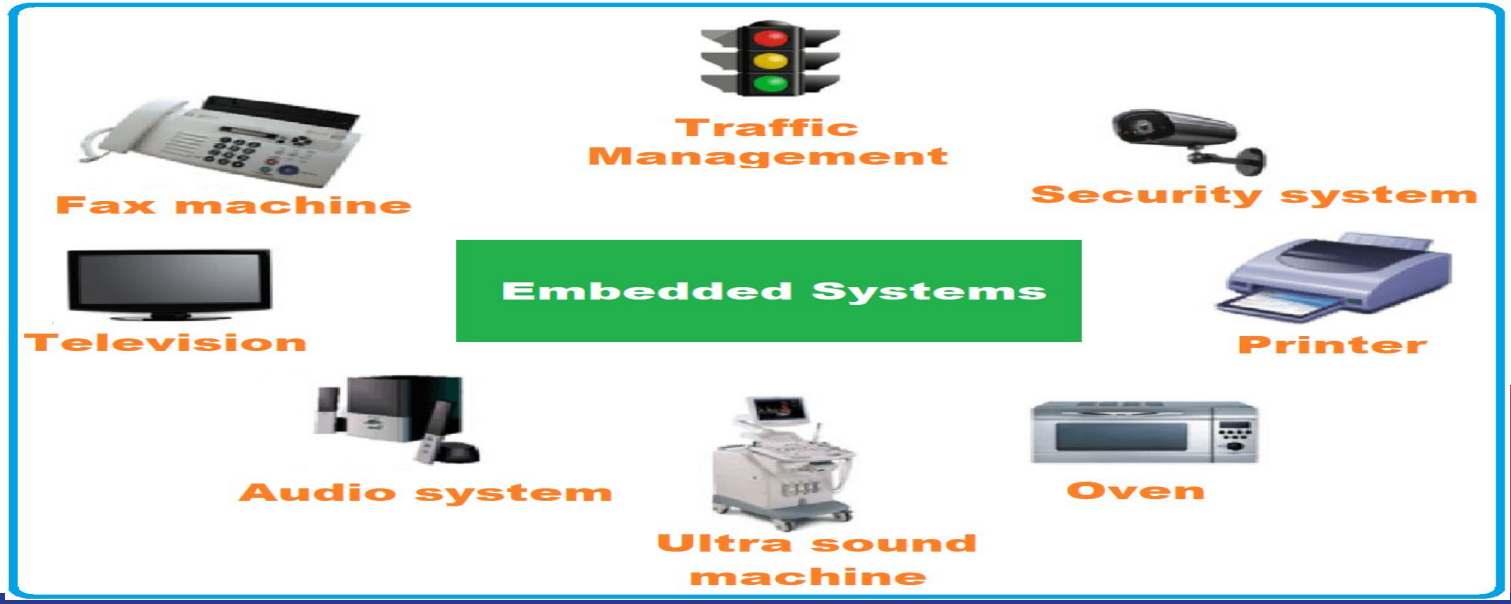
Embedded Systems

Presented by:
Sandesh Thapa Magar
ME computer

Embedded Systems

An embedded system is a microprocessor-based computer hardware system with software that is designed to perform a dedicated function, either as an independent system or as a part of a large system



Architecture of embedded systems

The basic components works as following:

1. Sensor: The sensor measures and converts the physical quantity to an electrical signal. Also stores the measured quantity to the memory.
2. A-D Converter: An analog-to-digital converter converts the analog signal sent by the sensor into a digital signal.
3. Processor: Processors assess the data to measure the output and store it to the memory.
4. D-A Converter: A digital-to-analog converter changes the digital data fed by the processor to analog data
5. 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.

Classification of Embedded Systems

Embedded Systems are classified based on the two factors:

1.Performance and Functional Requirements

Embedded Systems are divided into 4 types as follows:

i.Real-Time Embedded Systems

ii.Stand Alone Embedded Systems

iii.Networked Embedded Systems

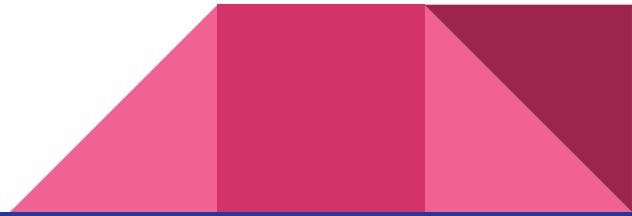
iv.Mobile Embedded Systems

2.Performance of Micro-controllers

i.Small Scale Embedded Systems

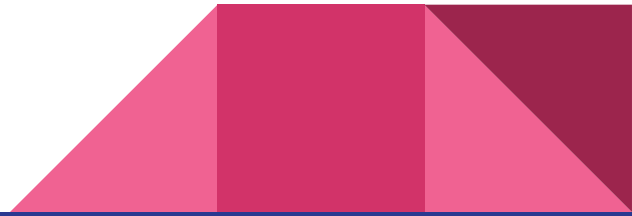
ii.Medium Scale Embedded Systems

iii.Sophisticated or Complex Embedded Systems



1.Real-Time Embedded Systems

- A Real-Time Embedded System is strictly time specific which means these embedded systems provides output in a particular/defined time interval.
- These type of embedded systems provide quick response in critical situations which gives most priority to time based task performance and generation of output.
- Real time embedded systems are used in defence sector, medical and health care sector, and some other industrial applications where output in the right time is given more importance.

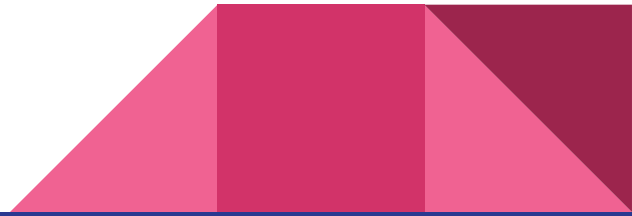


2.Stand Alone Embedded Systems

Stand Alone Embedded Systems are independent systems which can work by themselves they don’t depend on a host system. It takes input in digital or analog form and provides the output.

Examples:

- MP3 players
- Microwave ovens
- Calculator



3.Networked Embedded Systems

- Networked Embedded Systems are connected to a network which may be wired or wireless to provide output to the attached device. They communicate with embedded web server through network.

Examples:

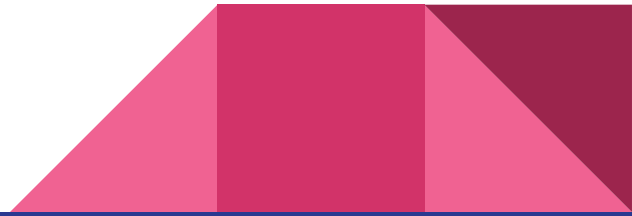
- Home security systems
- ATM machine
- Card swipe machine

4.Mobile Embedded Systems

Mobile embedded systems are small and easy to use and requires less resources. They are the most preferred embedded systems due its portability.

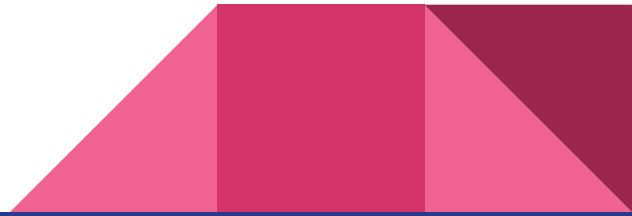
Examples:

- Mobile Phones
- Digital Camera



Embedded Operating System (EOS)

- An Embedded Operating System (EOS) is a specialized OS designed to manage hardware and software resources in embedded systems.
- Unlike general-purpose operating systems (e.g., Windows, Linux, macOS), embedded OSes are optimized for efficiency, reliability, and real-time performance in devices with limited resources.



Characteristics of an Embedded OS

1.Real-Time Performance

Some embedded OSes are real-time operating systems (RTOS), meaning they handle tasks on time, every time.
Example: A car's airbag system must deploy immediately when an accident happens—no delay is acceptable!

2. Small and Lightweight

Embedded OSes are tiny compared to regular OSes like Windows or Linux.
Example: A smartwatch OS is much smaller than a laptop OS because it only needs to do a few things.

3.Low Power Consumption

Embedded systems often run on batteries, so the OS is designed to use less energy.
Example: A fitness tracker needs an OS that doesn't drain the battery too quickly.

4.Works with Limited Resources

Embedded OSes run on devices with small memory (RAM) and slow processors compared to PCs.
Example: A washing machine OS doesn’t need powerful hardware like a gaming laptop.

5. Minimal or No User Interface

Many embedded OSes run in the background without a screen or user interaction.
Example: A traffic light control system runs automatically—you don’t interact with it.

6.Highly Reliable and Stable

Many embedded systems work 24/7 without stopping. The OS must be very reliable and rarely crash.
Example: A pacemaker in a patient’s heart must never fail, or it could be life-threatening.

Types of Embedded Operating Systems

. 1.Real-Time Operating System (RTOS)

- Used in time-sensitive applications.
- Provides deterministic task scheduling.
- Examples:
    - **FreeRTOS** – Open-source, lightweight RTOS for microcontrollers.

i.FreeRTOS

- **Type**: Real-Time Operating System (RTOS).
- **Designed For**: Microcontrollers (MCUs) with real-time constraints.
- **Task Management**: Uses a **preemptive scheduler**, meaning tasks run based on priority.
- **Best Used In**: Real-time embedded applications like **automotive, industrial automation, and robotics**.

TinyOS

-
- **Type**: Event-driven operating system.
- **Designed For**: Wireless Sensor Networks (WSNs) and IoT devices.
- **Task Management**: Uses an **event-driven model**, meaning it responds to events like sensor inputs.
- **Best Used In**: Low-power IoT applications like **environmental monitoring and smart home networks**.
- **sed In**: Low-power IoT applications like **environmental monitoring and smart home networks**.