Misr University for Science & Technology
Faculty Of Information Technology
Department of Computer Science

Embedded System

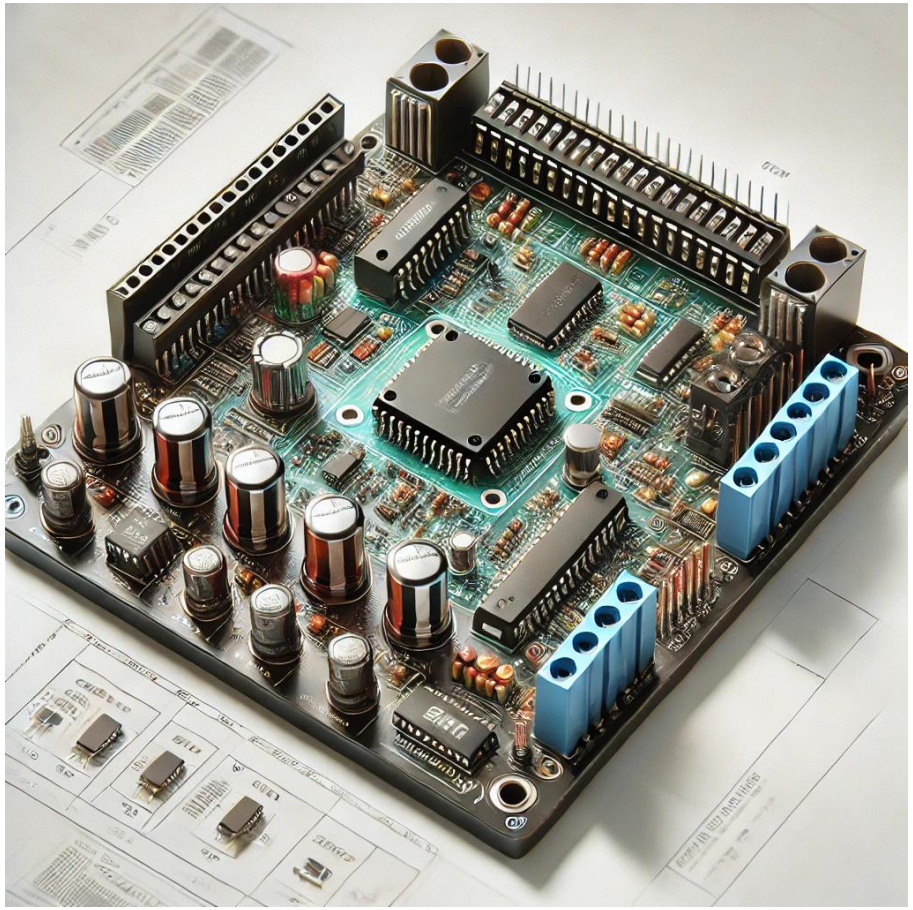
AI 230 / AI 302

Dr. Ahmed Zakaria

Introduction to Embedded System

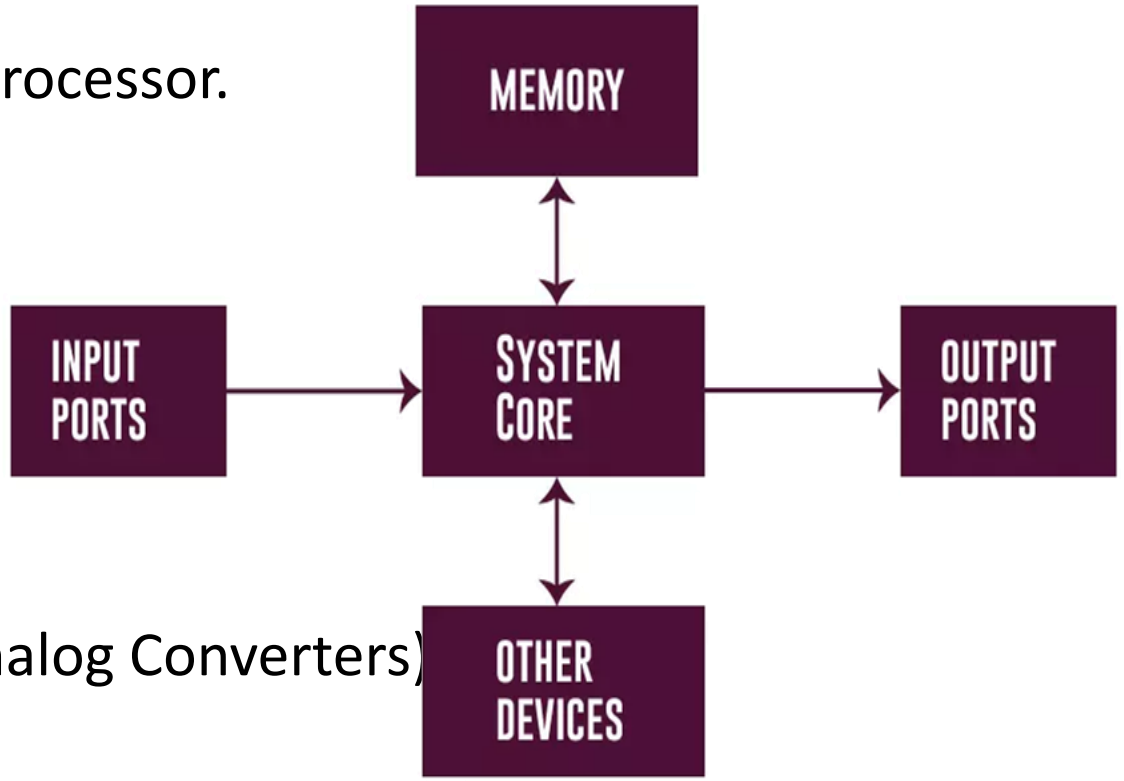
Introduction

- **What is an Embedded System?**
  - An embedded system is a specialized computing system that performs dedicated functions
  - An embedded system (ES) is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function.



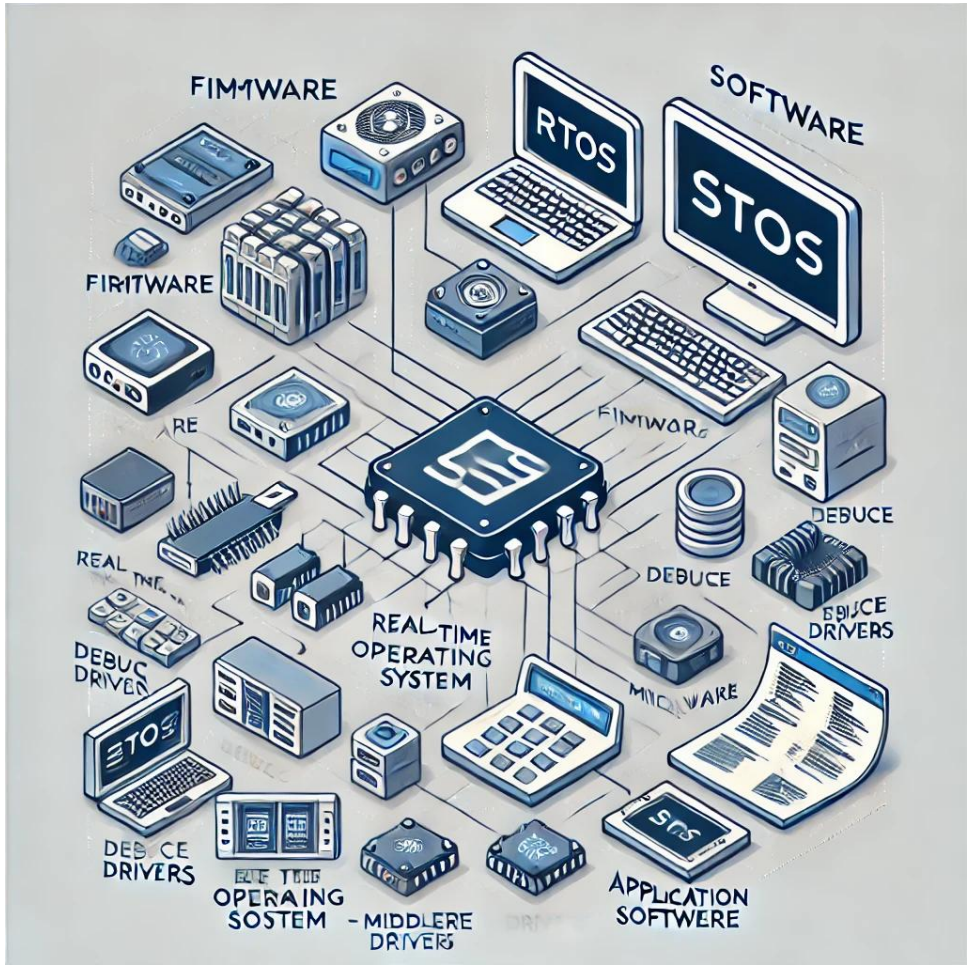
Introduction

- **The hardware in an embedded system consists of**
  - A programmed microcontroller or microprocessor.
  - Memory (RAM, ROM, Flash)
  - Input-output interfaces.
  - Display systems.
  - Communications modules.
  - Electronic and mechanical components.
  - Timers and Counters
  - ADC/DAC (Analog to Digital / Digital to Analog Converters)



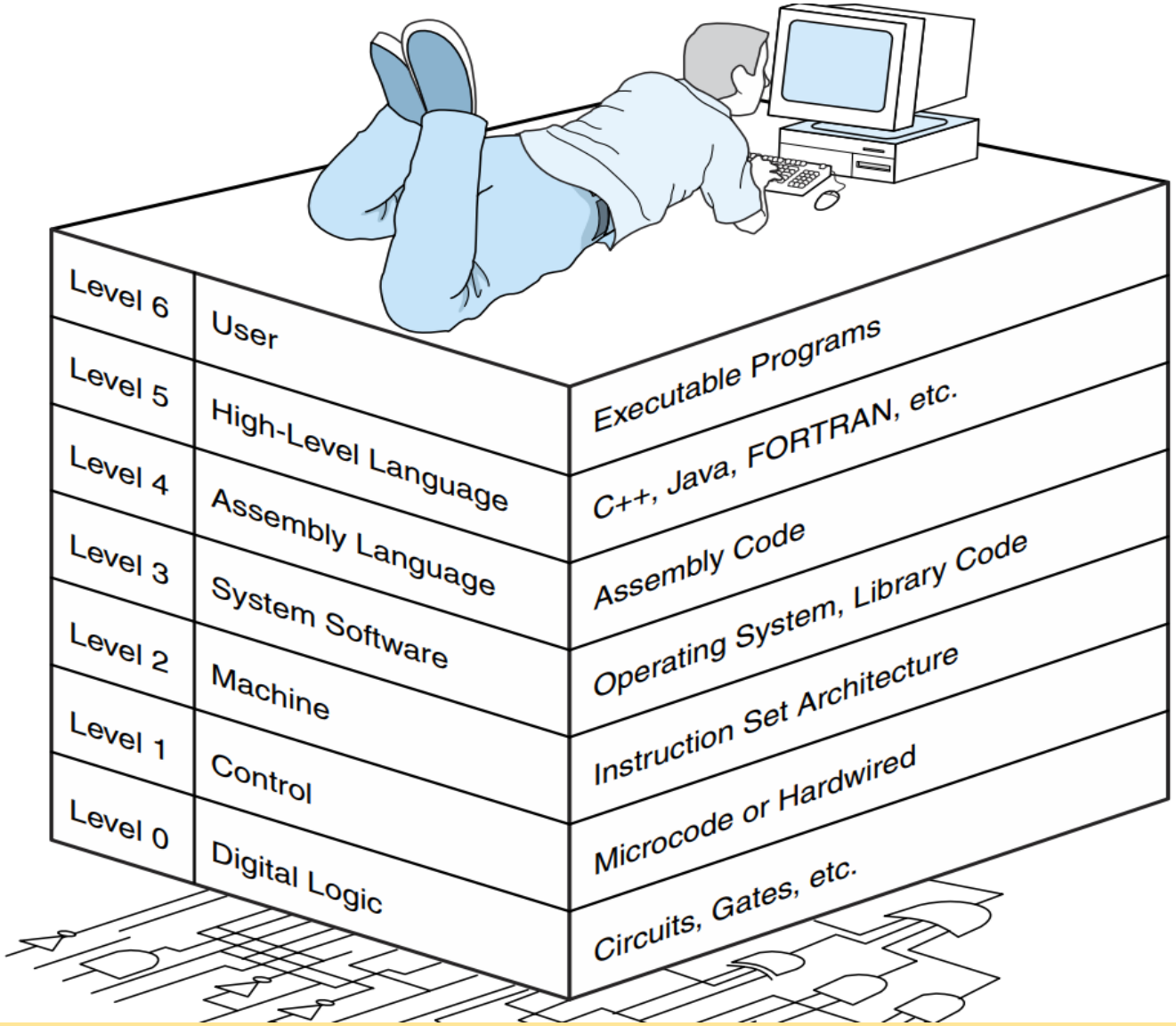
Introduction

- **The Software Components: in an embedded system**
  - Embedded Operating System (RTOS or Bare Metal) – Controls execution, manages tasks.
  - Firmware – Low-level software directly interacting with hardware.
  - Device Drivers – Interfaces between hardware and software.
  - Application Software – High-level software controlling the device.
  - Middleware – Software libraries for networking, encryption, etc.



Embedded System

Introduction



Introduction

- An embedded system is often confused with a general-purpose system.
- A general-purpose system performs multiple tasks at a time, e.g., personal computers and laptops can perform many tasks at a time.
- But an embedded system is destined to do a particular task at a time.

Introduction

Feature	Embedded System	General-Purpose Computer
Function	Dedicated task	Multipurpose
Hardware	Optimized for size and power	Standardized components
OS	Often uses RTOS or no OS	Full OS like Windows/Linux
Performance	Real-time constraints	Performance-focused
Power Consumption	Low	High

Early History of Embedded Systems

- The first multi-chip microprocessors, the Four-Phase Systems AL1 in 1969 and the Garrett AiResearch MP944 in 1970
- First single chip microprocessor was Intel 4004 in early 1971's.
- Automobiles used microprocessor-based engine
  - controllers starting in 1970's.
  - Control fuel/air mixture, engine timing, etc.
  - Multiple modes of operation: warm-up, cruise, etc
- The TMS1802NC was a single-chip microcontroller which was announced September 17, 1971 and implemented a four-function calculator.

Early History of Embedded Systems

- **1976**:
  - The Intel 8048, one of the first microcontrollers, was introduced.
  - It integrated a CPU, memory, and I/O peripherals on a single chip, making it ideal for embedded applications.
- **Growth and Standardization (1980s)**
  - The introduction of 8-bit microcontrollers (e.g., Intel 8051, Motorola 68HC11) revolutionized embedded systems.
  - These chips were cost-effective, energy-efficient, and widely adopted in consumer electronics, automotive systems, and industrial automation.
- **Note**
  - 8-bit microcontrollers are a type of microcontroller with an 8-bit data bus and processing capability.
  - This means they can process 8 bits of data (1 byte) at a time.

Early History of Embedded Systems

- **Growth and Standardization (1980s)**
  - Real-Time Operating Systems (RTOS): The need for multitasking and real-time processing led to the development of RTOS for embedded systems, such as VxWorks and QNX.
  - Embedded Programming Languages: C became the dominant programming language for embedded systems due to its efficiency and hardware-level control.

- **Expansion and Complexity (1990s)**
  - 1990s: The rise of 16-bit and 32-bit microcontrollers (e.g., ARM architecture) enabled more complex and powerful embedded systems.
  - Consumer Electronics: Embedded systems became integral to devices like mobile phones, digital cameras, and MP3 players.

Early History of Embedded Systems

- Real-Time Operating Systems (RTOS): The need for multitasking and real-time processing led to the development of RTOS for embedded systems, such as VxWorks and QNX.
- Embedded Programming Languages: C became the dominant programming language for embedded systems due to its efficiency and hardware-level control.

- **Modern Era (2000s–Present)**
  - The proliferation of 32-bit and 64-bit processors (e.g., ARM Cortex-M series) enabled high-performance embedded systems for applications like smartphones, wearables, and IoT devices.
  - IoT Revolution: Embedded systems became the backbone of IoT, connecting billions of devices worldwide. Examples include smart home devices, industrial sensors, and autonomous vehicles.
  - Advanced Technologies: Embedded systems now incorporate machine learning, artificial intelligence (AI), and edge computing to enable smarter and more autonomous functionality.
  - Open-Source Hardware and Software: Platforms like Arduino and Raspberry Pi

Early History of Embedded Systems

Year	Milestone
1960s	Apollo Guidance Computer (AGC)
1971	Intel 4004 microprocessor introduced
1976	Intel 8048 microcontroller released
1980s	Widespread adoption of 8-bit microcontrollers (e.g., Intel 8051)
1990s	Rise of 16-bit and 32-bit microcontrollers (e.g., ARM architecture)
2000s	IoT and networked embedded systems gain prominence
2010s–2020s	Integration of AI, machine learning, and edge computing in embedded systems

Thank You