Representation of `int` data

World Inside a Computer is Binary

Decimal Number System

- Basic symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Radix-10 positional number system. The radix is also called the base of the number system.

$$12304 = 1\times10^4+2\times10^3+3\times10^2+0\times10^1+4\times10^0$$

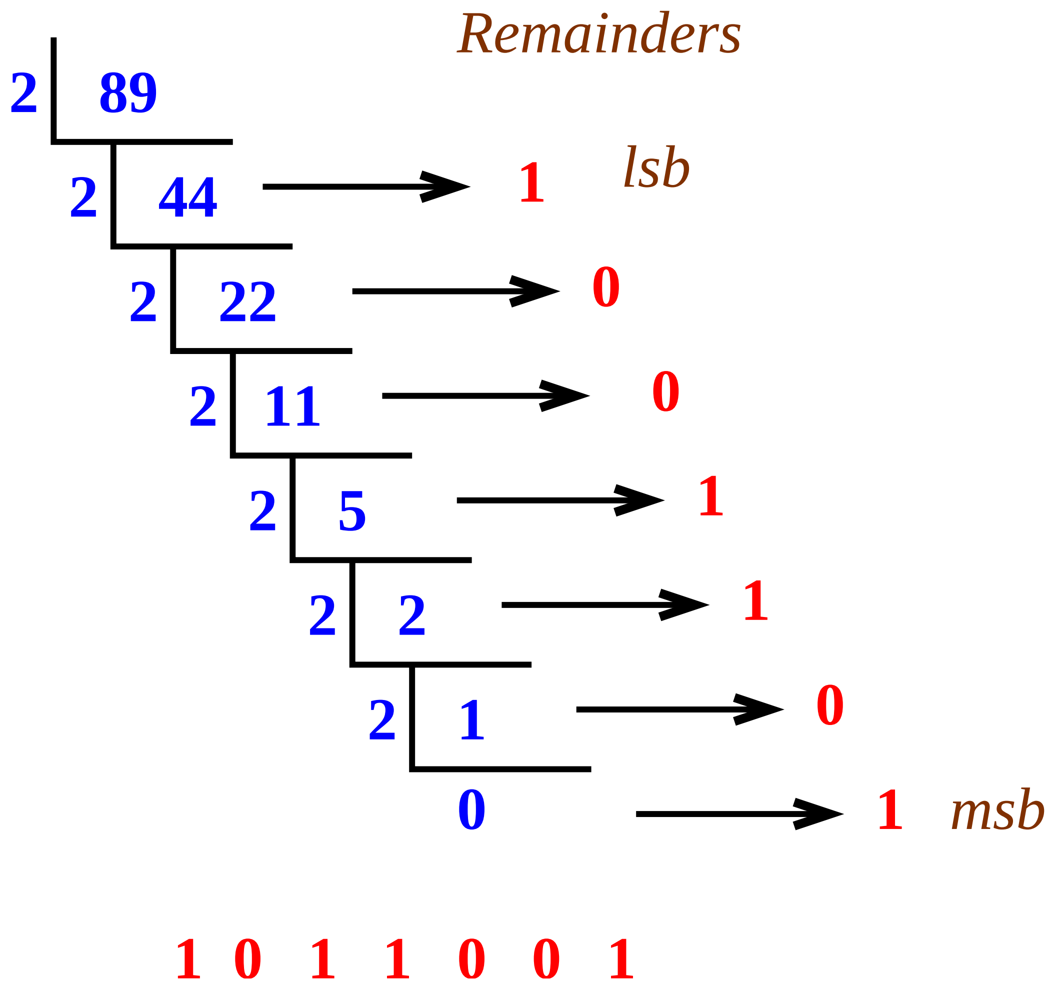
Unsigned Binary Number System

- Basic symbols: 0, 1
- Radix-2 positional number system.

$$10110 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

The value is 22 in decimal.

Decimal to Binary Conversion


Remainders
2 89
2 44
1 lsb
2 22
0
2 11
0
2 5
1
2 2
1
2 1
0
0
1 msb
1 0 1 1 0 0 1

Decimal to Binary Conversion

$$
\begin{aligned}
89_D &= 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\
&= 2 \cdot (2 \cdot (2 \cdot (2 \cdot (2 \cdot (2 \cdot 1 + 0) + 1) + 1) + 0) + 0) + 1
\end{aligned}
$$

Finite `word` Size of the CPU

- 32-bit word:

31	30		0
		...	

- Largest Number: $\sum_{i=0}^{31} 2^i = 4294967295$,

31	30		0
1	1	...	1

- Smallest Number: 0

31	30		0
0	0	...	0

Note

- The range of `unsigned int` or `unsigned` is (in *32-bits*) `0 - 4294967295`.
- The range of `unsigned short int` or `unsigned short` is (in *16-bits*) `0 - 65535`.

An Example with 4-bit Word Size

b_3	b_2	b_1	b_0

The Range of unsigned integer is:

$$0 \text{ to } 2^4 - 1 = 15$$

Bit String				*Decimal Value*
b_3	b_2	b_1	b_0	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7

Bit String				*Decimal Value*
b_3	b_2	b_1	b_0	
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

Signed Decimal Number

- We use '$+$' and '$-$' symbols to indicate sign in a decimal number.
- In a binary system only two symbols $\{0, 1\}$ are available to encode any information. So one extra bit is required to indicate the sign of a number.

Three Popular Schemes

- Signed Magnitude,
- 1's Complement,
- 2's Complement

Signed Magnitude

Consider a 4-bit word as an example.

b_3	b_2	b_1	b_0

b_3 is zero (0) for a positive number and it is one (1) for a negative number. Other three bits $b_2\ b_1\ b_0$ represent the magnitude of the number.

Bit String				*Decimal Value*
b_3	b_2	b_1	b_0	
0	0	0	0	+0
0	0	0	1	+1
0	0	1	0	+2
0	0	1	1	+3
0	1	0	0	+4
0	1	0	1	+5
0	1	1	0	+6
0	1	1	1	+7

Bit String				*Decimal Value*
b_3	b_2	b_1	b_0	
1	1	1	1	-7
1	1	1	0	-6
1	1	0	1	-5
1	1	0	0	-4
1	0	1	1	-3
1	0	1	0	-2
1	0	0	1	-1
1	0	0	0	-0

Signed magnitude

- There are two representations of zero: $+0, -0$,
- The range is $[-7 \cdots + 7]$ in 4-bits.
- The range is $[-(2^{n-1} - 1) \cdots + (2^{n-1} - 1)]$ for n-bits.

1's Complement Numeral

- Positive Numbers are same as the signed magnitude representation with the most significant bit[a] zero.
- If n is a number in 1's complement form, $-n$ is obtained by changing every bit to its complement. The result is called the 1's complement of n.

[a] b_3 in case of 4-bits and b_{n-1} in case of n-bits.

1's Complement Numeral

- A negative number has one (1) in its most significant bit.

$$\overbrace{0\ 1\ 1\ 0\ 1\ 1\ 0\ 1}^{+109} \xrightarrow{\text{1's complement}} \overbrace{1\ 0\ 0\ 1\ 0\ 0\ 1\ 0}^{-109}$$

$$\overbrace{1\ 0\ 0\ 1\ 0\ 0\ 1\ 0}^{-109} \xrightarrow{\text{1's complement}} \overbrace{0\ 1\ 1\ 0\ 1\ 1\ 0\ 1}^{+109}$$

Decimal Value	*Bit String*				*Bit String*				*Decimal Value*
	b_3	b_2	b_1	b_0	b_3	b_2	b_1	b_0	
+7	0	1	1	1	1	0	0	0	−7
+6	0	1	1	0	1	0	0	1	−6
+5	0	1	0	1	1	0	1	0	−5
+4	0	1	0	0	1	0	1	1	−4
+3	0	0	1	1	1	1	0	0	−3
+2	0	0	1	0	1	1	0	1	−2
+1	0	0	0	1	1	1	1	0	−1
+0	0	0	0	0	1	1	1	1	−0

1's Complement Representation

- Two representations of zero: $+0, -0$.
- The range is $[-7 \cdots + 7]$ in 4-bits.
- The range is $[-(2^{n-1} - 1) \cdots + (2^{n-1} - 1)]$.
- Positive number representation is identical to signed magnitude, but the negative number representations are different.

Signed Magnitude Verses 1's Complement

Decimal	Signed Magnitude	1's Complement
-0	1000	1111
-1	1001	1110
-2	1010	1101
-3	1011	1100
-4	1100	1011
-5	1101	1010
-6	1110	1001
-7	1111	1000

2's Complement

- Positive Numbers are same as the signed magnitude and 1's complement representations with the most significant bit[a] zero.
- If n is a number in 2's complement form, $-n$ is obtained by changing every bit to its complement and finally adding one to it. The result is called the 2's complement of n.

[a] b_3 in case of 4-bits and b_{n-1} in case of n-bits.

2's Complement Numeral

A negative number has a one (1) in the most significant position.

$$\overbrace{0\ 1\ 1\ 0\ 1\ 1\ 0\ 1}^{+109} \xrightarrow{\text{1's complement}} 10010010 + 1 \xrightarrow{\text{2's complement}} \overbrace{1\ 0\ 0\ 1\ 0\ 0\ 1\ 1}^{-109}$$

$$\overbrace{1\ 0\ 0\ 1\ 0\ 0\ 1\ 1}^{-109} \xrightarrow{\text{1's complement}} 01101100 + 1 \xrightarrow{\text{2's complement}} \overbrace{0\ 1\ 1\ 0\ 1\ 1\ 0\ 1}^{+109}$$

Direct 2's Complement

1 0 1 1 0 1 0 11 0 0 0 0 ⟶ 0 1 0 0 1 0 1 01 0 0 0 0

2's Comp

2's Complement Numeral

- Only one representations of zero: 0000.
- The range is $[-8 \cdots +7]$ for 4-bits.
- For n-bits, the range is i$[-(2^{n-1}) \cdots + (2^{n-1} - 1)]$.
- Positive representation is identical to signed magnitude and 1's complement, but the negative representation is different.

`int, short int`

1. The range of data of type `int` (32-bits) is `-2147483648` to `2147483647`.
2. The range of `short int` (16-bits) is `-32768` to `32767`.
3. The range of `long long int` (64-bits) is `-9223372036854775808` to `9223372036854775807`.

Decimal Value	*Bit String*				*Bit String*				*Decimal Value*
	b_3	b_2	b_1	b_0	b_3	b_2	b_1	b_0	
$+7$	0	1	1	1	1	0	0	0	-8
$+6$	0	1	1	0	1	0	0	1	-7
$+5$	0	1	0	1	1	0	1	0	-6
$+4$	0	1	0	0	1	0	1	1	-5
$+3$	0	0	1	1	1	1	0	0	-4
$+2$	0	0	1	0	1	1	0	1	-3
$+1$	0	0	0	1	1	1	1	0	-2
0	0	0	0	0	1	1	1	1	-1

Signed Magnitude, 1's and 2's Complements

Decimal	Sig. Mag.	1's Compl.	2's Compl.
-0	1000	1111	
-1	1001	1110	1111
-2	1010	1101	1110
-3	1011	1100	1101
-4	1100	1011	1100
-5	1101	1010	1011
-6	1110	1001	1010
-7	1111	1000	1001
-8			1000

Interpretation of Bits

Consider a 8-bit 2's complement number. If the number is positive e.g. 01101010, the value is as usual

$2^6 + 2^5 + 2^3 + 2^1 = 64 + 32 + 8 + 2 = 106$ (decimal).

Interpretation of Bits

If the number is negative e.g. 11101010, its value is $-$ (10000 0000 $-$ 1110 1010) i.e.

$$
\begin{aligned}
& -[2^8 - (2^7 + 2^6 + 2^5 + 2^3 + 2^1)] \\
= & -2^7 + 2^6 + 2^5 + 2^3 + 2^1 \\
= & -22(\text{in decimal}).
\end{aligned}
$$

Weight of msb is $-$ve

Sign Bit Extension

-39	25
1011001	0011001
11011001	00011001
111011001	000011001
1111011001	0000011001
11111011001	00000011001
111111011001	000000011001

2's Complement Addition

	Binary		Decimal
	0011		3
+	0010	+	2
	0101		5

	Binary		Decimal
	1101		−3
+	1110	+	−2
~~1~~	1011		−5

	Binary		Decimal
	0011		3
+	1011	+	−5
	1110		−2

	Binary		Decimal
	0100		4
+	0101	+	5
	1001		−7

Overflow

2's Complement Addition

	1101		-3
$+$	1010	$+$	-6
1	0111		7

Overflow

There may be carry-out without overflow. There may not be overflow even if there is carry-out.

`int` in Your Machine

The `int` in your machine (gcc-Linux on Pentium) is a 32-bit 2's complement number. Its range is -2147483648 to $+2147483647$. If one (1) is added to the largest positive number, the result is the smallest negative number.

0111 1111 1111 1111 1111 1111 1111 1111 +1	2147483647
1000 0000 0000 0000 0000 0000 0000 0000	−2147483648

10's Complement Number

The 2's complement numeral is nothing special. We can use radix-complement numerals for any radix to represent signed numbers without importing any new sign symbol. We consider radix-complement decimal or 10's complement numerals.

3-digit 10's Complement Numeral

There are one thousand patterns (000 to 999) with three decimal digits, we interpret them in the following way:

- If the most significant digit is any one of $\{0, 1, 2, 3, 4\}$, the denotation is a usual positive number e.g. 341 is same as usual decimal 341.

3-digit 10's Complement Numeral

- If the most significant digit is any one of $\{5, 6, 7, 8, 9\}$, the number is treated as a negative number (n).
- If n is a 10's complement number, $-n$ is obtained by ordinary decimal subtraction $1000 - n$.

10's Complement Numeral

Consider the 3-digit 10's complement numeral 725. It is a negative number whose magnitude is $1000 - 725 = 275$. The range of numbers represented in 3 digits is $-10^3/2$ to $+10^3/2 - 1$. In n-digits the range is $-10^n/2$ to $+10^n/2 - 1$.

Addition of 10's Complement Numeral

$$\begin{array}{rr} & 127 \\ + & 205 \\ \hline & 332 \end{array} \qquad \begin{array}{rl} & 127 \\ + & 821\ (-179) \\ \hline & 948\ (-52) \end{array} \qquad \begin{array}{lr} & 821\ (-179) \\ + & 753\ (-247) \\ \hline \not{1} \leftarrow & 574\ (-426) \end{array}$$

$$\begin{array}{rr} & 427 \\ + & 305 \\ \hline & 732 \end{array} \qquad \text{overflow} \qquad \begin{array}{lr} & 521\ (-179) \\ + & 753\ (-247) \\ \hline \not{1} \leftarrow & 274 \end{array}$$

Digit Extension

3-digit	4-digit	5-digit	Decimal value
234	0234	00234	234
721	9721	99721	−279