

20. LOW-LEVEL PROGRAMMING

Bitwise Shift Operators

- The bitwise shift operators are:

`<<` left shift

`>>` right shift

- The expression `i << j` represents `i` shifted left `j` positions, zero-filled.
- The expression `i >> j` represents `i` shifted right `j` positions. If `i` is of an unsigned type or if the value of `i` is not negative, then zero bits are added at the left as needed. If `i` is negative, the result depends on the implementation.

Bitwise Shift Operators

- For portability, it's best to perform shifts only on unsigned numbers.

- Examples:

```
unsigned short int i, j;
```

```
i = 13; /* i is now 13 (binary 00000000000000001101) */
```

```
j = i << 2; /* j is now 52 (binary 000000000000110100) */
```

```
j = i >> 2; /* j is now 3 (binary 00000000000000000011) */
```