Chapter 1

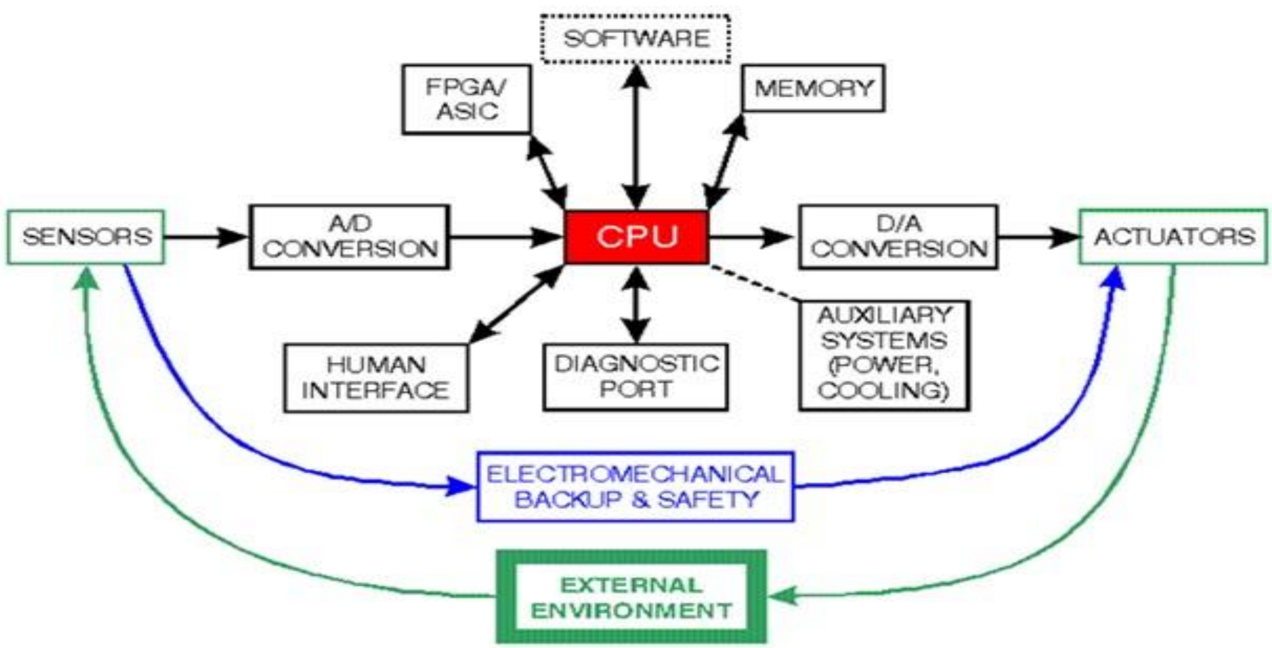
Introduction to Embedded System

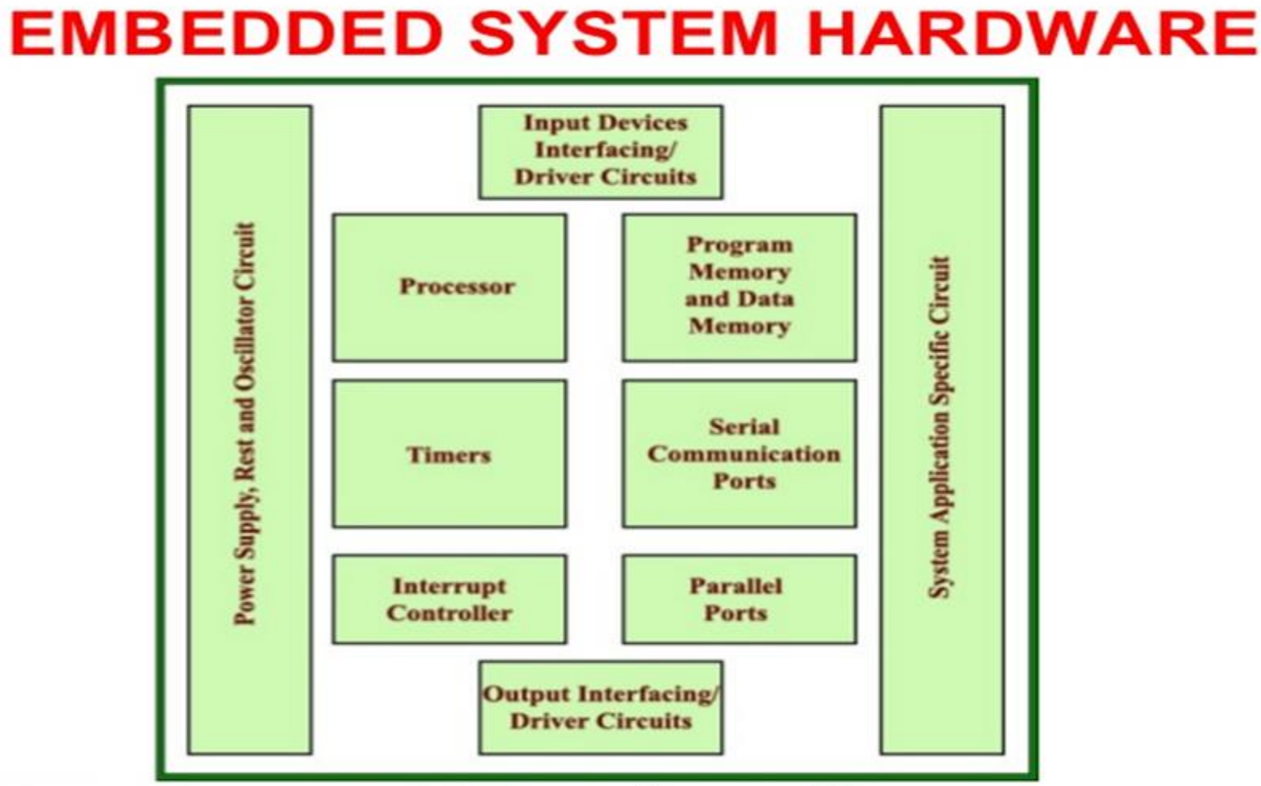
What is embedded system?

An embedded system is a system that has software embedded into hardware, which makes a system dedicated for an application (s) or specific part of an application or product or part of a larger system. It processes a fixed set of pre-programmed instructions to control electromechanical equipment which may be part of an even larger system (not a computer with keyboard, display, etc).

Three main embedded components are : (1). Embeds hardware to give computer like functionalities . (2). Embeds main application software generally into flash or ROM and the application software performs concurrently the number of tasks. (3). Embeds a real time operating system (RTOS), which supervises the application software tasks running on the hardware and organizes the accesses to system resources according to priorities and timing constraints of tasks in the system.

1.1 Block Diagram of embedded system:





The embedded system consist of different components embedded into it as follows: Embedded Processor 2) Power supply , reset and Oscillator circuits 3) System timers 4) Serial communication port 5) Parallel ports 6) Interrupt controller 7) Output and Input Interfacing and driver circuits 8) System application specific circuits 9) Program and Data memory

Processor: The processor is the heart of embedded system. The selection of processor is based on the following consideration 1. Instruction set 2. Maximum bits of operation on single arithmetic and logical operation 3. Speed 4. Algorithms processing and capability 5. Types of processor(microprocessor, microcontroller, digital signal processor, application specific processor, general purpose processor)

Power supply, Reset, Oscillator circuit and system timers:

Power source: Internal power supply is must. Es require from power up to power down to start time task. Also it can run continuously that is stay “On’ system consumes total power hence efficient real time programming by using proper ‘wait’ and ‘stop’ instruction or disable some unit which are not in use can save or limit power consumption.

Clock / oscillator Circuits: The clock ckt is used for CPU, system timers, and CPU machine cycles clock controls the time for executing an instruction. Clock oscillator may be internal or external .It should be highly stable.

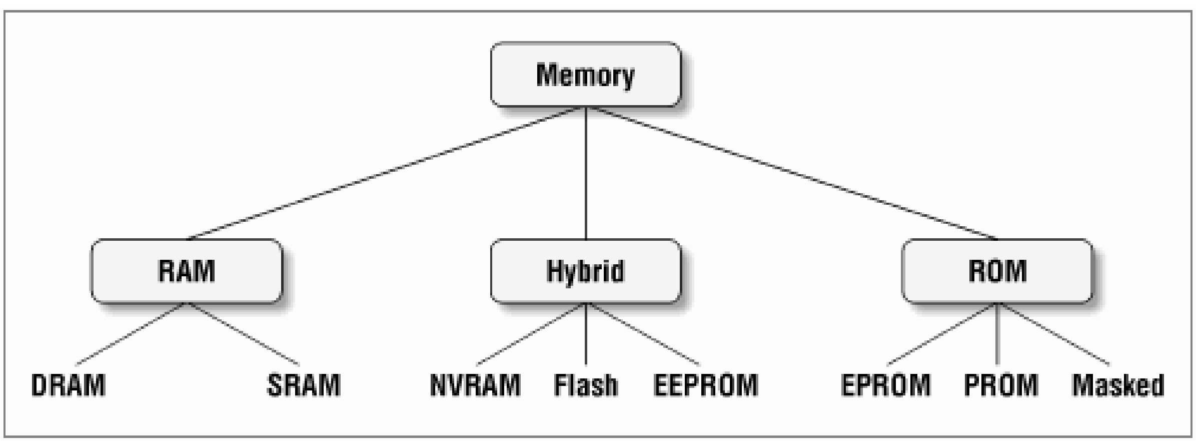
Real time clock(RTC): It require to maintain scheduling various tasks and for real time programming RTC also use for driving timers, counters needs in the system. Resets Ckts and power on reset: Reset process starts executing various instruction from the starting address. The address is set by the processor in the program counter. The reset step reset and runs the program in the following way 1. System program that execute from beginning 2. System boot up program 3. System initialization program

Serial and Parallel communication ports: Serial communication port and parallel communication ports are used to interface serial and parallel devices with the system and communicate between processor and devices.

Interrupt controller: It is used to receive interrupt from various sources and resolve the priority and provides the service to that interrupts.

Input and output interfacing and driver circuits: Characteristics of input or output devices may be different from the processor like voltage and current requirement to drive that specific device, hence driver circuits are needed to drive input or output devices.

Program and Data memory: The most microcontroller have inbuilt separate memory to store data and program. Following are types of memories used in embedded system.

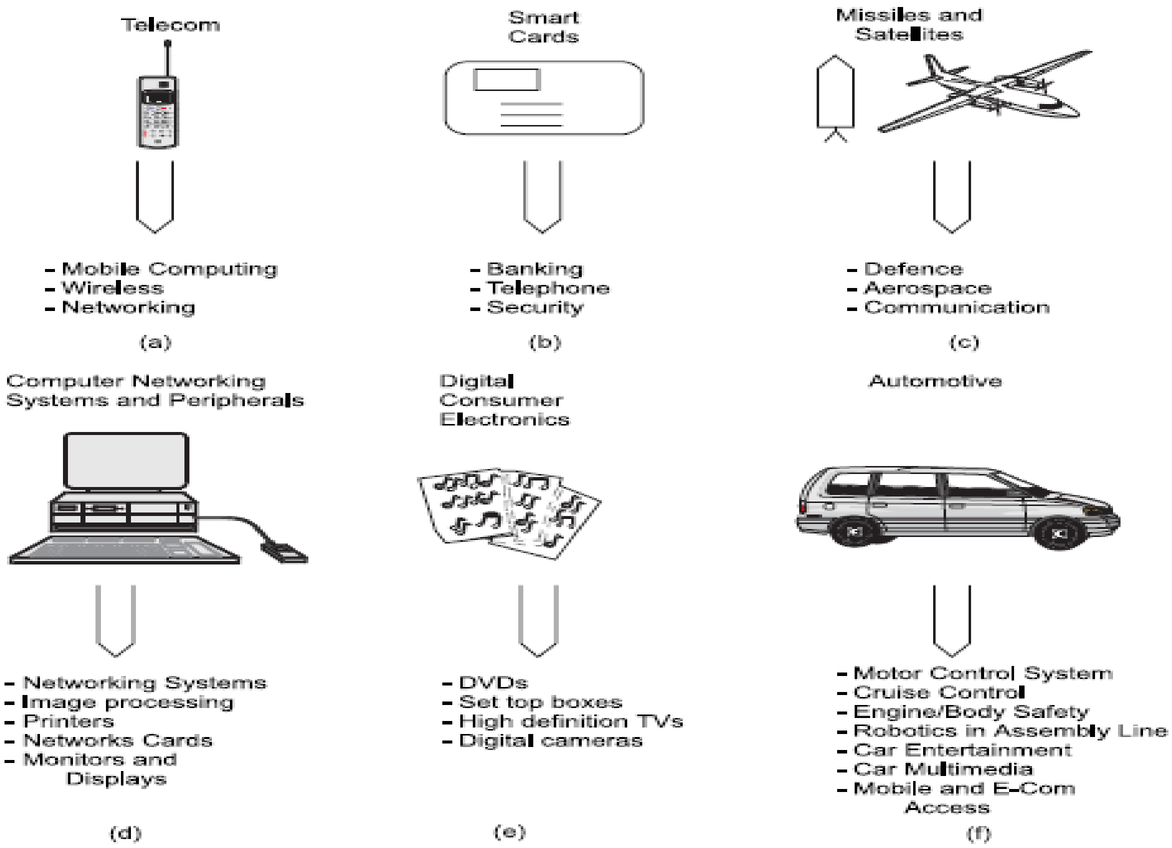


Applications of Embedded Systems:

Embedded systems are used in different applications like automobiles, telecommunications, smart cards, missiles, satellites, computer networking and digital consumer electronics.

Embedded Systems in Automobiles and in telecommunications

- Motor and cruise control system
- Body or Engine safety
- Entertainment and multimedia in car
- E-Com and Mobile access
- Robotics in assembly line
- Wireless communication
- Mobile computing and networking



Embedded Systems in Smart Cards, Missiles and Satellites

- Security systems
- Telephone and banking
- Defense and aerospace
- Communication

Embedded Systems in Peripherals & Computer Networking

- Displays and Monitors
- Networking Systems
- Image Processing
- Network cards and printers

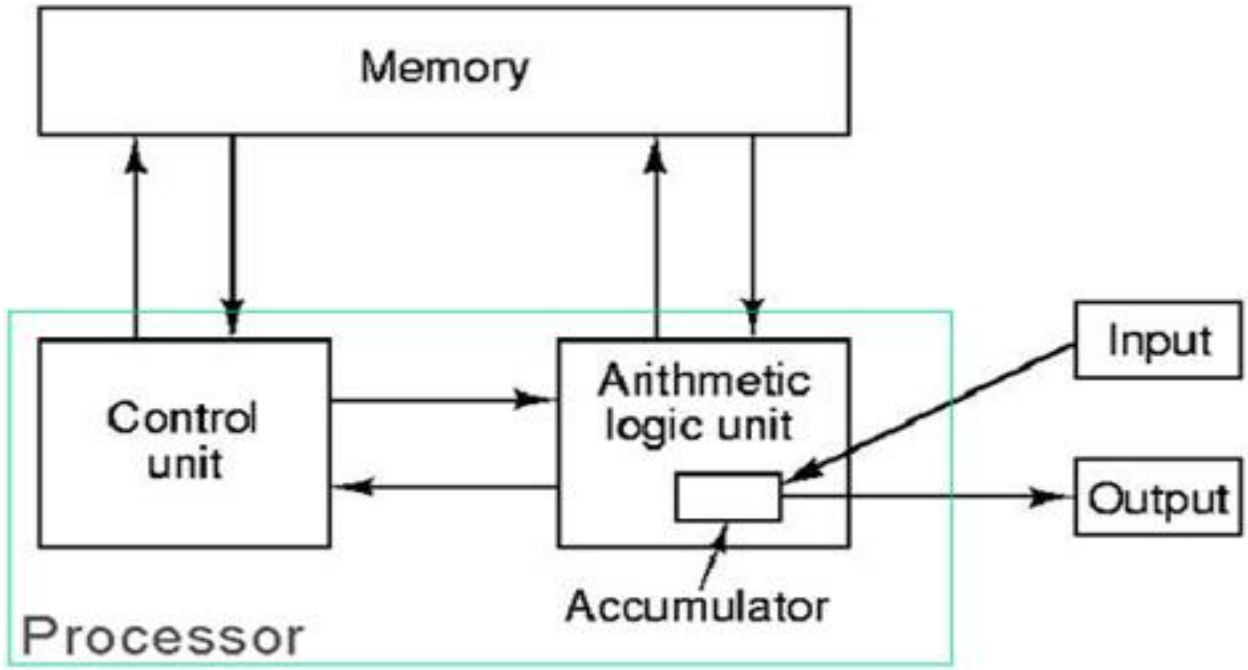
Embedded Systems in Consumer Electronics

- Digital Cameras
- Set top Boxes
- High Definition TVs
- DVDs

1.2 Processor Architectures

a) VON NEUMANN Architecture

- All parts of a computer are connected together by Bus,
- Computer structure is independent on the computed problem, a computer is programmed with content of memory,
- Every computing step depends on the previous step,
- Machine instruction and data are in the same memory,
- Memory is split to small cells with the same size. Their ordinal numbers are called address number
- Program consists of a sequence of instructions. Instructions are executed in order they are stored in memory.
- Sequence of instructions can be changed only by unconditional or conditional jump instructions.
- Instructions, characters, data and numbers are represented in binary form



ADVANTAGES

- Programmers organize the content of the memory and they can use the whole capacity of the installed memory.
- One bus is simpler for the Control Unit design.
- Development of the Control Unit is cheaper and faster.
- Computer with one bus is cheaper.
- Data and instruction are accessed in the same way

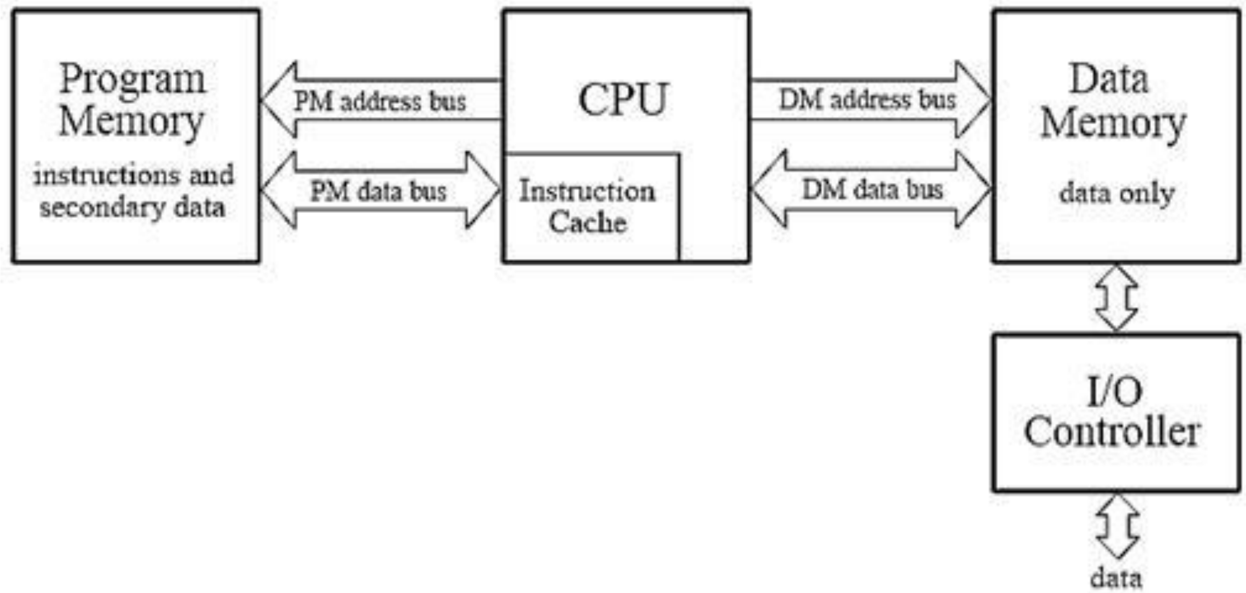
DISADVANTAGES

- Serial instruction processing does not allow parallel execution of program.
- One bus is a bottleneck. Only one information can be accessed at the same time.
- Instruction stored in the same memory as the data can be accidentally rewritten by an error in a program.

b) HARVARD architecture

- MARK II computer was finished at Harvard University in 1947.
- It wasn't so modern as the computer from von Neumann team. But it introduced a slightly different architecture.
- Memory for data was separated from the memory for instruction.

This concept is known as the Harvard architecture.



ADVANTAGES

- Two memories with two Buses allow parallel access to data and instructions. Execution can be 2x faster.
- Both memories can be produced by different technologies (Flash/EEPROM, SRAM/DRAM).
- Both memories can use different cell sizes.
- Program can't rewrite itself.

DISADVANTAGES

- Control unit for two Buses is more complicated and more expensive

- Production of a computer with two Buses is more expensive.
- Development of a complicated Control Unit needs more time.
- Free data memory can't be used for instruction and vice-versa.

c) CISC - Complex Instruction Set Computer

It has "High level" Instruction Set. For each instruction several "low level operations" are executed.

Ex: load, arithmetic operation, memory store.

Features:

- Instructions can operate directly on memory
- Small number of general purpose registers
- Instructions take multiple clocks to execute
- Few lines of code per operation

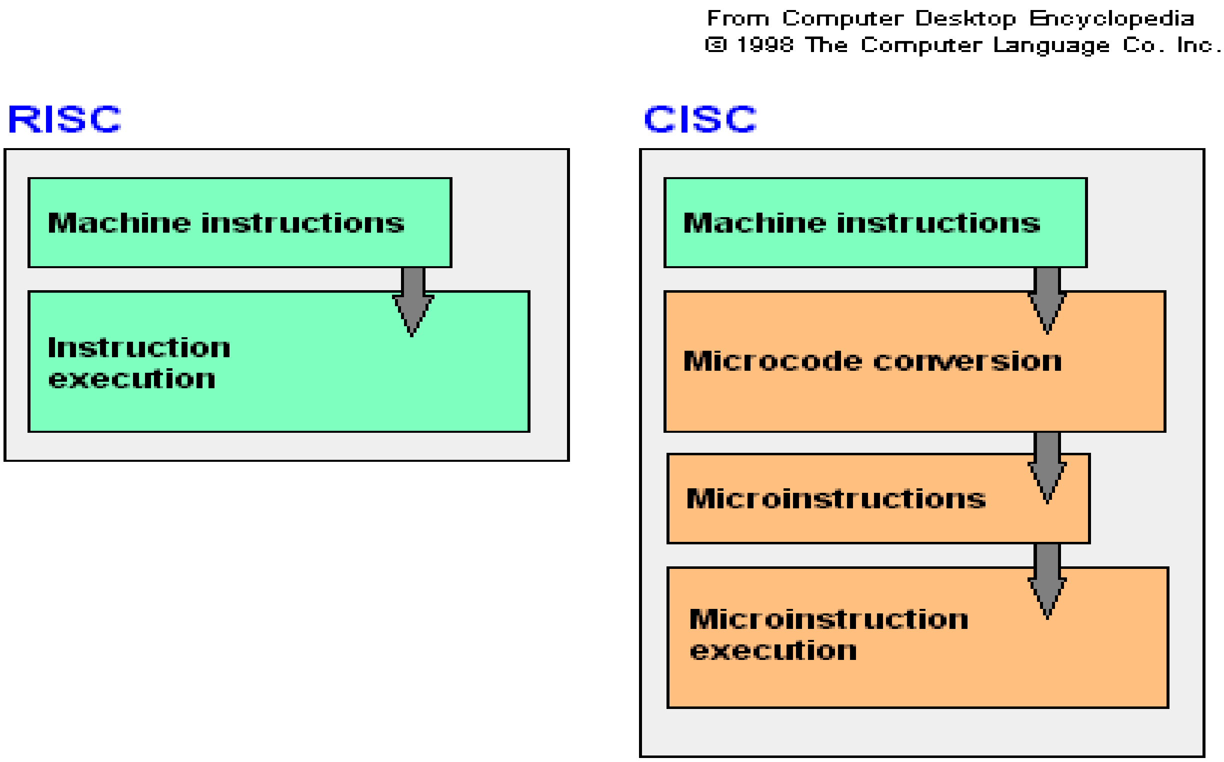
d) RISC - Reduced Instruction Set Computer

- RISC is a CPU design that recognizes only a limited number of instructions
  - Simple instructions
  - Instructions are executed quickly

Features:

Executes a series of simple instruction instead of a complex instruction

- Instructions are executed within one clock cycle
- Incorporates a large number of general registers for arithmetic operations to avoid storing variables on a stack in memory
- Only the load and store instructions operate directly onto memory
- Pipelining = speed



CISC	RISC
Complex instructions require multiple cycles	Reduced instructions take 1 cycle
Many instructions can reference memory	Only Load and Store instructions can reference memory
Instructions are executed one at a time	Uses pipelining to execute instructions
Few general registers	Many general registers
Emphasis on hardware	Emphasis on software

1.3 Features and Applications of Microcontrollers

a) 89C51

Features:

- The 89c51 is compatible with MCS 51 family.
- It has 8 bit data bus and 8 bit (arithmetic logic unit). *
- It has 4k bytes of on chip reprogrammable flash memory. *
- It supported three level program memory lock.
- It has 16 bit of address bus and 64 kb of RAM (random access memory) and ROM (read only memory). *
- It has on chip RAM 128 bytes data memory. *
- It has power saving mode.
- It has four 8 bit bidirectional input or output ports that is 32 programmable input or output lines. *
- It can execute 1 million one cycle instructions per second with a clock frequency of 12MHz.
- It has one UART programmable serial ports. *
- It has six interrupts source. *
- It has two multimode 16 bit timers. *

Applications:

Energy Management: aid in calculating energy consumption in domestic and industrialized applications.

Automobiles: The microcontroller 8051 discovers broad recognition in supplying automobile solutions. They are extensively utilized in hybrid motor vehicles to control engine variations. In addition, works such as cruise power and anti-brake mechanism has created it more capable with the amalgamation of micro-controllers.

Medical Devices: Handy medicinal gadgets such as glucose & blood pressure monitors bring into play micro-controllers, to put on view the measurements, as a result, offering higher dependability in giving correct medical results.

Consumer Electronics Products:

Toys, Cameras, Robots, Washing Machine, Microwave Ovens etc. [any automatic home appliance]

Instrumentation and Process Control:

Oscilloscopes, Multi-meter, Leakage Current Tester, Data Acquisition and Control etc.

b) PIC microcontrollers (Peripheral Interface Controller) – MICROCHIP

Features:

- RISC architecture : Only 35 instructions to learn, All single-cycle instructions except branches *
- Operating frequency 0-20 MHz*
- Precision internal oscillator : Factory calibrated, Software selectable frequency range of 8MHz to 31KHz
- Power supply voltage 2.0-5.5V*
- Power-Saving Sleep Mode
- Brown-out Reset (BOR) with software control option (processor to reset (or reboot) in the event of a brownout, which is a significant drop in the power supply output voltage.) *
- 35 input/output pins: High current source/sink for direct LED drive*
- 8K ROM memory in FLASH technology , Chip can be reprogrammed up to 100.000 times*

- *In-Circuit Serial Programming* Option: Chip can be programmed even embedded in the target device
- *256 bytes EEPROM memory*, Data can be written more than 1.000.000 times*
- *368 bytes RAM memory**
- *A/D converter:* 14-channels, 10-bit resolution*
- 3 independent timers/counters*
- Watch-dog timer*
- Analogue comparator module with
    - Two analogue comparators
    - Fixed voltage reference (0.6V)
    - Programmable on-chip voltage reference
- PWM output steering control*
- Enhanced USART module*
    - Supports RS-485, RS-232 and LIN2.0
    - Auto-Baud Detect
- Master Synchronous Serial Port (MSSP)*
    - supports SPI and I2C mode

Applications:

The PIC microcontrollers are commonly used in the following applications:

Household Appliances: Washing machine, Remote control, light control, video games, TV tuner, intercom and sewing machine

Office Equipment: Telephones, fax machines, printers and security systems

Instruments: Digital thermometers, level meters, and multi-meters

Peripheral Devices: Keyboard controllers, modems and plotters

Motor Control: Speed control of ac and dc motors, position control using servomotors and stepper motors.

Industry: Process control systems and automobile applications.

c) ARM microcontrollers (Advanced RISC Machines) – ARM holdings

- It is a Reduced Instruction Set Computing (RISC) controller. It has a high performance CPU of 32 bits and the pipelining is done through 3 stages.*
- The Thumb-2 technology has been integrated in these controllers, which means they can handle 16 bit as well as 32 bit instructions. This technology also provides high performance in operations and executions.*
- It has low power modes. Sleep modes are also supported by it.
- The NVIC, Nested Vectored interrupt controller provides low latency as well as low jitter interrupts response. Another advantage is that there is no need of assembly programming in it.*
- ISP (in system programming) or IAP (in application programming) using on-chip boot loader software.*
- On-chip static RAM is 8 kB-40 kB, on-chip flash memory is 32 kB-512 kB, the wide interface is 128 bit, or accelerator allows 60 MHz high-speed operation.*

- One or two 10-bit ADCs offer 6 or 14 analogs i/ps with low conversion time as 2.44 µs/ channel.*
- External event counter/32 bit timers-2, PWM unit, & watchdog.*
- Low power RTC (real time clock) *
- Several serial interfaces like two 16C550 UARTs, two I2C-buses with 400 kbit/s speed.*
- The incorporated oscillator on the chip will work by an exterior crystal that ranges from 1 MHz-25 MHz*
- There are 43 I/O lines*

Applications:

- ARM processors are extensively used in consumer electronic devices such as smartphones, tablets, multimedia players and other mobile devices, such as wearables.
- ARM microcontrollers can also be used in space and aerospace technologies.
- Used in many medical equipments such as MRI, CT scanner, ultrasound and implantable devices.
- Also used at the research level in particle accelerators, nuclear reactors and X-ray cargo scanning applications.

d) AVR Advanced Virtual RISC – ATMEL

Features:

- 32 x 8 general working purpose registers.*
- 32K bytes of in system self-programmable flash program memory*
- 2K bytes of internal SRAM*
- 1024 bytes EEPROM*
- Available in 40 pin DIP, 44 lead QTFP, 44-pad QFN/MLF

- 32 programmable I/O lines*
- 8 Channel, 10 bit ADC*
- Two 8-bit timers/counters with separate prescalers and compare modes and One 16-bit timer/counter with separate prescaler, compare mode and capture mode.*
- 4 PWM channels*
- In system programming by on-chip boot program
- Programmable watch dog timer with separate on-chip oscillator.*
- Programmable serial USART and Master/slave SPI serial interface*
- Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby and extended standby.
- Internal calibrated RC oscillator
- External and internal interrupt sources
- Power on reset and programmable brown-out detection.

Applications:

Atmel microcontroller programming is cheap and really tiny in size. Hence anyone can embed on a various device.

- Mobile Phones
- Auto Mobiles
- CD/DVD Players
- Washing Machines
- Cameras
- Modems and Keyboard Controllers
- Security Alarms

- Electronic Measurement Instruments
- Microwave Oven

1.4 Characteristics or Design metrics of Embedded Systems

The embedded-system designer must of course construct an implementation that fulfills desired functionality, but a difficult challenge is to construct an implementation that simultaneously optimizes numerous design metrics. A design metric is a measurable feature of a system's implementation. Common relevant metrics include:

1. Power Dissipation: For battery operated system this is important feature. Examples are mobile phone or digital camera where if power dissipation is small battery needs to be recharge less frequently.

2. Unit cost: the monetary cost of manufacturing each copy of the system, excluding NRE cost.

3. *NRE cost (Non-Recurring Engineering cost):* The monetary cost of designing the system. Once the system is designed, any number of units can be manufactured without incurring any additional design cost (hence the term "non-recurring").

4. Size: the physical space required by the system, often measured in bytes for software, and gates or transistors for hardware.

5. Memory: RAM in KB, internal flash memory requirements in MB or GB for running software and for data storage.

6. *Performance:* the execution time or throughput of the system. Instruction execution time in the system measures performance. Smaller execution time means higher performance. For eg. in mobile phone, voice signals are processed between antenna and speaker in 0.1s shows phone performance.

7. *Power:* the amount of power consumed by the system, which determines the lifetime of a battery, or the cooling requirements of the IC, since more power means more heat.

8. *Flexibility:* the ability to change the functionality of the system without incurring heavy NRE cost. Software is typically considered very flexible. Flexibility in design enables, without significant engineering cost, development of different versions or product or to develop advanced version later on. For example software enhancement by adding extra functions.

9. Reliability: It is measure of how much % you can rely upon the proper functioning of system. Mean Time Between Failure (MTBF) and Mean Time To Repair (MTTR) are used in determining reliability. MTBF gives the frequency of failures in hours/weeks/months.MTTR specifies how long the system is allowed to be out of order following a failure.

10. Maintainability: Deals with support and maintenance to the end user or client in case of technical issues and product failure. A more reliable system means with less maintainability.As reliability of the system increases chances of failure and non-functioning also reduces.

11. *Time-to-market:* The amount of time required to design and manufacture the system to the point the system can be sold to customers. The main contributors are design time, manufacturing time and testing time. There may be multiple players in the embedded industry who develop products of the same category (like mobile phones, portable media players etc.). If you come with new product and time to market is high competitor may take advantage of it with their product.

12. *Time-to-prototype:* The amount of time to build a working version of the system, which may be bigger or more expensive than the final system implementation, but can be used to verify the system's usefulness and correctness and to refine the system's functionality. If the prototype is developed faster, the actual estimated development time can be bought down.

13. *Correctness:* our confidence that we have implemented the system's functionality correctly. We can check the functionality throughout the process of designing the system, and we can insert test circuitry to check that manufacturing was correct.

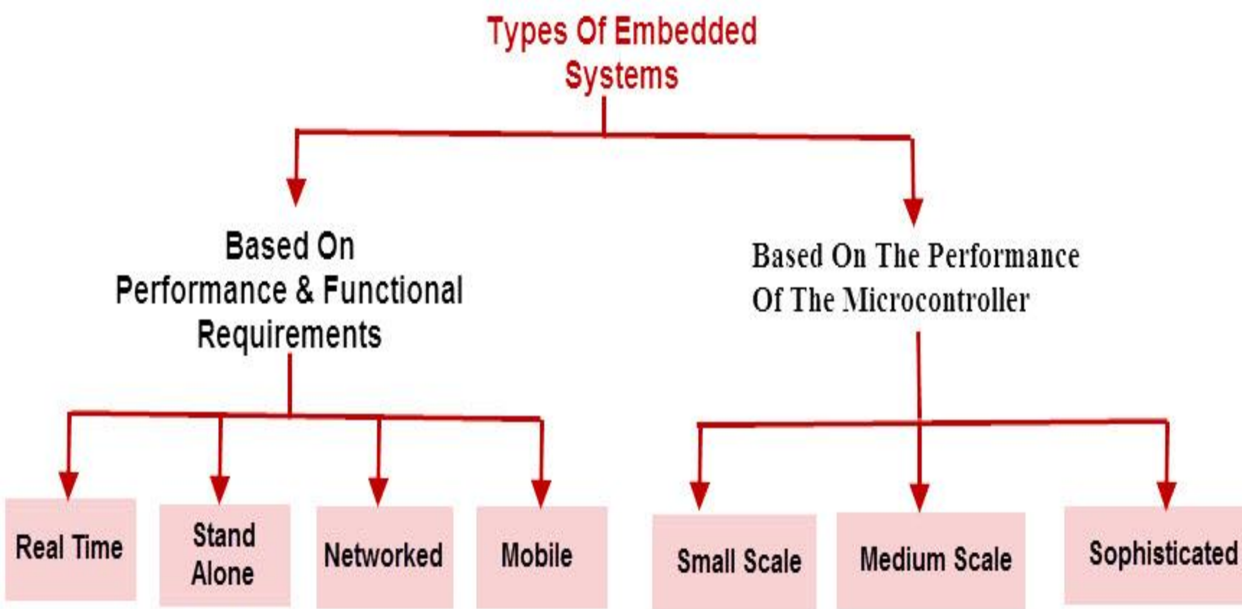
14. *Safety:* the probability that the system will not cause harm. It deals with possible damages that can happen to the operators, public and the environment due

to breakdown of embedded system, or due to the emission of radioactive or hazardous materials from embedded products. Safety analysis is a must in product engineering to evaluate the anticipated damages and determine best course of action.

15. Operating System : Embedded system should embed a real time operating system (RTOS), which supervises the application software tasks running on the hardware and organizes the accesses to system resources according to priorities and timing constraints of tasks in the system.

1.5 Types of Embedded Systems

Embedded systems can be classified into different types based on performance, functional requirements and performance of the microcontroller.



Types of Embedded systems

Embedded systems are classified into four categories based on their performance and functional requirements:

- Stand alone embedded systems
- Real time embedded systems
- Networked embedded systems
- Mobile embedded systems

Embedded Systems are classified into three types based on the performance of the microcontroller such as

- Small scale embedded systems
- Medium scale embedded systems
- Sophisticated embedded systems

Stand Alone Embedded Systems

Stand alone embedded systems do not require a host system like a computer, it works by itself. It takes the input from the input ports either analog or digital and processes, calculates and converts the data and gives the resulting data through the connected device, which either controls, drives and displays the connected devices. Examples for the stand alone embedded systems are mp3 players, digital cameras, video game consoles, microwave ovens and temperature measurement systems.

Real Time Embedded Systems

A real time embedded system is defined as a system which gives a required o/p in a particular time. These types of embedded systems follow the time deadlines for completion of a task. Real time embedded systems are classified into two types such as soft and hard real time systems .Example is autopilot system in a flight.Two types of real time systems are:

Hard Real time system

- This system guarantees that critical tasks complete on time.
- Many of these are found in industrial process control, avionics, and military and similar application areas.
- This goal says that all delays in the system must be restricted.

- These systems must provide absolute guarantees that a certain action will occur by a certain time.
- Examples
    - Chemical and nuclear plant control
    - Autopilot System In Plane
    - Pacemakers

Soft Real-Time System

- In this type of system, missing an occasional deadline, while not desirable, is acceptable and does not cause any permanent damage.
- Digital audio, digital telephone, and multimedia systems fall into this category.
- Examples
    - Mobile Communication
    - Music Playing Robots
    - Weather Station

Networked Embedded Systems

These types of embedded systems are related to a network to access the resources. The connected network can be LAN, WAN or the internet. The connection can be any wired or wireless. This type of embedded system is the fastest growing area in embedded system applications. The embedded web server is a type of system wherein all embedded devices are connected to a web server and accessed and controlled by a web browser. Example for the LAN networked embedded system is a home security system wherein all sensors are connected and run on the protocol TCP/IP.

Mobile Embedded Systems

Mobile embedded systems are used in portable embedded devices like cell phones, mobiles, digital cameras, mp3 players and personal digital assistants, etc.The basic limitation of these devices is the other resources and limitation of memory.

Small Scale Embedded Systems

These types of embedded systems are designed with a single 8 or 16-bit microcontroller that may even be activated by a battery. For developing embedded software for small scale embedded systems, the main programming tools are an editor, assembler, cross assembler and integrated development environment (IDE).

Medium Scale Embedded Systems

These types of embedded systems design with a single or 16 or 32 bit microcontroller, RISCs or DSPs. These types of embedded systems have both hardware and software complexities. For developing embedded software for medium scale embedded systems, the main programming tools are C, C++, JAVA, Visual C++, RTOS, debugger, source code engineering tool, simulator and IDE.

Sophisticated Embedded Systems

These types of embedded systems have enormous hardware and software complexities that may need ASIPs, IPs, PLAs, scalable or configurable processors. They are used for cutting-edge applications that need hardware and software Co-design and components which have to assemble in the final system.