

C/C++ bitwise operations		Single bit operations	
&	AND	$y = x \mid (1 < < n)$	Set the n^{th} bit
	OR	$y = x \& \sim(1 < < n)$	Unset the n^{th} bit
^	XOR	$y = x \wedge (1 < < n)$	Toggle the n^{th} bit
~	NOT	$\text{return } x \& (1 < < n)$	Test if the n^{th} bit is set
<<	SHIFT (left)	$y = x \& (x-1)$	Turn off rightmost 1bit
>>	SHIFT (right)	$y = x \& (-x)$	Isolate rightmost 1bit
Useful snippets		$y = x \mid (x-1)$	Right propagate rightmost 1bit (fill in ones)
Counting (c) bits set in x		$x \wedge= x >> 16; x \wedge= x >> 8; x \wedge= x >> 4; x \&= 0xf; \text{return } (0x6996 >> x) \& 1;$	Turn on rightmost 0bit
Computing parity in parallel (32 Bit)		$y = x \mid (x+1)$	Isolate rightmost 0bit
Integer arithmetics		$y = \sim x \& (x+1)$	Isolate rightmost 0bit
$x = y << n$	Multiply by n times 2		
$x = y >> n$	Divide by n times 2		
$\text{return } (x \& 1) == 0$	Is x even?		
$\text{return } (x \&\& !(x \& (x - 1)))$	Is x power of 2?		
$\text{return } (x \wedge y) < 0$	Has x opposite sign than y?		
$y \wedge ((x \wedge y) \& -(x < y))$	min(x,y)		
$x \wedge ((x \wedge y) \& -(x < y))$	max(x,y)		