

CSE 20—Discrete Math

Winter, 2006

January 12 (Day 2)
Number Systems
Computer Arithmetic
Algebraic Rules for Boolean Functions

Instructor: Neil Rhodes

Euclid's proof of the infinitude of primes

Givens

- Definition of prime
 - a number > 1 whose only factors are one and itself
- Fundamental Theorem of Arithmetic
 - Any positive integer > 1 is either:
 - prime
 - or, can be written as a unique product of primes

Given any finite set of primes $p_1, \dots, p_m$

- Compute $t = p_1 \cdot p_2 \cdot \dots \cdot p_m + 1$
- t is not divisible by any p_i
- Therefore, t is either:
 - prime itself
 - has a prime factor different from any p_i
- Thus, we've identified at least one new prime, so there are at least $m+1$ primes.
- Lather, rinse, repeat

2

Numbers in Different Bases

Given a number $d_{n-1}d_{n-2}d_{n-3}\dots d_1d_0$ in a base b , then:

- Each d_i is in the range $0 \dots b-1$
- The value of the number is $d_{n-1}b^{n-1} + d_{n-2}b^{n-2} + d_{n-3}b^{n-3} + \dots + d_1b^1 + d_0b^0$
- b is an integer > 1
- The largest such number with n digits is $b^n - 1$
- The smallest such number with n digits is b^{n-1}

Common bases in CS:

- 2 (binary)
 - binary digit = bit
 - Inconvenient to read/write
- 8 (octal)
 - No 8, 9, 10
- 16 (hexadecimal)
 - How to represent d_i in range $10 \dots 15$?
 - Very common

3

Converting between Bases

Converting a number n in base b to base c

- Divide n by c . The remainder is the rightmost digit
- Continue until n is 0

Shortcut for converting between common bases

- 2 $\rightarrow$ 8
 - Group in triples from RHS
- 2 $\rightarrow$ 16
 - Group in group of four from RHS
- 16 $\rightarrow$ 2
 - Convert each hex digit into 4 binary digits
- 8 $\rightarrow$ 2
 - Convert each octal digit into 3 binary digits

4

Binary Numbers

Using k bits to represent a non-negative number

- k is often a power of 2 (8, 16, 32, 64)
- Total unique numbers is 2^k
- Biggest number is $2^k - 1$
- Most-significant-bit
- Least-significant-bit

Adding numbers

- Add each pair of bits (including carry) from right to left, carrying as needed
- Overflow: If we end up with a carry, we've overflowed the result and can no longer represent it
 - $k=8$: $A8 + 86 =$

5

Negative Numbers

Sign-bit only

- MSB is now a sign bit
- Remaining bits represent the positive number
- Problems:
 - 2 representations of 0
 - Difficult to add/subtract

Ones-complement

- To obtain a negative number, complement (flip) each bit in the positive number
- Problems:
 - 2 representations of 0

Twos-complement

- To obtain the negative number, complement (flip) each bit in the positive number and add 1
 - $\text{neg}(n) = 2^k - n$
 - $\text{neg}(n) + n = 2^k - n + n = 2^k = 0$
- Negative of 0
- Problems:
 - More negative numbers than positive (not symmetric)

6

Adding Twos-Complement Numbers

Add the two (whether positive or negative)

If carry bit from MSB is set:

- If both were negative
 - If result looks negative, it is correct
 - Example: $-3 + -2$
 - If result looks positive, overflow occurred
- If both were positive
 - Not possible
- If signs were different, overflow can be ignored
 - Example: $5 + -3$

If carry bit from MSB is not set:

- If both were positive and result looks negative: overflow occurred

Summary:

- Ignore the carry bit from the MSB.
- If two numbers of the same sign yield a result of a different sign, overflow occurred.

7

Other Operations with Twos-Complement Numbers

Subtraction: convert subtraction problem to addition problem

- Convert $a - b$ to $a + (-b)$

8

Other Operations with Twos-Complement Numbers

Multiplication:

- Multiply by power of 2: **arithmetic** shift left
 - Shift (all but sign bit) 1 to the left (new bit on right is a zero)
- Arbitrary

```
result = 0
while b ≠ 0
  if (b and 1) ≠ 0
    result = result + a
  shift a left by one
  shift b right by one
return result
```

9

Other Operations with Twos-Complement Numbers

Division

- Divide by power of 2: **arithmetic** shift right
 - Shift all bits 1 to the right (what about new sign bit?)
- Example of arbitrary divisor and dividend using long division: 25/4

10

Algebraic Rules for Boolean Functions

Equivalences between boolean functions (similar to algebraic equivalences):

- Associative
- Distributive
- Idempotent
- Double negation
- DeMorgan's
- Commutative
- Absorption
- Bound
- Negation

11

Simplifying a Boolean Function

Example

- Show that $(p + \sim q)(\sim p + q) = \sim pq + \sim qp$

12