Number Systems

DR. TAREK A. TUTUNJI
PHILADELPHIA UNIVERSITY, JORDAN

Number Systems

- Programmable controllers use **binary** numbers in one form or another to represent various codes and quantities.
- Every number system has a base.
- The **base** of a number system determines the total number of unique symbols used by that system.

Decimal Number System

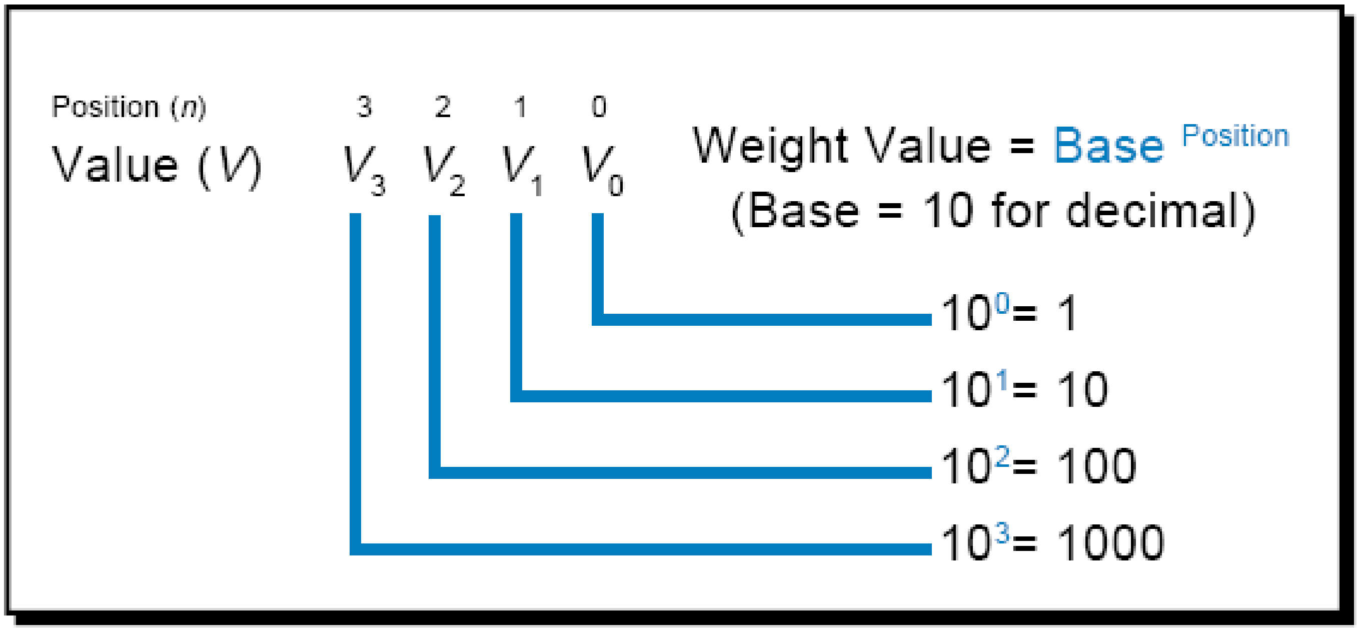
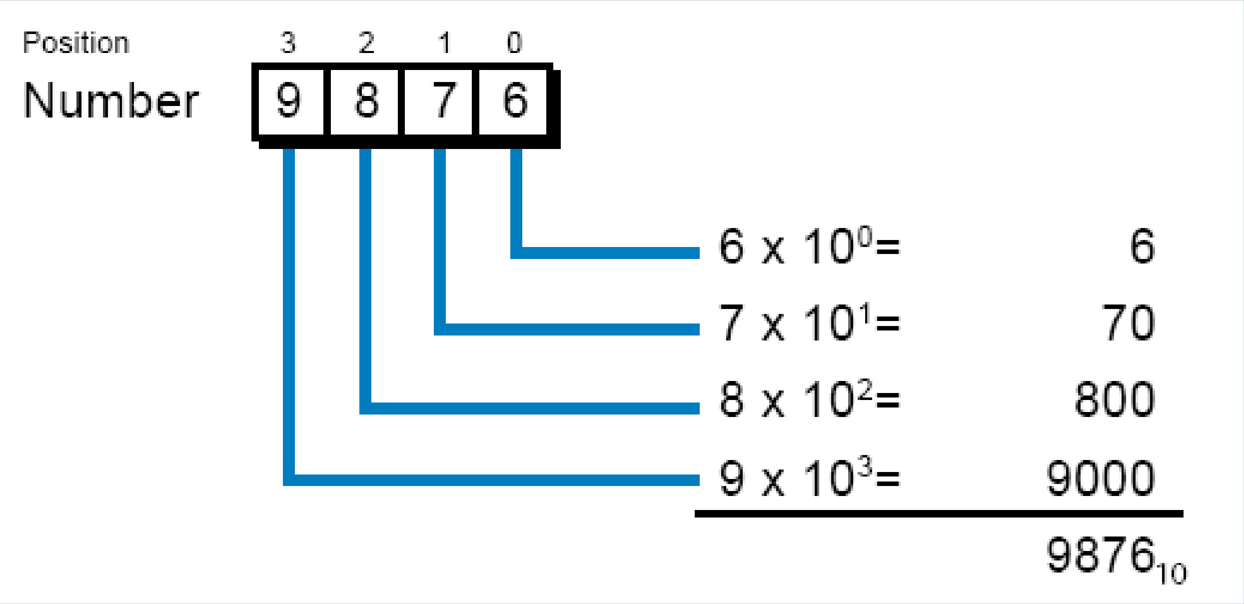


Figure 2-2. Weighted values.

Decimal Number System



Binary Number System

- The **binary number system** uses the number 2 as the base. Thus, the only allowable digits are:

  0 (Off) and 1 (On).

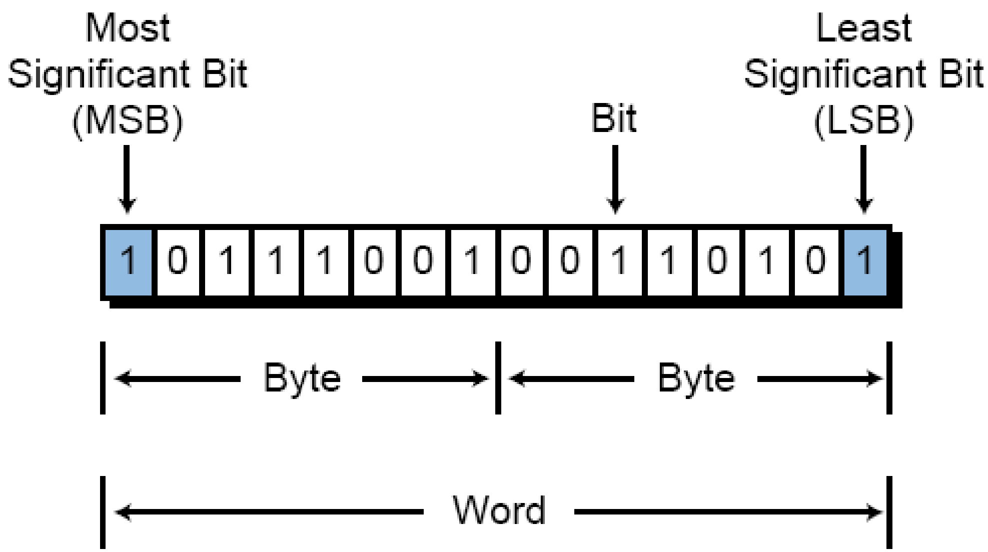
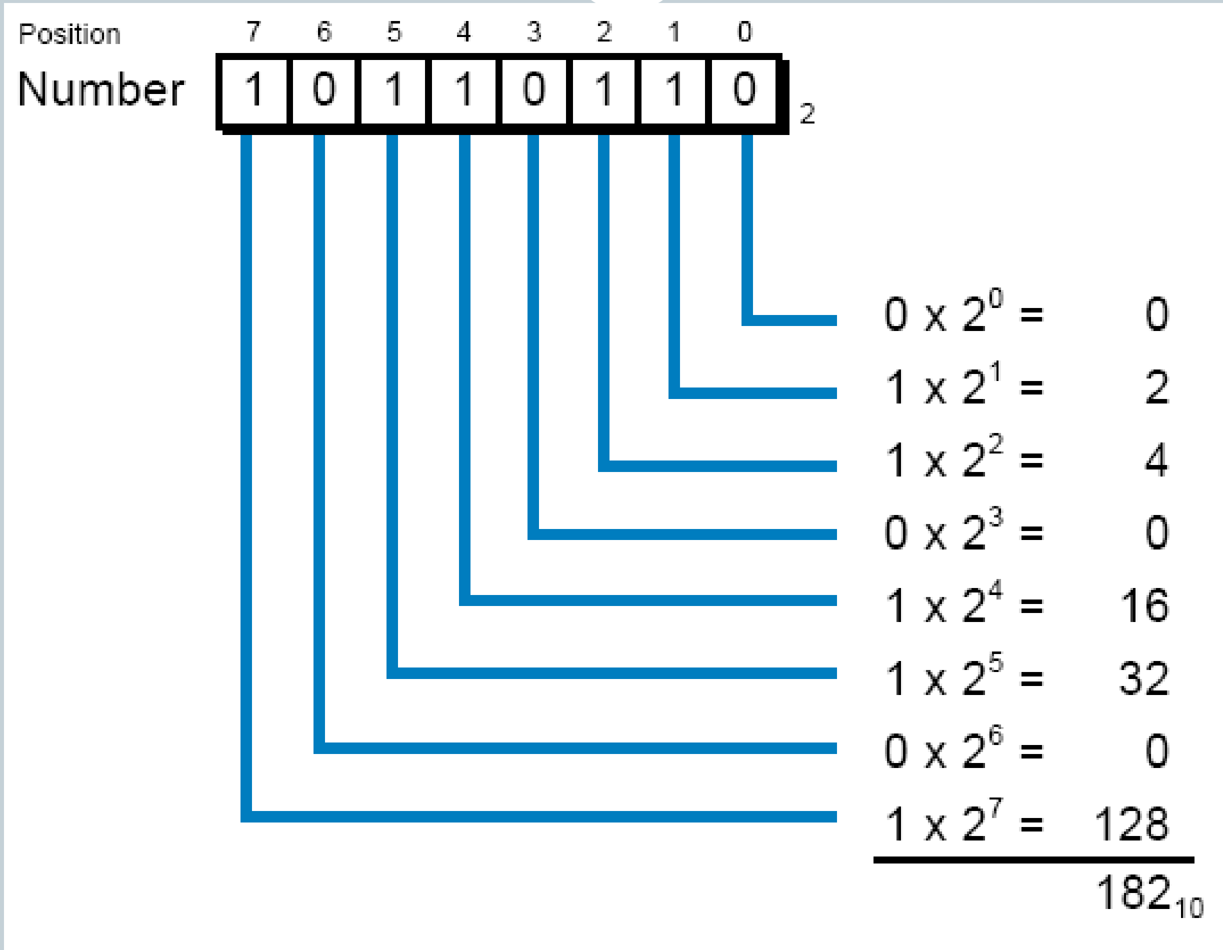


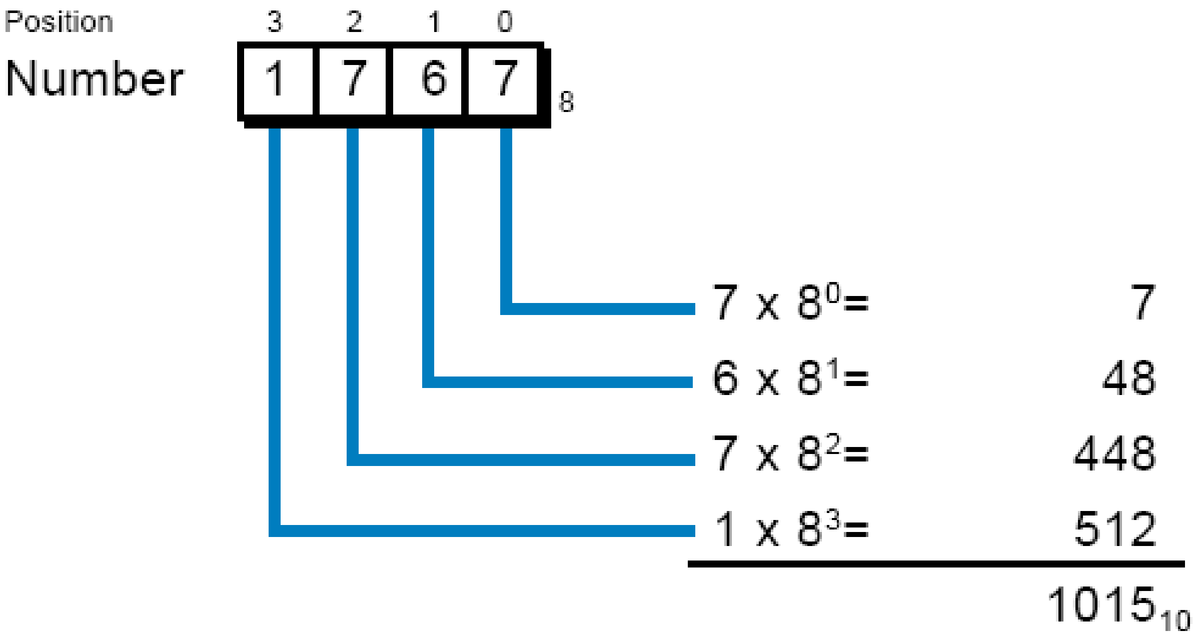
Figure 2-4. One word, two bytes, sixteen bits.

Binary Number System



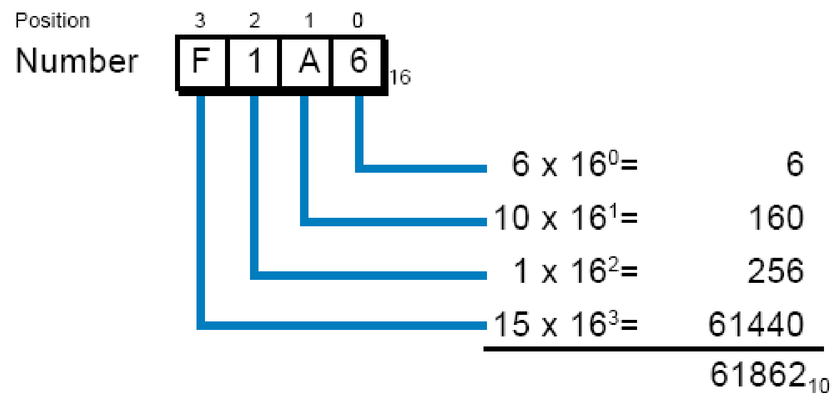
Octal Number System

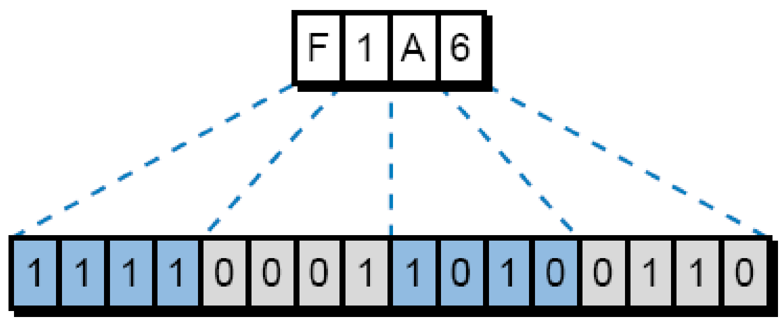
- The octal number system uses the **number 8 as its base**, with its eight digits being 0, 1, 2, 3, 4, 5, 6, and 7.



Hexadecimal Number System

- The **hexadecimal (hex) number system** uses 16 as its base.





Hexadecimal Number System

Binary	Decimal	Hexadecimal
0	0	0
1	1	1
10	2	2
11	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

Table 2-3. Binary, decimal, and hexadecimal counting.

Number Conversions

- To convert a decimal number to its equivalent in any base, you must perform a series of divisions by the desired base.
- The conversion process starts by dividing the decimal number by the base.
  - If there is a remainder, it is placed in the **least significant digit** (right-most) position of the new base number.
  - If there is no remainder, a 0 is placed in least significant digit position.
- The result of the division is then brought down, and the process is repeated until the final result of the successive divisions is 0.

Number Conversion Example: convert decimal to binary

- The binary equivalent of the decimal number 35 is 100011.

Division	Remainder
35 ÷ 2 = 17	1
17 ÷ 2 = 8	1
8 ÷ 2 = 4	0
4 ÷ 2 = 2	0
2 ÷ 2 = 1	0
1 ÷ 2 = 0	1

Number Conversion Example: convert decimal to hexadecimal

Division	Remainder
1355 ÷ 16 = 84	11
84 ÷ 16 = 5	4
5 ÷ 16 = 0	5

- The hexadecimal equivalent of 1355_{10} is $54B_{hex}$

Negative Numbers

- Consider the decimal number 23, or binary: 10111_2
- What about -23?
  - If a minus sign is placed in front of the number, as we do with decimal numbers: $-(10111)_2$
- This method is suitable for us, but it is impossible for programmable controllers and computers to interpret, since the only symbols they use are binary 1s and 0s.
- Therefore, two's compliment is used.

Two's Compliment

- The **two's complement** uses an extra digit to represent the sign.
- In the two's complement computation, each bit (from right to left) is inverted only after the first 1 is detected.
- Let's use the number +22 decimal as an example:
  - $+22_{10} = \mathbf{0}\ 10110_2$
- Its two's complement would be:
  - $-22_{10} = \mathbf{1}\ 01010_2$

Binary Codes

- An important requirement of programmable controllers is communication with various external (I/O) devices.
- This input/output function involves the transmission, manipulation, and storage of binary data that, at some point, must be interpreted by humans.
- Binary coding is the process of assigning a unique combination of 1s and 0s to each number, letter, or symbol that must be represented.
- The most common codes used in the industry are:
  - ASCII
  - BCD
  - Gray

ASC II

- **Alphanumeric codes** are used when information processing equipment, such as printers and cathode ray tubes (CRTs), must process the alphabet along with numbers and special symbols.

- These alphanumeric characters—26 letters (uppercase), 10 numerals (0-9), plus mathematical and punctuation symbols— can be represented using a 6-bit code (i.e., $2^6 = 64$ possible characters).

- The most common code for alphanumeric representation is **ASCII** (the American Standard Code for Information Interchange). Although a 6-bit code (64 possible characters) can accommodate the basic alphabet, numbers, and special symbols, standard ASCII character sets use a 7-bit code ($2^7 = 128$ possible characters), which provides room for lower case and control characters, in addition to the characters already mentioned.

BCD

- The **binary coded decimal (BCD)** system was introduced as a convenient way for humans to
  - Handle numbers that must be input to digital machines
  - Interpret numbers that are output from machines.

Decimal	Binary	BCD
0	0	0000
1	1	0001
2	10	0010
3	11	0011
4	100	0100
5	101	0101
6	110	0110
7	111	0111
8	1000	1000
9	1001	1001

Table 2-4. Decimal, binary, and BCD counting.

BCD

Figure 2-7. (a) A seven-segment indicator field device and **(b)** a thumbwheel switch.

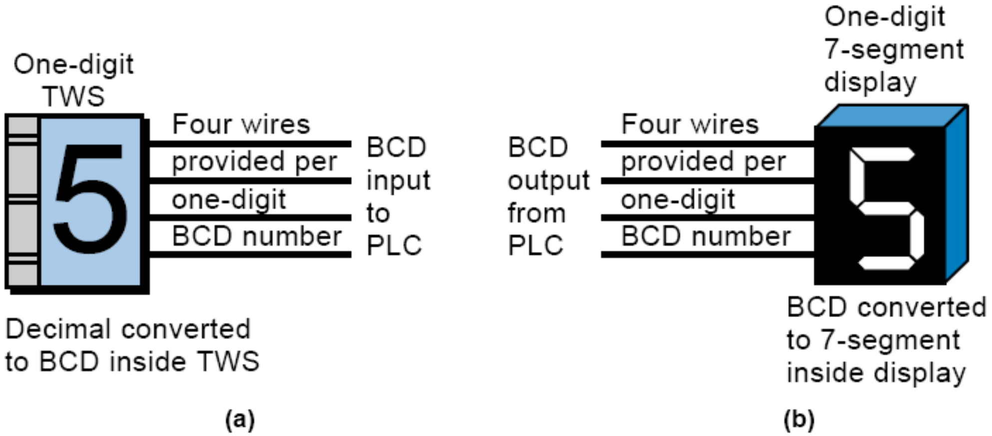


Figure 2-8. (a) Thumbwheel switch converts decimal numbers into BCD inputs for the PLC. **(b)** The seven-segment display converts the BCD outputs from the PLC into a decimal number.

Gray

- The **Gray code** is basically a modified binary code where only one bit changes as the counting number increases. This reduces the change of error. Therefore, it is suited primarily for position transducers

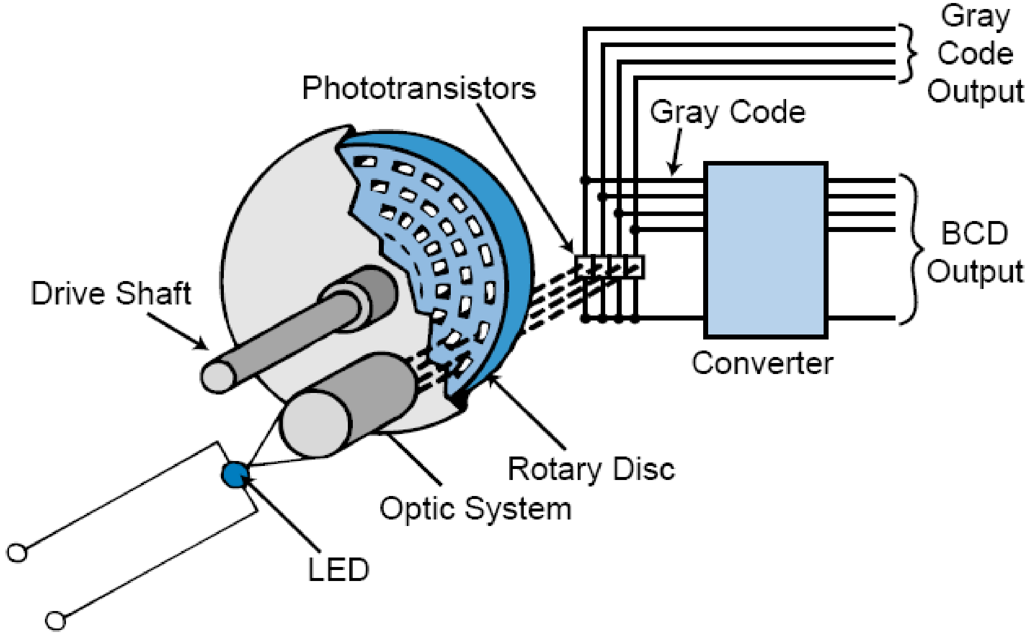


Figure 2-9. An absolute encoder with BCD and Gray outputs.

Gray

Gray Code	Binary	Decimal
0000	0	0
0001	1	1
0011	10	2
0010	11	3
0110	100	4
0111	101	5
0101	110	6
0100	111	7
1100	1000	8
1101	1001	9
1111	1010	10
1110	1011	11
1010	1100	12
1011	1101	13
1001	1110	14
1000	1111	15

Table 2-5. Gray code, binary, and decimal counting.

Register Word Format

- Programmable controller perform all internal operations in binary format using 1s and 0s. In addition, the status of I/O field devices is also read and written, in binary form, to and from the PLC's CPU.
- Generally, these operations are performed using a group of 16 bits.
- A PLC word (16-bits) is also called a **register.**

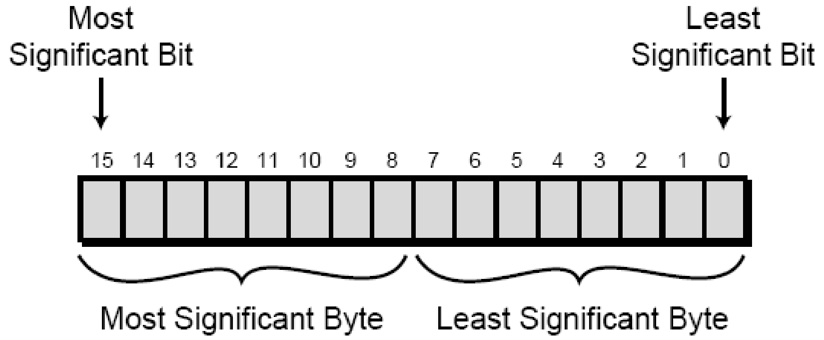


Figure 2-10. A 16-bit register/word.

REFERENCE: PROGRAMMABLE CONTROLLERS: THEORY AND IMPLEMENTATION BY BRYAN AND BRYAN