Microprocessor

Falcuty of Mechanical Engineering,
HCMC University of Technology and Education

HCMC, 08/2022

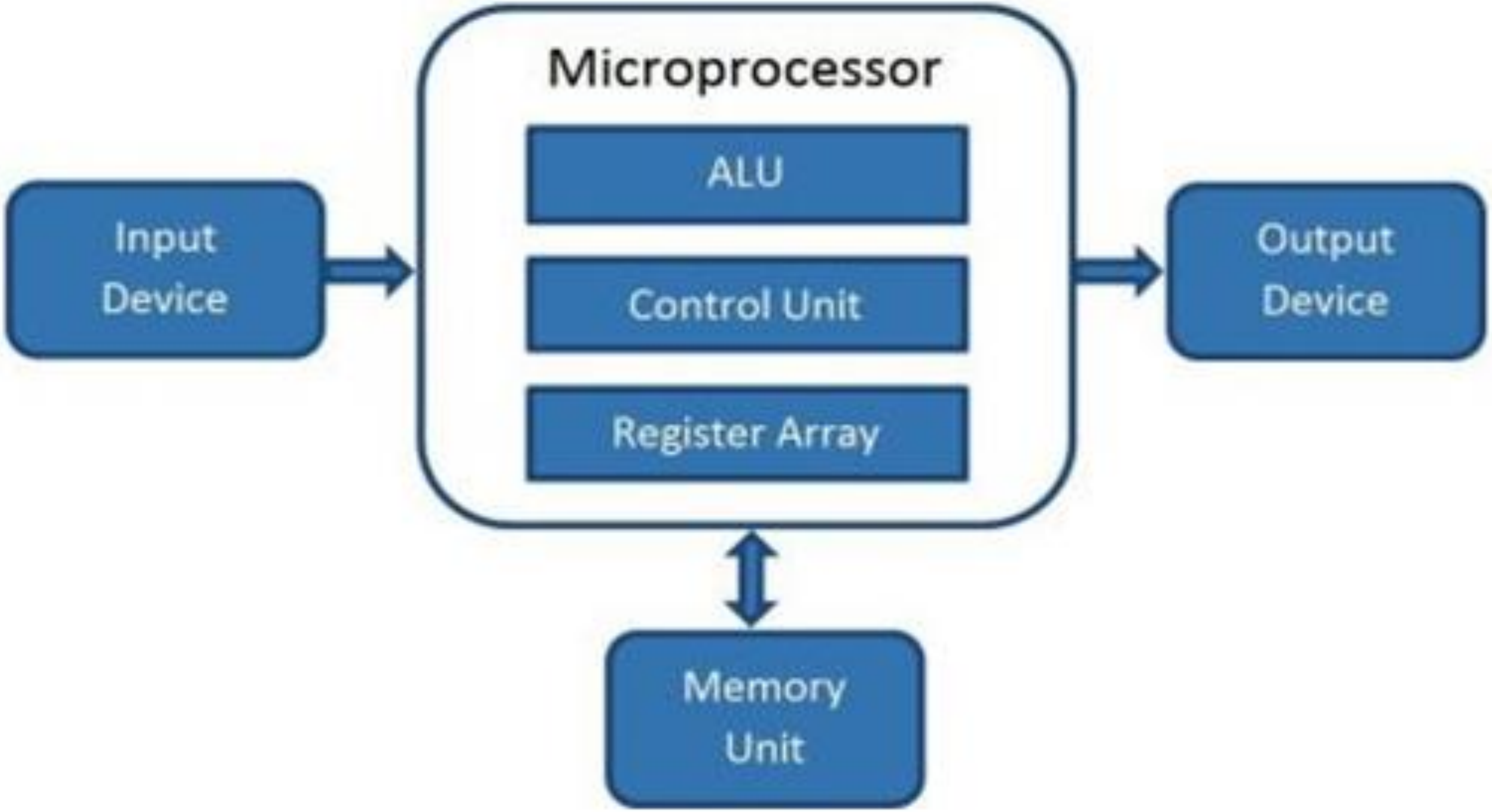
INTRODUCTORY CONCEPTS

https://en.wikipedia.org/wiki/List_of_Intel_processors

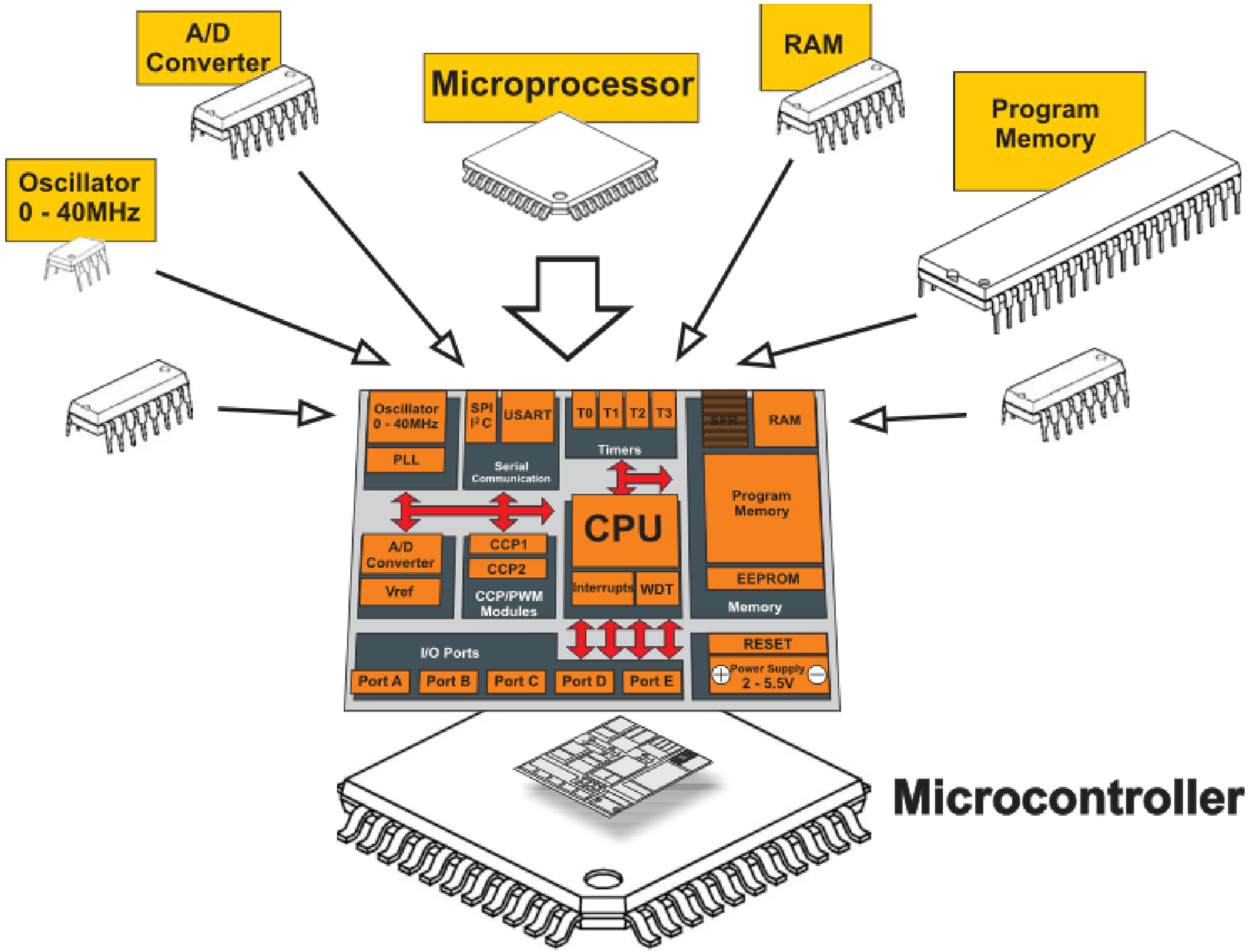
INTRODUCTORY CONCEPTS

Where we can find a microprocessor?

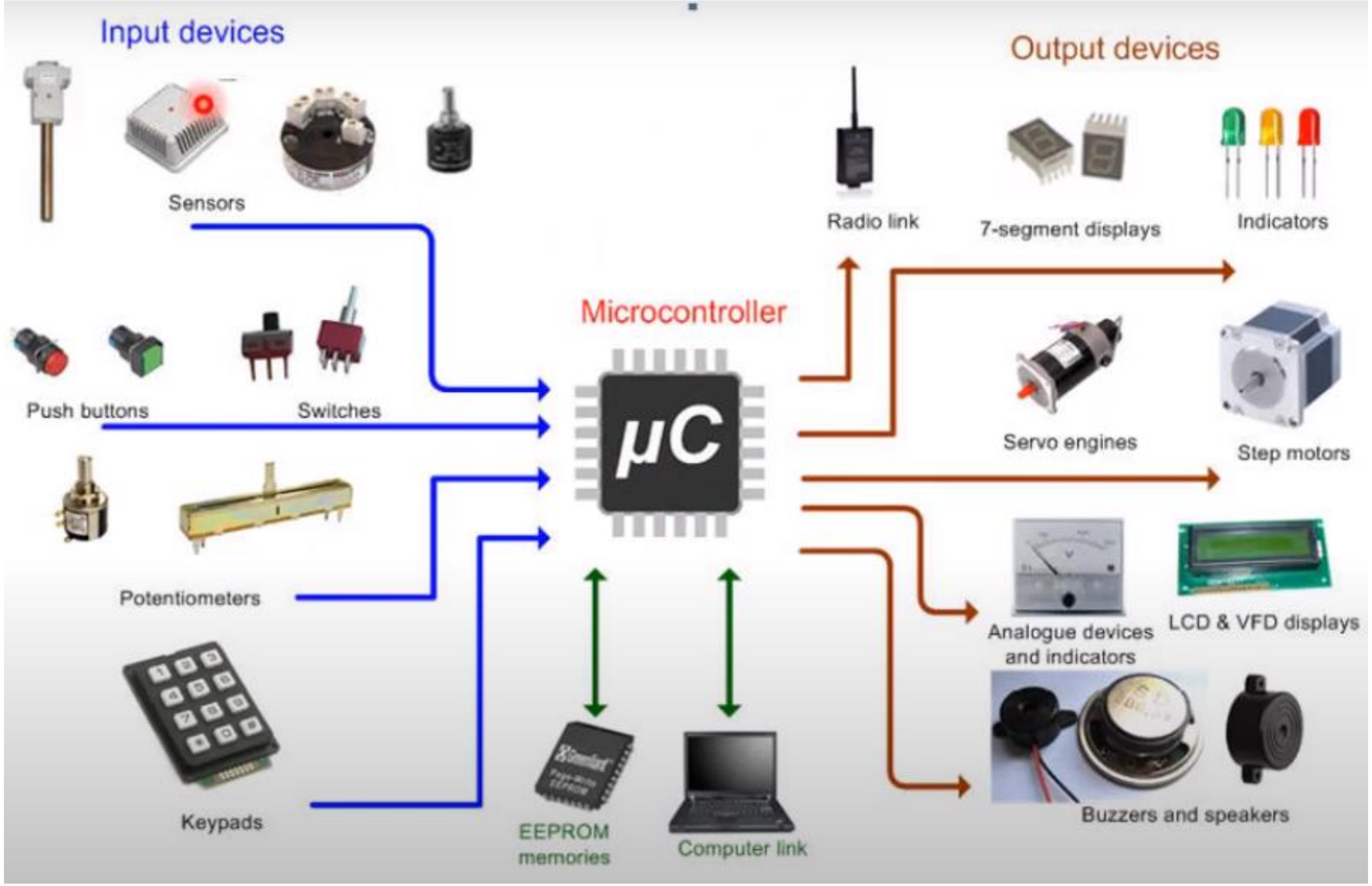
INTRODUCTORY CONCEPTS



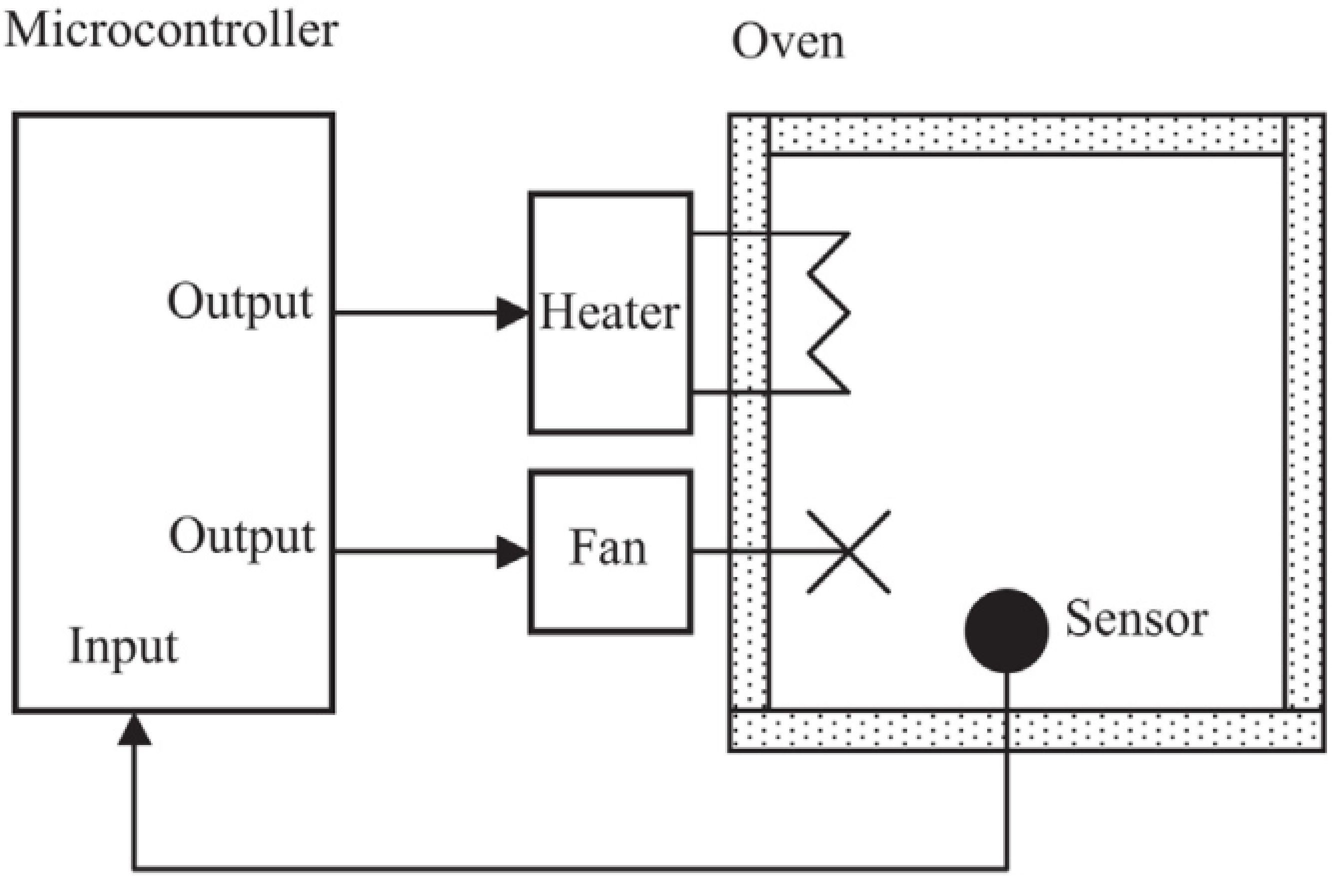
INTRODUCTORY CONCEPTS



INTRODUCTORY CONCEPTS

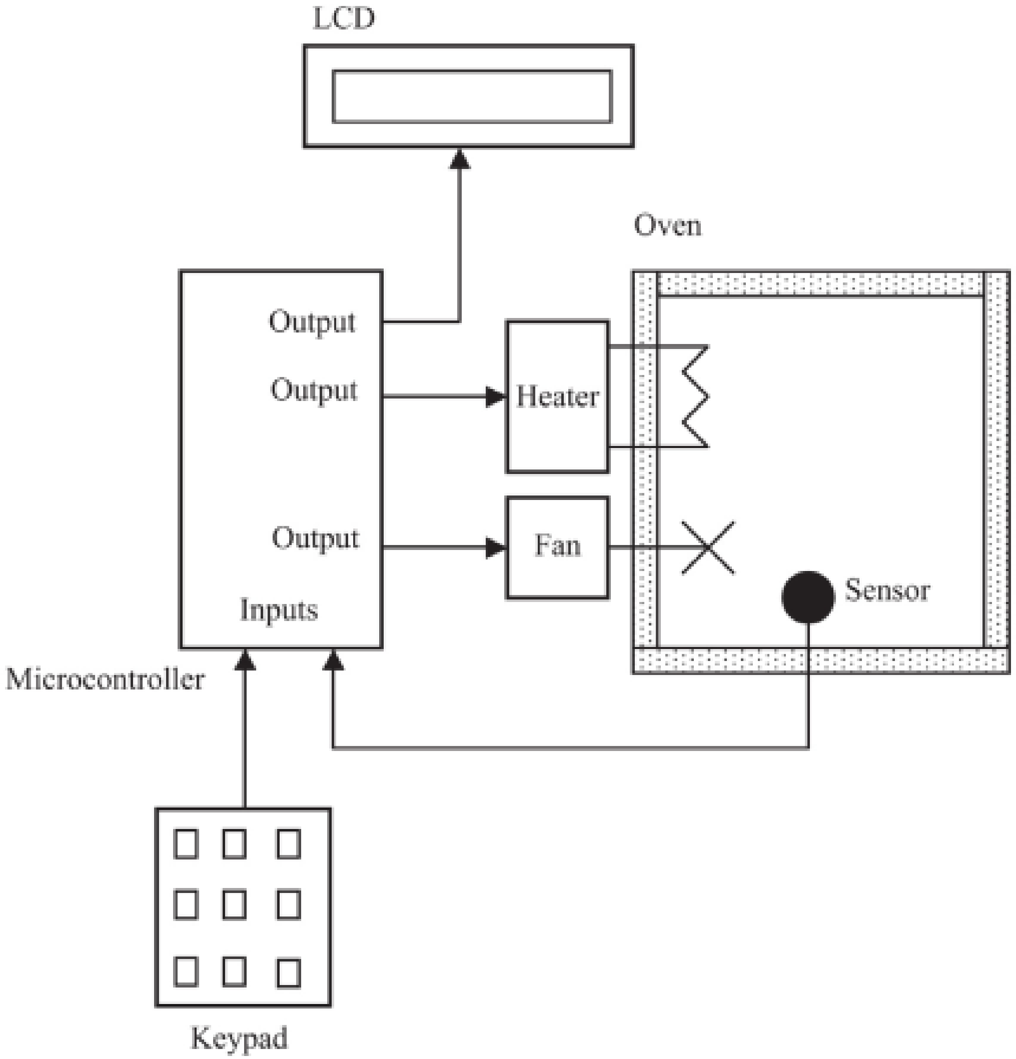


INTRODUCTORY CONCEPTS



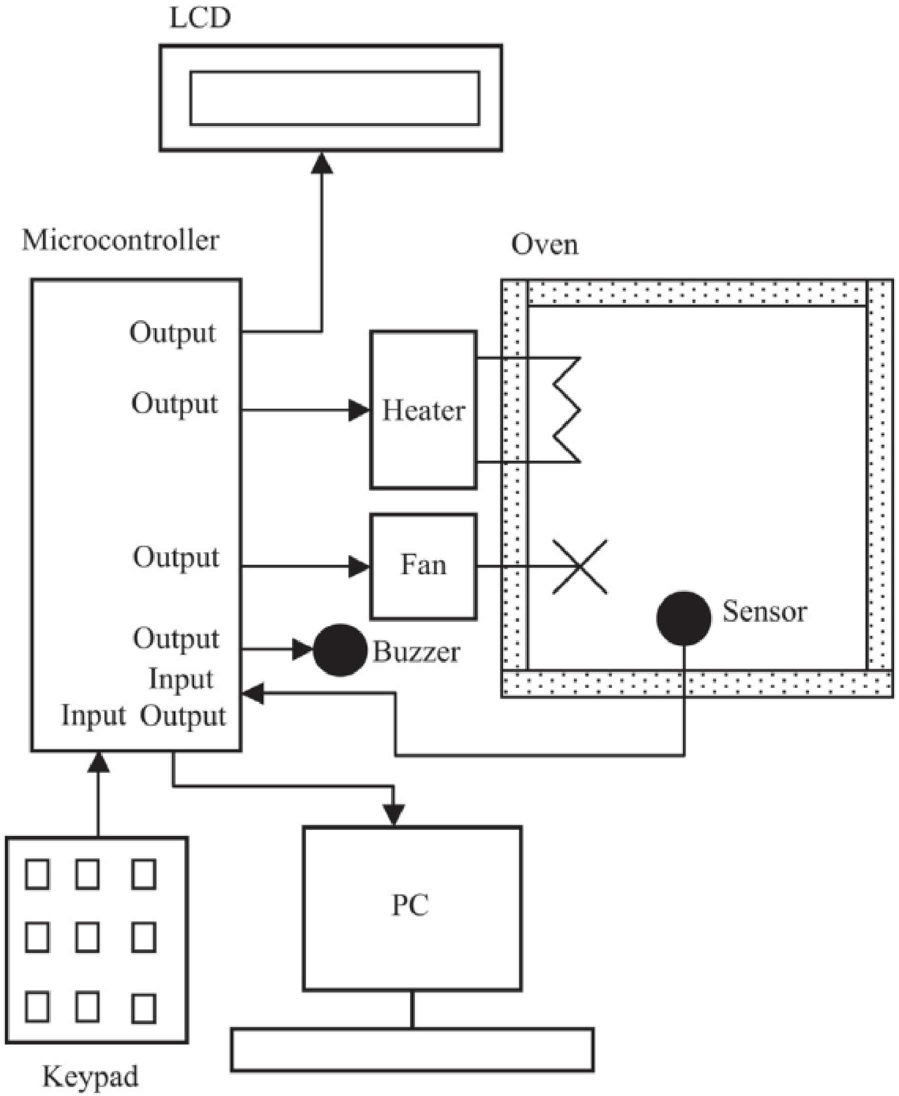
Microcontroller-based oven temperature control system.

INTRODUCTORY CONCEPTS



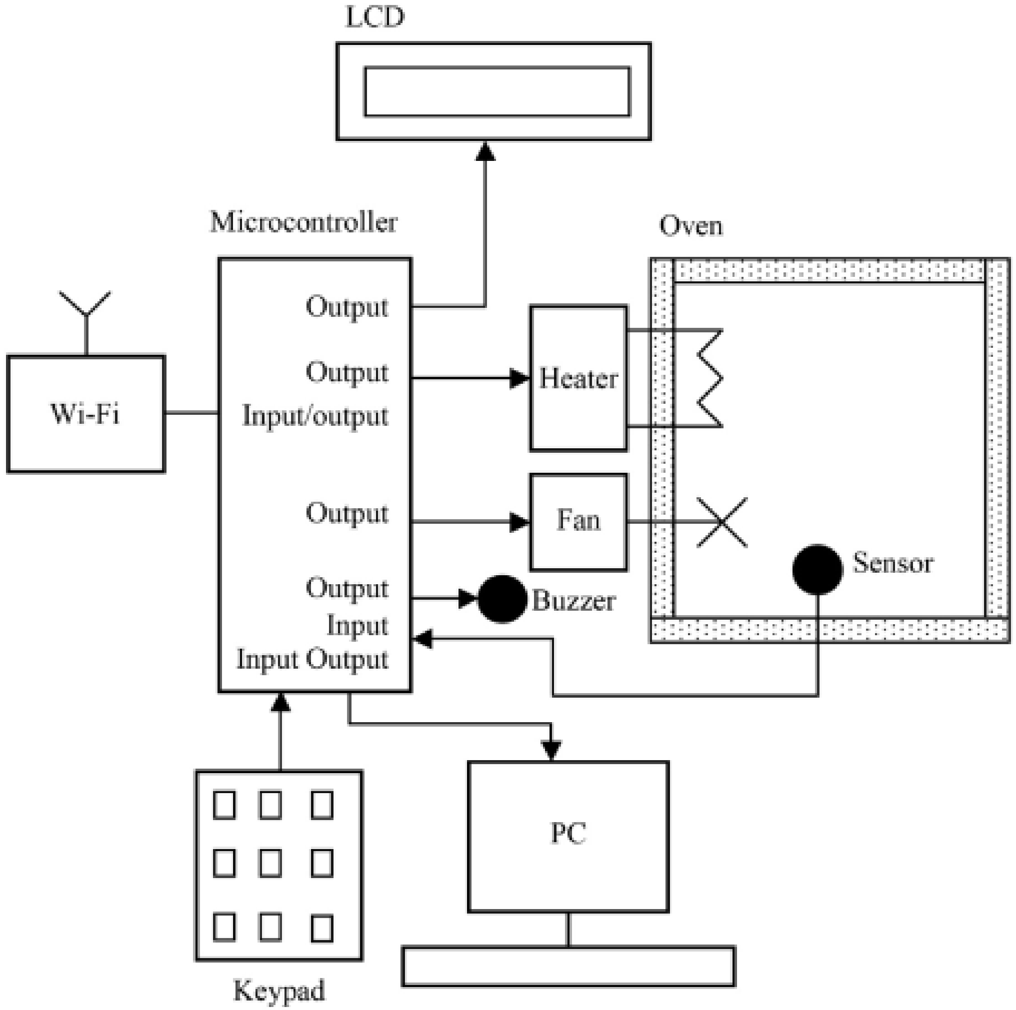
Temperature control system with a keypad and LCD.

INTRODUCTORY CONCEPTS



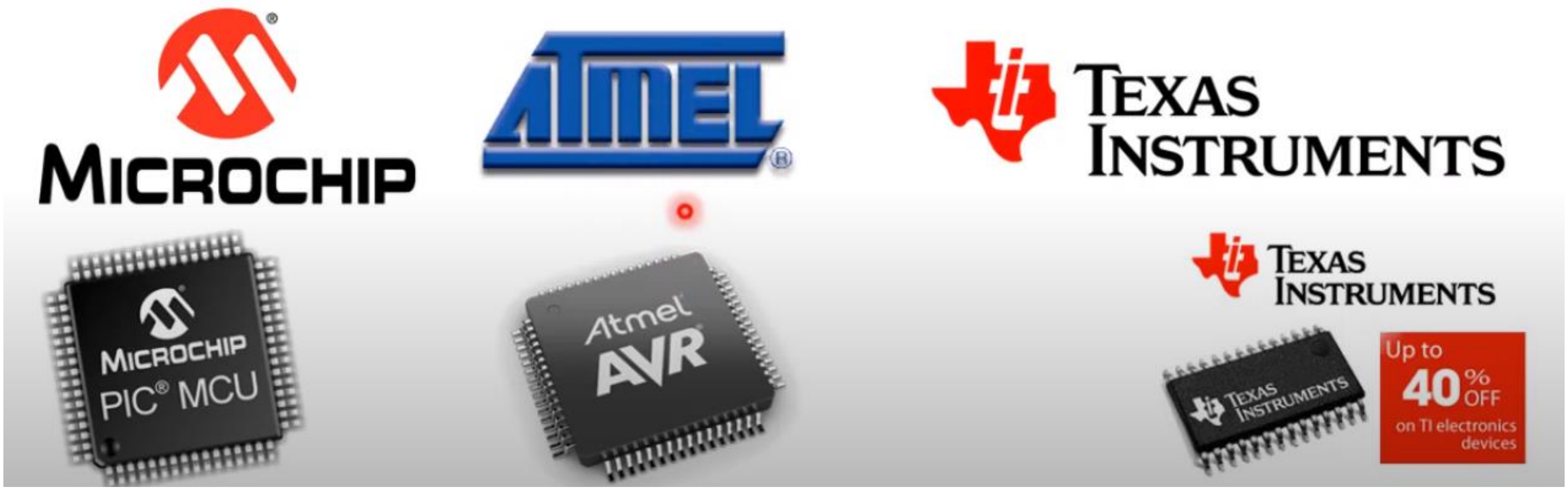
More sophisticated temperature controller.

INTRODUCTORY CONCEPTS



Temperature control system with Wi-Fi capability.

INTRODUCTORY CONCEPTS



INTRODUCTORY CONCEPTS

Main Difference between AVR, ARM, 8051 and PIC Microcontrollers

	8051	PIC	AVR	ARM
Bus width	8-bit for standard core	8/16/32-bit	8/32-bit	32-bit mostly also available in 64-bit
Communication Protocols	UART, USART,SPI,I2C	PIC, UART, USART, LIN, CAN, Ethernet, SPI, I2S	UART, USART, SPI, I2C, (special purpose AVR support CAN, USB, Ethernet)	UART, USART, LIN, I2C, SPI, CAN, USB, Ethernet, I2S, DSP, SAI (serial audio interface), IrDA
Speed	12 Clock/instruction cycle	4 Clock/instruction cycle	1 clock/ instruction cycle	1 clock/ instruction cycle
Memory	ROM, SRAM , FLASH	SRAM, FLASH	Flash, SRAM, EEPROM	Flash, SDRAM, EEPROM
ISA	CLSC	Some feature of RISC	RISC	RISC

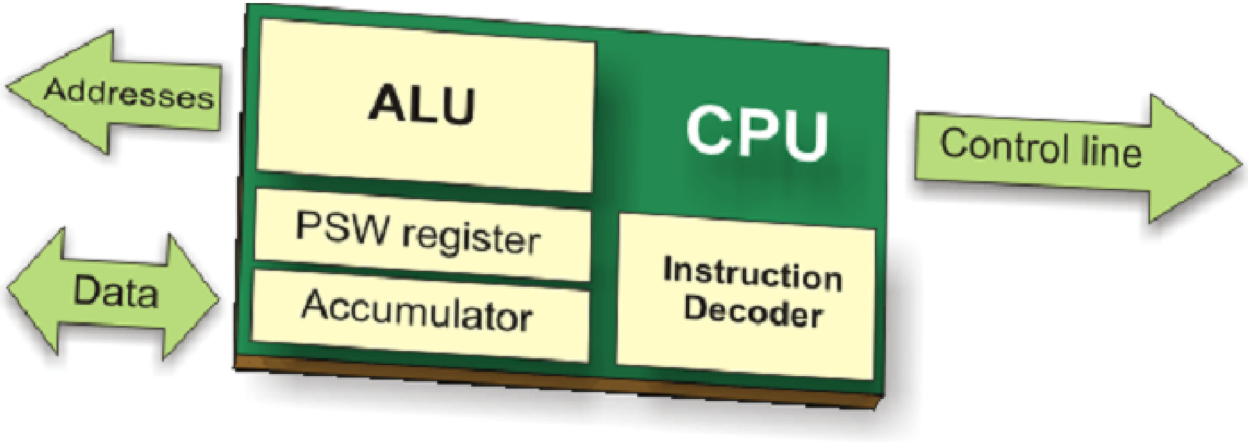
	8051	PIC	AVR	ARM
Memory Architecture	Harvard architecture	Von Neumann architecture	Modified	Modified Harvard architecture
Power Consumption	Average	Low	Low	Low
Families	8051 variants	PIC16,PIC17, PIC18, PIC24, PIC32	Tiny, Atmega, Xmega, special purpose AVR	ARMv4,5,6,7 and series
Community	Vast	Very Good	Very Good	Vast
Manufacturer	NXP, Atmel, Silicon Labs, Dallas, Cyprus, Infineon, etc.	Microchip Average	Atmel	Apple, Nvidia, Qualcomm, Samsung Electronics, and TI etc.
Cost (as compared to features provide)	Very Low	Average	Average	Low
Other Feature	Known for its Standard	Cheap	Cheap, effective	High speed operation Vast
Popular Microcontrollers	AT89C51, P89v51, etc.	PIC18fXX8, PIC16f88X, PIC32MXX	Atmega8, 16, 32, Arduino Community	LPC2148, ARM Cortex-M0 to ARM Cortex-M7, etc.

INTRODUCTORY CONCEPTS

CENTRAL PROCESSOR UNIT (CPU)

As its name suggests, this is a unit which monitors and controls all processes within the microcontroller. It consists of several subunits, of which the most important are:

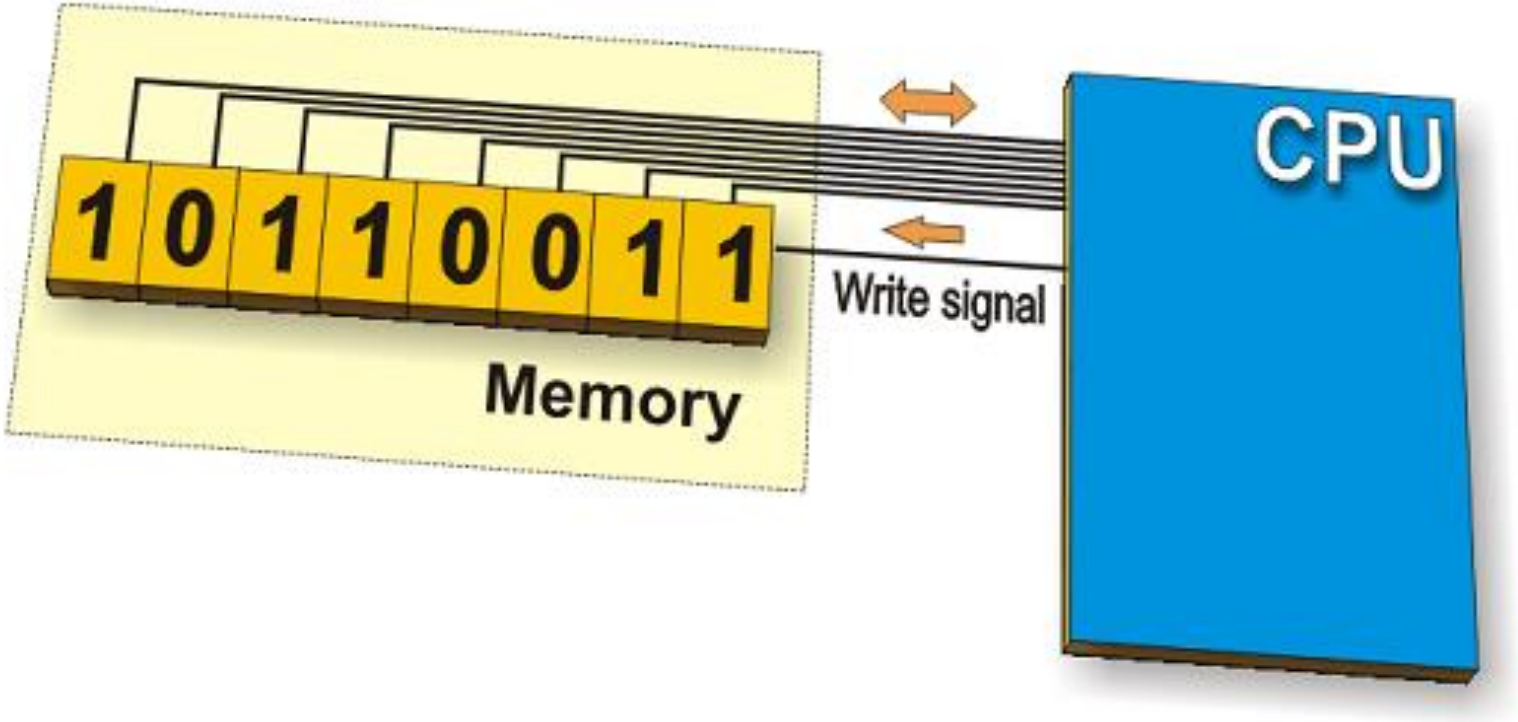
- **Instruction Decoder** is a part of electronics which decodes program instructions and runs other circuits on the basis of that. The 'instruction set' which is different for each microcontroller family expresses the abilities of this circuit.
- **Arithmetical Logical Unit (ALU)** performs all mathematical and logical operations upon data; and
- **Accumulator** is an SFR closely related to the operation of the ALU. It is a kind of working desk used for storing all data upon which some operation should be performed (addition, shift/move… etc.). It also stores results ready for use in further processing. One of the SFRs, called a *Status Register (PSW)*, is closely related to the accumulator. It shows at any given time the 'status' of a number stored in the accumulator (number is larger or less than zero… etc.). Accumulator is also called working register and is marked as W register or just W, therefore.



INTRODUCTORY CONCEPTS

REGISTER

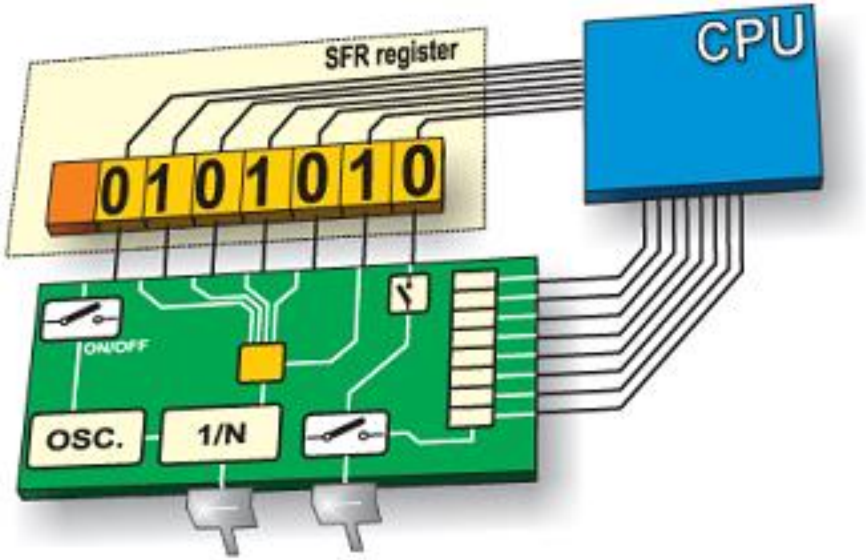
In short, a register or a memory cell is an electronic circuit which can memorize the state of one byte.



INTRODUCTORY CONCEPTS

SFR REGISTERS

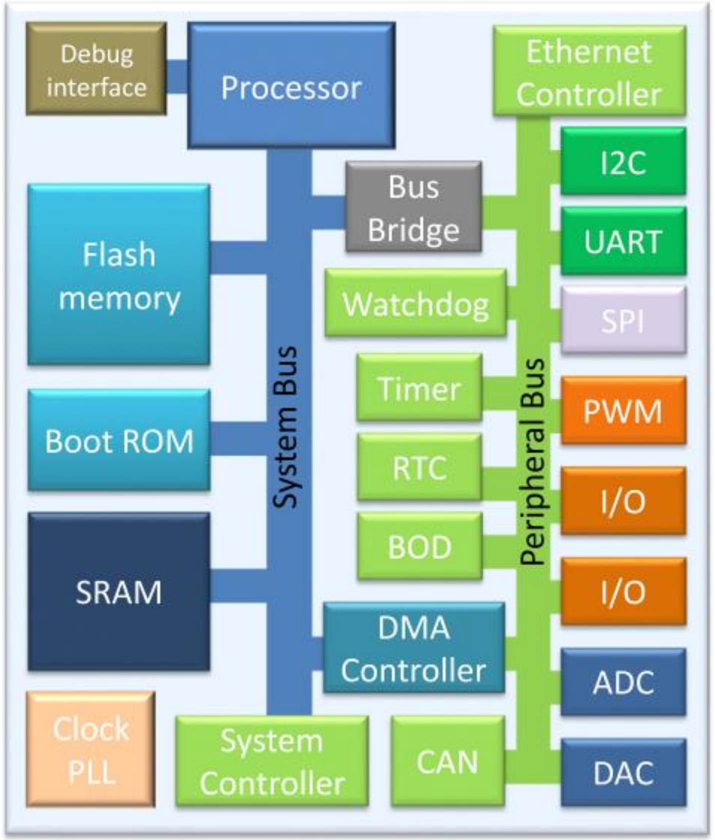
In addition to registers which do not have any special and predetermined function, every microcontroller has a number of registers (SFR) whose function is predetermined by the manufacturer. Their bits are connected (literally) to internal circuits of the microcontroller such as timers, A/D converter, oscillators and others, which means that they are directly in command of the operation of these circuits, i.e.: the microcontroller. Imagine eight switches which control the operation of a small circuit within the microcontroller -Special Function Registers do exactly that.



INTRODUCTORY CONCEPTS

BUS

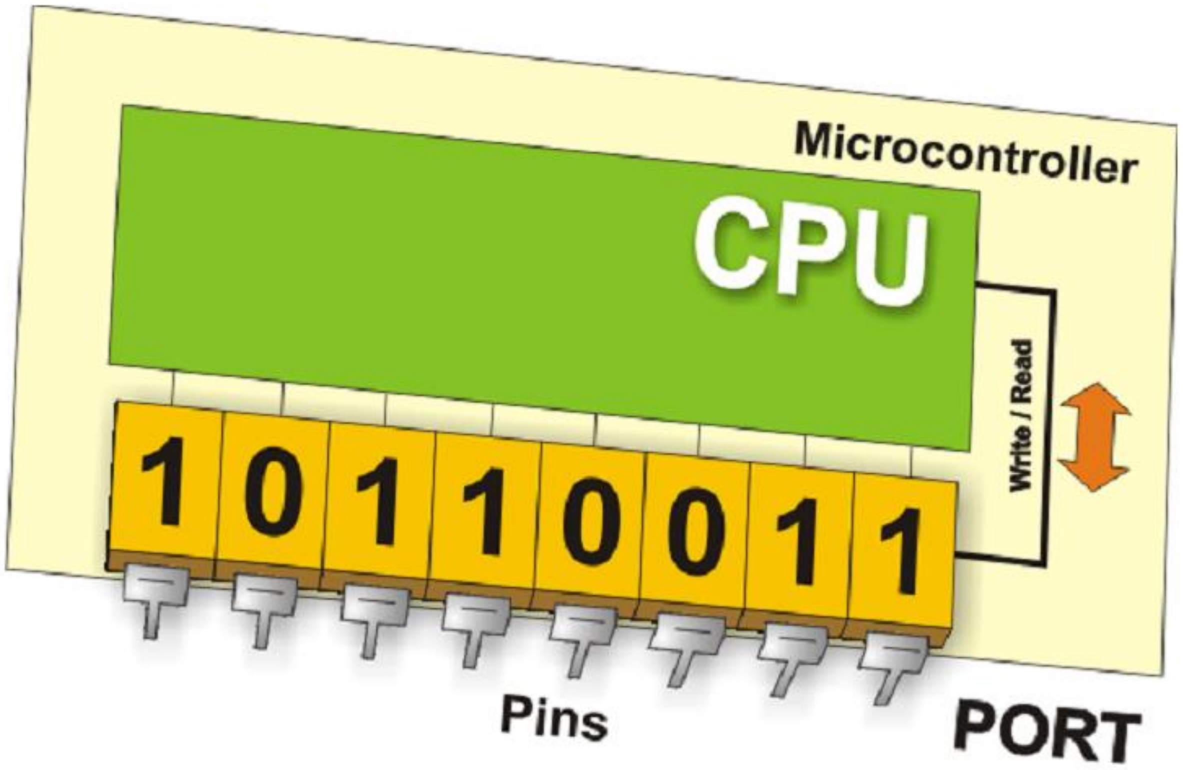
A bus consists of 8, 16 or more wires. There are two types of buses: the address bus and the data bus. The address bus consists of as many lines as necessary for memory addressing. It is used to transmit address from the CPU to the memory. The data bus is as wide as the data, in our case it is 8 bits or wires wide. It is used to connect all the circuits within the microcontroller.



INTRODUCTORY CONCEPTS

INPUT / OUTPUT PORTS

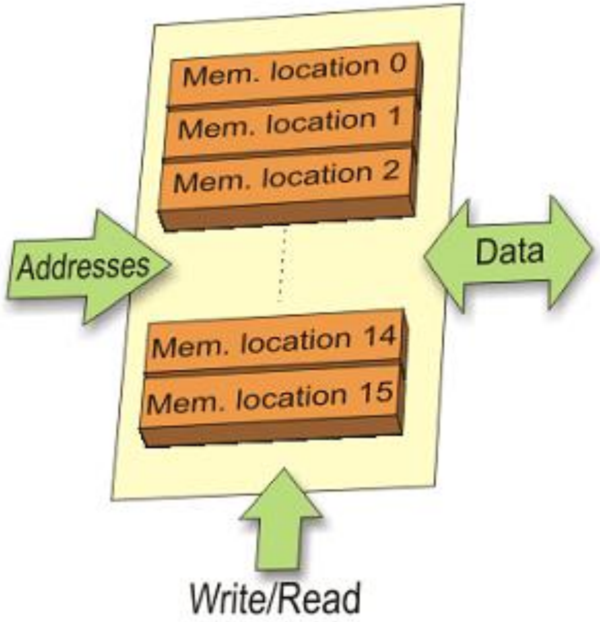
In order to make the microcontroller useful, it has to be connected to additional electronics, i.e.: peripherals. Each microcontroller has one or more registers (called ports) connected to the microcontroller pins. Why input/output? Because you can change a pin function as you wish. For example, suppose you want your device to turn on/off three signal LEDs and simultaneously monitor the logic state of five sensors or push buttons. Some of the ports need to be configured so that there are three outputs (connected to LEDs) and five inputs (connected to sensors). It is simply performed by software, which means that a pin function can be changed during operation.



INTRODUCTORY CONCEPTS

MEMORY UNIT

Memory is part of the microcontroller used for data storage. The easiest way to explain it is to compare it with a filing cabinet with many drawers. Suppose that the drawers are clearly marked so that their contents can be easily found out by reading the label on the front of the drawer.



Similarly, each memory address corresponds to one memory location. The contents of any location can be accessed and read by its addressing. Memory can either be written to or read from. There are several types of memory within the microcontroller:

INTRODUCTORY CONCEPTS

RANDOM ACCESS MEMORY (RAM):

Once the power supply is off the contents of RAM is cleared. It is used for temporary storing data and intermediate results created and used during the operation of the microcontroller. For example, if the program performs an addition (of whatever), it is necessary to have a register representing what in everyday life is called the ‘sum’. For this reason, one of the registers of RAM is called the ‘sum’ and used for storing results of addition.

READ ONLY MEMORY (ROM):

Read Only Memory (ROM) is used to permanently save the program being executed. The size of program that can be written depends on the size of this memory. Today’s microcontrollers commonly use 16-bit addressing, which means that they are able to address up to 64 Kb of memory, i.e.: 65535 locations. As a novice, your program will rarely exceed the limit of several hundred instructions. There are several types of ROM.

Masked ROM (MROM):

One Time Programmable ROM (OTP ROM):

PROM

Flash Memory:

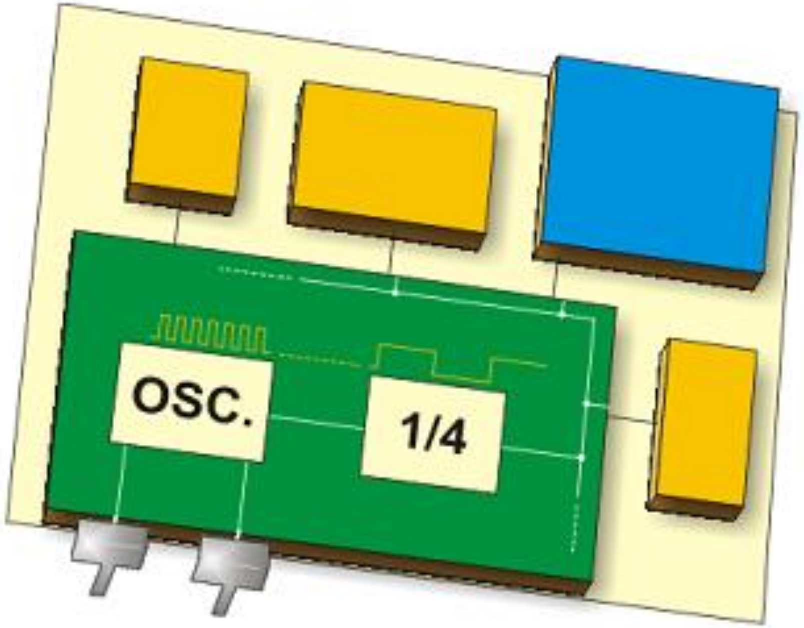
ELECTRICALLY ERASABLE PROGRAMMABLE ROM (EEPROM):

Flash EEPROM

UV Erasable Programmable ROM (UV EPROM):

INTRODUCTORY CONCEPTS

OSCILLATOR



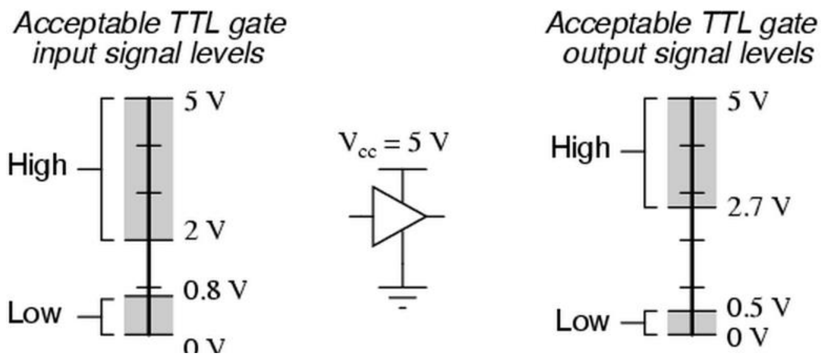
Even pulses generated by the oscillator enable harmonic and synchronous operation of all circuits within the microcontroller. The oscillator is usually configured so as to use quartz crystal or ceramic resonator for frequency stability, but it can also operate as a stand-alone circuit (like RC oscillator). It is important to say that instructions are not executed at the rate imposed by the oscillator itself, but several times slower. It happens because each instruction is executed in several steps. In some microcontrollers, the same number of cycles is needed to execute all instructions, while in others; the number of cycles is different for different instructions. Accordingly, if the system uses quartz crystal with a frequency of 20 MHz, the execution time of an instruction is not 50ns, but 200, 400 or 800 ns, depending on the type of MCU!

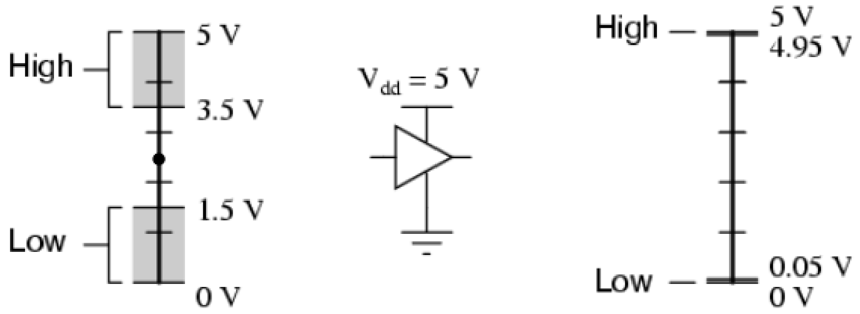
INTRODUCTORY CONCEPTS

Supply voltage

Most microcontrollers operate with the standard logic voltage of +5V or +3.3V. Some microcontrollers can operate at as low as +2.7V and some will tolerate +6V without any problems. You should check the manufacturers' data sheets about the allowed limits of the power supply voltage.

A voltage regulator circuit is usually used to obtain the required power supply voltage when the device is to be operated from a mains adaptor or batteries. For example, a 3.3V regulator is required if the microcontroller is to be operated using a 9V supply (e.g., a battery).





INTRODUCTORY CONCEPTS

POWER SUPPLY CIRCUIT

There are two things worth attention concerning the microcontroller power supply circuit:

- **Brown out** is a potentially dangerous condition which occurs at the moment the microcontroller is being turned off or when the power supply voltage drops to a minimum due to electric noise. As the microcontroller consists of several circuits with different operating voltage levels, this state can cause its out-of-control performance. In order to prevent it, the microcontroller usually has a built-in circuit for brown out reset which resets the whole electronics as soon as the microcontroller incurs a state of emergency.
- **Reset pin** is usually marked as MCLR (*Master Clear Reset*). It is used for external reset of the microcontroller by applying logic zero (0) or one (1) to it, which depends on the type of the microcontroller. In case the brown out circuit is not built in, a simple external circuit for brown out reset can be connected to the MCLR pin.

Power-on reset

Some microcontrollers have built-in power-on reset circuits which keep the microcontroller in the reset state until all the internal circuitry has been initialized correctly. This feature is very useful as it starts the microcontroller from a known state on power-up. An external reset can also be provided where the microcontroller can be reset when an external button is pressed.

INTRODUCTORY CONCEPTS

Low power operation

Low power operation is especially important in portable applications where the microcontroller-based equipment is operated from batteries. Some microcontrollers can operate with less than 2 mA with 5V supply, and around 15 µA at 3 V supply. Some other microcontrollers, especially microprocessor-based systems where there could be several chips may consume several hundred milliamperes of current or even more.

Sleep mode

Some microcontrollers offer power management functions such as built-in sleep modes, where executing this instruction puts the microcontroller into a mode where the internal oscillator is stopped and the power consumption is reduced to an extremely low level. The main reason for using the sleep mode is to conserve battery power when the microcontroller is not doing anything useful. The microcontroller usually wakes up from the sleep mode by an external reset or by a watchdog time-out.

INTRODUCTORY CONCEPTS

Current sink/source capability

USB interface

CAN interface

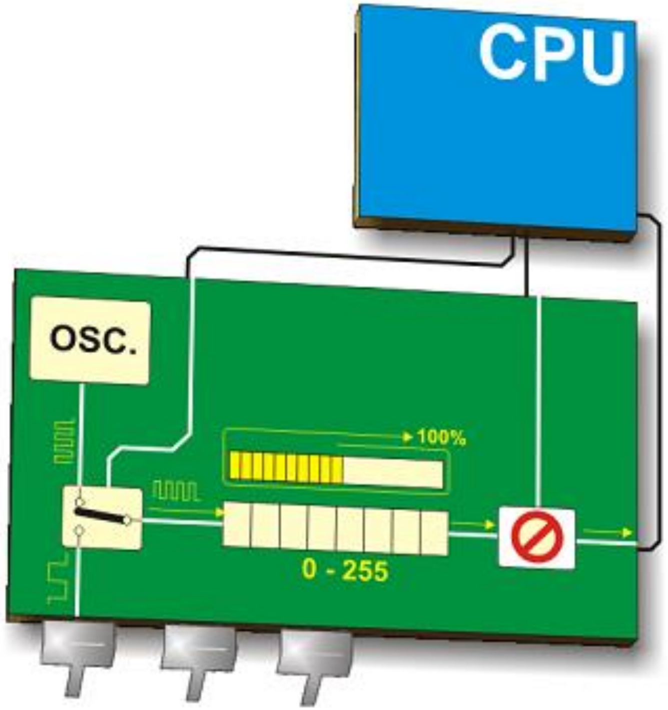
Ethernet interface

Wi-Fi and/or Bluetooth interface

INTRODUCTORY CONCEPTS

TIMERS/COUNTERS

The microcontroller oscillator uses quartz crystal for its operation. Even though it is not the simplest solution, there are many reasons to use it. The frequency of such oscillator is precisely defined and very stable, so that pulses it generates are always of the same width, which makes them ideal for time measurement. Such oscillators are also used in quartz watches. If it is necessary to measure time between two events, it is sufficient to count up pulses generated by this oscillator. This is exactly what the timer does.



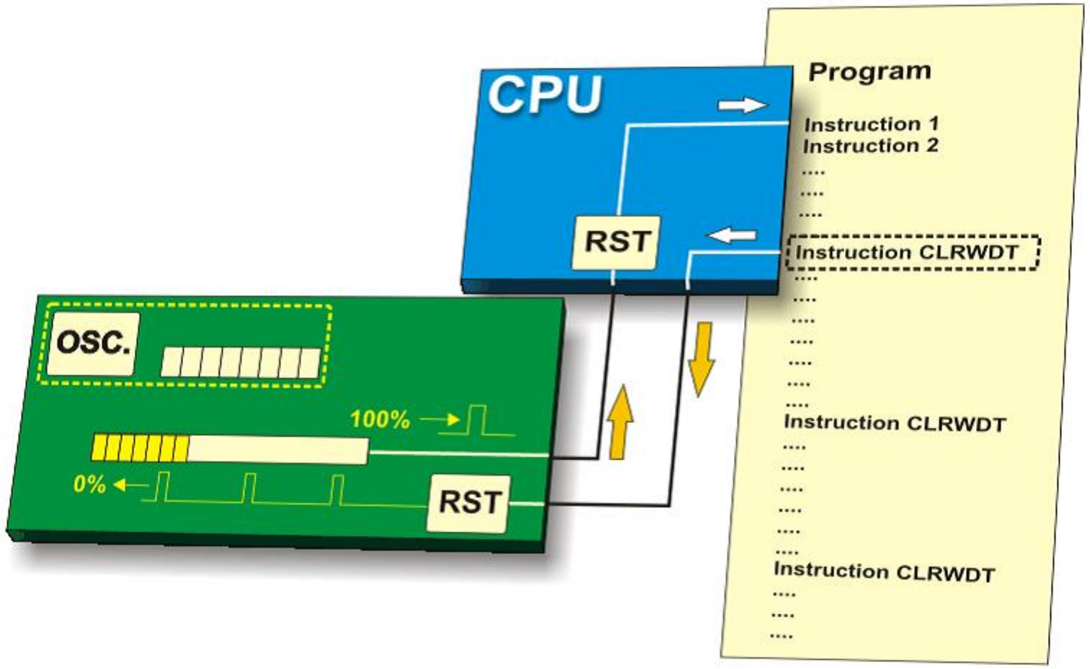
INTRODUCTORY CONCEPTS

WATCHDOG TIMER

A watchdog timer is a timer connected to a completely separate RC oscillator within the microcontroller.

If the watchdog timer is enabled, every time it counts up to the maximum value, the microcontroller reset occurs and the program execution starts from the first instruction. The point is to prevent this from happening by using a specific command.

Anyway, the whole idea is based on the fact that every program is executed in several longer or shorter loops. If instructions which reset the watchdog timer are set at the appropriate program locations, besides commands being regularly executed, then the operation of the watchdog timer will not affect the program execution. If for any reason, usually electrical noise in industry, the program counter 'gets stuck' at some memory location from which there is no return, the watchdog timer will not be cleared, so the register's value being constantly incremented will reach the maximum *et voila!* Reset occurs!



INTRODUCTORY CONCEPTS

INTERRUPT

Most programs use interrupts in their regular execution. The purpose of the microcontroller is mainly to respond to changes in its surrounding. In other words, when an event takes place, the microcontroller does something... For example, when you push a button on a remote controller, the microcontroller will register it and respond by changing a channel, turn the volume up or down... etc. If the microcontroller spent most of its time endlessly checking a few buttons for hours or days, it would not be practical at all.

This is why the microcontroller has learnt a trick during its evolution. Instead of checking each pin or bit constantly, the microcontroller delegates the ‘wait issue’ to a ‘specialist’ which will respond only when something attention worthy happens.

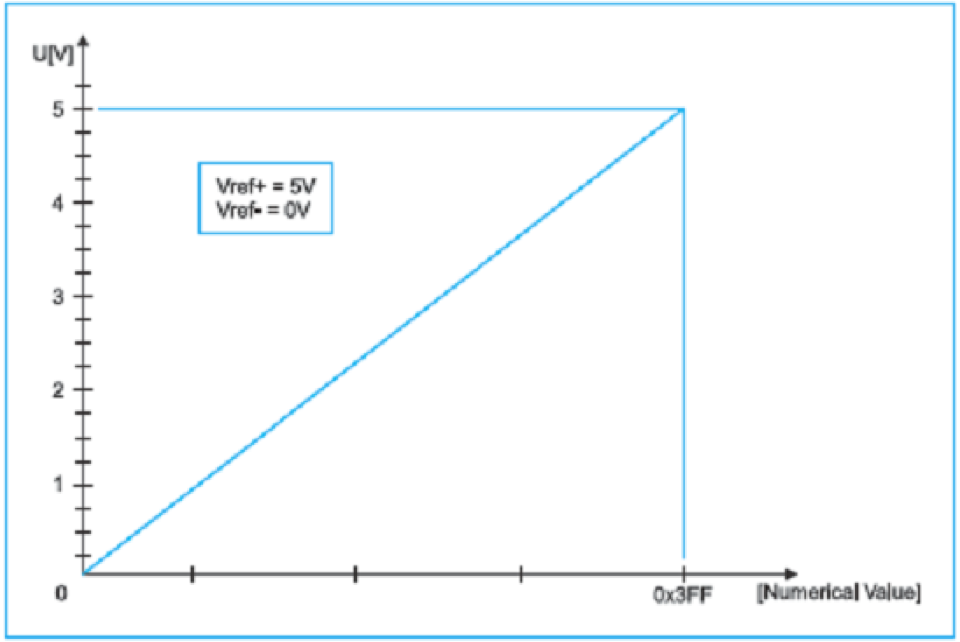
The signal which informs the central processor unit about such an event is called an INTERRUPT.

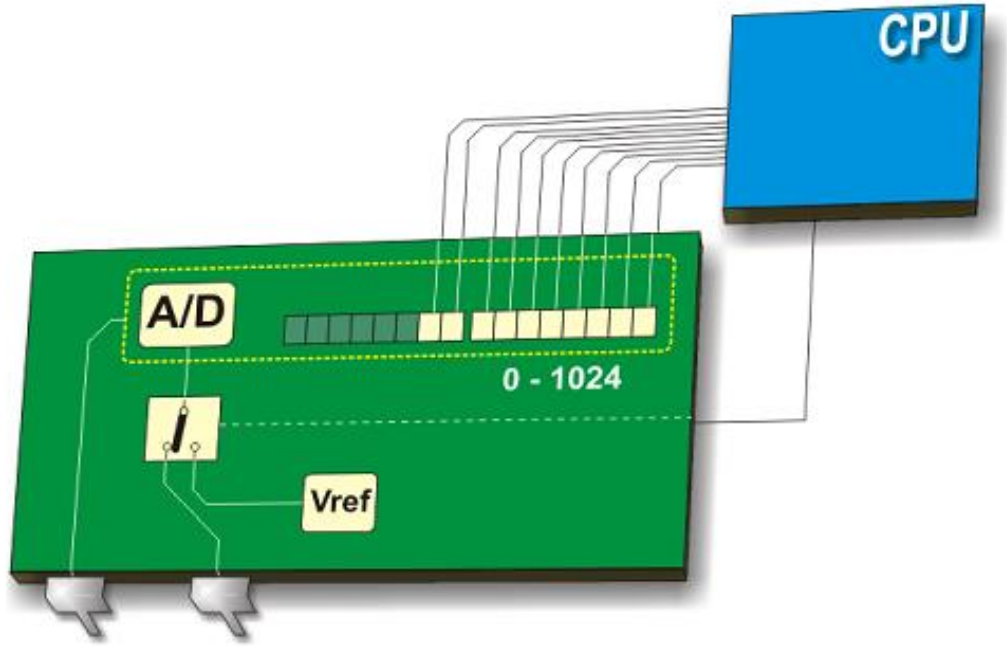
INTRODUCTORY CONCEPTS

A/D Converter

External signals are usually fundamentally different from those the microcontroller understands (ones and zeros) and have to be converted therefore into values understandable for the microcontroller. An analogue to digital converter is an electronic circuit which converts continuous signals to discrete digital numbers. In other words, this circuit converts an analogue value into a binary number and passes it to the CPU for further processing. This module is therefore used for input pin voltage measurement (analogue value).

The result of measurement is a number (digital value) used and processed later in the program.





INTRODUCTORY CONCEPTS

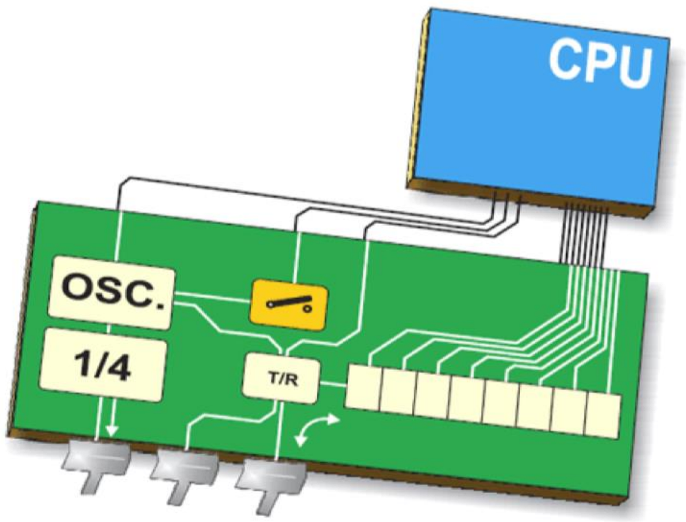
SERIAL COMMUNICATION

Parallel connection between the microcontroller and peripherals via input/output ports is the ideal solution on shorter distances up to several meters. However, in other cases when it is necessary to establish communication between two devices on longer distances it is not possible to use parallel connection. Instead, serial communication is used.

INTRODUCTORY CONCEPTS

Today, most microcontrollers have built in several different systems for serial communication as a standard equipment. Which of these systems will be used depends on many factors of which the most important are:

- How many devices the microcontroller has to exchange data with?
- How fast the data exchange has to be?
- What is the distance between devices?
- Is it necessary to send and receive data simultaneously?



One of the most important things concerning serial communication is the *Protocol* which should be strictly observed. It is a set of rules which must be applied in order that devices can correctly interpret data they mutually exchange. Fortunately, the microcontroller automatically takes care of this, so that the work of the programmer/user is reduced to simple write (data to be sent) and read (received data).

INTRODUCTORY CONCEPTS

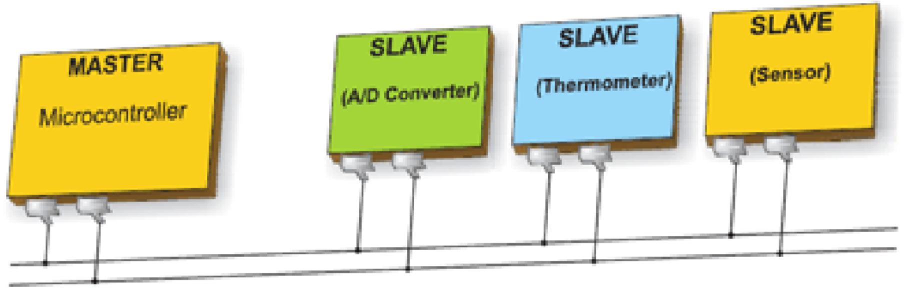
BAUD RATE

The term *baud rate* is used to denote the number of bits transferred per second [bps]. Note that it refers to bits, not bytes. It is usually required by the protocol that each byte is transferred along with several control bits. It means that one byte in serial data stream may consist of 11 bits. For example, if the baud rate is 300 bps then maximum 37 and minimum 27 bytes may be transferred per second.

The most commonly used serial communication systems are:

I²C (INTER INTEGRATED CIRCUIT)

Inter-integrated circuit is a system for serial data exchange between the microcontrollers and specialized integrated circuits of a new generation. It is used when the distance between them is short (receiver and transmitter are usually on the same printed board). Connection is established via two conductors. One is used for data transfer and the other is used for synchronization (clock signal). As seen in figure below, one device is always a master. It performs addressing of one slave chip before communication starts. In this way one microcontroller can communicate with 112 different devices. Baud rate is usually 100 Kb/sec (standard mode) or 10 Kb/sec (slow baud rate mode). Systems with the baud rate of 3.4 Mb/sec have recently appeared. The distance between devices which communicate over an I²C bus is limited to several meters.

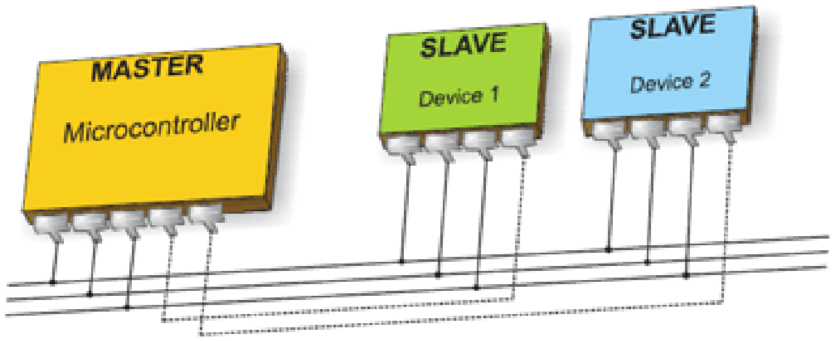


INTRODUCTORY CONCEPTS

SPI (SERIAL PERIPHERAL INTERFACE BUS)

A serial peripheral interface (SPI) bus is a system for serial communication which uses up to four conductors, commonly three. One conductor is used for data receiving, one for data sending, one for synchronization and one alternatively for selecting a device to communicate with. It is a full duplex connection, which means that data is sent and received simultaneously.

The maximum baud rate is higher than that in the I^2C communication system.



INTRODUCTORY CONCEPTS

UART (UNIVERSAL ASYNCHRONOUS RECEIVER/TRANSMITTER)

This sort of communication is asynchronous, which means that a special line for transferring clock signal is not used. In some applications, such as radio connection or infrared waves remote control, this feature is crucial. Since only one communication line is used, both receiver and transmitter operate at the same predefined rate in order to maintain necessary synchronization. This is a very simple way of transferring data since it basically represents the conversion of 8-bit data from parallel to serial format. Baud rate is not high, up to 1 Mbit/sec.

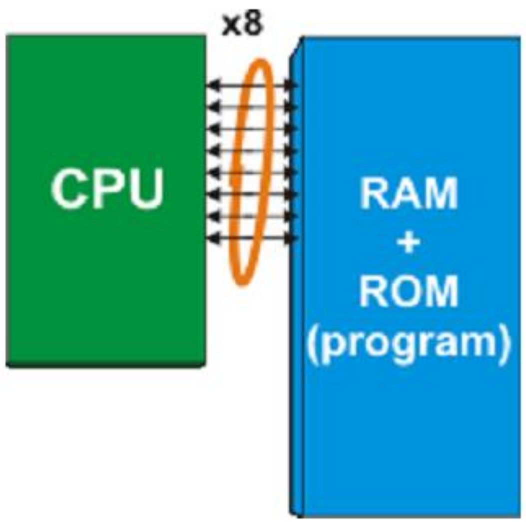
INTRODUCTORY CONCEPTS

INTERNAL ARCHITECTURE

All upgraded microcontrollers use one of two basic design models called *Harvard* and *von-Neumann* architecture.

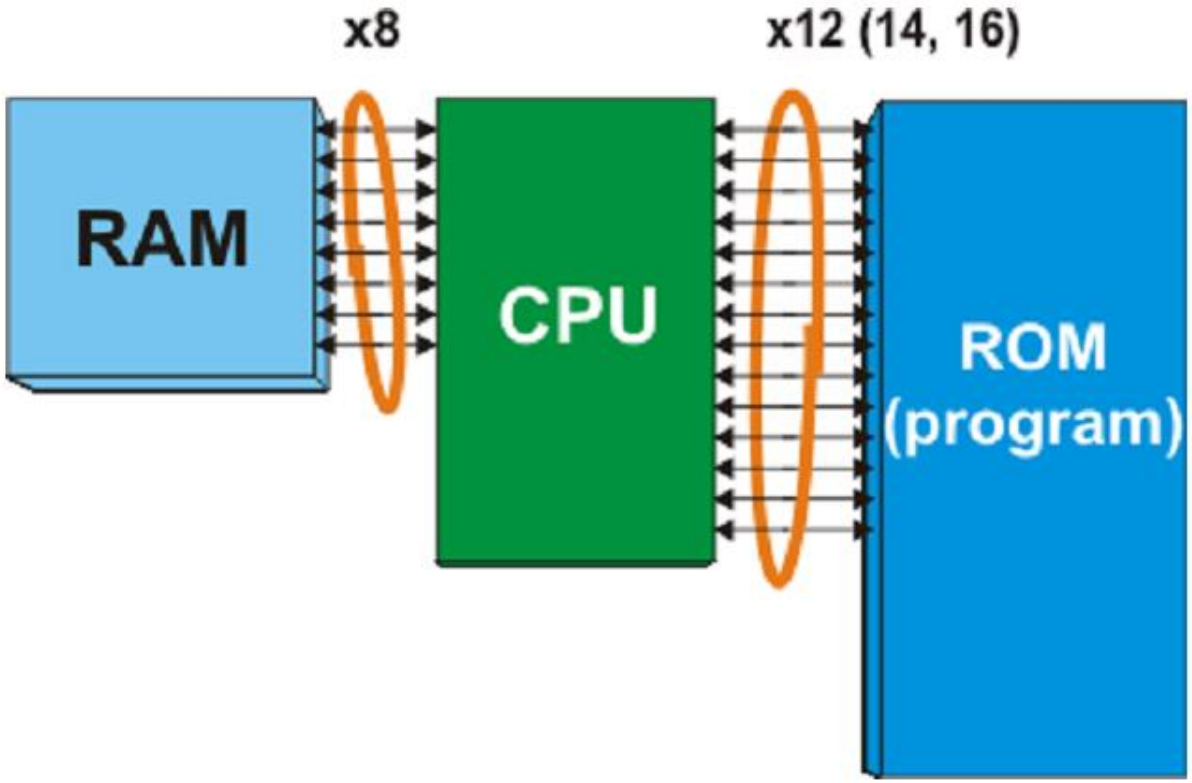
They represent two different ways of exchanging data between CPU and memory.

VON-NEUMANN ARCHITECTURE:



INTRODUCTORY CONCEPTS

HARVARD ARCHITECTURE:



INTRODUCTORY CONCEPTS

RISC and CISC

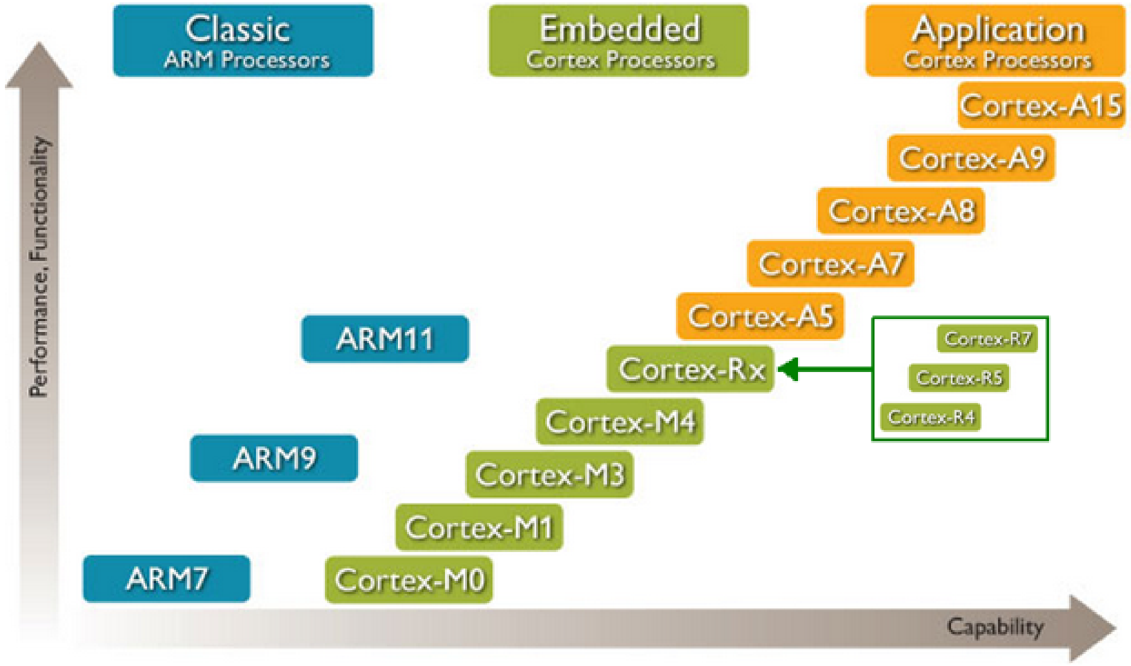
RISC (reduced instruction set computer) and CISC (complex instruction computer) refer to the instruction set of a microcontroller. In an 8-bit RISC microcontroller, data is 8-bit wide but the instruction words are more than 8-bit wide (usually 12-, 14- or 16-bits) and the instructions occupy one word in the program memory. Thus, the instructions are fetched and executed in one cycle, resulting in improved performance.

In a CISC microcontroller both data and instructions are 8-bit wide. CISC microcontrollers usually have over 200 instructions. Data and code are on the same bus and cannot be fetched simultaneously.

INTRODUCTORY CONCEPTS

Vi điều khiển STM32

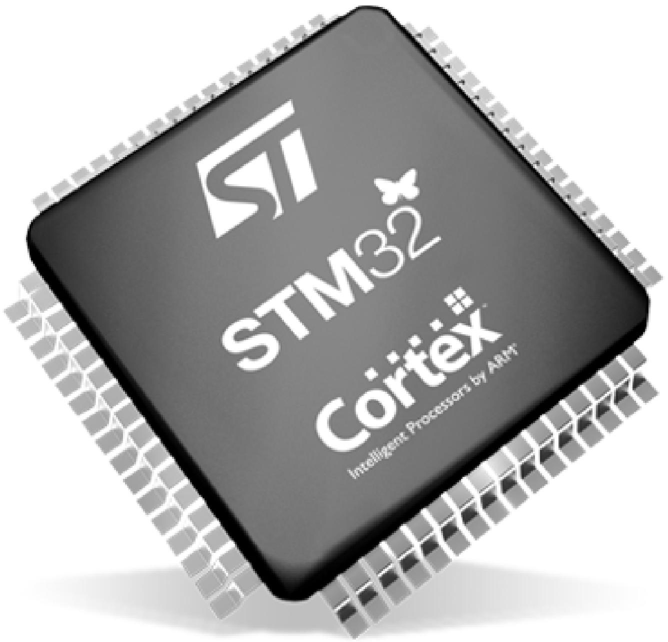
- Được thiết kế dựa trên dòng Arm-Cortex của hãng ST microelectronic.
- Được xây dựng trên kiến trúc Arm mới, tiết kiệm năng lượng, chi phí sản xuất thấp nên dễ dàng cạnh tranh với các dòng vi điều khiển 8/16 bit.
- Đáp ứng được các tiêu chí về hiệu suất, chi phí, ứng dụng tiêu thụ năng lượng thấp, đáp ứng thời gian thực khắt khe .



INTRODUCTORY CONCEPTS

Vi điều khiển STM32

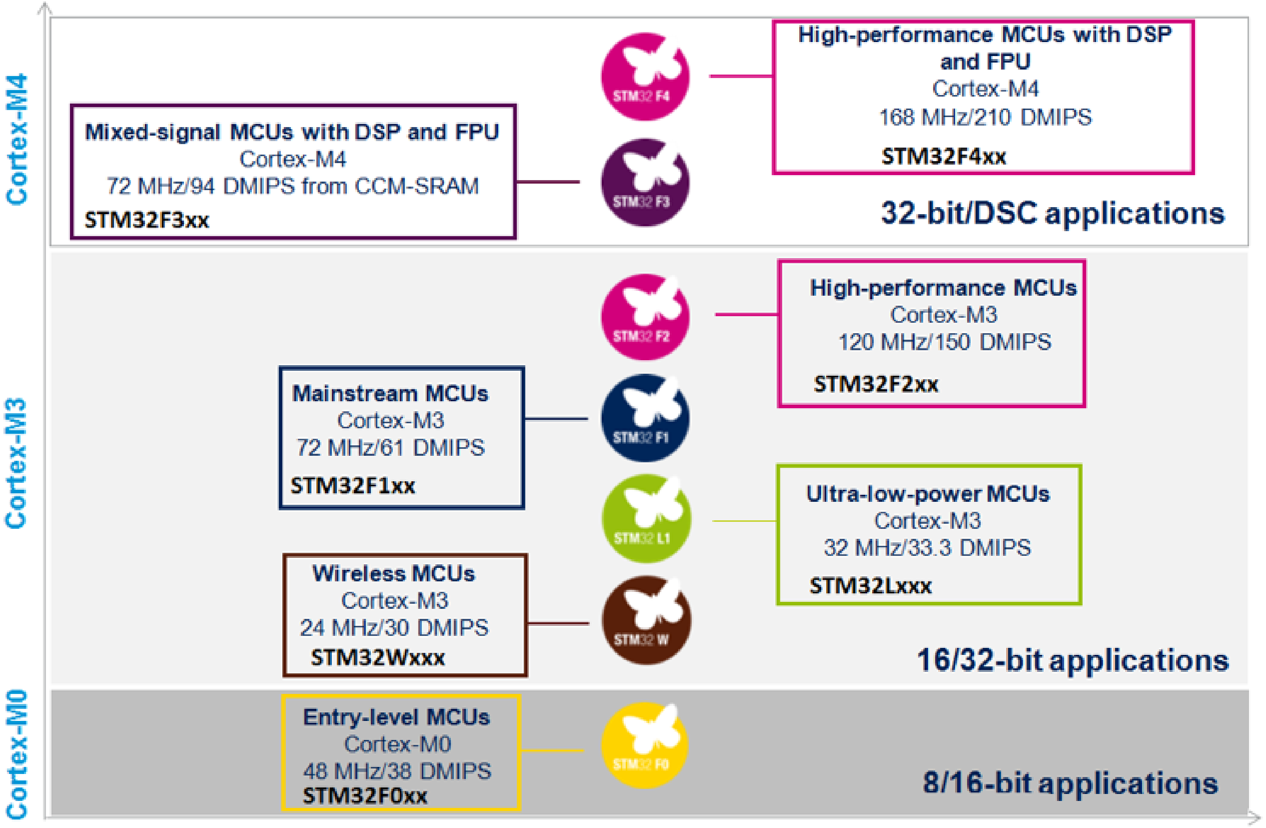
- Giá thành ngày rẻ. Vd. Stm32F100x giá khoảng 1 USD chạy tần số 24MHz)
- Khả năng hỗ trợ của hãng sản xuất, nhiều công cụ phát triển
- Sự đồ sộ về ngoài vi (GPIO, I2C, SPI, USB, Ethernet, CAN,…)



STM32 gồm nhiều biến thể: dòng Performance có tần số hoạt động của CPU lên tới 72Mhz và dòng Access có tần số hoạt động lên tới 36Mhz. Bộ nhớ FLASH ROM và SRAM lớn

INTRODUCTORY CONCEPTS

Các dòng vi điều khiển STM32



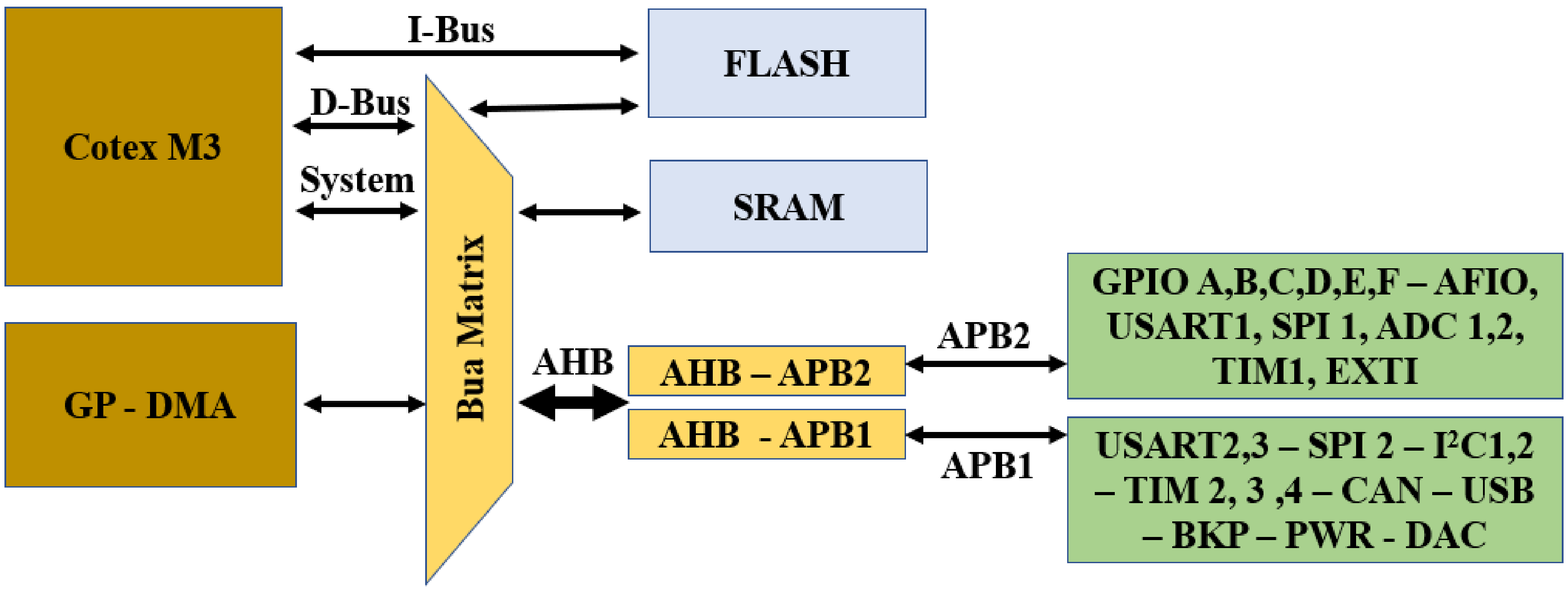
Mainstream MCUs
(Dòng vi điều khiển xu thế hiện nay)
STM32F1xx
(101xx,102xx,103xx,105xx,107xx)

STM32F103xx

- **Low – Desity Performance**
  Stm32F103x4, Stm32F103x6
- **Meium - DesityPerformance**
  Stm32F103x8, Stm32F103xB
- **High - DesityPerformance**
  Stm32F103xC, Stm32F103xD, Stm32F103xE
- **XL - DesityPerformance**
  Stm32F103xF, Stm32F103xG

INTRODUCTORY CONCEPTS

Kiến trúc dòng vi điều khiển STM32F103xx

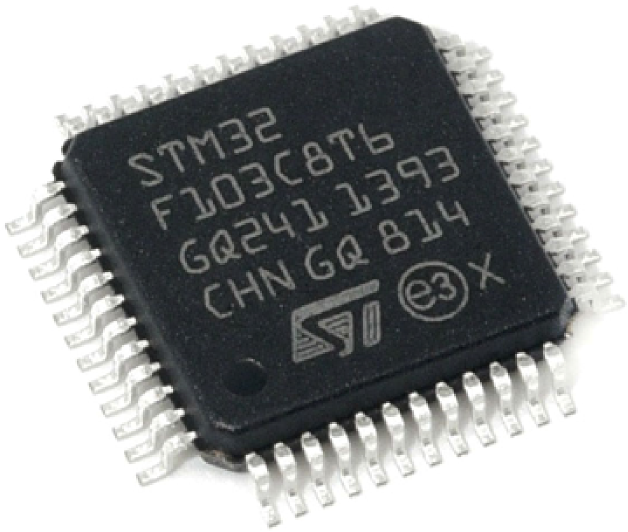


INTRODUCTORY CONCEPTS

Vi điều khiển STM32F103C8

STM32F103C8T6 là vi điều khiển 32bit, thuộc họ F1 của dòng chip STM32 hãng ST.

- Lõi ARM COTEX M3.
- Dòng Medium – density performance
- Tốc độ tối đa 72Mhz.
- Bộ nhớ :
- 64 kbytes bộ nhớ Flash
- 20 kbytes SRAM



INTRODUCTORY CONCEPTS

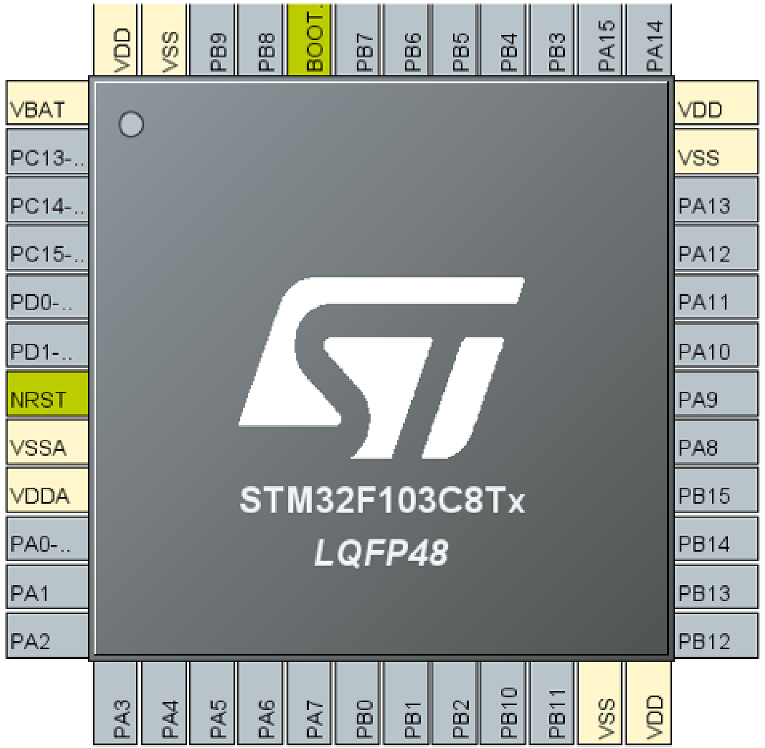
Các khối bên trong vi điều khiển STM32F103C8

- 2 Bộ ADC 12 bit với 16 kênh 2.4 MSPS
- 7 Timers 16 bit
- 2 Watchdog timers, RTC (Real Time Clock).
- 37 I/Os, 1 CAN, 2 I2C, 1 SPI 42Mbits/s, 2 I2S, 3 USART,.

Ngoài ra còn hỗ trợ DMA, 1 USB 2.0, Ethenet, camera.

INTRODUCTORY CONCEPTS

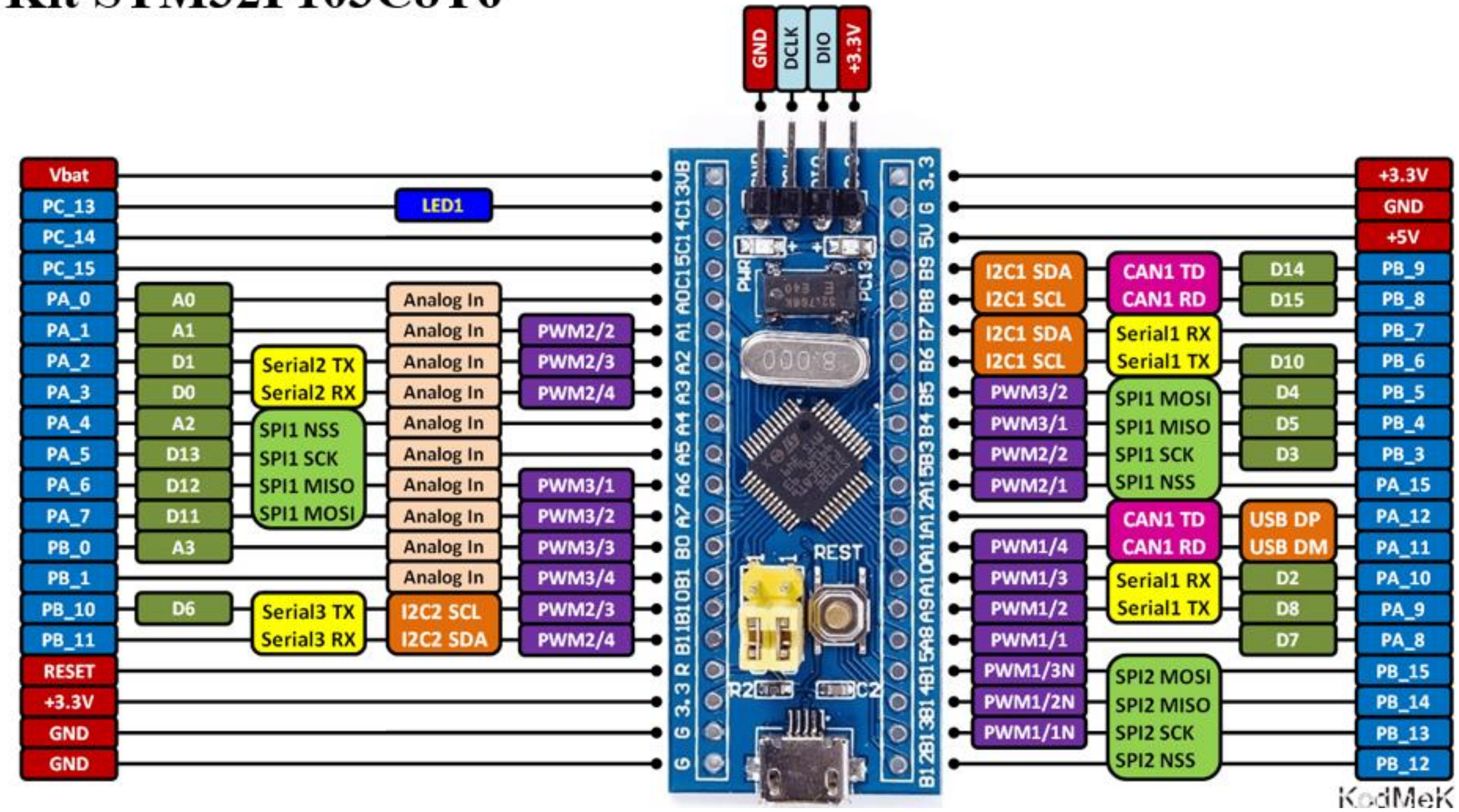
Khảo sát sơ đồ chân vi diều khiển STM32F103



Power	Chân VDD	Nguồn cấp 3.3v
	Chân VSS	Nguồn cấp 0v
Chân I/O	PA0 -> PA15 PB0 -> PB15 PC13 -> PC15	37 chân I/O đa chức năng
Ngắt ngoài	PA0 -> PA15 PB0 -> PB1 PC13 -> PC15	Chân cho phép thực hiện ngắt ngoài
Analog	PA0 -> PA7 PB0 -> PB1	ADC độ phân giải 12 bit
PWM	PA0 -> PA3 PA6 -> PS10 PB0, PB1, P6 -> PB9	15 chân điều chế độ rộng xung

INTRODUCTORY CONCEPTS

Kit STM32F103C8T6



Giá thành:
~200.000VND