The first step to enter modern embedded system (ES) world is simply knows the differences between microcontroller (µC) and microprocessor (µP); that will be true if we assume that the most important elements in any ES are the processing and controlling units.

Microprocessor: is an integrated circuit that performs the central processing and internal functions of a computer (Computers).

So, from the previous definition the µP is considered a CPU, it has only three repeated functions:

Fetch → Decode → Execute

And this implies that if we decide to use a µP in our design, we should use many supporting elements such as main memory, program memory and I/O controller show figure 1.1. There are many manufacturers for microprocessor table1.1 lists some of them:

Table1.1: The most popular microprocessor units (CPUs).

Manufacturer	Products
Intel	8085, 8086, Pentium, Core 2 Due.
AMD	K6, Athlon/Duron, Athlon XP
Other	ARM, VIA

Figure 1.1 shows a simple abstraction view for a temperature controlling system that designed using a µP, note that each element is a stand-alone chip and all of them are a must for the system.

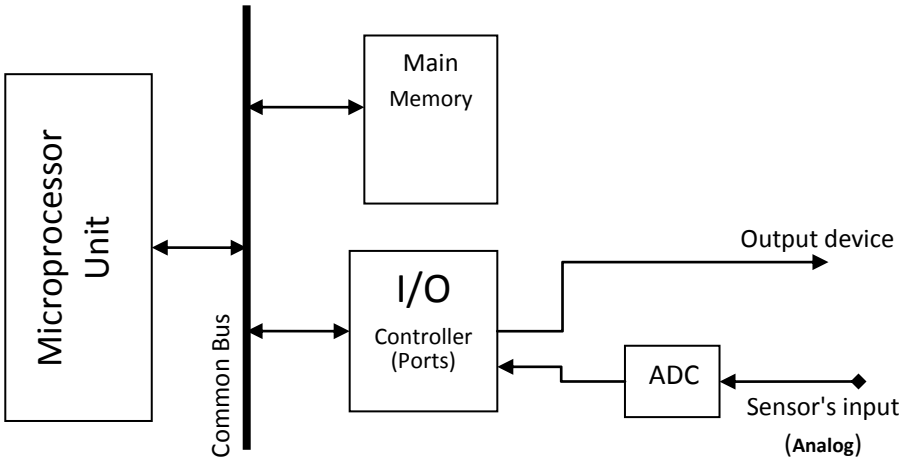

Figure 1.1: An embedded control system using µP unit.

Microcontroller: is a single integrated circuit that is designed to function as an embedded system (contains a CPU, RAM, ROM, I/O ports and timers).

What a nice sound, a single chip function as a whole ES, refer to the definition and read it again, a single die contains a CPU, RAM, ROM, I/O controller and these are the bases of any computer system i.e. CPU, Memory, I/O. So we can take the advantage of similarity between the µC and any computer system to give the µC a general and simple definition:

A microcontroller is a computer-on-a-chip, or a single- chip computer.

This means that if we use a µC in our design then we do not concern with the connections of main memory, I/O controller and program memory. All things that we must concern are writing system's software and how to make hardware connections between the µC as a whole and the external world!

To be more clearly, suppose that the previous temperature control system shown in figure1.1, is implemented using a µC has an on-chip ADC like PIC16F877A then all chips will be eliminated from the design, show figure 1.2.

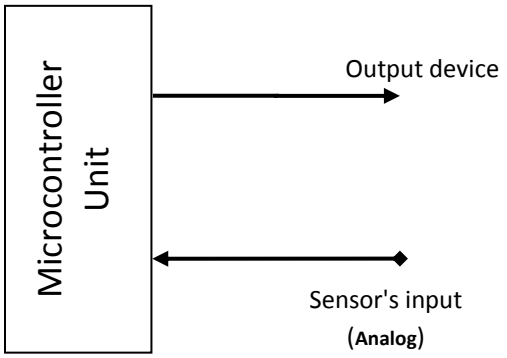

Figure 1.2: An embedded control system using µC unit.

As you see the design was greatly improved, not only in the simplicity of the circuitry; using µCs will affect cost, portability, size, power dissipation …..

Let us now re-arrange all things in graphical and a tabular form, figure 1.3 shows the internal architecture of a typical µC system, and its equivalent µP system, note that in the microcontroller side, all devices are packaged in the same chip, in contrast of µP system that each block is considered a stand-alone device.

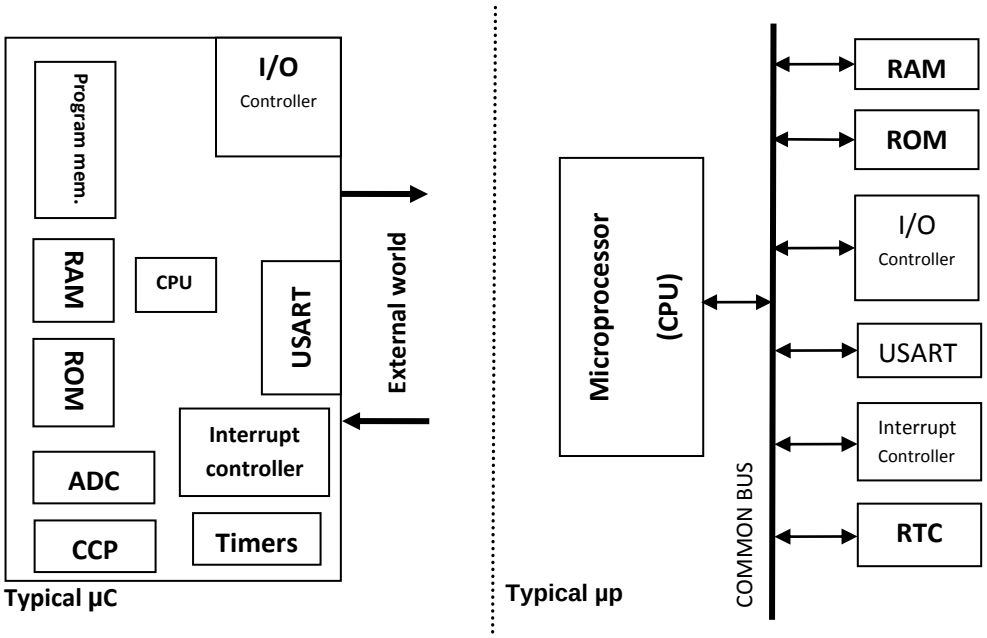

Figure 1.3: µC VS. µP, a graphical view.

Table 1.2 lists the most important differences.

Table 1.2: µC VS. µP.

COMPARISON FACTOR	MICROCONTROLLER	MICROPROCESSOR
Technology approach	RISC	CISC
Architecture	Harvard	Von Neumann
Memory size	Small (KB, a few MB)	Large (MB, GB, TB)
Applications	Simple	Complex, Advanced
Instruction cycle	Single	Multiple
Instruction format	Symmetric	Individual
Execution speed	Slow (KHz, <= 40MHz)	Fast (MHz, GHz)
Cost	Low	High
Power dissipation	Low	High

PIC MICROCONTROLLER UNIT (PIC MCU)

There are many companies in the world that considered the leader of a specific market in a specific product, like Intel (Pentium Microprocessors), Microsoft (Windows OS) and Nokia (Nokia Mobile phones).

Although there are many manufacturers for MCUs, look at table2.1; but the most familiar and well known is PIC's manufacturer; Microchip semiconductor.

Here we not concerned with how Microchip became the leader of MCUs market from a marketing view. We will look at the engineering side, and ask why do the majority of engineers, developers and hobbyists choose PIC in their projects?

Table 2.1: The most popular MCUs manufacturer.

Company	Products
Microchip	PIC12, PIC16, PIC18, dsPIC, PIC24
Intel, Philips	8051, P89C664
Atmel	ATtiny24,AT90S1200, ATmega
Motorola	68HC11

PIC16 identification:

PIC is generally assumed to mean **P**eripheral **I**nterface **C**ontroller, it comes with a variety of families; PIC10 and PIC12 (Base-line), PIC16 (Mid-range), PIC17 (High-end), PIC18 (enhancement), Finally PIC24 and dsPIC. Here we will deeply look at PIC16 family and highlights on other families' features if needed. PIC16F877A is our interest MCU in this family.

PIC16F877A:

PIC16F877A is a 40 pin chip, operating at a frequency up to 20MHz, it has five I/O ports A(6),B(8),C(8),D(8),E(3) mapping to 33 pins, the following points highlight the most important features:

1- 8Kx14bit Program memory space.
2- Five I\O ports.
3- 8 multiplexed analog ports, with internal 10bit resolution ADC.
4- 15 kinds of interrupts.
5- 256 Bytes of user EEPROM.
6- Two Capture\Compare\PWM modules (CCP).
7- Three timers with different capabilities.
8- RS-232, I^2C, and SPI interfaces (USART, MSSP).
9- 368B of RAM.
10- Wide operating frequency DC-20MHz.
11- Wide operating voltage 2.0v – 5.5v.

Note that a single pin can have many functions, for example pin2 can be a digital I/O (RA0) or analog input (AN0); the function of the pin will be controlled using software, figure 2.1 illustrates the PIC16F877A layout in more details.

PIC16F877A as other PIC families is implemented using RISC approach, with only 35 instructions; you can build great projects, security, control, talking with GSM system, linking to the internet, communicating with PCs and more.

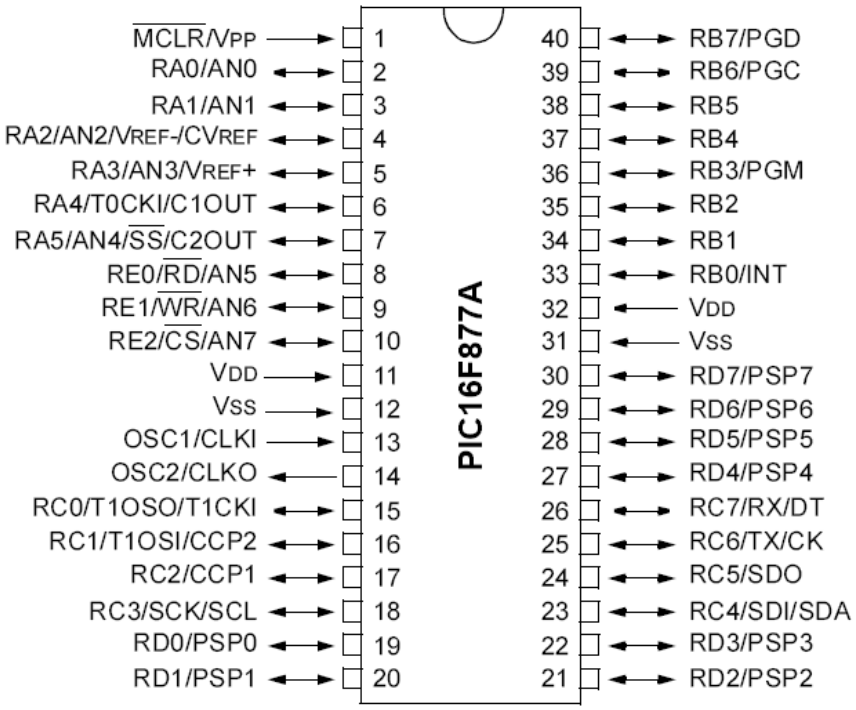

Figure 2.1: PIC16F877A pins layout.

Many engineers and developers choose it in their projects and designs for three main reasons:

1. PIC16F877A comes with a variety of embedded modules.
2. PIC16F877A is considered a low cost MCU.
3. It has wide supporting articles in the internet.

The **F** letter in its name stand for **F**lash technology, Flash EEPROM, a version of EEPROM memory, has become popular in microcontroller applications and is used to store the user program. Flash EEPROM is nonvolatile and usually very fast. The data can be erased and then reprogrammed using a suitable programming device thausands and thausands of times. Letter **A** at the end of PIC16F877**A** means that this MCU is an **A**dvanced and an improved version of a previous MCU i.e. PIC16F877.

With 8K program memory, PIC16F877A can store and run programs ranges from simple (a few lines), mid, up to complex programs with many handreds of lines, also 368B of general purpose registers (GPR) i.e. user RAM; this size of temporary storage aria can maintain and take complex operations either arithmetic or logic, float or integer, strings or characters.

Knowing the internal architecture of any MCU is a must if you want to use assembly for writing programs (in ES terminology called ***Firmware***), contrarily of using a high level language like C; a very little architecture knowledge is needed to write a perfect firmware.

Finally, all hardware, software aspects and the embedded modules of PIC16F877A will be taken and explained using examples and some projects.

PIC C Compiler

There are many compilers in many languages for PIC programming such as C, Pascal, BASIC; but the most popular language used in all MCUs programming is C. If you googled the net, you will find many C compilers for PIC MCUs from many companies, but the most popular is PIC C from CCS, it supports PIC12, 16, 18. Here we will learn how to write a body for any program.