What is an Embedded System?

- An embedded system is an electronic/electro-mechanical system designed to perform a specific function and is a combination of both hardware and firmware (software).
- Every embedded system is unique and the hardware as well as the firmware is highly specialised to the application domain.

Embedded Systems vs. General Computing Systems

- The computing revolution began with the general purpose computing requirements. Later it was realised that the general computing requirements are not sufficient for the embedded computing requirements.
- The embedded computing requirements demand ‘something special’ in terms of response to stimuli, meeting the computational deadlines, power efficiency, limited memory capability, etc.

General Purpose Computing System	Embedded System
A system which is a combination of a generic hardware and a General Purpose Operating System for executing a variety of applications	A system which is a combination of special purpose hardware and embedded OS for executing a specific set of applications
Contains a General Purpose Operating System (GPOS)	May or may not contain an operating system for functioning
Applications are alterable (programmable) by the user (It is possible for the end user to re-install the operating system, and also add or remove user applications)	The firmware of the embedded system is pre-programmed and it is non-alterable by the end-user (There may be exceptions for system supporting OS kernel image flashing through special hardware settings)
Performance is the key deciding factor in the selection of the system. Always, ‘Faster is Better’	Application-specific requirements (like performance, power requirements, memory usage, etc.) are the key deciding factors
Less/not at all tailored towards reduced operating power requirements, options for different levels of power management	Highly tailored to take advantage of the power saving modes supported by the hardware and the operating system
Response requirements are not time-critical	For certain category of embedded systems like mission critical systems, the response time requirement is highly critical
Need not be deterministic in execution behaviour	Execution behaviour is deterministic for certain types of embedded systems like ‘Hard Real Time’ systems

History of Embedded Systems

- Embedded systems were in existence even before the IT revolution.
  - Built around the old vacuum tube and transistor technologies.
- Advances in semiconductor and nanotechnology and IT revolution gave way to the development of miniature embedded systems.
- The first recognised modern embedded system is the Apollo Guidance Computer (AGC) developed by the MIT Instrumentation Laboratory for the lunar expedition.
  - It had 36K words of fixed memory and 2K words of erasable memory.
  - The clock frequency of was 1.024 MHz and it was derived from a 2.048 MHz crystal clock.
- The first mass-produced embedded system was the Autonetics D-17 guidance computer for the Minuteman-I missile in 1961.
  - It was built using discrete transistor logic and a hard-disk for main memory.
- The first integrated circuit was produced in September 1958 and computers using them began to appear in 1963.

Classification of Embedded Systems

- Some of the criteria used in the classification of embedded systems are:
  1. Based on generation
  2. Complexity and performance requirements
  3. Based on deterministic behaviour
  4. Based on triggering

Classification Based on Generation

- First Generation
- Second Generation
- Third Generation
- Fourth Generation
- Next Generation

Classification Based on Generation (continued)

- First Generation
  - Early embedded systems were built around 8-bit microprocessors like 8085 and Z80 and 4-bit microcontrollers.
  - Simple in hardware circuits with firmware developed in assembly code.
  - E.g.: Digital telephone keypads, stepper motor control units, etc.

Classification Based on Generation (continued)

- Second Generation
  - Embedded systems built around 16-bit microprocessors and 8-bit or 16-bit microcontrollers.
  - Instruction set were much more complex and powerful than the first generation.
  - Some of the second generation embedded systems contained embedded operating systems for their operation.
  - E.g.: Data acquisition systems, SCADA systems, etc.

Classification Based on Generation (continued)

- Third Generation
  - Embedded systems built around 32-bit microprocessors and 16-bit microcontrollers.
  - Application and domain specific processors/controllers like Digital Signal Processors (DSP) and Application Specific Integrated Circuits (ASICs) came into picture.
  - The instruction set of processors became more complex and powerful and the concept of instruction pipelining also evolved.
  - Dedicated embedded real time and general purpose operating systems entered into the embedded market.
  - Embedded systems spread its ground to areas like robotics, media, industrial process control, networking, etc.

Classification Based on Generation (continued)

- Fourth Generation
  - The advent of System on Chips (SoC), reconfigurable processors and multicore processors are bringing high performance, tight integration and miniaturisation into the embedded device market.
  - The SoC technique implements a total system on a chip by implementing different functionalities with a processor core on an integrated circuit.
  - They make use of high performance real time embedded operating systems for their functioning.
  - E.g.: Smart phone devices, Mobile Internet Devices (MIDs), etc.

Classification Based on Generation (continued)

- Next Generation
  - The processor and embedded market is highly dynamic and demanding.
  - The next generation embedded systems are expected to meet growing demands in the market.

Classification of Embedded Systems

- Some of the criteria used in the classification of embedded systems are:
  1. Based on generation
  2. Complexity and performance requirements
  3. Based on deterministic behaviour
  4. Based on triggering

Classification Based on Complexity and Performance

- Small-Scale Embedded Systems
- Medium-Scale Embedded Systems
- Large-Scale Embedded Systems

Classification Based on Complexity and Performance (continued)

- Small-Scale Embedded Systems
  - Simple in application needs and the performance requirements are not time critical.
  - E.g.: An electronic toy
  - Usually built around low performance and low cost 8-bit or 16-bit microprocessors/microcontrollers.
  - May or may not contain an operating system for its functioning.

Classification Based on Complexity and Performance (continued)

- Medium-Scale Embedded Systems
  - Slightly complex in hardware and firmware (software) requirements.
  - Usually built around medium performance, low cost 16-bit or 32-bit microprocessors/microcontrollers or digital signal processors.
  - Usually contain an embedded operating system (either general purpose or real time operating system) for functioning.

Classification Based on Complexity and Performance (continued)

- Large-Scale Embedded Systems
  - Highly complex in hardware and firmware (software) requirements.
  - They are employed in mission critical applications demanding high performance.
  - Usually built around high performance 32-bit or 64-bit RISC processors/controllers or Reconfigurable System on Chip (RSoC) or multi-core processors and programmable logic devices.
  - May contain multiple processors/controllers and co-units/hardware accelerators for offloading the processing requirements from the main processor of the system.
  - Decoding/encoding of media, cryptographic function implementation, etc. are examples of processing requirements which can be implemented using a co-processor/hardware accelerator.
  - Usually contain a high performance real time operating system (RTOS) for task scheduling, prioritization and management.

Classification of Embedded Systems

- Some of the criteria used in the classification of embedded systems are:
  1. Based on generation
  2. Complexity and performance requirements
  3. <u>Based on deterministic behaviour</u>
  4. Based on triggering

Classification Based on Deterministic Behaviour

- Applicable for ‘Real Time’ systems.
- The application/task execution behaviour can be either deterministic or non-deterministic.
- Based on the execution behaviour, real time embedded systems are classified into Hard Real Time and Soft Real Time systems.

Classification Based on Triggering

- Embedded systems which are ‘Reactive’ in nature (like process control systems in industrial control applications) can be classified based on the trigger.
- Reactive systems can be either event-triggered or time-triggered.

Major Application Areas of Embedded Systems

1. Consumer electronics: Camcorders, cameras, etc.
2. Household appliances: Television, DVD players, washing machine, refrigerators, microwave oven, etc.
3. Home automation and security systems: Air conditioners, sprinklers, intruder detection alarms, closed circuit television (CCTV) cameras, fire alarms, etc.
4. Automotive industry: Anti-lock braking systems (ABS), engine control, ignition systems, automatic navigation systems, etc.
5. Telecom: Cellular telephones, telephone switches, handset multimedia applications, etc.

Major Application Areas of Embedded Systems (continued)

6. Computer peripherals: Printers, scanners, fax machines, etc.
7. Computer networking systems: Network routers, switches, hubs, firewalls, etc.
8. Healthcare: Different kinds of scanners, EEG, ECG machines, etc.
9. Measurements & Instrumentation: Digital multimeters, digital CROs, logic analyzers, PLC systems, etc.
10. Banking & Retail: Automated teller machines (ATM) and currency counters, point of sales (POS), etc.
11. Card readers: Barcode, smart card readers, hand held devices, etc.

Purpose of Embedded Systems

- Each embedded system is designed to serve the purpose of any one or a combination of the following tasks:

1. Data Collection/Storage/Representation
2. Data Communication
3. Data (Signal) Processing
4. Monitoring
5. Control
6. Application Specific User Interface

Purpose of Embedded Systems (continued)

- Data Collection/Storage/Representation
  - Embedded systems designed for the purpose of data collection performs acquisition of data from the external world.
  - Data collection is usually done for storage, analysis, manipulation and transmission.
  - The term "data" refers all kinds of information, viz. text, voice, image, video, electrical signals and any other measurable quantities.
  - Data can be either analog (continuous) or digital (discrete).
  - The collected data may be stored or transmitted or it may be processed or it may be deleted instantly after giving a meaningful representation.

- A digital camera is a typical example of an embedded system with data collection/storage/representation of data.
- Images are captured and the captured image may be stored within the memory of the camera.
- The captured image can also be presented to the user through a graphic LCD unit.

Purpose of Embedded Systems (continued)

- Data Communication
  - Embedded data communication systems are deployed in applications ranging from complex satellite communication systems to simple home networking systems.
  - The transmission is achieved either by a wire-line medium or by a wireless medium.
  - The data collecting embedded terminal itself can incorporate data communication units like wireless modules (Bluetooth, ZigBee, Wi-Fi, EDGE, GPRS, etc.) or wire-line modules (RS-232C, USB, TCP/IP, PS2, etc.).

Fig: A wireless network router for data communication

- Network hubs, routers, switches, etc. are typical examples of dedicated data transmission embedded systems.
- They act as mediators in data communication and provide various features like data security, monitoring etc.

Purpose of Embedded Systems (continued)

- Data (Signal) Processing
  - The data (voice, image, video, electrical signals and other measurable quantities) collected by embedded systems may be used for various kinds of data processing.
  - Embedded systems with signal processing functionalities are employed in applications demanding signal processing like speech coding, synthesis, audio video codec, transmission applications, etc.

- A digital hearing aid is a typical example of an embedded system employing data processing.
- Digital hearing aid improves the hearing capacity of hearing impaired persons.

Purpose of Embedded Systems (continued)

- Monitoring
  - Almost embedded products coming under the medical domain are used for monitoring.
  - A very good example is the electro cardiogram (ECG) machine for monitoring the heartbeat of a patient.
    - The machine is intended to do the monitoring of the heartbeat.
    - It cannot impose control over the heartbeat.
    - The sensors used in ECG are the different electrodes connected to the patient's body.
  - Some other examples of embedded systems with monitoring function are measuring instruments like digital CRO, digital multimeters, logic analyzers, etc. used in Control & Instrumentation applications.

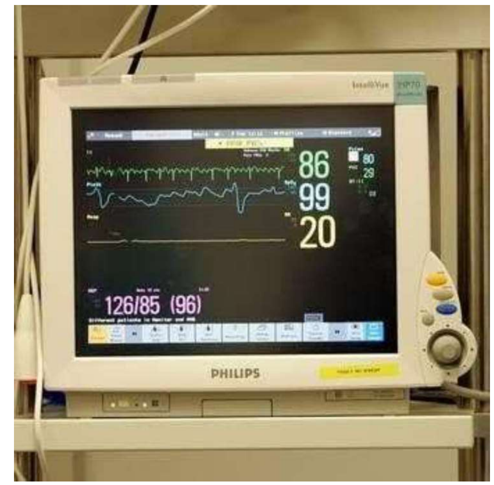

Fig: A patient monitoring system for monitoring heartbeat

Purpose of Embedded Systems (continued)

- Control
  - Embedded systems with control functionalities impose control over some variables according to the changes in input variables.
  - A system with control functionality contains both sensors and actuators.
  - Sensors are connected to the input port for capturing the changes in environmental variable or measuring variable.
  - The actuators connected to the output port are controlled according to the changes in input variable to put an impact on the controlling variable to bring the controlled variable to the specified range.

- An Air Conditioner System used to control the room temperature to a specified limit is a typical example for embedded system for control purpose.
- An air conditioner contains a room temperature-sensing element (sensor) which may be a thermistor and a handheld unit for setting up (feeding) the desired temperature.

Purpose of Embedded Systems (continued)

- Application Specific User Interface
  - These are embedded systems with application-specific user interfaces like buttons, switches, keypad, lights, bells, display units, etc.
  - Mobile phone is an example for this.
  - In mobile phone the user interface is provided through the keypad, graphic LCD module, system speaker, vibration alert, etc.

- A mobile phone is an example for embedded system with an application-specific user interfaces.