right).

Solution

Use the mask 00001000 to AND with the target to keep the fourth bit and clear the rest of the bits.

Continued on the next slide



Solution (continued)



Targeta b c d e f g hANDMask00001000

Result 0000e 000

Shift the new pattern three times to the right

.

0000e000 🗆 00000e00 🗆 000000e0 🗆 000000e

Now it is easy to test the value of the new pattern as an unsigned integer. If the value is 1, the original bit was 1; otherwise the original bit was 0.



Key terms

- AND operator
- Arithmetic operation
- Binary operation
- Binary operator
- Carry
- Clear
- Flip
- Floating-point number
- Force (強迫) to 0
- Force (強迫) to 1
- Logical operation

- Mantissa
- Mask
- NOT operator
- OR operator
- Overflow
- Set
- Truth table
- Two's complement
- Unary operation
- Unary operator
- Unset
- XOR operator



