Notes on Binary Operations

Michael Hamilton

April 29, 2007

Nearly all programming languages offer binary datatypes along with their type-specific operators. These notes are intended to provide some insight on their use and help demystify their cryptic nature.

Definition 1 (left shift): Given a decimal value, x, with an equivalent binary summation, $a_k 2^k + a_{k-1}2^{k-1} + \ldots + a_0 2^0,\ a_i \in \{0,1\}$, then the binary operator $\ll$ is defined as follows:

$x \ll i = a_k 2^{k+i} + a_{k-1}2^{k+i-1} + \ldots + a_0 2^{i-k} + 0 \cdot 2^{i-k+1} + \ldots + 0 \cdot 2^0$,

where $a_i \in \{0,1\}$

Example 1: Let $x = 5$, then the equivalent binary summation is: $1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$. In binary, the value is usually displayed as $b101$. Then, $x \ll 2 = 20 = 2 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = b10100$.

Theorem 1: $x \ll 1 = 2x$

proof:

- Let $x = a_k 2^k + a_{k-1}2^{k-1} + \ldots + a_0 2^0,\ a_i \in \{0,1\}$
- Then, $x \ll 1$
  $= a_k 2^{k+1} + a_{k-1}2^k + \ldots + a_0 2^1 + 0$
  $= 2(a_k 2^k + a_{k-1}2^{k-1} + \ldots + a_0 2^0)$ [factor a 2 from all the terms]

Q.E.D.

Theorem 2: $x \ll n = 2^n x$

proof (by induction):

Let $x = a_k 2^k + a_{k-1}2^{k-1} + \ldots + a_0 2^0,\ a_i \in \{0,1\}$

basis: $n = 1$: $x \ll 1 = 2x = 2^1 x$ [by Theorem 1]

inductive hypothesis: assume $P(i) : x \ll i = 2^i$ for some $i \geq 1$

(induction step)

$x \ll i + 1$

$= a_k 2^{k+i+1} + a_{k-1}2^{k+i} + \cdots + a_0 2^{i+1-k}$

$= 2(a_k 2^{k+i} + a_{k-1}2^{k+i-1} + \cdots + a_0 2^{i-k})$

$= 2(2^i)$ [by $P(i)$]
$= 2^{i+1}$

By method of mathematical induction, the claim holds.

Q.E.D.