Project Report

Basic Embedded System Project

Simple Embedded system for Beginners

Name: ANDAMNANDINI

Organization: SLASH MARK IT Solutions Submission Date:

24/3/25

Abstract

This project report presents three fundamental Embedded System projects:

1. Biometric Attendance System

2. Smart Irrigation System

3. LED Display

Each project demonstrates essential programming concepts such as data handling, user interaction, Sensors and automated controls and algorithm implementation.

The report details objectives, methodologies, results, testing, simulation and future enhancements.

Methodology

Approach: Modular development with structured coding principles.

Technologies Used: Embedded C, Arduino uno, and built-in libraries.

Process Flow: Identify requirements -> Develop code -> Simulation ->Test functionality -> Debug and optimize.

Implementation - Task 1: Biometric Attendance System

AEmbedded c program for managing daily tasks with fingerprint Recognition, facial Recognition, iris scanning, real time attendance.

Features:

- Fingerprint Recognition

- Facial Recognition

- Iris scanning

- Real-time attendance tracking

Code – Biometric Attendance System

```
#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>

SoftwareSerial mySerial(2, 3); // RX, TX for fingerprint sensor
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

void setup() {
  Serial.begin(9600);  // Start serial communication for debugging
  mySerial.begin(9600);  // Start software serial for fingerprint
sensor

  if (finger.begin()) {
    Serial.println(“Fingerprint sensor initialized.”);
  } else {
    Serial.println(“Fingerprint sensor not detected.”);
    while (1);
  }

  delay(1000);
  Serial.println(“Place your finger on the sensor...”);

  if (finger.getTemplateCount() == 0) {
    Serial.println(“No fingerprints enrolled. Enroll a fingerprint.”);
    enrollFinger();
  } else {
    Serial.println(“Ready to recognize fingerprints.”);
  }
}

void loop() {
  int fingerID = getFingerprintID();
  if (fingerID >= 0) {
    Serial.print(“Fingerprint matched with ID: ”);
    Serial.println(fingerID);
    logAttendance(fingerID);
  }
  delay(2000);  // Wait before checking for another fingerprint
}

void enrollFinger() {
  Serial.println(“Please place your finger on the sensor for
enrollment.”);
  int fingerID = -1;
  while (fingerID == -1) {
    fingerID = getFingerprintID();
    delay(1000);
  }

  Serial.print("Enroll finger ID: ");
  Serial.println(fingerID);

  if (finger.storeModel(fingerID) == FINGERPRINT_OK) {
    Serial.println("Fingerprint enrolled successfully.");
  } else {
    Serial.println("Failed to enroll fingerprint.");
  }
}

int getFingerprintID() {
  uint8_t result;
  uint8_t fingerID = -1;

  result = finger.getImage();
  if (result != FINGERPRINT_OK) {
    Serial.println("No finger detected.");
    return -1;
  }

  result = finger.image2Tz();
  if (result != FINGERPRINT_OK) {
    Serial.println("Failed to convert image.");
    return -1;
  }

  result = finger.fingerSearch();
  if (result == FINGERPRINT_OK) {
    fingerID = finger.fingerID;
  } else {
    Serial.println("Fingerprint not recognized.");
    fingerID = -1;
  }

  return fingerID;
}

void logAttendance(int id) {
  Serial.print("Logging attendance for user ID: ");
  Serial.println(id);
}
```

Implementation - Task 2: Smart Irrigation System

Smart Irrigation System involves creating an automated system that adjusts the watering schedule of plants based on real time environmental factors such as soil moisture, weather, humidity, temperature.

Features:

- Soil Moisture Monitoring
- Weather based adjustment
- Remote monitoring and controlling
- Water efficiency
- Real-time alert notification
- Leak detection
- Customizable Schedule
- Integration with smart home devices
- Data analytics and reports
- Energy efficient
- Automated irrigation control
- Remote control and mobile app
- Ai/ml optimizations

Code – Smart Irrigation System

```
// Pin Definitions
int soilMoisturePin = A0; // Soil moisture sensor connected to analog pin A0
int pumpRelayPin = 7; // Relay module connected to digital pin 7

int soilMoistureValue = 0; // Variable to store the moisture value
int moistureThreshold = 500; // Moisture threshold (adjustable based on your needs)
int pumpDelayTime = 5000; // Time (in milliseconds) to run the pump (e.g., 5 seconds)

void setup() {
  // Initialize the serial monitor for debugging
  Serial.begin(9600);

  // Set relay pin as output
  pinMode(pumpRelayPin, OUTPUT);

  // Initially turn off the pump
  DigitalWrite (pump Relay Pin, LOW);
}

void loop() {
  // Read the soil moisture level (0 to 1023, where 0 is completely dry and 1023 is fully wet)
  soilMoistureValue = analogRead(soilMoisturePin);

  // Print the moisture value to the serial monitor for debugging
  Serial.print(“Soil Moisture Value: ”);
  Serial.println(soilMoistureValue);

  // If soil moisture is below the threshold, turn on the pump
  if (soilMoistureValue < moistureThreshold) {
    Serial.println(“Soil is dry. Turning on the water pump…”);
    // Activate the water pump (Relay ON)
    digitalWrite(pumpRelayPin, HIGH);

    // Keep the pump on for a defined period to water the plants
    delay(pumpDelayTime);

    // Turn off the water pump
    digitalWrite(pumpRelayPin, LOW);
    Serial.println("Pump turned off.");
  } else {
    Serial. println("Soil is wet. No need to water.");
  }

  // Wait a bit before taking the next reading (optional)
  Delay (10000 ) ;   // Check every 10 seconds
}
```

Implementation - Task 3: LED Display

LED display like 7-segment display, matrix displays or LCD screen. LED contain 2 rows and 16 column .

Features:

- Low power consumption
- Cost saving

Code – LED Display

```
#include<LiquidCrystal.h>
LiquidCrystal lcd(13,12,11,10,9,8,7);
void setup() {
lcd.begin(16,2);

}

void loop() {
  lcd.setCursor(0,0);
  lcd.print("WELCOME");
  lcd.setCursor(0,1);
  lcd.print("KLMCOLLEGE");
}
```

Testing

Each project was simulation with multiple scenarios:

- Unit simulation for each function

-Simulation user inputs and boundary conditions

- Error handling verification

Results and Discussion

All projects successfully met their objectives, with proper implementation and simulation.

Performance metrics show efficient execution with minimal errors.

Future Scope

- Smart Irrigation System: Automates irrigation based on weather, soil moisture, remote control, optimizing water use and reducing waste.
- LED Display: Offers high brightness, energy efficiency, durability, flexibility, and real-time customizable content for various applications.
- Biometric Attendance System: Uses fingerprint facial, or iris recognition for secure, accurate, and automated employee attendance tracking, integrated with payroll.

Acknowledgments

Thanks to SLASH MARK IT Solutions and mentors for their guidance.