Chapter 05

Computer Arithmetic

Computer Fundamentals - Pradeep K. Sinha & Priti Sinha
In this chapter you will learn about:

- Reasons for using binary instead of decimal numbers
- Basic arithmetic operations using binary numbers
  - Addition (+)
  - Subtraction (-)
  - Multiplication (*)
  - Division (/)
Information is handled in a computer by electronic/electrical components.

Electronic components operate in binary mode (can only indicate two states – on (1) or off (0)).

Binary number system has only two digits (0 and 1), and is suitable for expressing two possible states.

In binary system, computer circuits only have to handle two binary digits rather than ten decimal digits causing:

- Simpler internal circuit design
- Less expensive
- More reliable circuits

Arithmetic rules/processes possible with binary numbers.
Examples of a Few Devices that work in Binary Mode

<table>
<thead>
<tr>
<th>Binary State</th>
<th>On (1)</th>
<th>Off (0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulb</td>
<td><img src="image" alt="Light Bulb On" /></td>
<td><img src="image" alt="Light Bulb Off" /></td>
</tr>
<tr>
<td>Switch</td>
<td><img src="image" alt="Switch On" /></td>
<td><img src="image" alt="Switch Off" /></td>
</tr>
<tr>
<td>Circuit Pulse</td>
<td><img src="image" alt="Circuit Pulse On" /></td>
<td><img src="image" alt="Circuit Pulse Off" /></td>
</tr>
</tbody>
</table>
Binary Arithmetic

- Binary arithmetic is simple to learn as binary number system has only two digits – 0 and 1.

- Following slides show rules and example for the four basic arithmetic operations using binary numbers.
Rule for binary addition is as follows:

\[
\begin{align*}
0 + 0 &= 0 \\
0 + 1 &= 1 \\
1 + 0 &= 1 \\
1 + 1 &= 0 \text{ plus a carry of 1 to next higher column}
\end{align*}
\]
Example

Add binary numbers 10011 and 1001 in both decimal and binary form

Solution

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>carry 11</td>
<td>carry 1</td>
</tr>
<tr>
<td>10011</td>
<td>19</td>
</tr>
<tr>
<td>+1001</td>
<td>+9</td>
</tr>
<tr>
<td>________</td>
<td>________</td>
</tr>
<tr>
<td>11100</td>
<td>28</td>
</tr>
<tr>
<td>________</td>
<td>________</td>
</tr>
</tbody>
</table>

In this example, carry are generated for first and second columns
### Example

Add binary numbers 100111 and 11011 in both decimal and binary form

### Solution

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>carry 11111</td>
<td>carry 1</td>
</tr>
<tr>
<td>100111</td>
<td>39</td>
</tr>
<tr>
<td>+11011</td>
<td>+27</td>
</tr>
<tr>
<td>_____________</td>
<td>_________</td>
</tr>
<tr>
<td>1000010</td>
<td>66</td>
</tr>
</tbody>
</table>

The addition of three 1s can be broken up into two steps. First, we add only two 1s giving 10 (1 + 1 = 10). The third 1 is now added to this result to obtain 11 (a 1 sum with a 1 carry). Hence, 1 + 1 + 1 = 1, plus a carry of 1 to next higher column.
Rule for binary subtraction is as follows:

0 - 0 = 0
0 - 1 = 1 with a borrow from the next column
1 - 0 = 1
1 - 1 = 0
Example

Subtract $01110_2$ from $10101_2$

Solution

\[
\begin{array}{c}
12 \\
0202 \\
10101 \\
-01110 \\
\hline \\
00111
\end{array}
\]

Note: Go through explanation given in the book
Complement of a Number

\[ C = B^n - 1 - N \]

- \( C \) = Complement of the number
- \( B^n \) = Base of the number
- \( 1 \) = The number
- \( N \) = Number of digits in the number
Example

Find the complement of $37_{10}$

Solution

Since the number has 2 digits and the value of base is 10,

$$(\text{Base})^{n} - 1 = 10^2 - 1 = 99$$  
Now $99 - 37 = 62$

Hence, complement of $37_{10} = 62_{10}$
Example

Find the complement of $6_8$

Solution

Since the number has 1 digit and the value of base is 8,

$$(\text{Base})^n - 1 = 8^1 - 1 = 7_{10} = 7_8$$

Now $7_8 - 6_8 = 1_8$

Hence, complement of $6_8 = 1_8$
Complement of a Binary Number

Complement of a binary number can be obtained by transforming all its 0’s to 1’s and all its 1’s to 0’s

Example

Complement of 1 0 1 1 0 1 0 is

↓ ↓ ↓ ↓ ↓ ↓ ↓

0 1 0 0 1 0 1

Note: Verify by conventional complement
Complementary Method of Subtraction

Involves following 3 steps:

Step 1: Find the complement of the number you are subtracting (subtrahend)

Step 2: Add this to the number from which you are taking away (minuend)

Step 3: If there is a carry of 1, add it to obtain the result; if there is no carry, recomplement the sum and attach a negative sign

Complementary subtraction is an additive approach of subtraction
Example:
Subtract $56_{10}$ from $92_{10}$ using complementary method.

Solution

Step 1: Complement of $56_{10}$

$$= 10^2 - 1 - 56 = 99 - 56 = 43_{10}$$

Step 2: $92 + 43$ (complement of 56)

$$= 135 \text{ (note 1 as carry)}$$

Step 3: $35 + 1$ (add 1 carry to sum)

Result

$$= 36$$

The result may be verified using the method of normal subtraction:

$$92 - 56 = 36$$
Example

Subtract $35_{10}$ from $18_{10}$ using complementary method.

Solution

Step 1: Complement of $35_{10}$

\[
10^2 - 1 - 35 = 99 - 35 = 64_{10}
\]

Step 2: $18 + 64$ (complement of $35$)

\[
\begin{array}{c}
\phantom{+}18 \\
+ 64 \\
\hline
82
\end{array}
\]

Step 3: Since there is no carry, re-complement the sum and attach a negative sign to obtain the result.

Result $= -(99 - 82) = -17$

The result may be verified using normal subtraction:

\[18 - 35 = -17\]
Example

Subtract $0111000_2 (56_{10})$ from $1011100_2 (92_{10})$ using complementary method.

Solution

$$
\begin{align*}
1011100 \\
+1000111 \quad \text{(complement of 0111000)} \\
\hline
10100011 \\
\uparrow 1 \quad \text{(add the carry of 1)} \\
\hline
0100100
\end{align*}
$$

Result = $0100100_2 = 36_{10}$
Binary Subtraction Using Complementary Method (Example 2)

Example

Subtract $100011_2$ ($35_{10}$) from $010010_2$ ($18_{10}$) using complementary method.

Solution

\[
\begin{align*}
010010 & \\
+011100 & \text{(complement of 100011)} \\
\hline
101110
\end{align*}
\]

Since there is no carry, we have to complement the sum and attach a negative sign to it. Hence,

\[
\text{Result} = -010001_2 \text{ (complement of 101110)} \\
= -17_{10}
\]
Table for binary multiplication is as follows:

\[
\begin{align*}
0 \times 0 &= 0 \\
0 \times 1 &= 0 \\
1 \times 0 &= 0 \\
1 \times 1 &= 1
\end{align*}
\]
**Binary Multiplication** (Example 1)

**Example**

Multiply the binary numbers 1010 and 1001

**Solution**

\[
\begin{array}{c}
1010 \quad \text{Multiplicand} \\
\times 1001 \quad \text{Multiplier} \\
\hline
1010 \quad \text{Partial Product} \\
0000 \quad \text{Partial Product} \\
0000 \quad \text{Partial Product} \\
1010 \quad \text{Partial Product} \\
\hline
1011010 \quad \text{Final Product}
\end{array}
\]
Whenever a 0 appears in the multiplier, a separate partial product consisting of a string of zeros need not be generated (only a shift will do). Hence,

\[
\begin{array}{c}
1010 \\
\times 1001 \\
\hline
1010 \\
1010SS \\
\hline
1011010
\end{array}
\]  
(S = left shift)
Table for binary division is as follows:

\[
\begin{align*}
0 \div 0 &= \text{Divide by zero error} \\
0 \div 1 &= 0 \\
1 \div 0 &= \text{Divide by zero error} \\
1 \div 1 &= 1
\end{align*}
\]

As in the decimal number system (or in any other number system), division by zero is meaningless.

The computer deals with this problem by raising an error condition called ‘Divide by zero’ error.
Rules for Binary Division

1. Start from the left of the dividend
2. Perform a series of subtractions in which the divisor is subtracted from the dividend
3. If subtraction is possible, put a 1 in the quotient and subtract the divisor from the corresponding digits of dividend
4. If subtraction is not possible (divisor greater than remainder), record a 0 in the quotient
5. Bring down the next digit to add to the remainder digits. Proceed as before in a manner similar to long division
**Example**

Divide $100001_2$ by $110_2$

**Solution**

```
  0101  (Quotient)
   __________
  110)10001  (Dividend)
        1       Divisor greater than 100, so put 0 in quotient
         110
        1000  Add digit from dividend to group used above
              110
             100  Subtraction possible, so put 1 in quotient
                110
               1001  Remainder from subtraction plus digit from dividend
                     110
                    100  Divisor greater, so put 0 in quotient
                        110
                       1001  Add digit from dividend to group
                             110
                            111  Subtraction possible, so put 1 in quotient
                               ________
                              11   Remainder
```
Most computers use the additive method for performing multiplication and division operations because it simplifies the internal circuit design of computer systems.

Example

\[4 \times 8 = 8 + 8 + 8 + 8 = 32\]
Rules for Additive Method of Division

1. Subtract the divisor repeatedly from the dividend until the result of subtraction becomes less than or equal to zero.

2. If result of subtraction is zero, then:
   - quotient = total number of times subtraction was performed
   - remainder = 0

3. If result of subtraction is less than zero, then:
   - quotient = total number of times subtraction was performed minus 1
   - remainder = result of the subtraction previous to the last subtraction
Example

Divide $33_{10}$ by $6_{10}$ using the method of addition

Solution:

$33 - 6 = 27$

$27 - 6 = 21$

$21 - 6 = 15$

$15 - 6 = 9$

$9 - 6 = 3$

$3 - 6 = -3$

Total subtractions = 6

Since the result of the last subtraction is less than zero, Quotient = $6 - 1$ (ignore last subtraction) = 5

Remainder = 3 (result of previous subtraction)
Key Words/Phrases

- Additive method of division
- Additive method of multiplication
- Additive method of subtraction
- Binary addition
- Binary arithmetic
- Binary division
- Binary multiplication
- Binary subtraction
- Complement
- Complementary subtraction
- Computer arithmetic