Bitweise Operationen in C

* NICHT  
  ~ausdruck  
  a = ~b  
  c = ~0x55; //0xAA
* UND  
  ausdruck1 & ausdruck2  
  a = b & c  
  a = a & 0xFE; // löscht das niederwertigste Bit in a
* ODER  
  ausdruck1 | ausdruck2  
  a = b | c;  
  a = a | 0x01; // setzt das niederwertigste Bit in a
* EXOR  
  ausdruck1 ^ ausdruck2  
  a = b ^ c;  
  a = a ^ 0x01; //toggelt das niederwertigste Bit in a
* Links-Schiebeoperator  
  ausdruck1 << ausdruck2  
  schiebt die Bits in ausdruck1 um   
  ausdruck2 Stellen  
  a = a << 2; //Multiplikation mit 4  
  a = 1 << 5; //Das Bit Nr. 5 ist 1
* Rechts-Schiebeoperator  
  schiebt die Bits in ausdruck1 um ausdruck2 Stellen nach rechts  
  a = a >> 2; //Division durch 4  
  a = 1 >> 2; //nicht sinnvoll  
  wenn ausdruck1 ein vorzeichenloser Datentyp ist wird mit 0-Bits aufgefüllt  
  wenn ausdruck1 ein vorzeichenbehafteter Datentyp ist wird mit 1-Bits aufgefüllt wenn das höchstwertige Bit 1 war (sonst mit 0-Bit)