École Polytechnique Fédérale de Lausanne

Fundamentals of Digital Systems

Notes taken during Prof. Mirjana Stojilovic's lectures at EPFL in Spring 2026.

Elie Baier (397222)
Computer Science Bachelor
February 16, 2026
Version: `723ebcf`

Contents

Chapter 1

Number Systems

1.1 Digital Representations

Definition 1.1.1 (Tuple).
In mathematics, a tuple is a finite ordered sequence of elements. An n-tuple is a sequence of n elements, where n is a non-negative integer.

In digital representations, a number is represented by an ordered n-tuple, where each element of the tuple is called a digit. The n-tuple is often referred to as a digit vector (or string of digits) and n is called the presicison of the representation.

1.1.1 Representation of Nonnegative Integers

Definition 1.1.2 (Digit-vector).
A digit-vector is an n-tuple of digits representing the integer x denoted as:

$$X = (X_{n-1}, X_{n-2}, \ldots, X_0)$$

where X_0 is the least significant digit (or low-order digit) and X_{n-1} is the most significant digit (or high-order digit).

The number system to represent the integer x consists of:

- The number of digits n
- A set D_i of possible values for each digit X_i
- A rule of interpretation that maps between the set of digit-vectors and the set of integers.

The size of the set of integers that can be represented by a number system is at most K elements, where K equals:

$$\prod_{i=0}^{n-1} |D_i|$$

Example. Let's consider the case where $n = 6$ and $D_i = \{0, 1, \ldots, 9\}$ of cardinality $|D_i| = 10$. The size of the set of integers that can be represented is at most:

$$K = \prod_{i=0}^{5} |D_i| = 10^6 = 1000000$$

Thus the number system can represent at most 1 million integers, which are the integers from 0 to 999999.

Definition 1.1.3 (Redundance and Non-redundance).
A number system is said to be redundant if there exist at least two distinct digit-vectors that represent the same integer. Otherwise, the number system is said to be non-redundant.

Definition 1.1.4 (Weighted (Positional) Number System).
A weighted number system is a non-redundant number system in which each digit position has a weight (or value) associated with it. It can be denoted as:

$$x = \sum_{i=0}^{n-1} X_i \cdot W_i$$

where $W = (W_{n-1}, W_{n-2}, \ldots, W_0)$ is the weight vector of size n of the number system.

Example. Let's consider the case where $n = 6$, the digit vector $X = (8, 5, 4, 7, 0, 3)$ and the weight vector $W = (10^5, 10^4, 10^3, 10^2, 10^1, 10^0)$. Thus we have:

$$x = 8 \cdot 10^5 + 5 \cdot 10^4 + 4 \cdot 10^3 + 7 \cdot 10^2 + 0 \cdot 10^1 + 3 \cdot 10^0 = 854703_{10}$$

Note that when the weight vector is of the form:

- $W_0 = 1$
- $W_i = W_{i-1} R_{i-1}$, for $1 \leq i \leq n-1$

it is said to be a radix number system.

Definition 1.1.5 (Radix Number System).
A radix number system is a weighted number system in which the weight vector is defined as follows:

- $W_0 = 1$
- $W_i = W_{i-1} R_{i-1}$, for $1 \leq i \leq n-1$

or equivalently:

$$W_i = \prod_{j=0}^{i-1} R_j$$

where $R = (R_{n-1}, R_{n-2}, \ldots, R_0)$ is the radix vector of size n of the number system.

Note that the radix can be fixed and thus the weight vector can be defined as:

$$W_i = R^i$$

In this case, the number system is called a fixed-radix number system. Otherwise, the number system is called a mixed-radix number system.

Example. Let's consider the weight vector in a fixed-radix number system with $r = 10$. We have:

$$W = (10^{n-1}, 10^{n-2}, \dots, 10, 1)$$

Thus:

$$x = \sum_{i=0}^{n-1} X_i \cdot 10^i$$

which is the familiar decimal number system.

Note that almost all common number systems are fixed-radix number systems, such as the binary number system (with $r = 2$), the octal number system (with $r = 8$) and the hexadecimal number system (with $r = 16$).

Example. Let's consider a mixed-radix number system, like the time system. The radix vector is $R = (24, 60, 60)$. Thus the weight vector is:

$$W = (3600, 60, 1)$$

Definition 1.1.6 (Canonical Number System).
A canonical number system is a radix number system in which the digit set D_i for each digit position i is defined as:

$$D_i = \{0, 1, \dots, R_i - 1\}$$

where $|D_i| = R_i$ the corresponding radix of the digit position i.

Example. In a n fixed-radix-r number system, the digit set for each digit position is:

$$D_i = \{0, 1, \dots, r - 1\}$$

and thus the range of integers that can be represented is from 0 to $r^n - 1$.

Definition 1.1.7 (Conventional Number System).
A conventional number system is a canonical number system in which the radix is fixed and positive.

Note that these are by far the most commonly used number systems.

1.1.2 Transformation of Nonnegative Numbers to and from Decimal

The conversion table (up to 15) for the most commonly used number systems is given below:

Decimal	**Binary**	**Octal**	**Hexadecimal**
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
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

Note that generally in programming languages the prefix 0x is used to denote hexadecimal numbers.

Example. Let's convert 10011_2 to decimal. We have:

$$10011_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 19_{10}$$

Let's now convert 179_{10} to binary. We have:

$$\begin{aligned}
179 \div 2 &= 89 \text{ remainder } 1 \\
89 \div 2 &= 44 \text{ remainder } 1 \\
44 \div 2 &= 22 \text{ remainder } 0 \\
22 \div 2 &= 11 \text{ remainder } 0 \\
11 \div 2 &= 5 \text{ remainder } 1 \\
5 \div 2 &= 2 \text{ remainder } 1 \\
2 \div 2 &= 1 \text{ remainder } 0 \\
1 \div 2 &= 0 \text{ remainder } 1
\end{aligned}$$

Thus we have:

$$179_{10} = 10110011_2$$

Example. Similarly, we can convert 751_{10} to octal. We have:

$$\begin{aligned} 751 \div 8 &= 93 \text{ remainder } 7 \\ 93 \div 8 &= 11 \text{ remainder } 5 \\ 11 \div 8 &= 1 \text{ remainder } 3 \\ 1 \div 8 &= 0 \text{ remainder } 1 \end{aligned}$$

Thus we have:

$$751_{10} = 1357_8$$

1.1.3 Octal or Hexadecimal to and from Binary

Definition 1.1.8 (Bit-Vector Representation).
A bit-vector representation of a number is a binary representation of the number in which each digit of the original number is represented by a fixed number of bits (binary digits).

Note that for powers of 2 radix, a digit d is represented by a bit-vector $(d_{k-1}, \dots, d_0)$ where $k = \log_2 r$ bits such that:

$$d = \sum_{i=0}^{k-1} d_i \cdot 2^i$$

For example, the digit D_{16} in the hexadecimal number system is represented by a 4-bit binary vector 1101_2.

Example. Let's convert 10000100110_2 to octal. Since $k = \log_2 8 = 3$, we group the bits into groups of 3 starting from the right:

$$10000100110_2 = 10\ 000\ 100\ 110_2 = 010\ 000\ 100\ 110_2 = 2046_8$$

Note that we added leading zeros to the leftmost group to make it a full group of 3 bits.

Example. Let's convert 2543_8 to binary. Since $k = \log_2 8 = 3$, each octal digit is represented by a 3-bit binary vector:

$$2543_8 = 010\ 101\ 100\ 011_2 = 1010111000011_2$$

The same process can be applied to convert between hexadecimal and binary, but with groups of 4 bits instead of 3 bits.

1.2 Representation of Signed Integers

Definition 1.2.1 (Sign-and-Magnitude (SM) Representation).
A signed integer is represented by a pair:

$$(x_s, x_m)$$

where x_s is the sign bit and x_m is the magnitude of the integer. The sign bit is typically defined as:

$$x_s = \begin{cases} 0 & \text{if } x \geq 0 \\ 1 & \text{if } x < 0 \end{cases}$$

and the magnitude can be represented as any nonnegative integer.

Note that in this representation, the integer 0 can be represented in two ways: $(0, 0)$ and $(1, 0)$, which makes it a redundant representation.

Example. Traditionally, in an n-bit sign-and-magnitude representation, the most significant bit (MSB) is used as the sign bit and the remaining $n - 1$ bits are used to represent the magnitude. For example:

$$\begin{aligned} 5_{10} &= (0, 101)_2 = 0101_2 \\ -5_{10} &= (1, 101)_2 = 1101_2 \\ 0_{10} &= (0, 000)_2 \text{ or } (1, 000)_2 \end{aligned}$$

Such a representation have some advantages such as being symmetrical around zero, the magnitude lies between $-(2^{n-1} - 1)$ and $2^{n-1} - 1$, and being easy to understand. However, it has some disadvantages such as having complex arithmetic operations and having two representations for zero.

1.2.1 True-and-Complement Representation

Definition 1.2.2 (True-and-Complement (TC) Representation)**.**
A signed integer is represented by a pair:

$$(x_s, x_m)$$

where x_s is the sign bit and x_m is the magnitude of the integer. The sign bit is typically defined as:

$$x_s = \begin{cases} 0 & \text{if } x \geq 0 \\ 1 & \text{if } x < 0 \end{cases}$$

and the magnitude can be represented as any nonnegative integer. However, in this representation, the magnitude of negative integers is represented by the complement of their absolute value.