Computer Arithmetic

Dr. Sheak Rashed Haider Noori

Learning Objectives

- **In this lecture you will learn about:**

✓Reasons for using binary instead of decimal numbers

✓Basic arithmetic operations using binary numbers

✓Addition (+)

✓Subtraction (-)

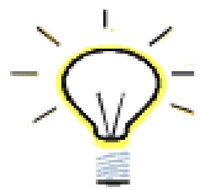
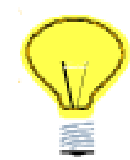
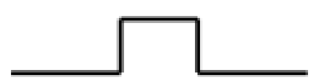
✓Multiplication (*)

✓Division (/)

Binary over Decimal

- 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

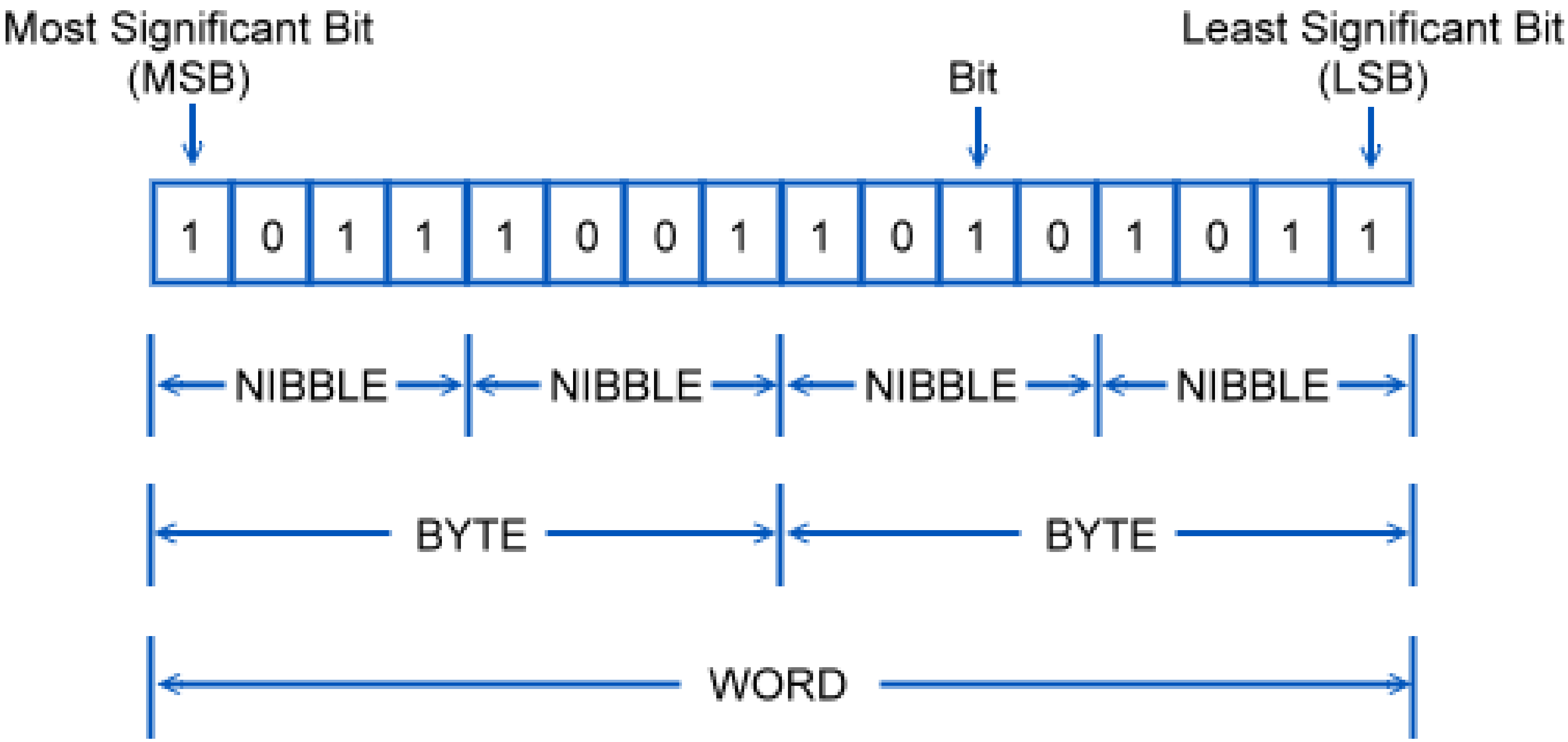
Examples of a Few Devices that work in Binary Mode Binary

Binary State	On (1)	Off (0)
Bulb		
Switch		
Circuit Pulse		

Binary Number System

- System Digits: 0 and 1
- Bit (short for binary digit): A single binary digit
- LSB (least significant bit): The rightmost bit
- MSB (most significant bit): The leftmost bit
- Upper Byte (or nybble): The right-hand byte (or nybble) of a pair
- Lower Byte (or nybble): The left-hand byte (or nybble) of a pair
- The term **nibble** used for 4 bits being a subset of byte.

Binary Number System



Binary Equivalents

- 1 Nybble (or nibble) = 4 bits
- 1 Byte = 2 nybbles = 8 bits
- 1 Kilobyte (KB) = 1024 bytes
- 1 Megabyte (MB) = 1024 kilobytes = 1,048,576 bytes
- 1 Gigabyte (GB) = 1024 megabytes = 1,073,741,824 bytes

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

Binary Addition

- Rule for binary addition is as follows:
  1. $0 + 0 = 0$
  2. $0 + 1 = 1$
  3. $1 + 0 = 1$
  4. $1 + 1 = 0$ plus a carry of 1 to next higher column

Binary Addition

- **Example 1:** $00011010_2 + 00001100_2 = 00100110_2$

			1	1						*carries*
	0	0	0	1	1	0	1	0	=	$26_{(base\ 10)}$
+	0	0	0	0	1	1	0	0	=	$12_{(base\ 10)}$
	0	0	1	0	0	1	1	0	=	$38_{(base\ 10)}$

Binary Addition

- Example 2: $00010011_2 + 00111110_2 = 01010001_2$

$$
\begin{array}{cccccccccl}
 & & \mathit{1} & \mathit{1} & \mathit{1} & \mathit{1} & \mathit{1} & & & \textit{carries} \\
 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & = 19_{(\text{base } 10)} \\
+ & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & = 62_{(\text{base } 10)} \\
\hline
 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & = 81_{(\text{base } 10)}
\end{array}
$$

Binary Addition (Example 3)

Example

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

Solution

	Binary	Decimal
carry	11111	1
	100111	39
	+11011	+27
	1000010	66

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.

Binary Subtraction

- Rule for binary subtraction is as follows:
  1. 0 - 0 = 0
  2. 0 - 1 = 1 with a borrow from the next column
  3. 1 - 0 = 1
  4. 1 - 1 = 0

Binary Subtraction

- Example 1: $00100101_2 - 00010001_2 = 00010100_2$

			0							*borrows*
	0	0	~~1~~	*1*0	0	1	0	1	=	$37_{(\text{base } 10)}$
-	0	0	0	1	0	0	0	1	=	$17_{(\text{base } 10)}$
	0	0	0	1	0	1	0	0	=	$20_{(\text{base } 10)}$

Binary Subtraction

- **Example 2:** $00110011_2 - 00010110_2 = 00011101_2$

			0	*^{1}0*	*1*					*borrows*
	0	0	~~1~~	~~1~~	~~0~~	^{1}0	1	1	=	$51_{(\text{base } 10)}$
-	0	0	0	1	0	1	1	0	=	$22_{(\text{base } 10)}$
	0	0	0	1	1	1	0	1	=	$29_{(\text{base } 10)}$

Binary Multiplication

- Table for binary multiplication is as follows:
  1. 0 x 0 = 0
  2. 0 x 1 = 0
  3. 1 x 0 = 0
  4. 1 x 1 = 1

Binary Multiplication

- **Example 1: $00101001_2 \times 00000110_2 = 11110110_2$**

		0	0	1	0	1	0	0	1	=	$41_{(\text{base } 10)}$
	×	0	0	0	0	0	1	1	0	=	$6_{(\text{base } 10)}$
		0	0	0	0	0	0	0	0		
	0	0	1	0	1	0	0	1			
0	0	1	0	1	0	0	1				
0	0	1	1	1	1	0	1	1	0	=	$246_{(\text{base } 10)}$

Binary Multiplication

- **Example 2: $00010111_2 \times 00000011_2 = 01000101_2$**

	0	0	0	1	0	1	1	1	=	$23_{\text{(base 10)}}$
×	0	0	0	0	0	0	1	1	=	$3_{\text{(base 10)}}$
		1	*1*	*1*	*1*	*1*				*carries*
	0	0	0	1	0	1	1	1		
0	0	0	1	0	1	1	1			
0	0	1	0	0	0	1	0	1	=	$69_{\text{(base 10)}}$

Binary Multiplication

- **Example 3:** Multiply the binary numbers 1010 and 1001

Solution

1010	Multiplicand
x1001	Multiplier
1010	Partial Product
0000	Partial Product
0000	Partial Product
1010	Partial Product
1011010	Final Product

Binary Multiplication

- **Example 4:**

  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,

```
   1010
  x1001
 ------
   1010
 1010SS   (S = left shift)
 -------
 1011010
```

Binary Division

- Table for binary division is as follows:
  1. 0 ÷ 0 = Divide by zero error
  2. 0 ÷ 1 = 0
  3. 1 ÷ 0 = Divide by zero error
  4. 1 ÷ 1 = 1
- 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

Binary Division (Example 1)

Example

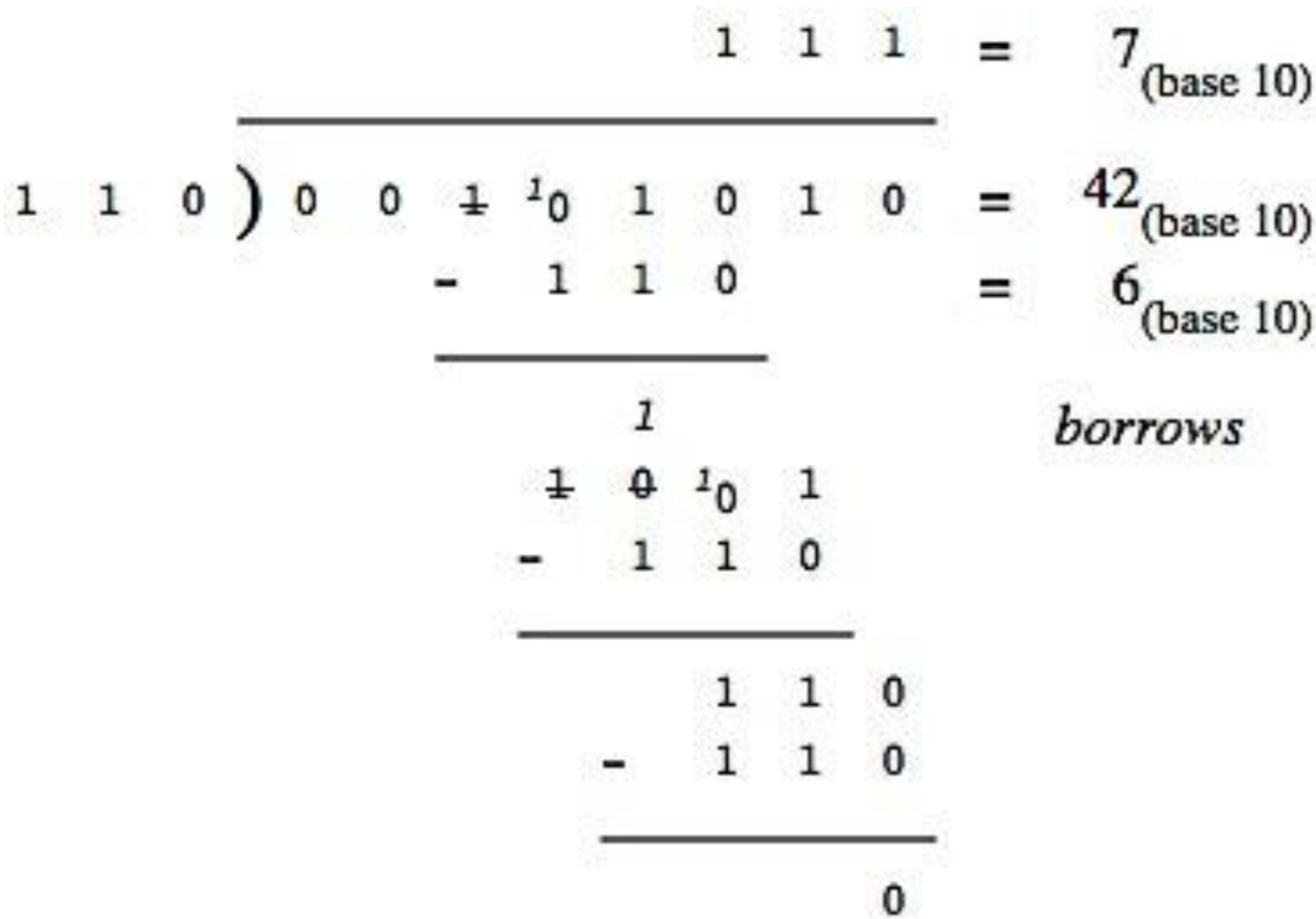
Divide 100001_2 by 110_2

Solution

```
          0101    (Quotient)
     ____________
110 ) 100001  (Dividend)
      110       1 <----   Divisor greater than 100, so put 0 in quotient
      ----
      1000      2 <----   Add digit from dividend to group used above
       110      3 <----   Subtraction possible, so put 1 in quotient
      ----
        100     4 <----   Remainder from subtraction plus digit from dividend
        110     5 <----   Divisor greater, so put 0 in quotient
      ----
        1001    6 <----   Add digit from dividend to group
         110    7 <----   Subtraction possible, so put 1 in quotient
      ----
          11      Remainder
```

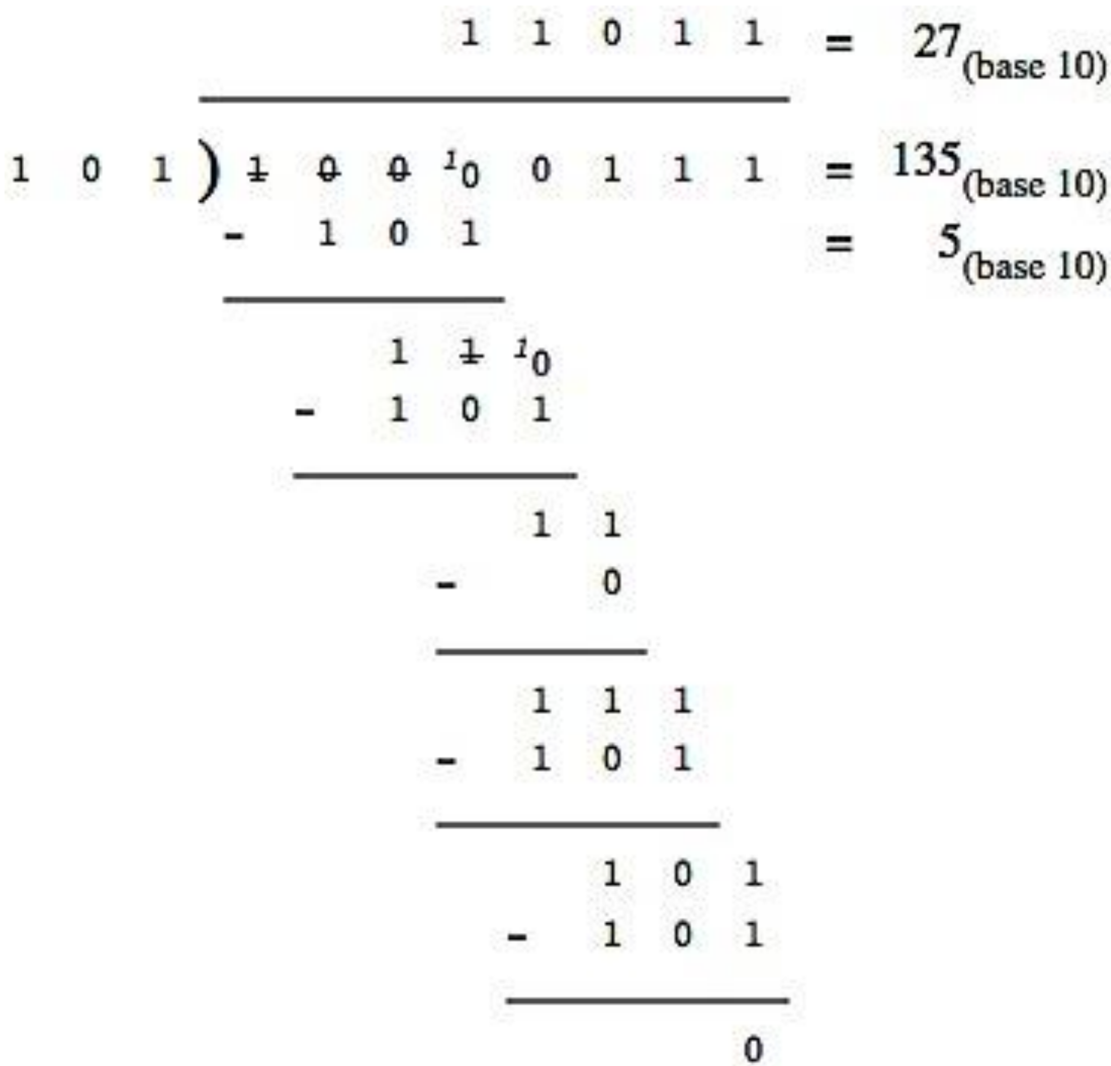
Binary Division (Example 2)

- **Example:** $00101010_2 \div 00000110_2 = 00000111_2$



Binary Division (Example 3)

- Example: $10000111_2 \div 00000101_2 = 00011011_2$



Complement of a Number

$$C = B^n - 1 - N$$

Number of digits in the number

Complement of the number

Base of the number

The number

Complement of a Decimal Number

Example

Find the complement of 37_{10}

Solution

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

$(Base)^n - 1 = 10^2 - 1 = 99$

Now $99 - 37 = 62$

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

Complement of a Octal Number

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

Complementary Subtraction (Example 1)

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)

$= 135$ (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:

$92 - 56 = 36$

Complementary Subtraction (Example 2)

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:

$$\begin{array}{r} 18 \\ +\ 64 \\ \hline 82 \\ \hline \end{array}$$

(complement of 35)

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:

$18 - 35 = -17$

Binary Subtraction Using Complementary Method (Example 1)

Example

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

Solution

```
 1011100
+1000111   (complement of 0111000)
--------
10100011
|
└──────→  1   (add the carry of 1)
--------
 0100100
--------
```

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

$$\begin{array}{r} 010010 \\ +011100 \\ \hline 101110 \\ \hline \end{array}$$ (complement of 100011)

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

Result = -010001_2 (complement of 101110_2)
= -17_{10}

Addition/Subtraction of Numbers in 2's Complement Notation

- Represent all negative numbers in 2's complement form.
- Now we have the same procedure for addition and subtraction.
- Subtraction of a number is achieved by adding the 2's complement of the number.
- This is illustrated in the following example where the carry, if any, from the most significant bit, during addition, should be ignored.
- The result has to be interpreted appropriately using the same convention.

Addition/Subtraction of Numbers in 2's Complement Notation…

Example 6.4

Using 2's complement representation, (a) subtract 3 from 5; (b) subtract (-3) from (-5); (c) add (- 5) and (- 2); and (d) add 5 and 4.

SOLUTION;

(a) 5 — 3

5	0101
— 3	1101
	10010

ignore carry

Answer = + 2

(b) - 5 + 3

- 5	1011
+ 3	0011
	1110

Answer = - 2

(c) - 5 - 2

- 5	1011
- 2	1110
	11001

ignore carry

Answer = - 7

(d) 5 4

5	0101
4	0100
	1001

incorrect answer

In the last example we get an incorrect answer because the sum 9 exceeds the range of numbers (0 to 7) we had stipulated in the beginning.

Key Words/Phrases

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

Binary addition

Binary division

Binary subtraction

Complement