REAL-TIME & EMBEDDED SYSTEMS

DEPARTMENT OF COMPUTER SCIENCE (YEAR 4-SEM II)

By: Kumela NN.

October 2025

CHAPTER ONE: INTRODUCTION

1.1 Definitions, characteristics and examples of real-time and embedded systems

1.2 Model of real time system

1.3 Types of real time tasks

1.4 Modeling timing constraints

1.5 Computer organization concepts and memory

1.6 Design process

EMBEDDED SYSTEMS: DEFINITION

- In computing disciplines, the term "**embedded system**" is used to refer to an electronic system that is designed to perform a ***dedicated function*** and is often ***embedded*** within a larger system.
- A system that has **embedded software & hardware** that makes it a dedicated system for a specific application or a part of an application or product .
- Any system that has a microprocessor or microcontroller ***embedded*** into the system to control system.
- ***Combination*** of a computer hardware with the software to control, Monitor, Communicate or do multiple operations

EMBEDDED SYSTEMS: DEFINITION…

- An Electronic/Electro mechanical system which is designed to perform a **specific function** and is a **combination** of both hardware and firmware (Software)

  E.g. Electronic Toys, Mobile Handsets, Washing Machines, Air Conditioners, Automotive Control Units, Set Top Box, DVD Player etc…

- **Embedded Systems are:**
  - Unique in character and behavior
  - With specialized hardware and software

HISTORY OF EMBEDDED SYSTEM

First Recognized Modern Embedded System: Apollo Guidance Computer (AGC) developed by Charles Stark Draper at the MIT Instrumentation Laboratory.

- It has two modules
- 1.Command module(CM) 2.Lunar Excursion module(LEM)
- RAM size 256 , 1K ,2K words
- ROM size 4K,10K,36K words
- Clock frequency is 1.024MHz
- 5000 ,3-input RTL NOR gates are used
- User interface is DSKY(display/Keyboard)

DO YOU THINK ES IS SAME WITH COMPUTER?

Features of General Purpose Computer

- Microprocessor
- Semiconductor/Secondary Memories
- Can do multiple tasks
- More Costly
- Requires number of peripherals & their controllers
- Uses Complicated OS which requires more space

Features of Embedded system

- Microcontroller
- Uses only semiconductor memories
- Designed only for a particular predefined task
- Cheaper
- Less peripherals
- Controllers available in Microcontrollers
- Uses RTOS
- Uses less memory space

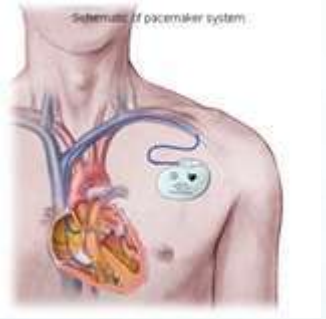
EMBEDDED SYSTEMS VS GENERAL COMPUTING SYSTEMS

General Purpose Computing System	Embedded System
A system which is a combination of generic hardware and 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
Contain a General Purpose Operating System (GPOS)	May or may not contain an operating system for functioning
Applications are alterable (programmable) by user (It is possible for the end user to re-install the Operating System, and add or remove user applications)	The firmware of the embedded system is pre-programmed and it is non-alterable by end-user
Performance is the key deciding factor on 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 hardware and 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 behavior	Execution behavior is deterministic for certain type of embedded systems like „Hard Real Time" systems

SOPHISTICATED EMBEDDED SYSTEM

- Washing Machine
- Digital Camera
- Power window of a vechicle
- Power steering of a vechicle
- Fuel Injection system of a vechicle
- AC
- Smart Phones
- Music player
- Set Top Box

EXAMPLES OF ES



CLASSIFICATION OF EMBEDDED SYSTEM

1. Based on Generation
   - G1, G2, G3, G4
2. Based on Complexity & Performance Requirements
   - Small, Medium, Large Scale
3. Based on deterministic behavior
   - deterministic or non-deterministic (Real time)
4. Based on Triggering
   - Event and Time Trigger

1. EMBEDDED SYSTEMS - CLASSIFICATION BASED ON GENERATION

- **First Generation:** The early embedded systems built around 8-bit microprocessors like 8085 and Z80 and 4-bit microcontrollers **EX. stepper motor control units, Digital Telephone Keypads etc.**
- **Second Generation:** Embedded Systems built around 16-bit microprocessors and 8 or 16-bit microcontrollers, following the first generation embedded systems **EX.SCADA, Data Acquisition Systems etc.**
- **Third Generation:** Embedded Systems built around high performance 16/32 bit Microprocessors/ controllers, Application Specific Instruction set processors like Digital Signal Processors (DSPs), and Application Specific Integrated Circuits (ASICs).The instruction set is complex and powerful.
  EX. Robotics, industrial process control, networking etc.
- **Fourth Generation:** Embedded Systems built around System on Chips (SoC's), Reconfigurable processors and multicore processors. It brings high performance, tight integration and miniaturization into the embedded device market **EX Smart phone devices, MIDs etc.**

2. EMBEDDED SYSTEMS - CLASSIFICATION BASED ON COMPLEXITY & PERFORMANCE

- **Small Scale:** The embedded systems built around low performance and low cost 8 or 16 bit microprocessors/ microcontrollers. It is suitable for simple applications and where performance is not time critical. It may or may not contain OS.
- **Medium Scale:** Embedded Systems built around medium performance, low cost 16 or 32 bit microprocessors / microcontrollers or DSPs. These are slightly complex in hardware and firmware. It may contain GPOS/RTOS.
- **Large Scale/Complex:** Embedded Systems built around high performance 32 or 64 bit RISC processors/controllers, RSoC or multi-core processors and PLD. It requires complex hardware and software. These system may contain multiple processors/controllers and co-units/hardware accelerators for offloading the processing requirements from the main processor. It contains RTOS for scheduling, prioritization and management.

3. EMBEDDED SYSTEMS - CLASSIFICATION BASED ON DETERMINISTIC BEHAVIOR

- It is applicable for Real Time systems. The application/task execution behavior for an embedded system can be either deterministic or non-deterministic

These are classified in to two types

1. Soft Real time Systems: Missing a deadline may not be critical and can be tolerated to a certain degree

2. Hard Real time systems: Missing a program/task execution time deadline can have catastrophic consequences (financial, human loss of life, etc.)

4. EMBEDDED SYSTEMS - CLASSIFICATION BASED ON TRIGGERING

- **These are classified into two types**

1. Event Triggered : Activities within the system (e.g., task run-times) are dynamic and depend upon occurrence of different events .

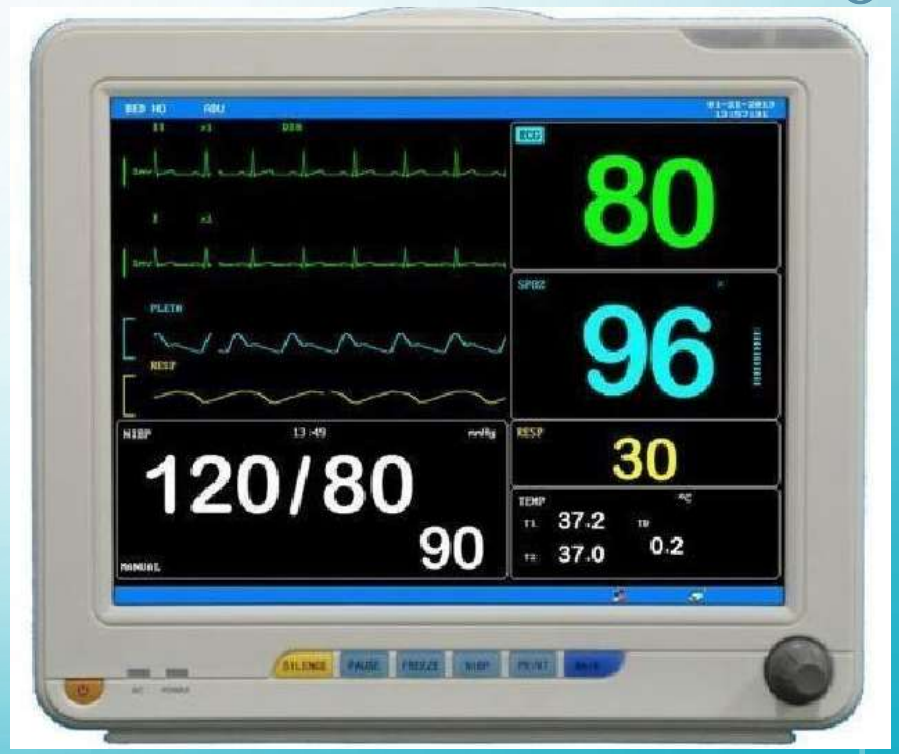
2. Time triggered: Activities within the system follow a statically computed schedule (i.e., they are allocated time slots during which they can take place) and thus by nature are predictable.

MAJOR APPLICATION AREAS OF EMBEDDED SYSTEMS

- **Consumer Electronics:** Camcorders, Cameras etc.
- **Household Appliances:** Television, DVD players, washing machine, Fridge, Microwave Oven etc.
- **Home Automation and Security Systems:** Air conditioners, sprinklers, Intruder detection alarms, Closed Circuit Television Cameras, Fire alarms etc.
- **Automotive Industry:** Anti-lock breaking systems (ABS), Engine Control, Ignition Systems, Automatic Navigation Systems etc.
- **Telecom:** Cellular Telephones, Telephone switches, Handset Multimedia Applications etc.
- **Computer Peripherals:** Printers, Scanners, Fax machines etc.
- **Computer Networking Systems:** Network Routers, Switches, Hubs, Firewalls etc.
- **Health Care:** Different Kinds of Scanners, EEG, ECG Machines etc.
- **Measurement & Instrumentation:** Digital multi meters, Digital CROs, Logic Analyzers PLC systems etc.
- **Banking & Retail:** Automatic Teller Machines (ATM) and Currency counters, Point of Sales (POS)
- **Card Readers:** Barcode, Smart Card Readers, Hand held Devices etc.

PURPOSE OF EMBEDDED SYSTEMS:

- Each Embedded Systems is designed to serve the purpose of any one or a combination of the following tasks.
  - o Data Collection/Storage/Representation
  - o Data Communication
  - o Data (Signal) Processing
  - o Monitoring
  - o Control
  - o Application Specific User Interface



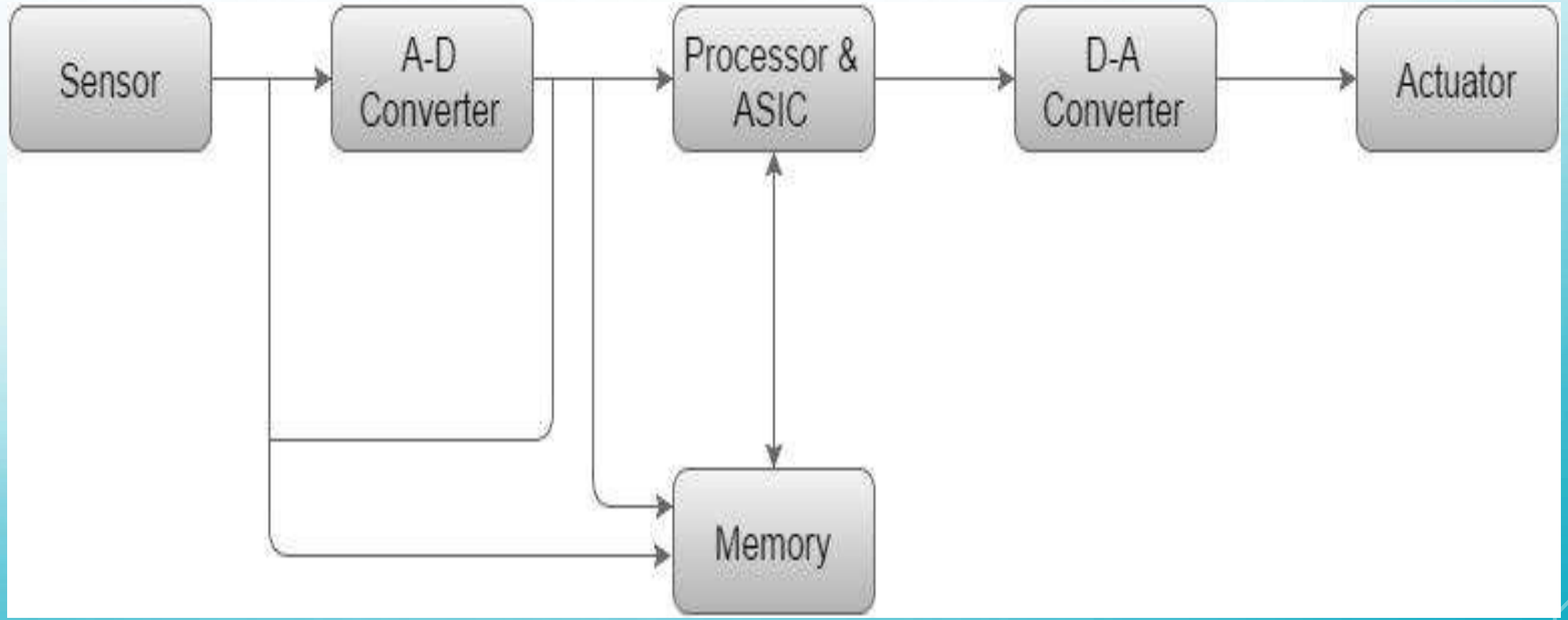
CHARACTERISTICS OF ES

- **Single-functioned** – An embedded system usually performs a specialized operation and does the same repeatedly. For example: A pager always functions as a pager.
- **Tightly constrained** – All computing systems have constraints on design metrics, but those on an embedded system can be especially tight.
- **Reactive and Real time** – Many embedded systems must continually react to changes in the system's environment and must compute certain results in real time without any delay.

CHARACTERISTICS OF ES…

- **Microprocessors based** – It must be microprocessor or microcontroller based.
- **Memory** – It must have a memory, as its software usually embeds in ROM. It does not need any secondary memories in the computer.
- **Connected** – It must have connected peripherals to connect input and output devices.
- **HW-SW systems** – Software is used for more features and flexibility. Hardware is used for performance and security.

BASIC STRUCTURE OF ES



- **Sensor** – It measures the physical quantity and converts it to an electrical signal which can be read by an observer or by any electronic instrument like an A2D converter. A sensor stores the measured quantity to the memory.
- **A-D Converter** – An analog-to-digital converter converts the analog signal sent by the sensor into a digital signal.
- **Processor & ASICs** – Processors process the data to measure the output and store it to the memory.

- **D-A Converter** – A digital-to-analog converter converts the digital data fed by the processor to analog data

- **Actuator** – An actuator compares the output given by the D-A Converter to the actual (expected) output stored in it and stores the approved output.

REAL-TIME SYSTEMS

- There are systems that need to **respond to a service request** within a certain amount of time: they are called real-time systems.
- To a real-time system, each incoming service request imposes a task (job) that is typically associated with a **real-time computing constraint**, or simply called its timing constraint.
- The **timing constraint** of a task is normally specified in terms of its **deadline**, which is the time instant by which its execution (or service) is required to be completed.

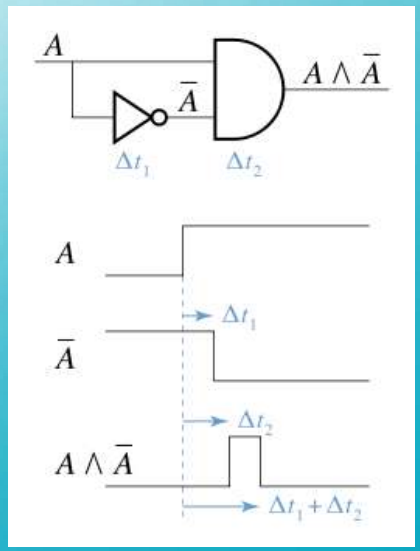
- A **timing constraint is *hard*** if the consequence of a missed deadline is fatal. A late response (completion of the requested task) is useless, and sometimes totally unacceptable.
- A **timing constraint is *soft*** if the consequence of a missed deadline is undesirable but tolerable. A late response is still useful as long as it is within some acceptable range (say, it occurs occasionally with some acceptably low probability).

REALTIME CHARACTERISTICS

- RTS have to respond to events in a certain pre-detemined amount of time.
  - The time constraints have to be considered during planning, design, implementation and testing phases.
- Internal failures due to software and hardware fault have be handled satisfactorily.
  - You cannot simply pop-up a dialog error box that says “send report” or “don’t send report”.
  - Also external failures due to outside sources need to be handled.

REALTIME CHARACTERISTICS (CONTD.)

- Typical interaction in an RTS is asynchronous. Thus an RTS should have features to handle asynchronous events such as interrupt handlers and dispatcher and associated resources.
- Potential for race condition: when state of resources are timing dependent race condition may occur.
- Periodic tasks are common.

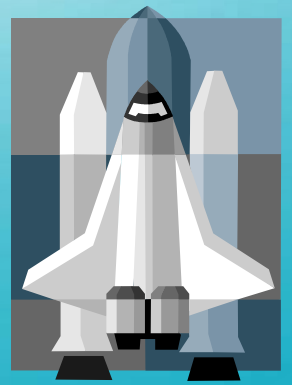


REAL-TIME SYSTEM CONCEPTS

- A system is a mapping of a set of input into a set of outputs.
- A digital camera is an example of a realtime system: set of input including sensors and imaging devices producing control signals and display information.
- Realtime system can be viewed as a sequence of job to be scheduled.
- Time between presentation of a set of inputs to a system and the realization of the required behavior, including availability of all associated outputs, is called the response time of the system.

REAL-TIME SYSTEM CONCEPTS (CONTD.)

- Real-time system is the one in which logical correctness is based on both the correctness of the output as well as their timeliness.
- A soft real-time system is one in which performance is degraded by failure to meet response-time constraints.
- A hard real-time system is one in which failure to meet a single deadline may lead to complete and catastrophic failure.
- More examples:
  - Automatic teller: soft
  - Robot vacuum cleaner: firm
  - Missile delivery system: hard



SPECTRUM OF REAL-TIME SYSTEMS

- A real-time system is called a *real-time embedded system* if it is designed to be **embedded** within some larger system.