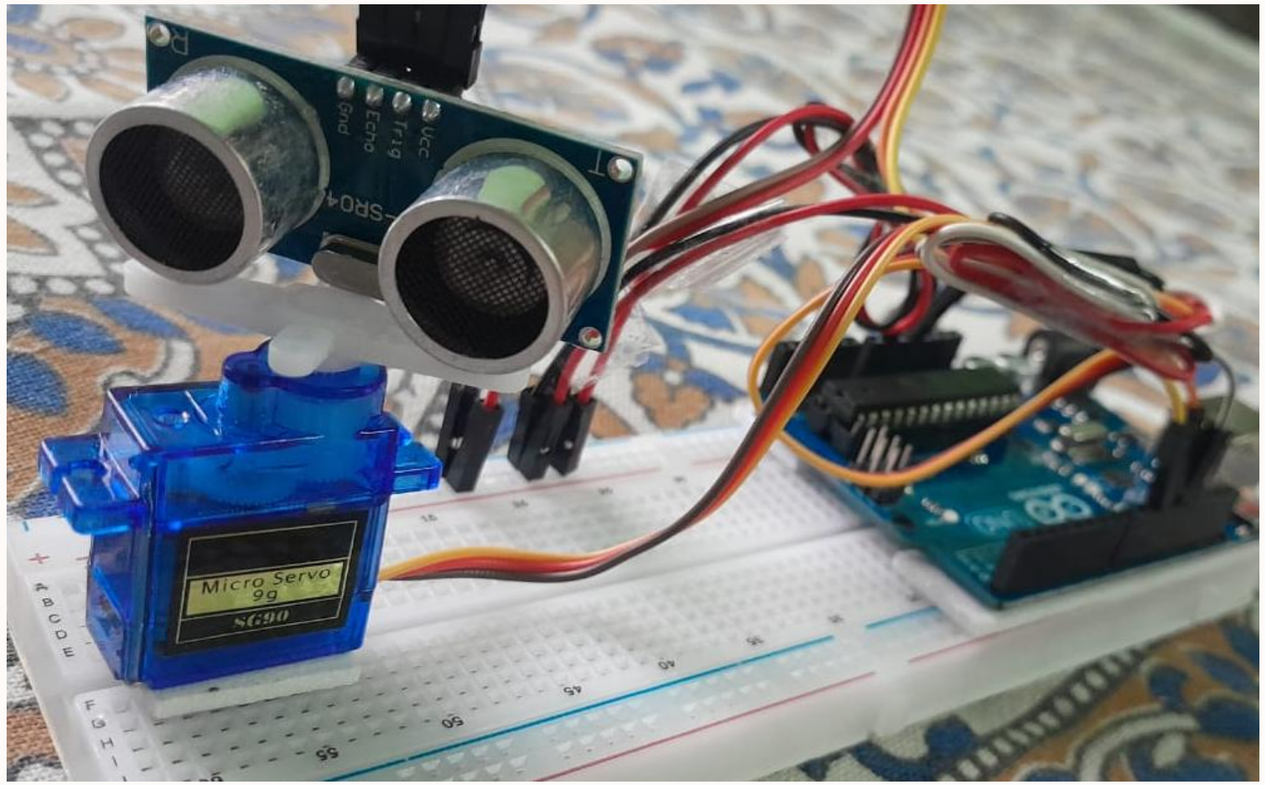
APSSDC

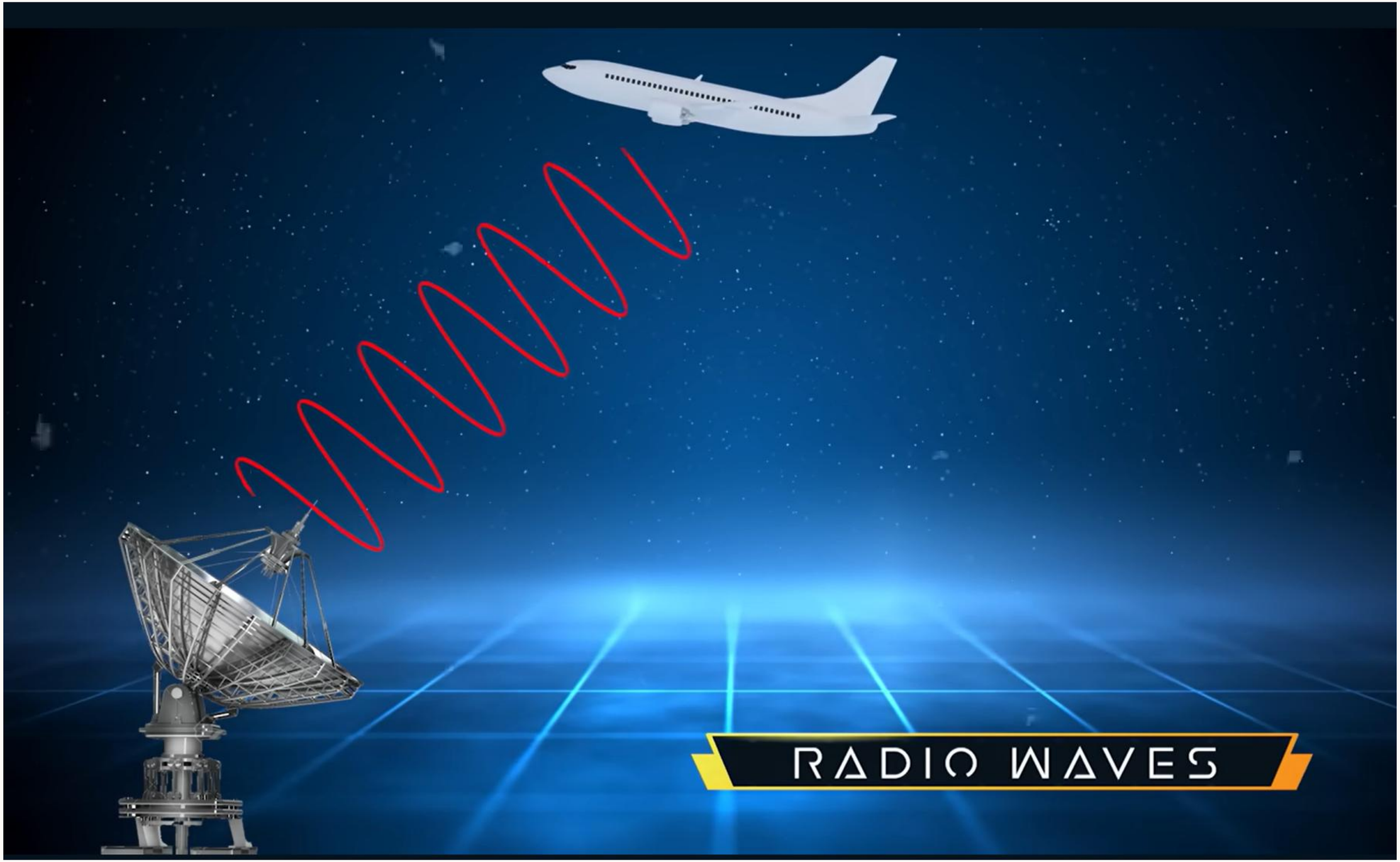
Andhra Pradesh State Skill Development Corporation

ROSHINISHA BEGUM- 322114214071

Radar-based Obstacle Detection System using Arduino



This mini-project demonstrates an automated environment sensing system. It uses an Arduino microcontroller for real-time data processing. The system detects objects and measures their distance. This forms the core of many automated applications.


RADIO WAVES

Introduction to Embedded Systems

Definition

A computer system with a dedicated function. It is part of a larger mechanical or electrical system.

Characteristics

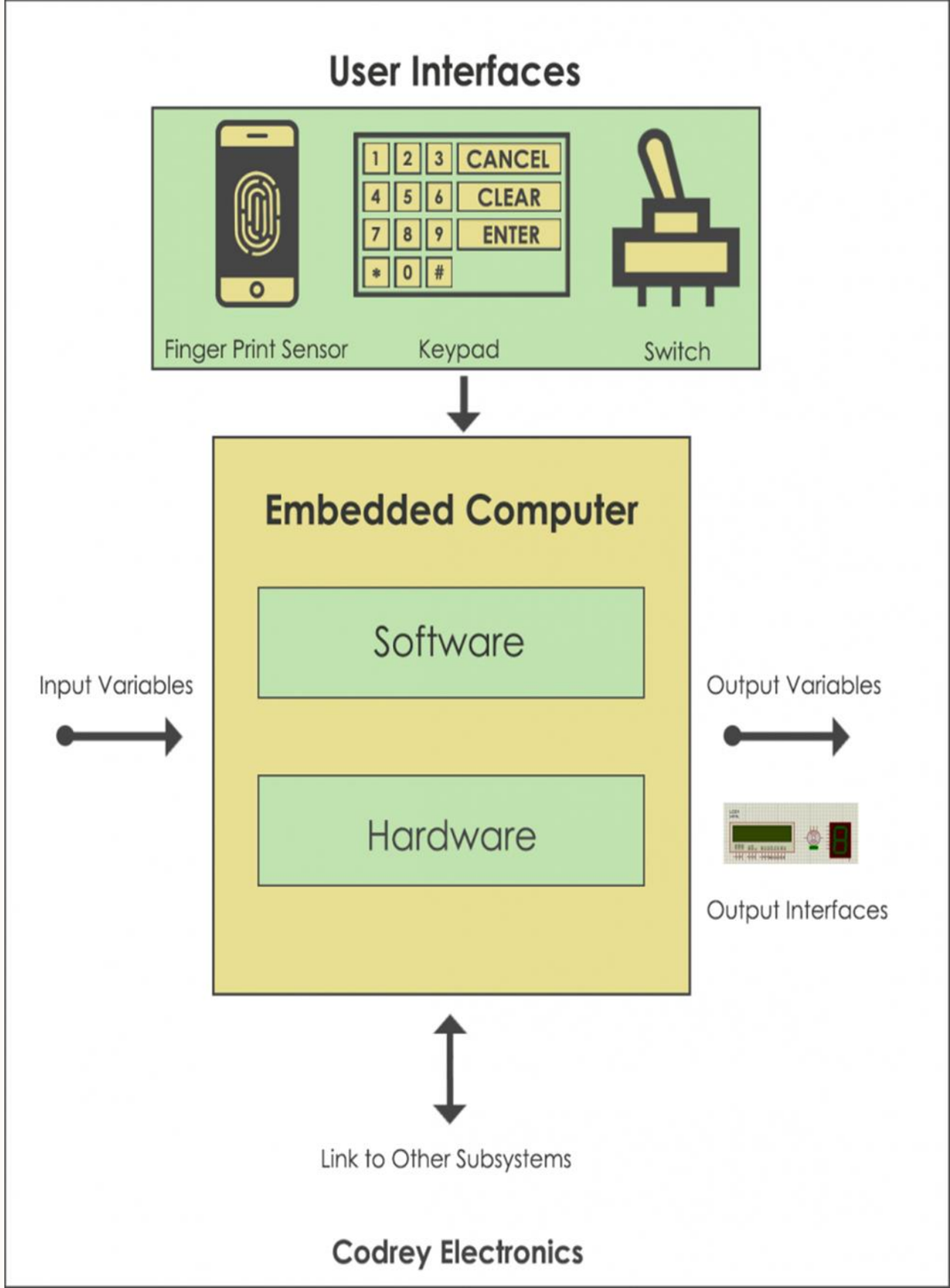
Real-time operation, low power consumption, compact size, and high reliability.

Project Advantages

Efficient resource use and cost-effective for dedicated tasks. Direct interaction with hardware.

Applications

Anti-lock Braking System (ABS)Airbag control industrial control, and consumer electronics like smartphones.



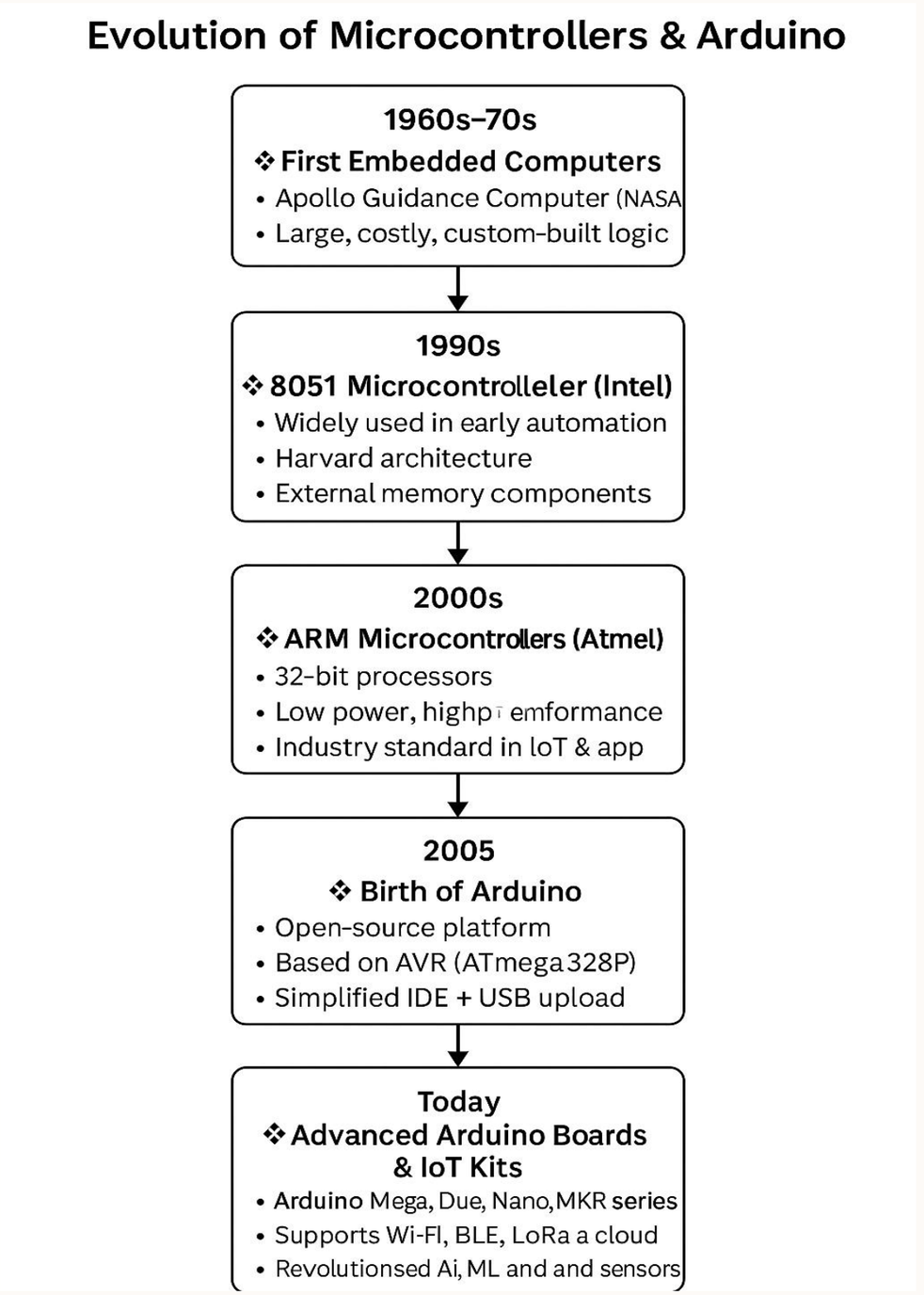
Evolution of Microcontrollers & Arduino

Microcontroller Journey

- From early 8-bit Intel 8051 (1980).
- To modern ARM Cortex-M series.
- Increasing power and efficiency.

Arduino's Impact

- Open-source platform for rapid prototyping.
- Simplifies hardware-software interaction.
- Great for beginners and experts.



System Architecture & Hardware

Hardware Integration

1

Sensor Integration

HC-SR04 Trigger/Echo pins connected to Arduino digital I/O.

2

Servo Motor Control

Servo motor connected to Arduino PWM pin for angular movement (0-180 degrees).

3

Power Supply

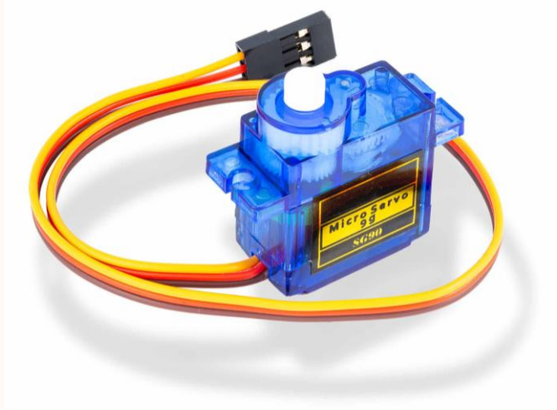
5V from Arduino board or external supply for stability.

4

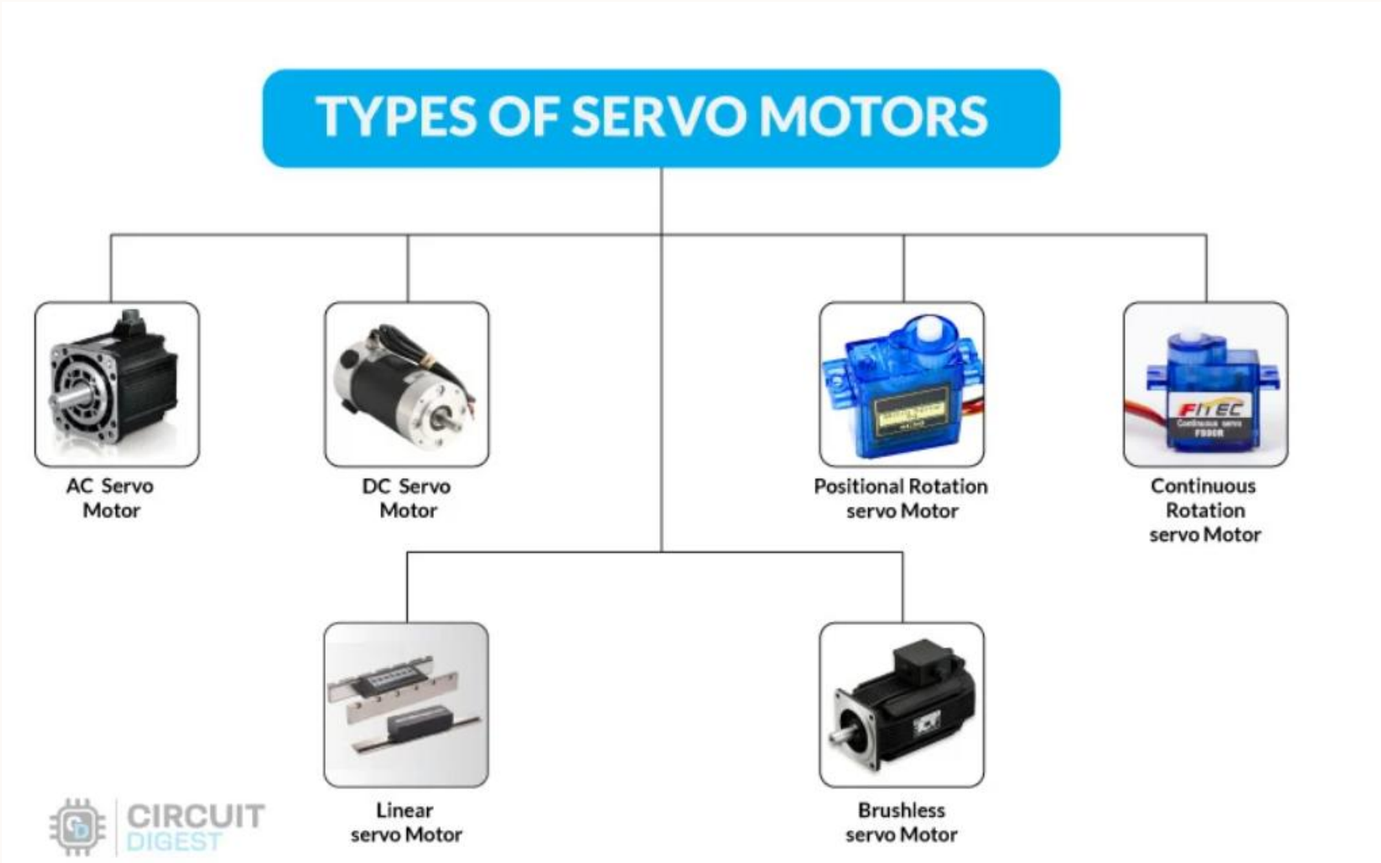
Data Flow

Sensor data (distance) to Arduino for processing. Serial Monitor provides output visualization.

SERVO MOTOR:



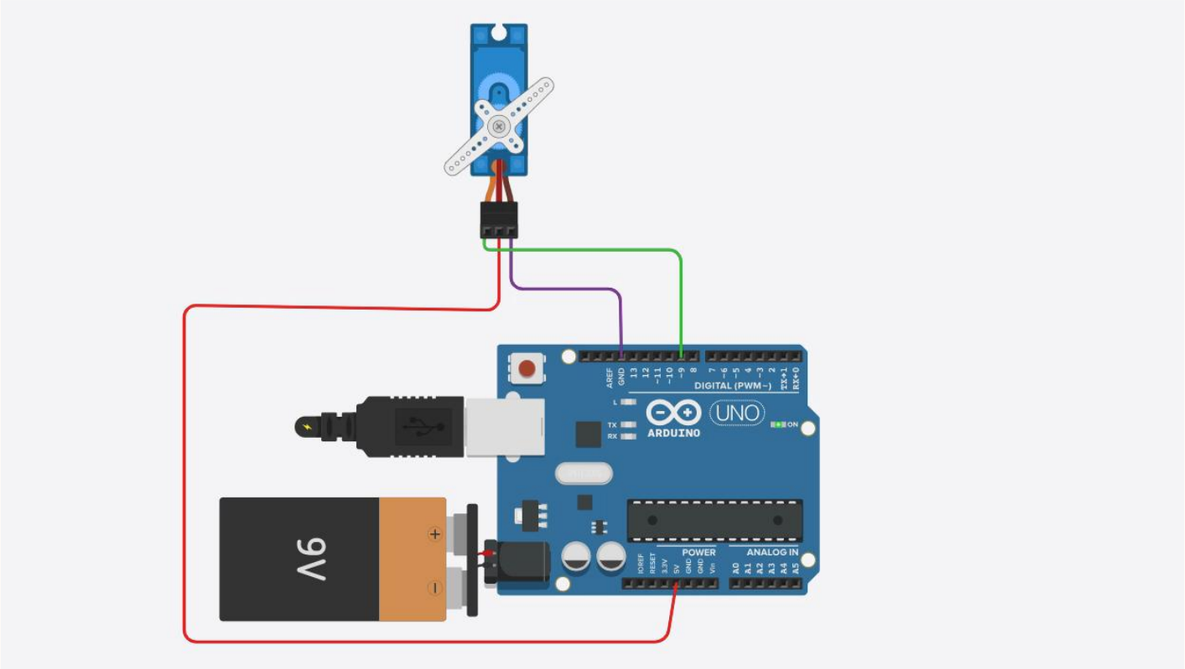
A **servo motor** is a rotary actuator designed to precisely control angular position. It consists of a motor coupled with a sensor for position feedback. Servo motors are commonly used in robotics, RC vehicles, and various applications requiring precise motion control.



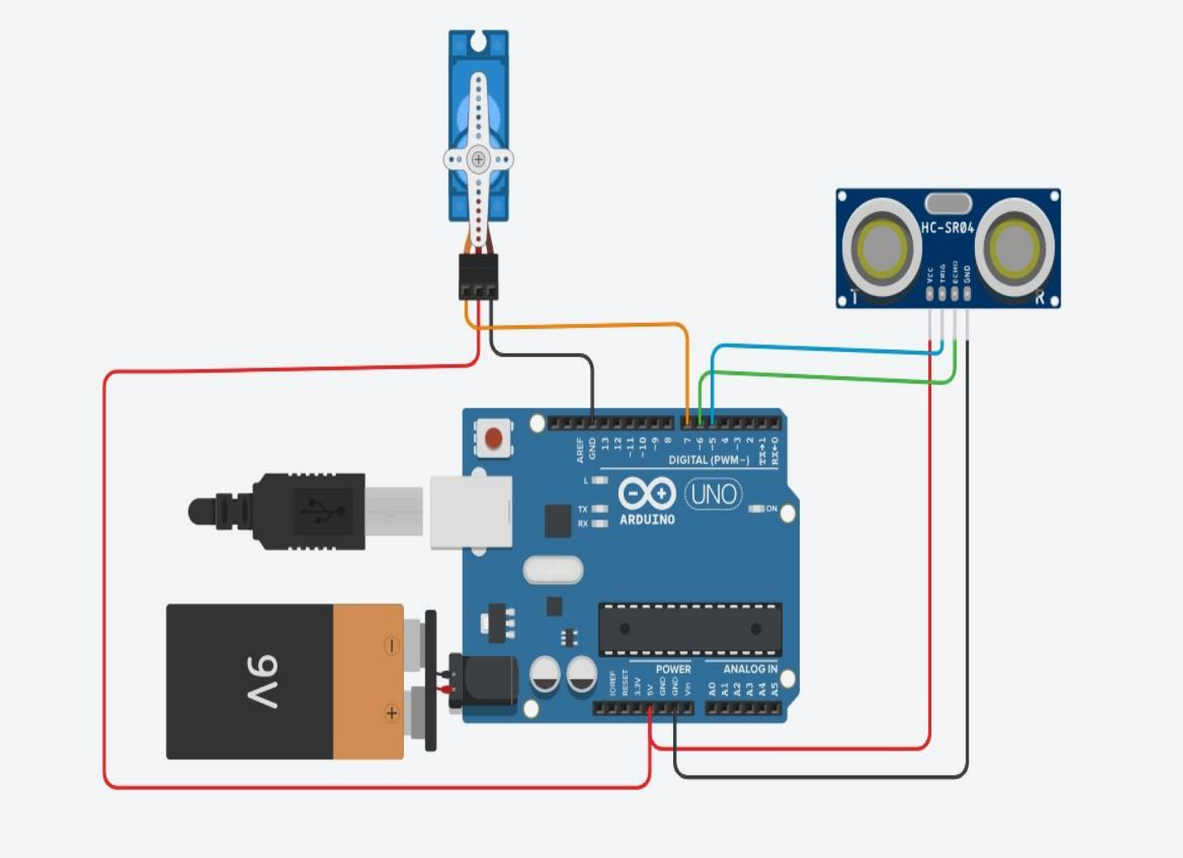
They rotate to a specific angle based on input signals, making them ideal for applications like robotic arms, antenna positioning, and camera gimbals.

CODE:

```
#include <Servo.h> // Include the Servo library

Servo servo1; // Create a servo object to control the servo motor
int angle = 0; // Variable to store the servo position

void setup() {
  servo1.attach(9); // Attach the servo to PWM pin 9
}

void loop() {
  // Rotate the servo from 0 to 180 degrees
  for (angle = 0; angle <= 180; angle += 1)
 {
    servo1.write(angle); // Set the servo to the current angle
    delay(20); // Wait for the servo to reach the position
  }

  // Rotate the servo from 180 to 0 degrees
  for (angle = 180; angle >= 0; angle -= 1) {
    servo1.write(angle); // Set the servo to the current angle
    delay(20); // Wait for the servo to reach the position
  }
}
```



Radar-based Obstacle Detection: The Core Concept



Working Principle (Ultrasonic)

Emits high-frequency sound pulses at 40kHz. Measures time for the echo to return.

"Radar-like" Function

Maps surroundings by sweeping, similar to radar principles using sound waves.

Key Components

HC-SR04 Ultrasonic Sensor, SG90 Servo Motor, Arduino Uno R3.

Distance Calculation

`(Time x Speed of Sound) / 2`. Speed of sound is approx. 343 m/s.

Working Principle of Ultrasonic Sensor

What is Ultrasonic Sensor?
Ultrasonic sensors are electronic devices that calculate the target's distance by emission of ultrasonic sound waves and convert those waves into electrical signals. The speed of emitted ultrasonic waves traveling speed is faster than the audible sound.
To know the distance between the target and the sensor, the sensor calculates the amount of time required for sound emission to travel from transmitter to receiver. The calculation is done as follows:
D = 1/2 T * C

Advantages and Limitations of the System

Advantages

- Cost-effective (under ₹1500).
- Simple implementation.
- Non-contact detection.
- Real-time feedback.

Limitations (Ultrasonic)

- Affected by temperature/humidity.
- Limited range (2cm-400cm).
- Susceptible to soft surfaces.
- Narrow beam angle (15-30 degrees).

Environmental Impact

- Degrades in noisy environments.
- Poor performance against irregular surfaces.

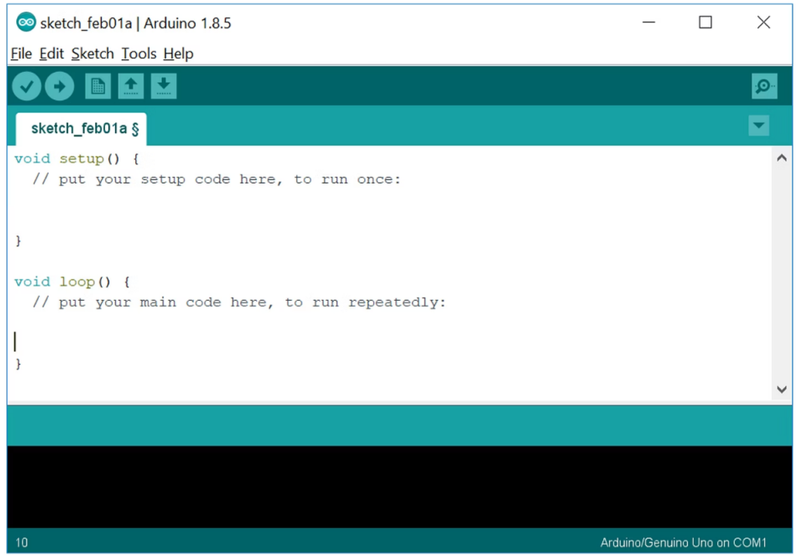
True Radar Comparison

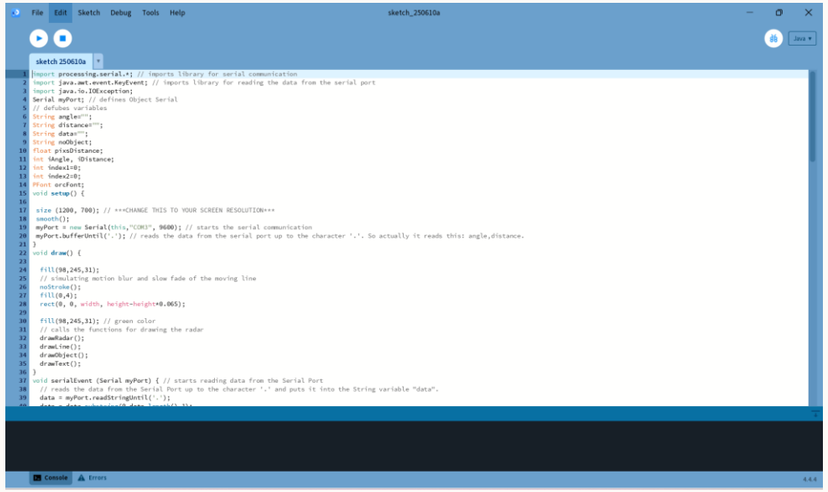
- Longer range, better penetration.
- Less environmental sensitivity.
- More complex and expensive.

Software Development: Arduino IDE & TinkerCAD

SETUP
Called once at the very beginning of execution. Use this for initialization

LOOP
Called over and over again as fast as possible by the microcontroller. As soon as your code completes here, loop() will be called again forever





Arduino IDE & Processing IDE

Integrated Development Environment for coding in C/C++& JAVA. User-friendly for microcontroller programming.

Key Libraries

- **Servo.h** for motor control.
- **NewPing.h** for advanced ultrasonic readings.

TinkerCAD Simulation

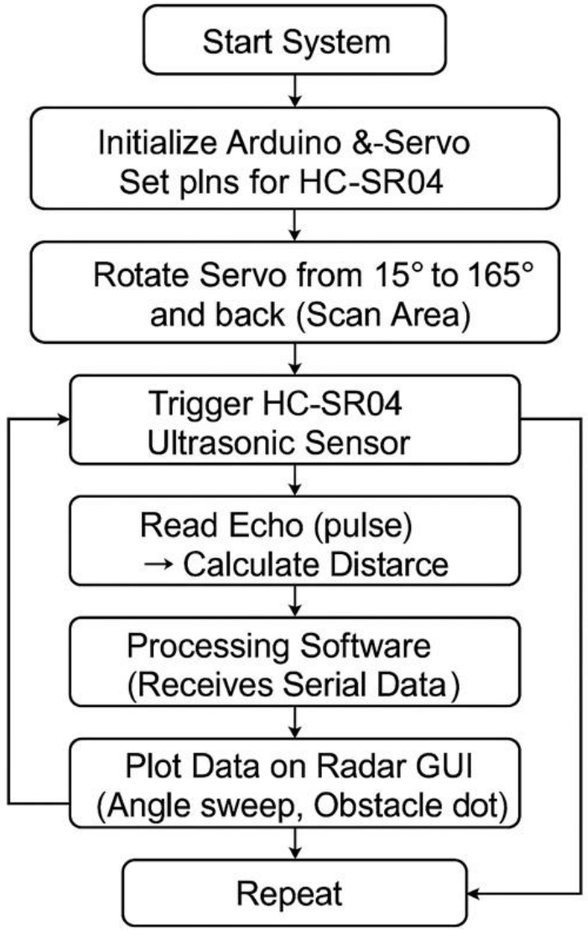
Web-based platform for circuit design and code simulation.

Benefits

Virtual testing, code debugging without physical hardware. Supports iterative design.

Visual representation of sensor sweep and detected distances.

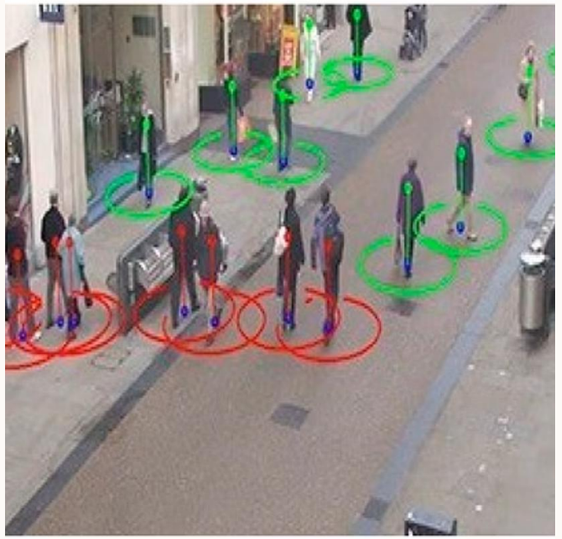
BLOCK DIAGRAM OF HOW IT WORKS?



Radar-based Obstale Detection System using Arduino

Real-time Applications: Defense & Security



- **Perimeter Security:** Intruder detection for borders and restricted areas. Ensures high-level security.
- **Cosmic objects:** finding cosmic objects far from our earth.
- **Drone-based Reconnaissance:** air traffic patterns recognition.
- **Public Safety:** Crowd density monitoring and vehicle traffic flow analysis. Aids in disaster management.

Conclusion & Future Scope

Summary

We successfully developed a low-cost, functional obstacle detection system. It uses Arduino and ultrasonic sensing. This project effectively demonstrated key embedded system principles. It also showcased microcontroller applications and real-time object detection.

Future Enhancements

Sensor Upgrades

Integrate LiDAR or actual radar modules for better performance.

Signal Processing

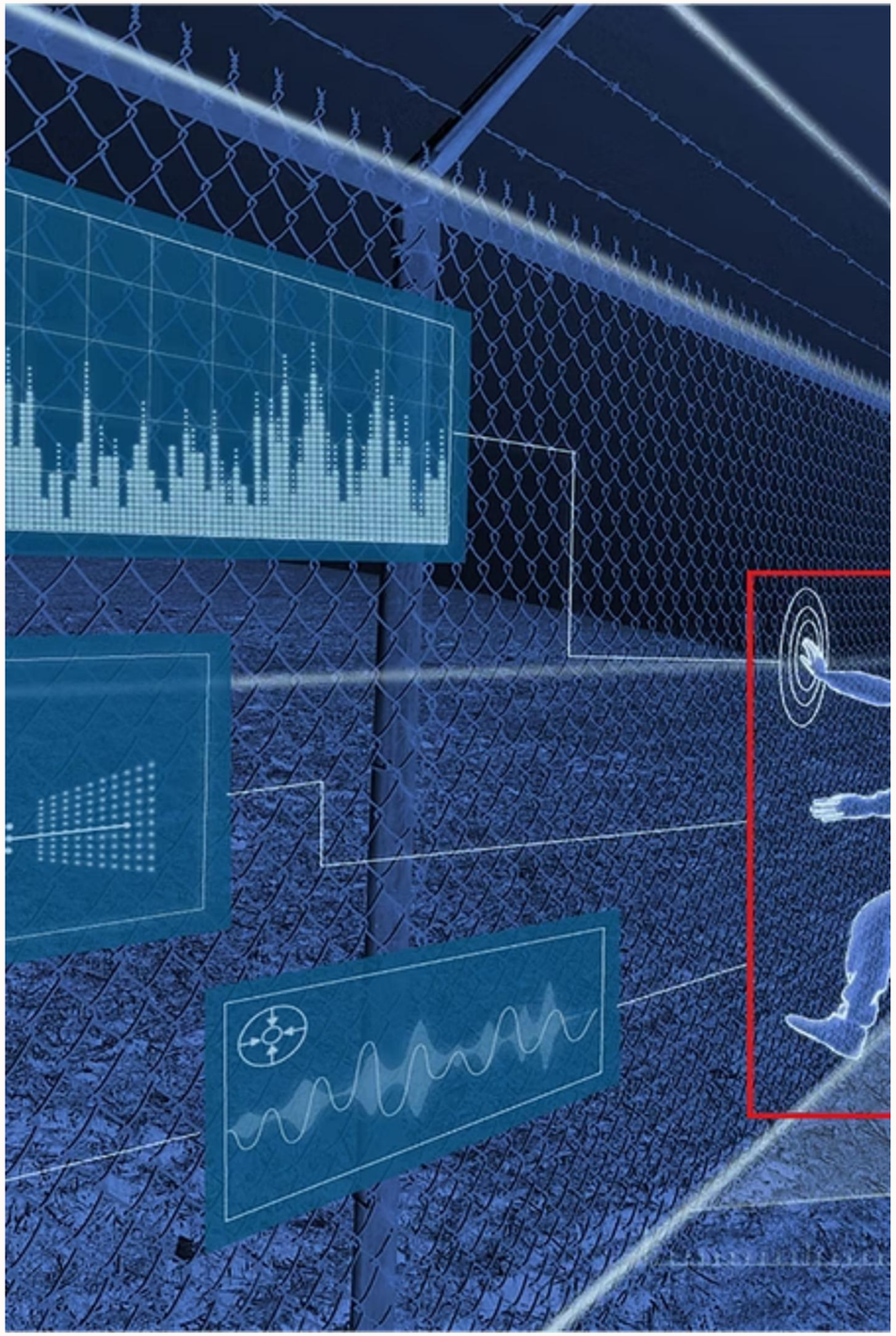
Implement advanced algorithms for noise reduction. Enhance data accuracy.

GUI Development

Create a graphical user interface using Python with Pygame. Improve data visualization.

3D Mapping

Extend capabilities by adding vertical servo movement for comprehensive 3D mapping.




THANKYOU