**KHOA Công nghệ thông tin.**

**MÔN HỌC: KIẾN trúc máy tính.**

**Họ tên SV: Lê Hoàng Thuận** **MSSV: 2311552855**

|  |
| --- |
| **LAB 2: BIỂU DIỄN SỐ HỌC TRÊN MÁY TÍNH**  (SV thực hiện tại lớp)  **\*\*\*** |

1. **Các đơn vị đo lường trong máy tính**

* **Yêu cầu bài tập:**
  1. Tính giá trị theo đơn vị đo dung lượng của những tập tin sau (*chỉ đổi từ đơn vị lớn => nhỏ*):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Dung lương tập tin** | **Đơn vị GB** | **Đơn vị**  **MB** | **Đơn vị**  **KB** | **Đơn vị**  **Byte** | **Đơn vị**  **bit** |
| **189 KB** | 0.00018432 | 0.018432 | 189 | 193,536 | 1,548,288 |
| **6.7 MB** | 0.0066 | 6.7 | 67 | 65,536 | 10,485,760 |
| **500 MB** | 0.49152 | 500 | 500 | 5,242 | 805,306,368 |
| **1.2 GB** | 1.2 | 1,200 | 1,200 | 7,372,800 | 983,040,000 |

* 1. Từ thông số về tần số (*Frequency*) hoạt động của các đường truyền (hoặc thiết bị) dưới đây, hãy diễn giải lại số lần *truyển* (hoặc *xử lý*) dữ liệu trong thời gian 1 giây (*second*):

|  |  |
| --- | --- |
| **Thông số thiết bị** | **Số lần truyền / xử lý trong thời gian 1 second** |
| RAM bus 400 MHz | 400,000,000 lần truyền / 1 sec |
| RAM bus 1333 MHz | 1,333,000,000 lần truyền / 1 sec |
| CPU có FSB 800 MHz | 800,000,000 xung nhịp/ 1 sec |
| CPU 2.2 GHz | 2,200,000,000 xung nhịp/ 1 sec |

* 1. Tính băng thông (*Bandwith* - *dung lượng dữ liệu truyền trong thời gian 1 second*) của các thiết bị sau:

| **Thông số thiết bị** | **Độ rộng đường truyền** | **Dung lượng dữ liệu truyền trong thời gian 1 second** |
| --- | --- | --- |
| RAM bus 400 MHz | 64 bit | 3,200 MB/s |
| RAM bus 1333 MHz | 64 bit | **10,664 MB/s**. |
| CPU có FSB 800 MHz | 64 bit | **5,120 MB/s**. |
| CPU có FSB 1333 MHz | 64 bit | **10,664 MB/s**. |
| USB 2.0 (480MHz) | Serial | **480 MB/s**. |
| HDD SATA-2 (3GHz) | Serial | **300 MB/s** |
| HDD SATA-3 (6GHz) | Serial | **750 MB/s**. |
| Card mạng Fast Ethernet | Serial | **100 MB/s**. |
| Card mạng Gigabit Ethernet | Serial | **125 MB/s**. |
| Cáp quang 15 “mê” | Serial | **12,5 GB/s**. |
| CPU có QPI **4.8 GT/s**  (GT/s = *Giga transfer / sec*) | Lane = 1 bit | 480MB/s |
| Lane = 8 bits | 4.8GB/s |
| Lane = 16 bits | 8,192 MB/s |
| CPU có DMI **2.5 GT/s**  (GT/s = *Giga transfer / sec*) | Lane = 1 bit | 312.5 MB/s |
| Lane = 8 bits | 2,500 MB/s |
| Lane = 16 bits | 4,096 MB/s |

* 1. Giải thích các thông số thiết bị sau:

| **Thông số thiết bị** | **Thuật ngữ của  thông số** | **Giải thích** |
| --- | --- | --- |
| Máy in 600 **DPI** | *Dots per inch* | 600 điểm mực trong khoảng cách 1 inch |
| Máy in 1200 **DPI** | Dots per inch | 1200 điểm mực trong khoảng cách 1 inch |
| HDD 7200 **RPM** | Rotations per minute | 7200 vòng quay mỗi phút |
| Máy ảnh 5 **MP** | Megapixels | 5 triệu điểm ảnh |
| Ảnh màu 24 **bits** | Bits per pixel | 24 bit dữ liệu cho mỗi điểm ảnh, cho phép hiển thị 16,7 triệu màu |
| Màn hình 1080x1920 | Pixel | 1080 điểm ảnh theo chiều ngang và 1920 điểm ảnh theo chiều dọc |
| Audio 192 **Kbps** | Kilobits per second | 192.000 bit dữ liệu âm thanh mỗi giây |
| Video **HD** 720p | Pixel | 720 điểm ảnh theo chiều ngang và 480 điểm ảnh theo chiều dọc |
| Video **Full HD** 1080p | Pixel | 1080 điểm ảnh theo chiều ngang và 720 điểm ảnh theo chiều dọc |
| Video **Ultra HD** 4K | Pixel | 3840 điểm ảnh theo chiều ngang và 2160 điểm ảnh theo chiều dọc |

1. **Chuyển đổi các hệ đếm**

* **Yêu cầu bài tập:**
  1. Thực hiện chuyển đối giá trị hệ *Binary* => *Hexa* và *Decimal*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Số hệ  Binary** | **Số hệ Hexa** | **Số hệ Decimal** |  | **Số hệ  Binary** | **Số hệ Hexa** | **Số hệ Decimal** |
| **1100 1010** | 1A | 162 |  | **0011 1011** | 3B | 59 |
| **0100 0001** | 41 | 65 |  | **0011 1100** | 3C | 60 |
| **1010 1100** | A4 | 164 |  | **1111 1111** | FF | 255 |

* 1. Thực hiện chuyển đối giá trị hệ *Decimal* => *Binary* và *Hexa*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Số hệ  Decimal** | **Số hệ Binary** | **Số hệ  Hexa** |  | **Số hệ  Decimal** | **Số hệ Binary** | **Số hệ  Hexa** |
| **65** | 0100 0001 | 41 |  | **128** | 1000 0000 | 80 |
| **97** | 0110 0001 | 61 |  | **192** | **1100 0000** | C0 |
| **150** | 1001 0100 | 96 |  | **224** | 1110 0000 | E0 |
| **168** | 1010 1000 | A8 |  | **240** | 1100 0000 | C0 |

* 1. Chuyển đổi địa chỉ IP: **192.168.1.129** thành dãy 32 số Nhị phân (4 *octet*):

|  |  |  |  |
| --- | --- | --- | --- |
| **192** | **168** | **1** | **129** |
| **1100 0000 0000 0000** | **1010 1000 0000 0000** | **0000 0000 0000 0001** | **0111 1111 0000 0001** |

1. **Biểu diễn và tính toán số âm dưới dạng bù 2:**

* **Yêu cầu bài tập:**
  1. Biểu diễn các số âm dưới dây dưới dạng số bù 2:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Số hệ  Decimal** | **Biểu diễn  Binary** | **Biểu diễn  bù 2** |  | **Số hệ  Decimal** | **Biểu diễn  Binary** | **Biểu diễn  bù 2** |
| **-9** | **0000 1001** | **1111 0111** |  | **-56** | **0011 1000** | **1100 1000** |
| **-20** | **0001 0100** | **1110 1100** |  | **-129** | **1000 0001** | **0111 1111** |
| **-16** | **0001 0000** | **1111 0000** |  | **-200** | **1100 1000** | **0011 1000** |

* 1. Thục hiện các phép toán có số âm (số âm được biểu diễn dạng bù 2):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Phép cộng** | **Số Binary hoặc bù 2** |  | **Phép cộng** | **Số Binary hoặc bù 2** |  | **Phép cộng** | **Số Binary hoặc bù 2** |
| **16** | 0001 0000 |  | **-5** | **1111 1011** |  | **18** | 1001 0010. |
| **-20** | 1110 1100 |  | **7** | **0000 0011** |  | **-30** | 1111 1010. |
| **KQ: -4** | 1111 1100 |  | **KQ: 2** | **0000 0010** |  | **KQ: -12** | 1000 0011. |

* **Hướng dẫn:**
* Biểu diễn các số dưới dạng bit:
  + Biểu diễn **nhị phân** số **+16**: **0001 0000**
  + Biểu diễn **nhị phân** số **-20**: **1001 0100**
  + Biểu diễn **bù 2** của số **-20**: **1110 1100**
* Khi đưa số cho CPU tính toán, số **âm** **-20** được đưa vào dưới **dạng bù 2**, số **dương** **+16** giữ nguyên mẫu.
  + Biểu diễn ***nhị phân*** số **16**: **0001 0000**
  + Biểu diễn ***bù 2*** của số **-20**: **1110 1100**
  + Kết quả tính **16 - 20**: **1111 1100**
* Xử lý kết quả sau tính toán:
  + Nếu kết quả tính toán trả về ***số âm*** (bit đầu là **1**): phải **tính bù 2** cho *kết quả* này để có giá trị **số âm biểu diễn dạng *Binary***:
    - Minh họa: số bù 2: **1111 1100** => viết dạng *Binary*: **1000 0100** => giá trị là **-4**)
  + Nếu kết quả tính toán trả về ***số dương*** (bit đầu là **0**) thì *kết quả* đó cũng là số biểu diễn dạng *Binary*.
  + Với hệ xử lý **8 bit**, nếu kết quả tính nhiều hơn 8 bit thì **loại bỏ** bit cao nhất (bit thứ 9).

1. **Các phép tính logic trên số nhị phân (bit).**

* **Yêu cầu bài tập:**
  1. Tính kết quả của các phép tính *Logic* sau:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **AND** | **1010 1010** |  | **AND** | **1010 1010** |  | **OR** | **1010 1010** |  | **OR** | **1010 1010** |
| **1111 1111** |  | **0000 0000** |  | **1111 1111** |  | **0000 0000** |
| **KQ** | 1010 1010 |  |  | 0000 0000 |  |  | 1111 1111 |  |  | 1010 1010 |

* 1. Dùng phép **XOR** để mã hóa dữ liệu.
* Giả sử: ta có 16 bit dữ liệu gốc **X** = **1110 0101 0011 1011**
* Ta mã hóa dữ liệu trên bằng khóa 4 bit **K** = **1010** dùng thuật giải *XOR*.
* Ta được dữ liệu đã mã hóa **Y** = **X** ^ **K**

|  |  |
| --- | --- |
| **X** | **1110 0101 0011 1011** |
| **K** | **1010 1010 1010 1010** |
| **Y** = **X** ^ **K** | **0100 1111 1001 0001** |

*Yêu cầu*: hãy dùng thuật giải ***XOR*** để **giải mã** dữ liệu mã hóa **Y** trong 2 trường hợp:

* Giải mã dùng khóa **K** đúng (giống với khóa đã dùng mã hóa dữ liệu **X**).
* Giải mã dùng khóa **K** sai (không giống với khóa đã dùng mã hóa dữ liệu **X**).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Trường hợp khóa K sai** | |  | **Trường hợp khóa K đúng** | |
| **Y** | **0100 1111 1001 0001** |  | **Y** | **0100 1111 1001 0001** |
| **KT** | **1100 1100 1100 1100** |  | **KF** | **1010 1010 1010 1010** |
| **X** = **Y** ⊕ **K** | **1110 0101 0011 1011** |  | **X** = **Y** ⊕ **K** | **0000 0000 0000 0000** |