Fixed point

Goldsmiths Computing

January 13, 2019

Motivation

Representing:

- integers within a range
- continuous sequence of place-value digits

Useful and practical:

- simple to implement in hardware
- reasonable properties
- … but some unexpected behaviours too

Definition

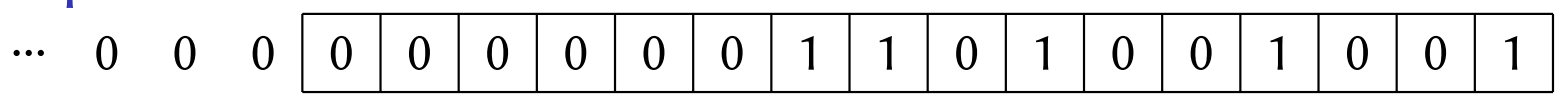
- data type for a number
- fixed number of digits
- implicit scaling factor
  - number represented is integer value multiplied by scaling factor

Unsigned integers

Fixed-width of binary digits

- assumed infinite leftward zeros
- radix (decimal) point at the right-hand end of the fixed-width field
- scaling factor of 1

Representation



(16-bit fixed point integer)

Overflow

What happens when a calculation would put a 1 in the infinite sea of zeros?

hardware	wraparound, carry flag
C++	wraparound
Java	bad luck, no unsigned integers

Signed integers

Many possible representations:

- sign-magnitude

sign	high bit (0: positive; 1: negative)
magnitude	remaining bits

- ones complement

  -x $\neg$x

- twos-complement

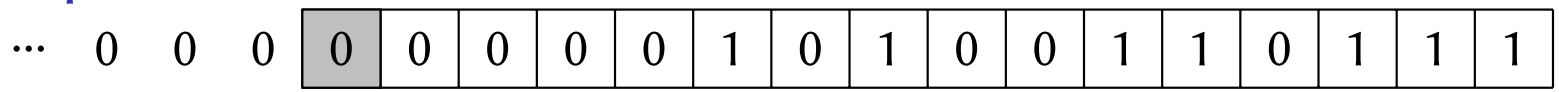
  -x $\neg$x + 1

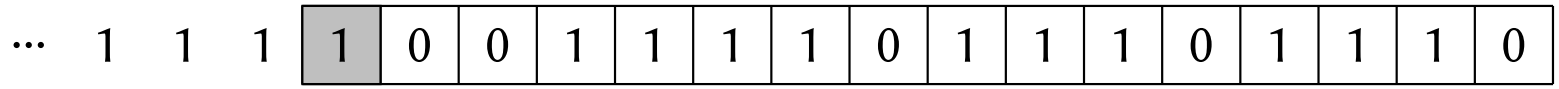
Twos complement

What's so good about twos-complement?

- only one zero
- addition, subtraction and multiplication all the same as unsigned

Representation





In practice, all current systems use twos-complement.

```
function NEG(x)
    r ← ¬x
    return r + 1
end function
```

Overflow

What happens when a calculation would put a non-sign bit in the infinite sea of sign bits?

hardware	wraparound, carry flag
Java	wraparound
C++	bad luck, undefined behaviour

Absolute value

`abs` (C++ cstdlib) or `Math.abs` (Java):

- return (as an `int`) the absolute (non-negative) value of its argument

```
function ABS(x)
    if x < 0 then
        return NEG(x)
    else
        return x
    end if
end function
```

- does this always return a non-negative answer?

Population count

`popcnt` (x86 instruction)

- return how many one bits are set in the (unsigned) integer argument.

Divide-and-conquer implementation:

function POPCNT(x)
    return POPCNTW(x,W) ▷ W is the integer width
end function
function POPCNTW(x,w)
    if w = 1 **then**
        return x
    else
        nw ⟵ $\frac{w}{2}$ ▷ w assumed to be a power of 2
        lo ⟵ POPCNTW(x & $2^{nw} - 1$, nw)
        hi ⟵ POPCNTW($\left\lfloor \frac{x}{2^{nw}} \right\rfloor$, nw)
        return lo + hi
    end if
end function

Population count

Parallel divide-and-conquer implementation:

function POPCNT(x) ▷ Assume W = 32

$x \leftarrow (x \;\&\; \texttt{0x55555555}) + (\left\lfloor \frac{x}{2} \right\rfloor \;\&\; \texttt{0x55555555})$

$x \leftarrow (x \;\&\; \texttt{0x33333333}) + (\left\lfloor \frac{x}{4} \right\rfloor \;\&\; \texttt{0x33333333})$

$x \leftarrow (x \;\&\; \texttt{0x0f0f0f0f}) + (\left\lfloor \frac{x}{16} \right\rfloor \;\&\; \texttt{0x0f0f0f0f})$

$x \leftarrow (x \;\&\; \texttt{0x00ff00ff}) + (\left\lfloor \frac{x}{256} \right\rfloor \;\&\; \texttt{0x00ff00ff})$

$x \leftarrow (x \;\&\; \texttt{0x0000ffff}) + (\left\lfloor \frac{x}{65536} \right\rfloor \;\&\; \texttt{0x0000ffff})$

return x

end function

Work

1. Reading
    - Henry S. Warren, Jr. *Hacker's Delight*, Addison-Wesley (2003)
        - sections 5-1, 7-1