

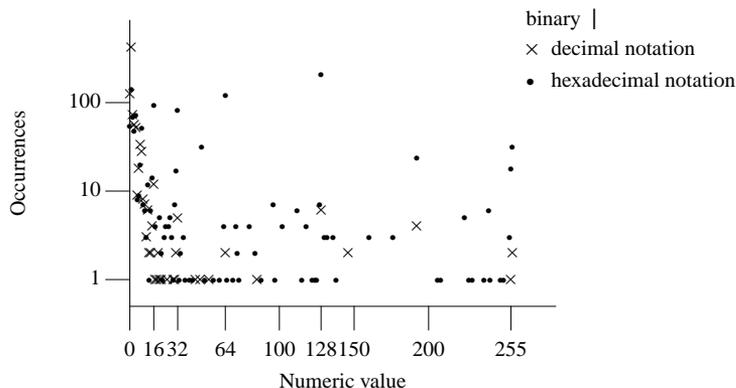
# **The New C Standard** (Excerpted material)

---

**An Economic and Cultural Commentary**

**Derek M. Jones**

derek@knosof.co.uk



**Figure 1244.1:** Number of *integer-constants* having a given value appearing as the right operand of the bitwise-OR operator. Based on the visible form of the `.c` files.

## 6.5.12 Bitwise inclusive OR operator

inclusive-OR-expression  
syntax

*inclusive-OR-expression*:

*exclusive-OR-expression*

*inclusive-OR-expression* | *exclusive-OR-expression*

1244

### Commentary

The discussion on AND-expression is applicable here.

### Coding Guidelines

The `|` and `||` operators differ from their corresponding AND operators in that the zero/nonzero status of their result is always the same, even though the actual result values are likely to differ.

`0x10 | 0x01`  $\Rightarrow$  `0x11`

`0x10 || 0x01`  $\Rightarrow$  `1`

While it is possible to use either operators in a context where the result is compared against zero, the guideline recommendation dealing with matching an operator to the role of its result still applies. The pattern of operator context usage (see Table 1244.1) is similar to that of the two AND operators.

**Table 1244.1:** Occurrence of the `|` and `||` operator (as a percentage of all occurrences of each operator; the parenthesized value is the percentage of all occurrences of the context that contains the operator). Based on the visible form of the `.c` files.

Context		
<b>if</b> control-expression	8.8 ( 0.7)	86.0 ( 6.9)
other contexts	90.7 (—)	11.9 (—)
<b>while</b> control-expression	0.3 ( 0.5)	1.9 ( 2.7)
<b>for</b> control-expression	0.0 ( 0.0)	0.3 ( 0.2)
<b>switch</b> control-expression	0.1 ( 0.3)	0.0 ( 0.0)

### Constraints

Each of the operands shall have integer type.

1245

### Commentary

The discussion for the various subsections is the same as those for the bitwise AND operator.

### Semantics

& binary  
operand type

---

1246 The usual arithmetic conversions are performed on the operands.

**Commentary**

The discussion in the various subsections is the same as that for the bitwise exclusive-OR operator.

^  
operands converted

---

1247 The result of the | operator is the bitwise inclusive OR of the operands (that is, each bit in the result is set if and only if at least one of the corresponding bits in the converted operands is set).

**Commentary**

This information is usually expressed in tabular form.

	0	1
0	0	1
1	1	1

**Common Implementations**

The Unisys A Series<sup>[1]</sup> uses signed magnitude representation. If the operands have an unsigned type, the sign bit is not affected by the bitwise-OR operator. If the operands have a signed type, the sign bit does take part in the bitwise-OR operation.

**Coding Guidelines**

Although the result of the bitwise inclusive-OR operator is the common type derived from the usual arithmetic conversions, for the purpose of these guideline recommendations its role is the same as that of its operands.

usual arithmetic conversions object role

# References

1. Unisys Corporation. *C Programming Reference Manual, Volume 1:*

*Basic Implementation.* Unisys Corporation, 8600 2268-203 edition, 1998.