Embedded Concept

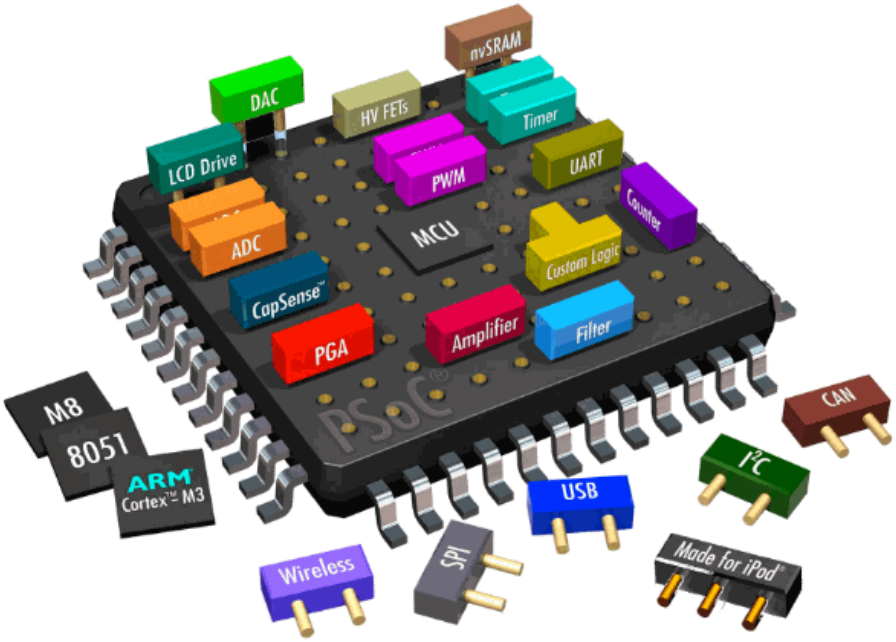
- **Embedded System Definition**

هو نظام مدمج (مدمج تعني صغير أو مخفي) يقوم بوظيفة محدده (قد تم تحديدها له مسبقا) داخل نظام كبير.
فمثلا لو قمنا بعمل دائرة تتحكم ف التكييف لتشعر بدرجة الحرارة ثم وضعناها ف السيارة للتحكم ف السرعة فان هذه الدائرة لن تعمل ولن يكون لها خرج لان وظيفتها محددة وهي التحكم ف التكييف وليس التحكم في سرعة السيارة.

- **Embedded System Design Techniques And Challenges**

Design	Challenges
⇒ Embedded System Is Computing System But Limited Resources. ⇒ Computing System Consists Of : • Processor • Memory • Input/Output Peripherals ⇒ Embedded System Design Techniques: • System On Board • System On Chip	⇒ Performance ⇒ Size ⇒ Cost ⇒ Power Consumption ⇒ Limited Resources ⇒ Real Time Constraints

❖ System On Chip



❖ System On Board

	System On Board	System On Chip
Performance	———	———
Size	↑	↓
Cost	↑	↓
Power Consumption	↑	↓
Configurability	Easy (يمكن التجربة وتغيير المكونات)	NA
When To Use?	For Development and Design phase	For Production

- **CPU vs Processor vs Micro Processor**

  - **CPU**

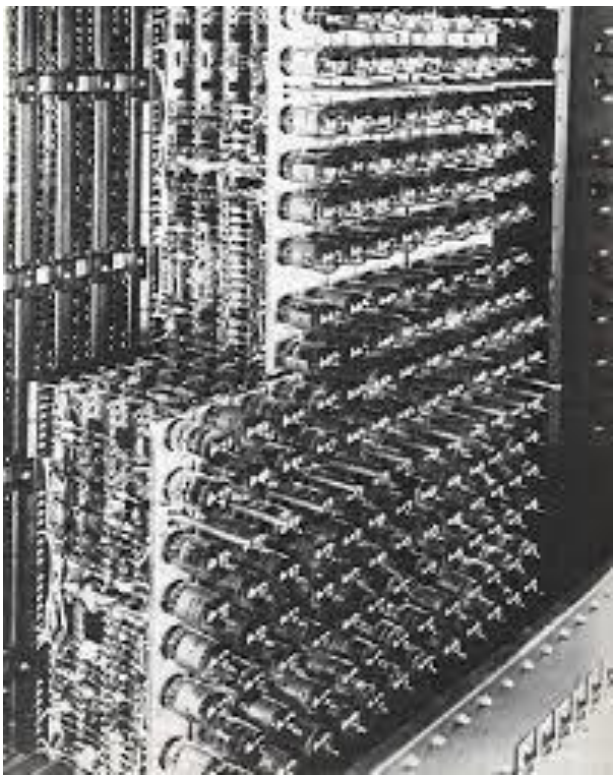
    When you have a system that contains more than one processor, one of them would be the master one, it would control all other processor. This one is called the **Central Processing Unit CPU.**

    The CPU is the brain of a computer, containing all the circuitry needed to process input, store data, and output results. The CPU is constantly following instructions of computer programs that tell it which data to process and how to process it.

- **Processor**

  processor was made of **vacuum tubes**, it was very huge like a whole building.

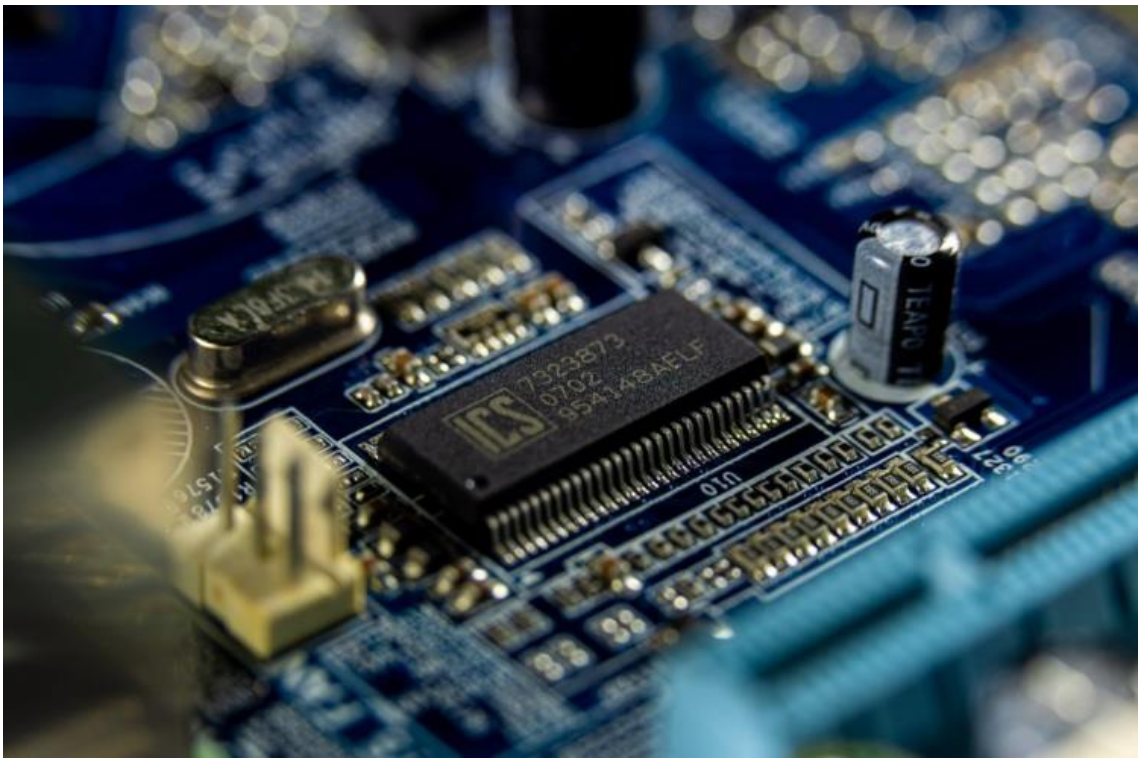
  A processor is an integrated electronic circuit that performs the calculations that run a computer. A processor performs arithmetical, logical, input/output (I/O) and other basic instructions that are passed from an operating system (OS). Most other processes are dependent on the operations of a processor.



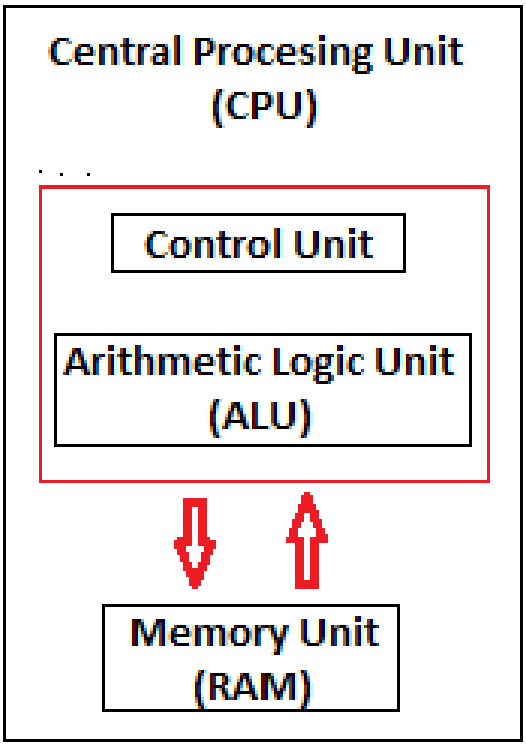
- **Micro Processor**

It’s silicon based IC, Micro refers to its small size.

The main difference between the two is their functionality and purpose within a computer system. A CPU is a type of processor tasked with a variety of roles. A microprocessor is generally tasked with one specific task and does that one task exceedingly well.



- **Deep Inside The Processor**



- **Control Unit**

  It fetches the instructions from main memory and decodes them by ID (Instruction Decoder).

- **ALU**

  It receives the instruction after decoding it in CU and execute it.

- **Register File (RAM)**

  It consists of:

- **PC (Program counter):** It's value is the address of the next instruction to be executed.
- **IR(Instruction Register):** its value is the address of instruction being executed at the current time.

- **ACC(Accumulator):** into which the ALU will automatically load the resultant of a processing operation.
- **PSW(Processor Status Word):** it's a 8-bit register, also known as a flag.
- **GPR(General Purpose Register):** is used for either memory address or data whenever needed.

- **Instruction Set Architecture (ISA)**

It defines how CPU is controlled by software, it acts as interface between hardware and software. It help the developers understand the output of the compiler.

RISC: Reduce Instruction Set Architecture

CISC: Complex Instruction Set Architecture

	RISC	CISC
Performance	Same	Same
Cost	Same	Same
Size	Same	Same
Power Consumption	Same	Same

RISC: execute more one line code.

CISC: execute one line code.

Intel uses CISC Machine.
ARM uses RISC Machine.

- **Cache Memory**

  Used to reduce the access time to main memory.

- **Pipe Line**

  It plays an important role in increasing the efficiency of data processing by keeping the processor in a continuous process of fetching, decoding and executing called (F&E Cycle).

- **Little Endian**

  It stores the least significant byte first.

- **Big Endian**

  It stores the most significant byte first.

- **Real Time Operating System**

  It is used in an environment where a large number of events.

- **Memory**

There are two types of memory:

1. Volatile: it loses its data when the applied voltage is removed.
2. Non-Volatile: it keeps its data when the applied voltage is removed.

Volatile Memory

RAM (Random Access Memory) : it called random because the time which it takes to reach the address is constant.

⇒ **RAM is used in run time as it faster then ROM.**

SRAM (Static Random Access Memory) : it based on transistor.

DRAM (Dynamic Random Access Memory): it based on capacitor.

	SRAM (Transistor)	DRAM (Capacitor)
cost	High	Low
size	Low	High
Performance	High	Low
Power consumption	Low	Needs a refresh circuit to recharge capacitor

⇒ They don't use DRAM because the refresh circuit as it increases the power consumption.

Non-Volatile Memory

ROM (Read Only Memory) : Designed and programmed by manufacture.

⇒ ROM is used for Code as it doesn't lose its data.

⇒ It called ROM as the processor can't write on it.

OTP (One Time Programmable ROM) : Can be Programmed only one time.

EPROM (Erase Programmable ROM) : Can be Programmed many times and erased by UV (Ultra Violet).

EEPROM(Electrical Erase Programmable ROM) : Can Be Programmed many time and erased by electrical signals.

Flash ROM : it is a type of EEPROM.

Flash Vs EEPROM

Flash memory	EEPROM
Based on NAND gates	Based on NOR gates
Memory density is more compared to EEPROM	Memory density is less compared to flash memory
Access to this memory is slower as the architecture is based on NAND	Access to this memory is faster as the architecture is based on NOR
Supports Erase, Write, Read operations	Supports Erase, Write, Read operations
Flash is erased block-wise	EEPROM is erased byte-wise
Write cycles are faster than EEPROM Read cycles are slower than EEPROM Erase cycles are faster than EEPROM	Write cycles are slower than flash Read cycles are faster than flash Erase cycles are slower than flash
Memory access in sequential. So, read is slower	Memory access in random. So, read is faster
Cheaper	Costly
Mainly used for program storage and data storage	Mainly used in applications to store configuration data.
Less endurance than EEPROM	More endurance than flash memory
Maximum erase/write cycles are less than EEPROM	Maximum erase/write cycles are more than flash
Size of this range up to GB	Size of this range from KB to MB.
USB thumb drives, hard disk and other mass storage media use flash memory	Examples of this memory usage include configuration storage in embedded boards`
Life cycle is more than EEPROM	Life cycle is lesser than flash
Parallel (D0-D7 along with control lines and address lines) interface for the microcontroller/processor	I2C, SPI interface for the microcontroller/processor
Example: S34ML16G202TFI200 from Cypress	Example: AT24C512C from ATMEL

	Flash	EEPROM
Access	**Blocks of bytes**	**Byte access**
Performance	**Good for blocks**	**Good for bytes**
Endurance	**10,000 Time to die**	**100,000 to 11000,000 Times to die**

Every Embedded System must have FLASH and SRAM and may have EEPROM.

I/O Peripherals

- It means that the way by which the user and computer or the user and processor communicate with each other.

Input / Output Peripherals between the user and computer like:

- Keyboard
- Mouse
- Webcam
- Speakers
- Fax

Input / Output Peripherals between the user and Processor like:

- Digital Input Output (DIO)
- Analog to Digital Converter (ADC)
- Digital to Analog Converter (DAC)
- Timers and Pulse Width Modulators (PWM)
- Universal Asynchronous Receiver Transmitter (UART)
- Serial Peripheral Interface (SPI)
- Inter Integrated Circuit (I2C)

ES Architecture

- **Von Numann Architecture**

  All memories have the same buses (Single Memory).

- **Harvard Architecture**

  Each Memory has its own buses (has separate memory for instructions and data).

  - Address Bus : its value is the address of the instruction to be execute.
  - Data Bus: it takes the data of the instruction whose address is in Address Bus.
  - Control Bus : it takes (write) or (read), write for writing data in the memory, read for reading data from the memory.