**Number Systems Used In Computers**

Complete the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Number System** | **Number Base** | **Digits Used** | **Purpose** |
| Decimal | Base 10 | 0 to 9 | Used for communicating with human users |
| Binary |  |  |  |
| Octal |  |  |  |
| Hexadecimal |  |  |  |

**Decimal vs. Binary System**

Write down the definition of a **Bit** in computer terminology:

Complete the following table:

|  |  |  |
| --- | --- | --- |
|  | **Decimal System** | **Binary System** |
| Uses Digits |  |  |
| Addition Examples |  |  |
| Powers of 10 |  |  |
| Evaluate 111 |  |  |

**Powers of 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Power  of 2** | **Decimal Value** | **Integer Range** | **Number  of Bits** | **Computer Binary** |
| 21 | 2 | 0 - 1 | 1 | 0001 |
| 22 | 4 | 0 - 3 | 2 | 0011 |
| 23 |  |  |  | 0111 |
| 24 |  |  |  | 1111 |
| 25 |  |  |  | 0001 1111 |
| 26 |  |  |  | 0011 1111 |
| 27 |  |  |  | 0111 1111 |
| 28 | 256 | 0 - 255 | 8 | 1111 1111 |
| 29 |  |  |  | 0001 1111 1111 |
| 210 |  |  |  | 0011 1111 1111 |
| 211 |  |  |  | 0111 1111 1111 |
| 212 |  |  |  | 1111 1111 1111 |
| 213 |  |  |  | 0001 1111 1111 1111 |
| 214 |  |  |  | 0011 1111 1111 1111 |
| 215 |  |  |  | 0111 1111 1111 1111 |
| 216 | 65,536 | 0 – 65,535 | 16 | 1111 1111 1111 1111 |

**Conversions**

Convert the following binary numbers to decimal:

11 binary =

101 binary =

1010 binary =

Convert the following decimal numbers to binary:

6 decimal =

13 decimal =

**Binary Addition**

Add the following binary numbers. (verify your answers using decimal)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **0101** | **(decimal 5)** | | **+ 0010** | **(decimal 2)** | |  |  | | |  |  | | --- | --- | | **0101** | **(decimal 5)** | | **+ 1010** | **(decimal 10)** | |  |  | |
| |  |  | | --- | --- | | **0011** | **(decimal 3)** | | **+ 0010** | **(decimal 2)** | |  |  | | |  |  | | --- | --- | | **0110** | **(decimal 6)** | | **+ 0011** | **(decimal 3)** | |  |  | |

**Prefixes**

* Kilo (K) means \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Multiply by x\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Mega (M) means \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Multiply by x\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Giga (G) means \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Multiply by x\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Tera (T) means \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Multiply by x\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Examples**

1. 64 Kbps (Kilo-bits per seconds) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ bps (bits per second)
2. 256 Giga-bytes = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ bytes
3. 256 Giga-bytes = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ bits

**Computer Memory Structures**

|  |  |  |  |
| --- | --- | --- | --- |
| **Structure** | **Number of Bits** | **Largest Value** | **Used For...** |
| Bit |  |  |  |
| Byte |  |  |  |
| Word |  |  |  |
| Short Integer |  |  |  |
| Double Word |  |  |  |
| Long Integer |  |  |  |
| Memory Address |  |  |  |

**Memory Organization**

Bit (Binary Digit)

* One bit can have a value of \_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_
* Or a Boolean logic value of \_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | |

Byte (8 bits)

* One byte has a size of \_\_\_\_\_\_\_\_ bits
* A byte has an unsigned integer value range from 0 to \_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | |

Word (2 bytes or 16 bits)

* One word has a size of \_\_\_\_\_\_\_\_ bits
* One word has a size of \_\_\_\_\_\_\_\_ bytes
* A byte has an unsigned integer value range from 0 to \_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | |
|  | |  | |

Long or Double Word (4 bytes or 32 bits)

* One Long word has a size of \_\_\_\_\_\_\_\_ words
* A Long word has an unsigned integer value range from 0 to \_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | |
|  | |  | |
| |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | | |  |  |  |  | | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | |
|  | |  | |

**Questions**

1. What is the smallest memory object that can represent a character of information?
   1. Think… How many upper case letters in the alphabet (A to Z)?

26

* 1. Think… How many lower case letters in the alphabet (a to z)?

25

* 1. Think… How many number digits (0 to 9)?

10

* 1. Think… How many punctuation marks?

7

* 1. Add them all up

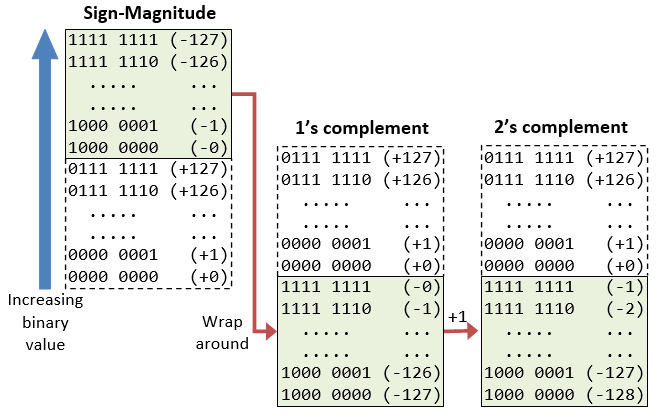
68

1. Research the ASCII characters set. What is it and how is it related to computer memory?  
   ASCII is the acronym for the American Standard Code for Information Interchange. It is a code for representing 128 English characters as numbers, with each letter assigned a number from 0 to 127
2. How are strings of characters (Google “String”) represented in computer memory?

Representing Characters in Memory. The data in a computer are not recognized with the help of alphabets. To represent the data there are many code representations available. It has 26 lower case characters, 25 uppercase characters, 10 digits, 7 punctuation marks and anywhere from 20 to 40 other characters.

1. How are negative integers represented in computer memory? (Include a diagram)  
   In computing, signed number representations are required to encode negative numbers in binary number systems. In mathematics, negative numbers in any base are represented by prefixing them with a minus ("−") sign. However, in computer hardware, numbers are represented only as sequences of bits, without extra symbols.
2. How are decimal numbers (Google “Floating Point”) represented in computer memory? (Include a diagram)

The computer memory is organized into strings of bits called words of same length. Decimal numbers are first converted into their binary equivalents and then are represented in either integer or floating point form



1. A Pixel is computer memory structure used to store image information. How is a Pixel represented in memory? (Include a diagram).

Pixels are the smallest individual element in an image, holding antiquated values that represent the brightness of a given color at any specific point. Typically, the pixels are stored in computer memory as a raster image or raster map, a two-dimensional array of small integers.

