

- **Undefiniert für  $n < 0$  (Shiften um eine negative Anzahl Bits)**

```
int shifted = 1 << -1; /* Undefiniert, nicht notwendigerweise Rechts-Shift */
```

- **Undefiniert für  $n \geq$  Bitbreite des Typs von AusdruckL (nach Konvertierung)**

```
int shifted = 1 << (sizeof(1 << 0) * CHAR_BIT); /* Undefiniert */
```

- **Beispiele: Bit auf bestimmte Position shiften**

```
unsigned char uc = 0;
signed char   sc = 0;
int           sint = 0;

uc = 1 << 0;          /* 00000001      (1) */
uc = 1 << 5;          /* 00100000      (32) */
uc = 1 << 7;          /* 10000000     (128) */
sc = 1 << 7;          /* 10000000    (-128) */
uc = 1 << 12;         /* 00000000      (0) */
sint = 1 << 12;       /* 00000000 00000000 00010000 00000000 (4096) */

/* Zusammengesetzte Zuweisung: i <<= n ⇔ i = i << n */
/* Wiederholte Anwendung, um sukzessive eine Maske für jedes Bit zu erhalten */
uc = 1;
while( uc ) { ... uc <<= 1; }
```

Annahmen  
char 8bit  
Zweierkomplement-  
Darstellung