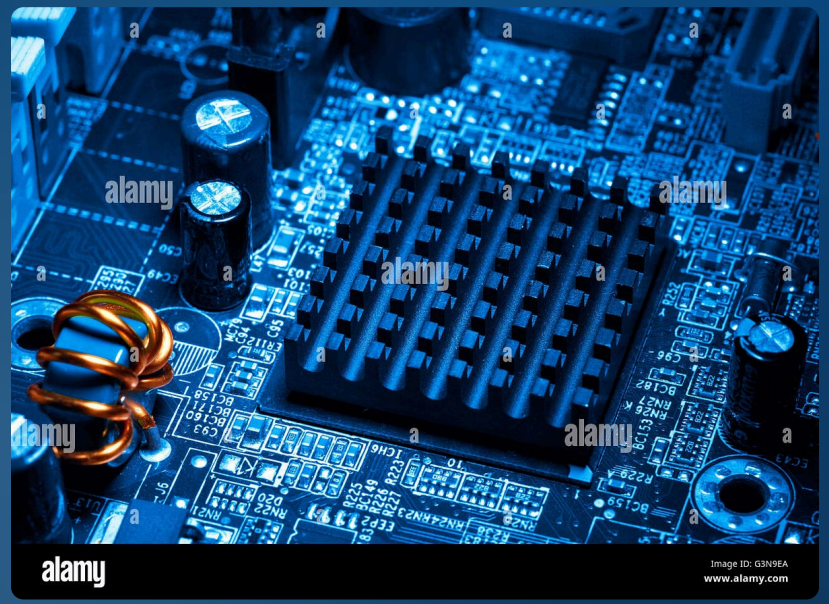


Introduction to Embedded Systems

Embedded systems are computer systems that are designed for a specific purpose and often operate autonomously. They are integrated into devices to control their functionality.

by Ranjeet Kaur

Hardware Components

Microcontroller

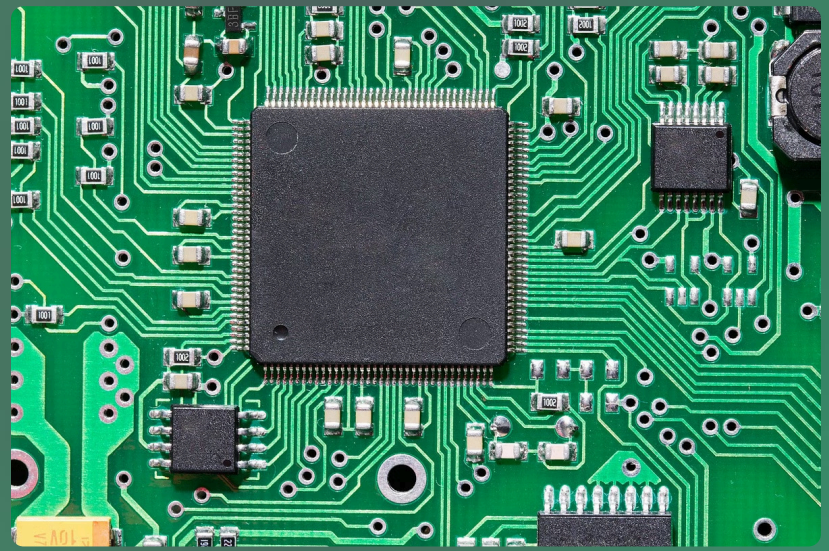
The heart of an embedded system. It's a specialized integrated circuit that controls and manages the system's operations.

Memory

Provides storage for program instructions and data. Common types include ROM, RAM, and flash memory.

Input/Output Devices

Allow the embedded system to interact with the external world. Examples include sensors, actuators, and displays.



Microcontrollers and Processors

1 Microcontrollers

Specialised chips designed for embedded applications. They typically have built-in peripherals, such as timers, ADCs, and communication interfaces.

2 Processors

More general-purpose computing units. They are often used in systems requiring higher processing power or complex tasks.

3 Choosing the Right Processor

The selection depends on factors like performance requirements, power consumption, and cost.

Embedded Software Development

Programming Languages

C, C++, and assembly language are widely used for embedded development.

Development Environments

Integrated development environments (IDEs) provide tools for writing, compiling, and debugging embedded software.

Debugging and Testing

Essential to ensure the software functions correctly and meets the system's requirements.

Real-Time Operating Systems

1

Kernel

The core of the RTOS. It manages resources and provides services to applications.

2

Task Scheduling

RTOSs schedule tasks to ensure efficient use of system resources and timely responses.

3

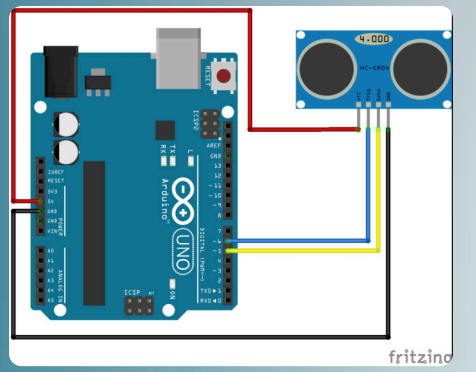
Inter-Process Communication

RTOSs provide mechanisms for tasks to communicate and synchronize with each other.

4

Memory Management

RTOSs handle memory allocation and deallocation for applications and system processes.



Sensors and Actuators

1

2

3

Sensors

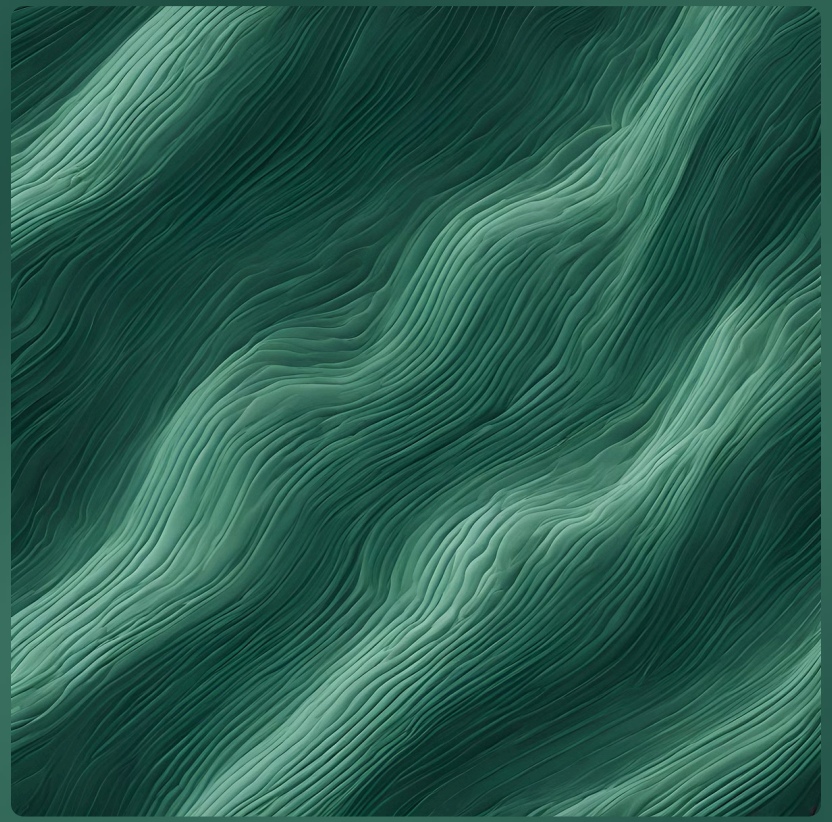
Devices that convert physical quantities into electrical signals. Examples include temperature sensors, pressure sensors, and light sensors.

Actuators

Devices that convert electrical signals into physical actions. Examples include motors, solenoids, and valves.

Interaction

Sensors and actuators work together to allow embedded systems to interact with their surroundings.



Embedded System Design Principles

Power Management	Efficient use of power is crucial for battery-powered systems.
Reliability	Embedded systems must function reliably in challenging environments.
Security	Protection against unauthorized access and data breaches is important.
Maintainability	Systems should be easy to diagnose, repair, and upgrade.

Applications and Trends

Automotive

Engine control, anti-lock braking systems, and infotainment systems.

Consumer Electronics

Smartphones, tablets, and wearable devices.

Home Automation

Smart appliances, security systems, and lighting control.

Robotics and Automation

Industrial robots, drones, and autonomous vehicles.