

Lesson 11: Embedded Systems & IoT

1. The "Thing" - Embedded Systems & Arduino

Core Concepts

- **Embedded System:** A computer system inside a larger system. Also known as **Physical Computing**.
- **IPO Model:** Input (Sensors) → Process (Microcontroller) → Output (Actuators).
- **Microcontroller vs. Microprocessor:** A microcontroller is a single chip with integrated CPU, memory, and I/O. A microprocessor-based system has these as separate components.
- **IEEE Definition:** A computer system that is part of a larger system and performs some of its requirements.

The Development Board: Arduino Uno

- **Key Features:** USB Port (for programming/power), Analog Pins (A0-A5), Microcontroller (ATmega328P), Digital I/O Pins (0-13).

The Software: Arduino IDE

- Arduino programs are called **Sketches**.
- **IDE Components:** Verify, Upload, Code Editor, Console Window.

Firmware Structure & Logic

- Must have two main functions: `void setup()` (runs once) and `void loop()` (runs over and over).
- An **endless loop** is used because there is no OS to return control to.

Commonly Used Components (Practicals)

- **Outputs:** LED, Piezo Buzzer, DC Motor.
- **Inputs:** LDR, LM35 Temp Sensor, Reed Switch.
- **Other:** Resistor (220Ω, 10kΩ), Transistor (BC547), Diode (1N4001).

Core Arduino Functions

- `pinMode(pin, MODE);` - Sets pin as INPUT or OUTPUT.
- `digitalWrite(pin, STATE);` - Sets pin to HIGH or LOW.
- `digitalRead(pin);` - Reads state (HIGH/LOW) from pin.
- `analogRead(pin);` - Reads value (0-1023) from analog pin.
- `delay(ms);` - Pauses program for milliseconds.
- `tone(pin, frequency);` - Generates a sound tone.
- `noTone(pin);` - Stops the tone.
- `Serial.begin(baudRate);` - Starts serial communication.

2. The "Network" - Practical IoT with Arduino

Core Concepts

- **IoT Definition:** A network of interconnected embedded systems communicating over the Internet.
- **Goal:** To create a "Smart World" for convenience and comfort.

Building an IoT Device (Smart Light Example)

- **Key Hardware:** **Arduino Ethernet Shield** - an add-on module for internet connectivity.
- **Important:** When the Ethernet Shield is used, digital pins 4, 10, 11, 12, and 13 are reserved and cannot be used for other purposes.

Core IoT/Ethernet Functions

- `#include <Ethernet2.h>` - Include library at the start.
- `byte mac[] = {...};` - Holds the shield's MAC address.
- `EthernetServer s(80);` - Creates a server on port 80.
- **In `setup()`:**
 - `Ethernet.begin(mac);` - Connects to network (DHCP).
 - `Serial.println(Ethernet.localIP());` - Displays IP address.
 - `server.begin();` - Starts the web server.
- **In `loop()`:**
 - `EthernetClient c = s.available();` - Checks for a client.
 - `c.connected();` - Checks if client is still connected.
 - `c.read();` - Reads data sent from the client.
 - `c.stop();` - Disconnects the client.

3. Applications & Implications

Example Systems (from Practical Book)

- **Blinker:** Blinks an LED at a regular interval.
- **AutoLight:** Controls an LED based on LDR light intensity.
- **AutoFan:** Controls a motor based on LM35 temperature.
- **Door-Alarm:** Triggers a buzzer using a reed switch.
- **Smart Light (IoT):** Controls an LED remotely via HTTP.

Social & Security Consequences

- Social Isolation.
- **Security & Privacy:** Risk of unauthorized control of devices and access to personal data.