Embedded Systems Concepts

1. What is an Embedded System?

An Embedded System is a combination of hardware and software designed for a specific function within a larger system.

Examples: washing machines, cars, medical devices, and industrial robots.

Key points:

- Low power consumption
- Small size
- Real-time operation
- Dedicated purpose

2. Embedded System Design

- Requirements analysis: Understand what the system must do.
- Hardware selection: Choose processor, memory, and peripherals.
- Software development: Write firmware or embedded software.
- Integration & Testing: Combine hardware and software and test them together.

3. Processor Building Blocks

A processor (CPU) typically contains:

- ALU (Arithmetic Logic Unit): Performs calculations and logic operations.
- Control Unit: Controls the flow of data and instructions.
- Registers: Small, fast storage for temporary data.

- Buses: Transfer data between processor, memory, and peripherals.

4. Register Types

Registers are fast memory inside the CPU:

- General Purpose Registers (GPRs): Store temporary data and operands.

- Special Purpose Registers: E.g., Program Counter (PC), Stack Pointer (SP).

- Status Registers: Hold flags for CPU status (Zero flag, Carry flag, etc).

5. Instruction Life Cycle

Instruction execution involves:

1. Fetch: Get instruction from memory.

2. Decode: Identify what to do.

3. Execute: Perform the operation.

4. Write Back: Save results.

5. Repeat for next instruction.

6. Instruction Set Architecture (ISA)

- Defines the commands a processor can execute.

- Examples: ARM, AVR, x86.

- Includes data types, instructions, registers, addressing modes.

- Determines software compatibility.

7. Memory Types

- ROM (Read-Only Memory): Stores firmware (permanent).

- RAM (Random Access Memory): Temporary storage for data & instructions.

- Cache: Fast memory between CPU and RAM.

- EEPROM/Flash: Rewritable non-volatile storage.

8. System Architecture & Timing

- Embedded systems are often real-time systems.

- Timing constraints must be met to avoid failure.

- Example: Airbag controller must deploy within milliseconds.

- System design must consider timing diagrams, clock speed, and synchronization.

Prepared by: Ibrahim Mohamed Ibrahim Ahmed Eldaker