FROM ZERO TO ONE

Number Systems

- Decimal numbers

1000's column 100's column 10's column 1's column

$5374_{10} =$

- Binary numbers

8's column 4's column 2's column 1's column

$1101_2 =$

ELSEVIER

FROM ZERO TO ONE

Number Systems

- Decimal numbers

1000's column 100's column 10's column 1's column

$5374_{10} = 5 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 4 \times 10^0$

five thousands, three hundreds, seven tens, four ones

- Binary numbers

8's column 4's column 2's column 1's column

$1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 13_{10}$

one eight, one four, no two, one one

ELSEVIER

FROM ZERO TO ONE

Powers of Two

- $2^0 =$
- $2^1 =$
- $2^2 =$
- $2^3 =$
- $2^4 =$
- $2^5 =$
- $2^6 =$
- $2^7 =$
- $2^8 =$
- $2^9 =$
- $2^{10} =$
- $2^{11} =$
- $2^{12} =$
- $2^{13} =$
- $2^{14} =$
- $2^{15} =$

ELSEVIER

FROM ZERO TO ONE

Powers of Two

- $2^0 = 1$
- $2^1 = 2$
- $2^2 = 4$
- $2^3 = 8$
- $2^4 = 16$
- $2^5 = 32$
- $2^6 = 64$
- $2^7 = 128$
- $2^8 = 256$
- $2^9 = 512$
- $2^{10} = 1024$
- $2^{11} = 2048$
- $2^{12} = 4096$
- $2^{13} = 8192$
- $2^{14} = 16384$
- $2^{15} = 32768$
- Handy to memorize up to 2^9

ELSEVIER

Number Conversion

FROM ZERO TO ONE

- Decimal to binary conversion:
  - Convert 10011_2 to decimal
- Decimal to binary conversion:
  - Convert 47_{10} to binary

Number Conversion

FROM ZERO TO ONE

- Decimal to binary conversion:
  - Convert 10011_2 to decimal
  - $16\times1 + 8\times0 + 4\times0 + 2\times1 + 1\times1 = 19_{10}$
- Decimal to binary conversion:
  - Convert 47_{10} to binary
  - $32\times1 + 16\times0 + 8\times1 + 4\times1 + 2\times1 + 1\times1 = 101111_2$

Binary Values and Range

FROM ZERO TO ONE

- *N*-digit decimal number
  - How many values?
  - Range?
  - Example: 3-digit decimal number:
- *N*-bit binary number
  - How many values?
  - Range:
  - Example: 3-digit binary number:

ELSEVIER

Binary Values and Range

FROM ZERO TO ONE

- *N*-digit decimal number
  - How many values? **10^N**
  - Range? **$[0, 10^N - 1]$**
  - Example: 3-digit decimal number:
    - **10^3 = 1000 possible values**
    - **Range: [0, 999]**
- *N*-bit binary number
  - How many values? **2^N**
  - Range: **$[0, 2^N - 1]$**
  - Example: 3-digit binary number:
    - **2^3 = 8 possible values**
    - **Range: [0, 7] = [000_2 to 111_2]**

ELSEVIER

FROM ZERO TO ONE

Hexadecimal Numbers

Hex Digit	Decimal Equivalent	Binary Equivalent
0	0	
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
A	10	
B	11	
C	12	
D	13	
E	14	
F	15	

ELSEVIER

FROM ZERO TO ONE

Hexadecimal Numbers

Hex Digit	Decimal Equivalent	Binary Equivalent
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

ELSEVIER

FROM ZERO TO ONE

Hexadecimal Numbers

- Base 16
- Shorthand for binary

ELSEVIER

FROM ZERO TO ONE

Hexadecimal to Binary Conversion

- Hexadecimal to binary conversion:
  - Convert $4AF_{16}$ (also written 0x4AF) to binary
- Hexadecimal to decimal conversion:
  - Convert 0x4AF to decimal

ELSEVIER

Hexadecimal to Binary Conversion

- Hexadecimal to binary conversion:
  - Convert $4AF_{16}$ (also written 0x4AF) to binary
  - $0100\ 1010\ 1111_2$
- Hexadecimal to decimal conversion:
  - Convert $4AF_{16}$ to decimal
  - $16^2 \times 4 + 16^1 \times 10 + 16^0 \times 15 = 1199_{10}$

Bits, Bytes, Nibbles…

- Bits

10010110

most significant bit — least significant bit

- Bytes & Nibbles

byte

10010110

nibble

- Bytes

CEBF9AD7

most significant byte — least significant byte

FROM ZERO TO ONE

Large Powers of Two

- 2^{10} = 1 kilo ≈ 1000 (1024)
- 2^{20} = 1 mega ≈ 1 million (1,048,576)
- 2^{30} = 1 giga ≈ 1 billion (1,073,741,824)

FROM ZERO TO ONE

Estimating Powers of Two

- What is the value of 2^{24}?
- How many values can a 32-bit variable represent?

ELSEVIER

FROM ZERO TO ONE

Estimating Powers of Two

- What is the value of 2^{24}?

  - $2^4 \times 2^{20} \approx$ **16 million**

- How many values can a 32-bit variable represent?

  -$2^2 \times 2^{30} \approx$ **4 billion**

ELSEVIER

FROM ZERO TO ONE

Addition

- Decimal

```
   11  ← carries
  3734
+ 5168
------
  8902
```

- Binary

```
   11  ← carries
  1011
+ 0011
------
  1110
```


ELSEVIER

Binary Addition Examples

FROM ZERO TO ONE

- Add the following 4-bit binary numbers

```
   1001
+  0101
```

- Add the following 4-bit binary numbers

```
   1011
+  0110
```


Binary Addition Examples

FROM ZERO TO ONE

- Add the following 4-bit binary numbers

```
      1
   1001
+  0101
   1110
```

- Add the following 4-bit binary numbers

```
   111
   1011
+  0110
  10001
```

Overflow!

Overflow

FROM ZERO TO ONE

- Digital systems operate on a **fixed number of bits**
- Overflow: when result is too big to fit in the available number of bits
- See previous example of 11 + 6

Signed Binary Numbers

FROM ZERO TO ONE

- Sign/Magnitude Numbers
- Two's Complement Numbers

Sign/Magnitude Numbers

- 1 sign bit, *N*-1 magnitude bits
- Sign bit is the most significant (left-most) bit
  - Positive number: sign bit = 0
  - Negative number: sign bit = 1

$$A : \{a_{N-1}, a_{N-2}, \cdots a_2, a_1, a_0\}$$

$$A = (-1)^{a_{n-1}} \sum_{i=0}^{n-2} a_i 2^i$$

- Example, 4-bit sign/mag representations of ± 6:

  +6 =

  - 6 =

- Range of an *N*-bit sign/magnitude number:

Sign/Magnitude Numbers

- 1 sign bit, *N*-1 magnitude bits
- Sign bit is the most significant (left-most) bit
  - Positive number: sign bit = 0
  - Negative number: sign bit = 1

$$A : \{a_{N-1}, a_{N-2}, \cdots a_2, a_1, a_0\}$$

$$A = (-1)^{a_{n-1}} \sum_{i=0}^{n-2} a_i 2^i$$

- Example, 4-bit sign/mag representations of ± 6:

  +6 = **0110**

  - 6 = **1110**

- Range of an *N*-bit sign/magnitude number:

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

Sign/Magnitude Numbers

- Problems:
  - Addition doesn't work, for example -6 + 6:

```
   1110
 + 0110
 ------
  10100 (wrong!)
```

  - Two representations of 0 (± 0):

```
   1000
   0000
```


Two's Complement Numbers

- Don't have same problems as sign/magnitude numbers:
  - Addition works
  - Single representation for 0

FROM ZERO TO ONE

Two's Complement Numbers

- Msb has value of -2^{N-1}

$$A = a_{n-1}\left(-2^{n-1}\right) + \sum_{i=0}^{n-2} a_i 2^i$$

- Most positive 4-bit number:
- Most negative 4-bit number:
- The most significant bit still indicates the sign (1 = negative, 0 = positive)
- Range of an N-bit two's comp number:

FROM ZERO TO ONE

Two's Complement Numbers

- Msb has value of -2^{N-1}

$$A = a_{n-1}\left(-2^{n-1}\right) + \sum_{i=0}^{n-2} a_i 2^i$$

- Most positive 4-bit number: **0111**
- Most negative 4-bit number: **1000**
- The most significant bit still indicates the sign (1 = negative, 0 = positive)
- Range of an N-bit two's comp number:

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

"Taking the Two's Complement"

- Flip the sign of a two's complement number
- Method:
  1. Invert the bits
  2. Add 1
- Example: Flip the sign of $3_{10} = 0011_2$

"Taking the Two's Complement"

- Flip the sign of a two's complement number
- Method:
  1. Invert the bits
  2. Add 1
- Example: Flip the sign of $3_{10} = 0011_2$

1. 1100

2. + 1

$1101 = -3_{10}$

Two's Complement Examples

- Take the two's complement of $6_{10} = 0110_2$

- What is the decimal value of 1001_2?

Two's Complement Examples

- Take the two's complement of $6_{10} = 0110_2$
  1. 1001
  2. \+ 1

  $1010_2 = -6_{10}$

- What is the decimal value of the two's complement number 1001_2?
  1. 0110
  2. \+ 1

  $0111_2 = 7_{10}$, so $1001_2 = -7_{10}$

Two's Complement Addition

FROM ZERO TO ONE

- Add 6 + (-6) using two's complement numbers

```
   0110
+  1010
```

- Add -2 + 3 using two's complement numbers

```
   1110
+  0011
```


ELSEVIER

Two's Complement Addition

FROM ZERO TO ONE

- Add 6 + (-6) using two's complement numbers

```
   111
   0110
+  1010
  10000
```

- Add -2 + 3 using two's complement numbers

```
  111
   1110
+  0011
  10001
```


ELSEVIER

Increasing Bit Width

- **Extend number from *N* to *M* bits (*M* > *N*) :**
  - Sign-extension
  - Zero-extension

Sign-Extension

- Sign bit copied to msb's
- Number value is same

- **Example 1:**
  - 4-bit representation of 3 = 0011
  - 8-bit sign-extended value: 00000011
- **Example 2:**
  - 4-bit representation of -5 = 1011
  - 8-bit sign-extended value: 11111011

FROM ZERO TO ONE

Zero-Extension

- Zeros copied to msb's
- Value changes for negative numbers

- **Example 1:**
  - 4-bit value = $0011_2 = 3_{10}$
  - 8-bit zero-extended value: 00000011 = 3_{10}
- **Example 2:**
  - 4-bit value = 1011 = -5_{10}
  - 8-bit zero-extended value: 00001011 = 11_{10}

FROM ZERO TO ONE

Number System Comparison

Number System	Range
Unsigned	$[0, 2^N-1]$
Sign/Magnitude	$[-(2^{N-1}-1), 2^{N-1}-1]$
Two's Complement	$[-2^{N-1}, 2^{N-1}-1]$

For example, 4-bit representation:

	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Unsigned									0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Two's Complement	1000	1001	1010	1011	1100	1101	1110	1111	0000	0001	0010	0011	0100	0101	0110	0111								
Sign/Magnitude		1111	1110	1101	1100	1011	1010	1001	0000 1000	0001	0010	0011	0100	0101	0110	0111								