Heaven's Light is Our Guide

Rajshahi University of Engineering & Technology

Department of Computer Science & Engineering

Course Code: CSE 2205
Course Title: Microprocessor, Microcontroller and Assembly Language

Assignment No. : 01
Assignment Title : Comparison between microprocessor and microcontroller

Submitted by:
Md. Mostafizur Rahaman
Roll: 2103034
Section: A
Session: 2021-22

Submitted to:
Md. Sozib Hossain
Lecturer
Dept. of Computer Science & Engineering, RUET

Date of Submission : 18 February 2025

Introduction:

Both microprocessors and microcontrollers are types electronic devices that come in the form of integrated circuits and are used in different modern electronic equipment such as computers, laptops, washing machines, air conditioners, and many other automated electronic gadgets [1].

Microprocessor:

A microprocessor is the predominant type of modern computer processor. It combines the components and function of a central processing unit (CPU) into a single integrated circuit (IC) or a few connected ICs [1]. The computer circuits consist of three parts: the central processing unit (CPU), the memory circuits, and the I/O circuits. In a microcomputer, the CPU is a single-chip processor called a microprocessor. So we can say, a microprocessor is the central processing unit of a computer system, designed to execute instructions and process data. It requires external components like memory (RAM, ROM) and input/output (I/O) peripherals to function. Microprocessors are commonly used in personal computers, servers, and high-performance systems. Some key features of microprocessor are:

1. Fetch: The microprocessor retrieves or fetches instructions from computer memory. The fetch process can be initiated by automatic or manual input.
2. Decode: The microprocessor "decode" the instructions, essentially interpreting the input or command into a request and instigating a specific process or computation.
3. Execute: The microprocessor performs the required or requested operation.
4. Store: The result of the execution is committed to the computer's memory.

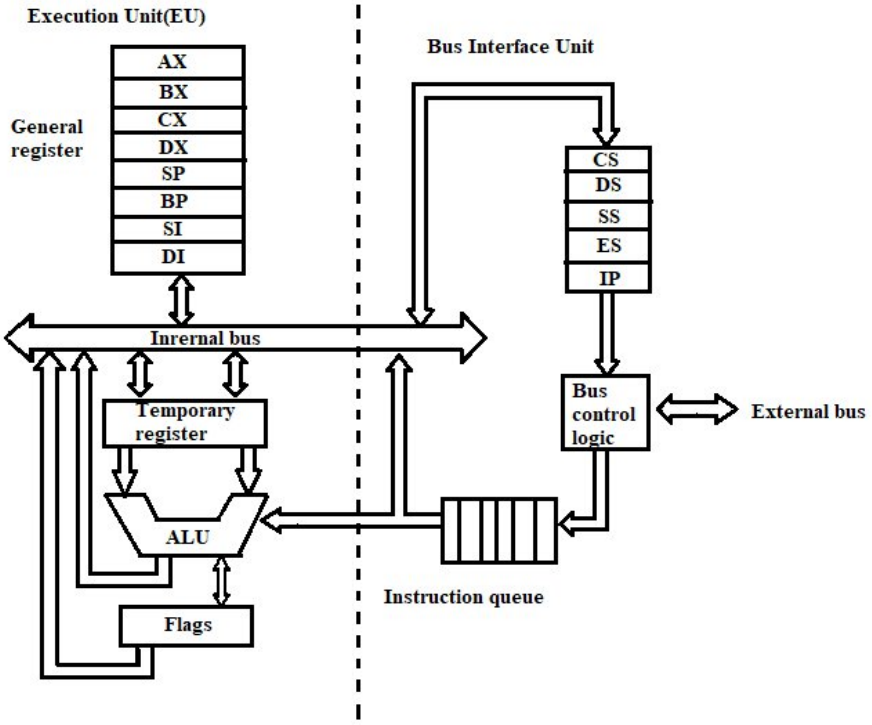
Microcontroller:

A microcontroller is system that has a CPU (a microprocessor) in addition to a fixed amount of RAM, ROM, I/O ports, and a timer all on a single chip. In other words, the processor, RAM, ROM, I//O ports, and timer are all embedded together on one chip; therefore, the designer cannot add any external memory, I/O, or timer to it. The fixed amount of on-chip ROM, RAM, and number of I/O ports in microcontrollers make them ideal for many applications in which cost and space are critical. Microcontrollers usually must have low-power requirements since many devices they control are battery-operated. Microcontrollers are used in many consumer electronics, car engines, computer peripherals, and test or measurement equipment.

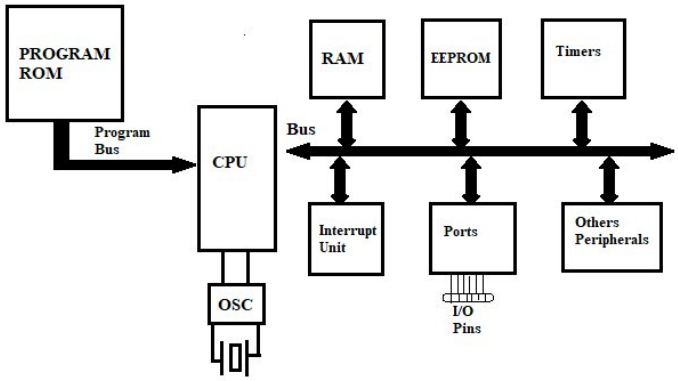
So both microprocessors and microcontrollers are types electronic devices that come in the form of integrated circuits and are used in different modern electronic equipment. Microprocessor is an electronic component that acts as a processing device in various computing systems such as computers, laptops, smartphones, etc. On the other hand, a microcontroller is a small microcomputer acts as a controlling device in different embedded systems such as washing machines, microwave ovens, etc.

Structural Comparison:

[2] An example of a microprocessor is Intel 8086. There are two main components: the execution unit and the bus interface unit. The execution unit contains a circuit called the arithmetic and logic unit (ALU). The ALU can perform arithmetic and logic operations. The data for the operations are stored in circuit called registers. A register is like a memory location except that we normally refer to it by a name rather than a number. The BIU facilitates communication between the EU and the memory or I/O circuits. It is responsible for transmitting addresses, data, and control signals on the buses. Its registers are named CS, ES, SS and IP, they hold addresses of memory locations. The diagram of an Intel 8086 microprocessor, [5]



An example of microcontroller is AVR (Alf and Vegard's RISC Processor). It is an 8-bit RISC single-chip microcontroller. It has on-chip program ROM, data RAM, data EEPROM, timers and I/O ports which follows Harvard architecture. It also has additional features like ADC, PWM, and different kinds of serial interface such as USART, SPI, I2C, CAN, USB, and so on. The ROM stores the microcontroller programs. The data is stored in RAM. AVR I/O pins range from 3 to 86, dependent on the package's pin count. The simplified view of an AVR microcontroller is, [6]



So, the structural comparison of microprocessor and microcontroller are [3],

Feature	**Microprocessor**	**Microcontroller**
CPU	Present	Present
Memory	External	Built-in(on-chip RAM and ROM)
I/O Ports	External	Built-in
Circuit complexity	The circuit is complex due to external connection	Microcontrollers are present on chip memory. The circuit is less complex
Power Consumption	Higher (requires more external components)	Lower (optimized for embedded system)
Number of Register	Microprocessor have less number of register	Microcontroller have more number of registers
Zero status flag	Microprocessors have a zero status flag	Microcontroller doesn't have a zero status flag
Applications	Computers, servers, high-end applications	Embedded systems, IoT, automation

Assembly Code:

There are so many differences in assembly code between microprocessor and microcontroller. The number of register in them is different, also there are I/O ports in microcontroller which is absent in microprocessor. Microcontroller can receive input and show output through those ports, so we have to write code for that too. There is a simple code of finding sum first 10 number in both microprocessor and microcontroller,

Microprocessor's code:

```
.MODEL SMALL
```

```
.STACK 100H
.CODE
MAIN PROC
   MOV BL,0     ;initializing sum to 0
   MOV BH,1      ;starting point of loop
   MOV CX,10    ;number of time loop will run

   LEVEL:
   ADD BL,BH   ;adding number
   INC BH          ;increasing the number
   LOOP LEVEL

   MOV AH,2        ;printing the sum
   MOV DL,BL
   INT 21H

   MAIN ENDP
END MAIN
```

Microcontroller's code:

```
LDI R16, 10    ;assigning immediate value 10
LDI R20, 0       ;initializing the sum to 0
AGAIN:
ADD R20, R16   ;adding sum
BRNE AGAIN     ;continuing the loop
OUT SPL, R20     ;move to output port address
```

If we compare the codes, we can see the microcontroller's code is smaller than the microprocessor's code. In x86 microcontroller MOV opcode was used to assign value to a register, in AVR microcontroller LDI was used to assign immediate value to register, but we can

only use register from R16 to R31 if we use LDI. For both code to add same instruction was used. To loop in x86 ‘LOOP’ opcode was used and in AVR ‘BRNE’ was used. For output, in AVR ‘OUT’ opcode was used. It move the output to a I/O ports address and in x86 the result is first moved to DL register and then the output was printed.

Conclusion:

A microprocessor is the central component of a computer system, responsible for executing operations using its Arithmetic Logic Unit (ALU) and generating output. However, it requires external components such as RAM, ROM, input/output ports, and other peripherals to function properly. This makes microprocessors ideal for general-purpose computing, where high processing power and multitasking capabilities are essential. They are commonly used in personal computers, laptops, servers, gaming consoles, and networking systems.

On the other hand, a microcontroller is a compact integrated device that contains a microprocessor along with built-in RAM, ROM, I/O ports, timers, and other peripherals within a single chip. This integration makes microcontrollers highly efficient for embedded systems that require real-time control, low power consumption, and cost-effectiveness. They are widely used in automobiles, home appliances, medical devices, consumer electronics, IoT devices, and robotics.

While microprocessors are optimized for complex computation and multitasking, microcontrollers excel in applications that demand real-time processing, efficiency, and ease of integration. Microcontrollers often support low-power modes to extend battery life in portable and IoT applications, whereas microprocessors are designed for high-performance computing with advanced processing capabilities.

References:

1. Introduction is collected from https://www.tutorialspoint.com/difference-between-microprocessor-and-microcontroller.
2. Definition source is https://www.ibm.com/think/topics/microprocessor and book “Assembly Language Programming and Organization of the IBM PC by Ytha Yu and Charles Marut”.
3. To gather information about the structural difference the books “Assembly Language Programming and Organization of the IBM PC by Ytha Yu and Charles Marut” and “The Avr Microcontroller and Embedded System” was used.
4. The table data was collected from the website “TutorialsPoint(https://www.tutorialspoint.com/difference-between-microprocessor-and-microcontroller)”.
5. The diagram is collected from the book “Assembly Language Programming and Organization of the IBM PC by Ytha Yu and Charles Marut”, page 8.

6. The microcontroller diagram is collected from the book “The Avr Microcontroller and Embedded System”, page 45.