**Các phép toán thao tác trên hệ nhị phân**

Nếu trong hệ thập phân ta có các phép toán như cộng, trừ, nhân, chia,… thì trong hệ nhị phân chúng ta có các phép toán AND, OR, XOR, NOT, dịch trái (bits shift left) và dịch phải (bits shift right).

**1.AND (&)**

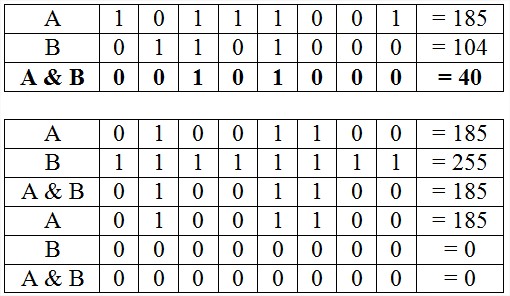
Giả sử ta có 2 bit 0 và 1 thì:

|  |  |
| --- | --- |
| 0 AND 0=0 | 0 |
| 1 AND 1 =1 | 1 |
| 0 AND 1=0 | 0 |
| 1 AND 0=0 | 0 |

Như vậy chỉ khi nào 2 bit đều là 1 thì kết quả trả về mới là 1, các trường hợp còn lại đều là 0.

Phát biểu bằng lời: nếu cả 2 điều kiện cùng đúng thì kết quả là đúng, và dĩ nhiên những trường hợp còn lại là sai.

Ví dụ:

[](http://k2.arduino.vn/img/2014/06/07/0/523_8821-1402154365-0-bitmath.doc---microsoft-word-10.jpg)

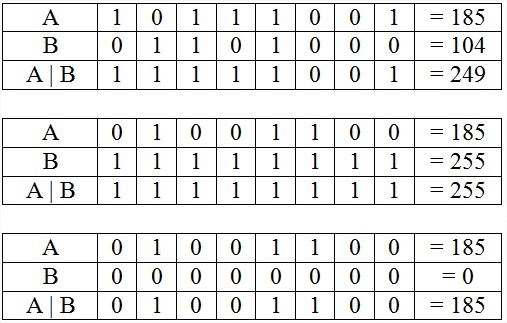
**2. OR ( | )**

Giả sử ta có 2 bit 0 và 1 thì:

|  |  |
| --- | --- |
| 0 OR 0 | 0 |
| 1 OR 1 | 1 |
| 0 OR 1 | 1 |
| 1 OR 0 | 1 |

Như vậy chỉ cần 1 trong 2 bit là 1 thì kết quả trả về sẽ là 1.

Phát biểu bằng lời: nếu có một trong 2 điều kiện là đúng thì kết quả là đúng

[](http://k3.arduino.vn/img/2014/06/07/0/521_8121-1402154417-0-bitmath.doc---microsoft-word-11.jpg)

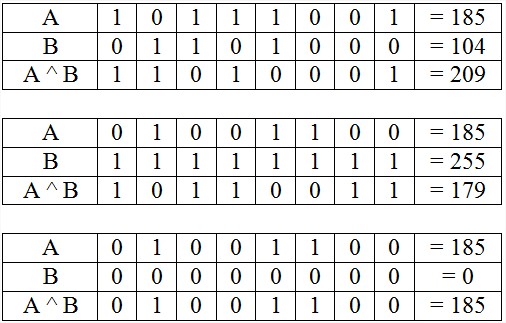
**3.XOR (^)**

Giả sử ta có 2 bit 0 và 1 thì:

|  |  |
| --- | --- |
| 0 XOR 0 | 0 |
| 1 XOR 1 | 0 |
| 0 XOR 1 | 1 |
| 1 XOR 0 | 1 |

Như vậy nếu 2 bit khác nhau sẽ cho ra kết quả 1 và ngược lại, 2 bit giống nhau sẽ cho ra kết quả 0. Từ đó ta thấy nếu A xor B = 0 thì A = B

Phát biểu bằng lời: nếu 2 điều kiện mang giá trị đúng – sai khác nhau thì kết quả trả về là đúng.

[](http://k2.arduino.vn/img/2014/06/07/0/524_8821-1402154515-0-bitmath.doc---microsoft-word-12.jpg)

**4. NOT (~)**

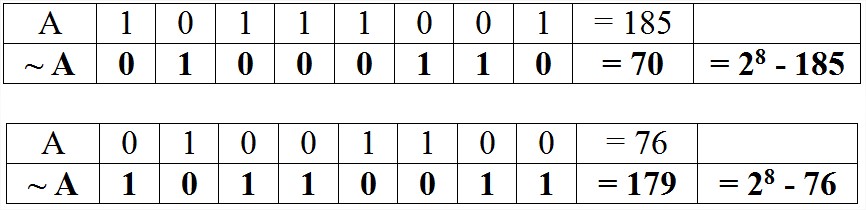
Phép toán not thay đổi bit 0 thành bit 1 và ngược lại, bit 1 thành bit 0.

Tức là:

|  |  |
| --- | --- |
| NOT 0 | 1 |
| NOT 1 | 0 |

Phép toán này còn được gọi là phép đảo bit.

Ví dụ:

[](http://k2.arduino.vn/img/2014/06/04/0/507_8821-1401856421-0-bitmath.doc---microsoft-word-8.jpg)

\*Bài tập ví dụ nháy 8 led với các phép thao tác bit:

#include"main.h"

#include"delay.h"

#define LED P2 //define port P2=”LED”

void main**()**

**{** **while(**1**)**

**{**

LED**=** 0xFF**^**0xAA**;** // LED= 0x55 or LED= 01010101 (use function XOR "^")

Delay**(**500**);**

LED**=~**0x55**;** //LED=0xAA or LED= 10101010 (use function NOT "~")

Delay**(**500**);**

LED**=**0xFF**&**0x49**;** //LED=0x49 or LED= 1001001 (use function AND "&")

Delay**(**500**);**

LED**=**0x00**|**0x66**;** //LED=0x66 or LED=0110110 (use function OR "|")

Delay**(**500**);**

**}**

**}**

Lưu ý:0xFF=1111 1111;0xAA=1010 1010;