CPE201
Digital Design

By Benjamin Haas

Class 3: Arithmetic and Complements

University of Nevada, Reno

Binary Arithmetic

- By hand, it uses all the same rules you know for decimal arithmetic

Binary Addition

- 4 Rules

0 + 0 = 0

0 + 1 = 1

1 + 0 = 1

1 + 1 = 10   The sum is 0 and the carry is 1

Binary Addition

- Carry is the same as for decimal

19 $\quad\quad$ $1^{1}9$ Put carry here

+ 1 $\quad\quad$ + 1

0 First col adds to 10 $\quad\quad$ 20

Binary Addition Examples

- 100 + 110
- 010 + 1
- 11 + 111
- 011.1 + 111.01

Binary Multiplication

- Easiest multiplication table ever!

0 x 0 = 0

0 x 1 = 0

1 x 0 = 0

1 x 1 = 1

Binary Multiplication

- Makes partial products a breeze

```
   100        11.1
  x110       x 11
   000        111
  100        +111
+100         1010.1
 11000
```

Binary Subtraction

- The usual rules for positive numbers where you subtract a smaller number from a bigger number
- Subtract each column and do a borrow, if necessary

Binary Subtraction

11000	1010
- 100	- 111
10100	0011

Binary Subtraction

- The rules work well for a small subset of arithmetic problems
  - What about negative numbers?
  - What about subtracting a larger # from a smaller one?
- Complements!

Binary Division

- Just like decimal, same as the other operators

Unsigned Numbers

- The number of bits n used to store the # determine the max value, $2^n - 1$
- For 4 bits, max value = $2^4 = 15$
  - Not 16 because of including zero
- For 8 bits, max value = 255
- For 16 bits, max value = 65,535

Complements

- Used to express or store negative numbers
- Turns subtraction into addition, which is easier
- In general, there are 2 types of complements
  - Diminished radix complement (1’s complement)

1’s Complement

- Invert every bit (1 to 0, 0 to 1)
  - Keep the same number of bits

11000

00111

2's Complement

- Take a 1's Complement of a # and add 1
- 2's Complement = (1's Complement) + 1

11000

00111

+ 1

01000

2's Complment

- Another way to get it is to start at the LSB and move left
- When you get to the first 1, invert all bits to the left of it

Invert 11000 No inversion

01000

2’s Complement Examples

111

10101

10111

100

9’s and 10’s Complement

- Like 1’s and 2’s Complement, but for decimal
  - Feel free to look at it
  - Not used in this class

Signed Numbers

- Splits the value range for neg and pos #'s
  - Max value = $2^{n-1} - 1$
  - Min value = $-(2^{n})$
- For 4 bits, -8 to 7
- For 8 bits, max value = -128 to 127
- For 16 bits, max value = -32,768 to 32,767

Signed Numbers

- Highest bit (MSB) is a sign bit
  - 0 is positive
  - 1 is negative
- Negative #'s are the 2's complement of the equivalent positive #

-1 = 1 = 0000 0001 = 1111 1110 = 1111 1111 = -1

Pos # 1's Comp 2's Comp

Signed Numbers

- Weights are slightly modified
  - MSB is negative
- $2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$ for unsigned becomes
- $-2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$ for signed
- 1011 is $-2^3 + 2^1 + 2^0 = -8 + 2 + 1 = -5$
- 1111 is $-2^3 + 2^2 + 2^1 + 2^0 = -8 + 4 + 2 + 1 = -1$

Signed Number Addition

- Pad with leading zeros to fill the bit size
- Add the two numbers
- Throw away carry

0000 1111	15	0100 1010	74
+1111 0001	+(-15)	+1000 1111	+(-113)
1 0000 0000	0	1101 1001	-39

Signed Number Subtraction

- Take the 2's complement of the 2nd #, then add

0000 1111 15 0100 1010 74

- 0000 1111 -15 - 0111 0001 -113

0000 0000 0 1101 1001 -39

Reading

- This lecture
  - Sections 2.4-2.7
- Next lecture
  - Sections 2.10-2.12