**Bitwise operators.**

( &, |, ^, ~, <<, >> )

Bitwise operators modify variables considering the bit patterns that represent the values they store.

|  |  |  |
| --- | --- | --- |
| **operator** | **asm equivalent** | **description** |
| & | AND | Bitwise AND |
| | | OR | Bitwise inclusive OR |
| ^ | XOR | Bitwise exclusive OR |
| ~ | NOT | Unary complement (bit inversion) |
| << | SHL | Shift bits left |
| >> | SHR | Shift bits right |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11  12  13  14  15  16  17  18  19  20  21  22 | #include <iostream>  using namespace std;  int main()  {  // a = 5(00000101), b = 9(00001001)  int a = 5, b = 9;  cout << "a = " << a << "," << " b = " << b << endl;  // The result is 1(00000001)  cout << "a & b = " << (a & b) << endl;  // The result is 13(00001101)  cout << "a | b = " << (a | b) << endl;  // The result is 12(00001100)  cout << "a ^ b = " << (a ^ b) << endl;  // The result is 250(11111010) => 250 - 256 = -6  cout << "~(" << a << ") = " << (~a) << endl;  // The result is 36(00100100)  cout << "b << 2" << " = " << (b << 2) << endl;  // The result is 1(00000001)  cout << "b >> 3 " << "= " << (b >> 3) << endl;  system("pause");  return 0;  } | a = 5, b = 9  a & b = 1  a | b = 13  a ^ b = 12  ~(5) = -6  b << 2 = 36  b >> 3 = 1 | [Edit & Run](https://cplusplus.com/doc/tutorial/operators/) |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11  12  13  14  15  16  17  18  19  20  21  22 | #include <iostream>  using namespace std;  int main()  {  // a = 1(00000001), b = 8(00001000)  int a = 1, b = 8;  cout << "a = " << a << "," << " b = " << b << endl;  // The result is 0(00000000)  cout << "a & b = " << (a & b) << endl;  // The result is 9(00001001)  cout << "a | b = " << (a | b) << endl;  // The result is 9(00001001)  cout << "a ^ b = " << (a ^ b) << endl;  // The result is 254(11111110) => 254 - 256 = -2  cout << "~(" << a << ") = " << (~a) << endl;  // The result is 16(00010000)  cout << "b << 1" << " = " << (b << 1) << endl;  // The result is 4(00000100)  cout << "b >> 1" << " = " << (b >> 1) << endl;  system("pause");  return 0;  } | a = 1, b = 8  a & b = 0  a | b = 9  a ^ b = 9  ~(1) = -2  b << 1 = 16  b >> 1 = 4 | [Edit & Run](https://cplusplus.com/doc/tutorial/operators/) |