RUTGERS

THE STATE UNIVERSITY OF NEW JERSEY

Computer Architecture (CS-211) Recitation 5

Siwei Mai

Topics

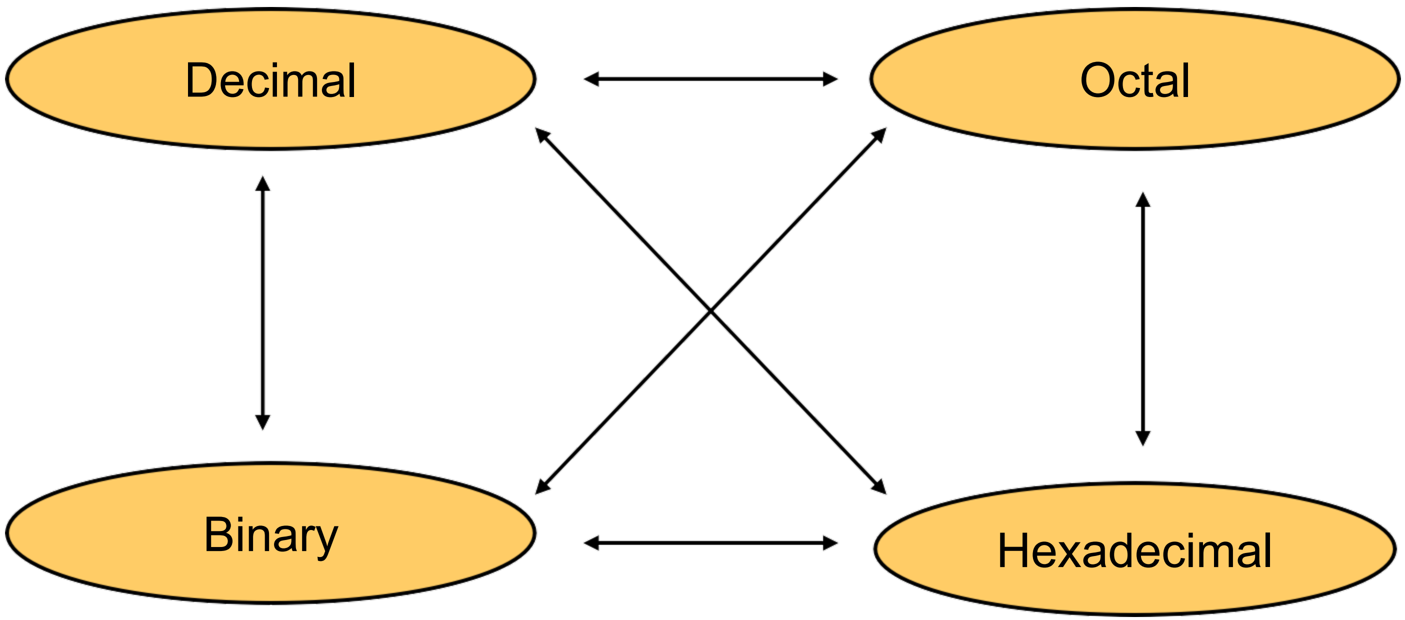
Number System

- Decimal, Binary, Octal, Hexadecimal
- Fractions
- Endianness (Big and Little)
- One’s Complement and Two’s Complement

* Some materials are collected and compiled from previous year's CS 211 lectures and TAs

Number Systems

- Possible conversion among bases



Number System

Decimal	Binary	Octal	Hexa-decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

Decimal	Binary	Octal	Hexa-decimal
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

$$25_{10} = 11001_2 = 31_8 = 19_{16}$$

Converting Hex to Binary

- Each hex digit can be represented by 4 binary digits
  - Why?
- Example
  - 0x2A8C (hex) = 0b0010101010001100 (binary)
- So to convert hex to binary, just convert each digit and concatenate

Converting Binary to Hex

- Do it reverse
  - Group each set of 4 digits and change to corresponding digit in hex
  - Go from right to left
- Example
  - 0b1011011110011100 = 0xB79C

Bin-Hex Example

- 0xFA01
  - 0b 1111 1010 0000 0001

- 0x370E
  - 0b 0011 0111 0000 1110

- 0xBA52
  - 0b 1011 1010 0101 0010

- 0b1101111110101000
  - 0xDFA8

- 0b1100000101110
  - 0x182E

Converting Octal to Binary

- Each octal digit can be represented by 3 binary digits
  - Why?
- Example
  - 0o276 (hex) = 0b010111110 (binary)
- So to convert octal to binary, just convert each digit and concatenate

Converting Binary to Octal

- Do it reverse
  - Group each set of 3 digits and change to corresponding digit in octal
  - Go from right to left
- Example
  - 0b11111100110 = 0o3746

Decimal and Binary Fractions

- In decimal, digits to the right of radix point have value $1/10^i$ for each digit in the i^{th} place
- Similarly, in binary, digits to the right of radix point have value $1/2^i$ for each i^{th} place
- Example
  - 0.625 (decimal) = 0.101 (binary)
- How to convert?

Decimal and Binary Fractions Example

- Begin with the decimal fraction and multiply by 2. Keep the first binary digit
  - 0.625 * 2 = **1**.25
  - So far 0.625 =0.1?? (binary)

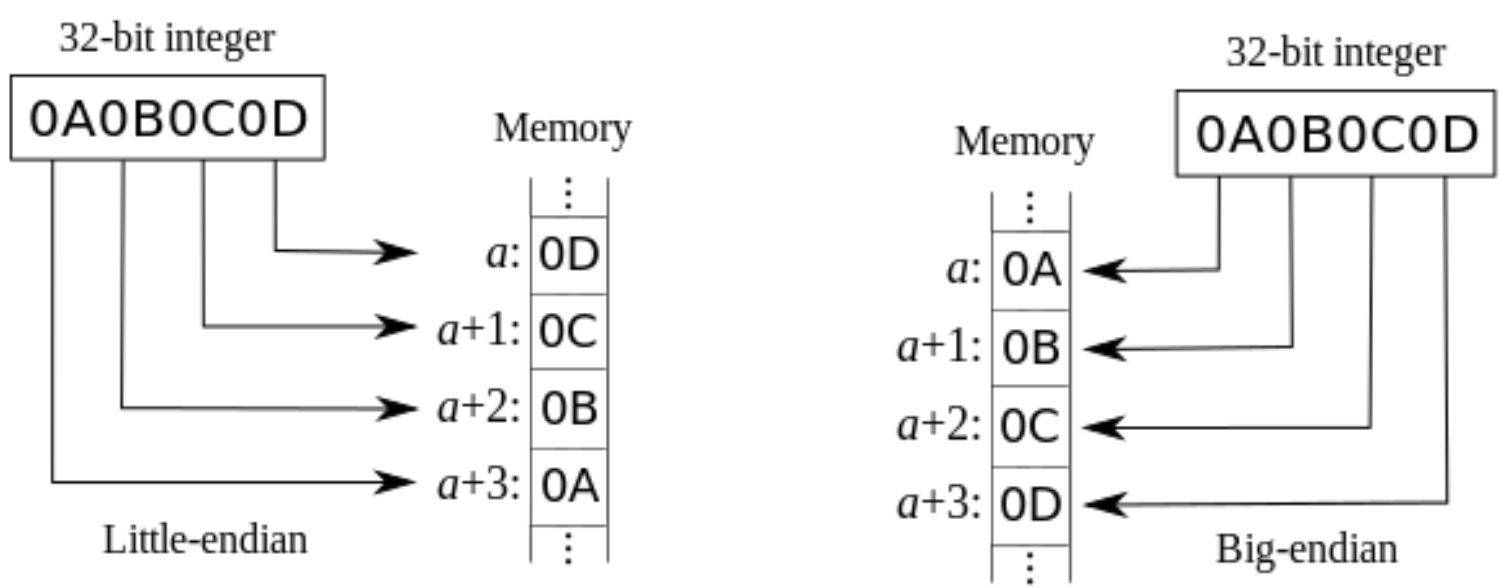
- Follow the same rule by multiplying 2 with the remaining digit
  - 0.25*2 = **0**.50
  - So far 0.625 = 0.10? (binary)

- Continue this process until we get a zero as our decimal part
  - 0.50*2 = **1**.00
  - So far 0.625 = 0.101 (binary)

- Because we have 0 as a fractional part we can end this process

Endianness

- Endianness - the order of the bytes stored in memory
- Big endian
  - MSB is stored at a particular address and the subsequent bytes are stored in the following higher memory addresses
  - LSB is stored at the highest memory address
- Little endian
  - LSB is stored at the lower memory address and the subsequent bytes are stored in the following higher memory addresses
  - MSB is stored at the highest memory address



Endianness (Byte Ordering Example)

- Consider the following word (32 bit) of memory

MSB = Most significant bit

the bit position in a binary number having the greatest value

LSB = Least significant bit

	AB	**CD**	**00**	**00**
Memory address	**0**	**1**	**2**	**3**

- Big Endian interprets as
  - AB CD 00 00 (2882338816)
- Little Endian interprets as
  - 00 00 CD AB (52651)

How to Represent Negative Integers?

- Use a sign bit
  - Positive number: MSB(left most bit) is 0
  - Negative number: MSB is 1

- In 4 bits
  - **0**100 = 4
  - **1**100 = – 4
  - But 1000 = – 0 and 0000 = 0 (two zero values)

- Range: $[-2^{N-1}, 2^{N-1}]$

One's Complement

- Represent negative numbers by complementing positive numbers

- Range: $[\,-(2^{N-1}-1),\ 2^{N-1}-1\,]$

- It has two zero representation

000	001	010	011	100	101	110	111
0	1	2	3	-3	-2	-1	-0

Two's Complement

- Advantages – only 1 zero & convenient for arithmetic computation

- Flip the bits and add 1 (One's complement + 1)
- Example

  40 = 0010 1000
  Flip 1101 0111
  Add 1 1101 1000 (-40 in two's complement form)

000	001	010	011	100	101	110	111
0	1	2	3	-4	-3	-2	-1

- What is the range that can represent with n bits?

$$[-2^{n-1}, 2^{n-1}-1]$$

Arithmetic of Two's Complement

- Arithmetic addition

+6	0000 0110	-6	1111 1010
<u>+13</u>	<u>0000 1101</u>	<u>+13</u>	<u>0000 1101</u>
+19	0001 0011	+7	0000 0111

Arithmetic of Two’s Complement

- Arithmetic subtraction

-5	1111 1011	-5	1111 1011
+ -6	0000 0110	- -6	1111 1010
-11	1111 0101	+ 1	0000 0001

Two’s Complement Overflow

- It needs one extra bit but the sign bit will be wrong
- How to detect an overflow?
  - Adding 2 positive numbers -> But negative result
  - Adding 2 negative numbers -> But positive result

6	0110	-6	1010
+ 5	0101	+ -6	1010
-5	1011	4	0100

Q&A

Thanks!