**Introduction and Number System**

**SLP Assignment**

**By Odiscious Dozier**

**Trident University**

**Why do digital computers use binary numbers for their operation?**

Digital computers use binary numbers for their operation because, increasing the base will decrease the number of digits required to represent any given number; thus, less storage is needed and the machine runs more efficiently, proficiently.

**Convert the following binary numbers to equivalent decimal numbers: a) 11010 b) 1010001 c) 1101011 and d) 10011010. Show your work.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 16 | 8 | 4 | 2 | 1 |
| 1 | 1 | 0 | 1 | 0 |

The decimal result is 🡺16 + 8 + 2 = 26

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
|  |  |  | 1 | 0 | 1 | 0 | 0 | 0 | 1 |

The decimal result is 🡺64 + 16 + 1 = 81

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
|  |  |  | 1 | 1 | 0 | 1 | 0 | 1 | 1 |

The decimal result is 🡺64 + 32 + 8 + 2 + 1 = 107

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
|  |  | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |

The decimal result is 🡺128 + 16 + 8 + 2 = 154

**Convert the following decimal numbers to equivalent binary numbers: a) 17.71875 b) 50.7 C) 74.635 and d) 100.5. Show your work.**

A)

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Number /2 | Remainder New Fraction | Integer |
| 17.71875 | 17 / 2 = 8 |  | 1(LSB) |
| 8 | 8/2 = 4 |  | 0 |
| 4 | 4/2 = 2 |  | 0 |
| 2 | 2/2 = 1 |  | 0 |
| 1 | 1/1 = 1 |  | 1 |
| Fraction portion | Fraction \* 2 |  |  |
| .71875 | .71875 \* 2 = 1.4375 | .4375 | 1(MSB) |
| .4375 | .4375 \* 2 =0.875 | .875 | 0 |
| .875 | .875\*2 = 1.75 | .75 | 1 |
| .75 | .75\*2 = 1.5 | .5 | 1 |
| .5 | .5\*2 = 1.0 | 0 | 1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

The result is 🡺 10001.10111

B)

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Number /2 | Remainder New Fraction | Integer |
| 50.7 | 50 / 2 = 25 |  | 0(LSB) |
| 25 | 25/2 =12 |  | 1 |
| 12 | 12/2 = 6 |  | 0 |
| 6 | 6/2 = 3 |  | 0 |
| 3 | 3/2 = 1 |  | 1 |
| 1 | ½ = 0 |  | 1 |
| Fraction portion | Fraction \* 2 |  |  |
| .7 | .7 \* 2 = 1.4 | .4 | 1(MSB) |
| .4 | .4 \* 2 =0.8 | .8 | 0 |
| .8 | .8 \* 2 = 1.6 | .6 | 1 |
| .6 | .6 \* 2 = 1.2 | .2 | 1 |
| .2 | .2 \* 2 = 0.4 | .4 | 0 |
| .4 | .4 \* 2 = 0.8 | .8 | 0(LSB) |
| repeat |  |  |  |
|  |  |  |  |

The result is 🡺110010.101100

C)

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Number /2 | Remainder New Fraction | Integer |
| 74.635 | 74 / 2 = 37 |  | 0(LSB) |
| 37 | 37/2 =18 |  | 1 |
| 18 | 18/2 = 9 |  | 1 |
| 9 | 9/2 = 4 |  | 1 |
| 4 | 4/2 = 2 |  | 0 |
| 2 | 2/2 = 1 |  | 0 |
| 1 | ½ = 0 |  | 1(MSB) |
| Fraction portion | Fraction \* 2 |  |  |
| .635 | .635 \* 2 = 1.27 | .27 | 1(MSB) |
| .27 | .27 \* 2 =0.54 | .54 | 0 |
| .54 | .54 \* 2 = 1.08 | .08 | 1 |
| .08 | .08 \* 2 = 0.16 | .16 | 0 |
| .16 | .16 \* 2 = 0.32 | .32 | 0 |
| .32 | .32 \* 2 = 0.64 | .64 | 0 |
| .64 | .64 \* 2 = 1.28 | .28 | 1 |
| .28 | .28 \* 2 = 0.56 | .56 | 0 |
| .56 | .56 \* 2 = 1.12 | .12 | 1 |
| .12 | .12 \* 2 = 0.24 | .24 | 0(LSB) |
|  |  |  |  |
| Continues… |  |  |  |
|  |  |  |  |

The result is 🡺1001110.1010001010

D)

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Number /2 | Remainder New Fraction | Integer |
| 100.5 | 100 / 2 = 50 |  | 0(LSB) |
| 50 | 50/2 =25 |  | 0 |
| 25 | 25 / 2 = 12 |  | 1 |
| 12 | 12/2 = 6 |  | 0 |
| 6 | 6/2 = 3 |  | 0 |
| 3 | 3/2 = 1 |  | 1 |
| 1 | ½ = 0 |  | 1(MSB) |
| Fraction portion | Fraction \* 2 |  |  |
| .5 | .5 \* 2 = 1.0 | .0 | 1(MSB) |

The result is 🡺1100100.1

**Discuss overflow and underflow phenomena which occur in a digital computer**

With the mantissa, in addition or subtraction where a number is too small or too large to be shown by a computer is known as *mantissa* *underflow* or *mantissa* *overflow*,respectively. Likewise, with multiplying exponents, they are added together. When a situation exists where the number is too large to be displayed, this is known as *exponent* *overflow*. If the multiplication renders an exponent that is too small to be displayed, this is known as *exponent* *underflow*.

REFERENCES

Blinnikov, L. (n.d.). Why Computers Use Binary. Retrieved July 28, 2015, from <http://nookkin.com/articles/computer-science/why-computers-use-binary.ndoc>

Rom, B. (2007). Chapter 2: Introduction. Computer Fundamentals, Architecture and Organization. New Age International, New Delhi (Ebrary)