CSE360: Computer Interfacing

Final Project Report

Topic: Smart Coffee Machine

Lab Section: 02

Submitted by:
Group 9

Name	ID
Golam Sarwar Sami	21101276
Farhan Tanvir Niloy	22301329
Md. Tahmid Iqbal	21201701
Abrar Fahim	21301073

Submitted to:
Pratick Roy Chowdhuruy (CSE)
Md. Irtiza Hossain (CSE)

May 9, 2025

1 Abstract

This project showcases a Smart Coffee Machine alongside its implementation. It automates detecting mug, brewing, water pouring and mixing process alongside with monitoring water level and temperature.Communication Protocols such as I2C, GPIO, PWM and OneWire enables efficient interfacing within components. The system aims to minimize human effort and remove human error while providing a practice application of embedded system using interfacing techniques.

2 Introduction

This section introduces the motivation behind Smart Coffee Machine, Objectives and its significance while focusing on communication and interfacing elements.

2.1 Problem Statement:

Making a good cup of coffee requires quite some manual labor like grinding, mixing ingredients and water pouring in equal amount. All of these are time demanding and requires precision. This manual process introduces human error and also waste of valuable time which could be used in other productive activities. There is a clear need for an automated coffee machine. This project aims to address the problem of coffee making which millions enjoy by automating the entire coffee making process using micro controllers and interfacing technologies.

2.2 Objectives:

The primary objective of this project is to implement an automated coffee machine which will be controlled via the Arduino Uno R3 micro controllers. Specific interfacing and protocol implementation include:

- Use of sensors like IR sensor for object detection
- Control of motor and pump using LN98N motor driver module
- Real-time user interfacing using the OLED display module using **I2C communication protocol**
- Component sequencing and timing of functions using **Digital I/O and Pulse Width Modulation (PWM)**
- Ensuring isolated and stable power supply for all components to maintain operational stability.

2.3 Significance:

This project showcases how automated coffee machine is creating integrating components through effective interfacing. It also shows the working of micro-controllers their communication with other I/O components and multi-stage automation. Coffee is one thing millions brew and enjoy everyday and using a Smart Coffee Machine altogether can help save huge amount of time and effort from people's lives. Furthermore, this project provides an hands on experience of embedded system with practical application in interfacing and protocols.

3 Interfacing Design

This section discusses the interfacing design, components used, block diagram of the coffee machine and challenges faced.

3.1 Interfacing Components:

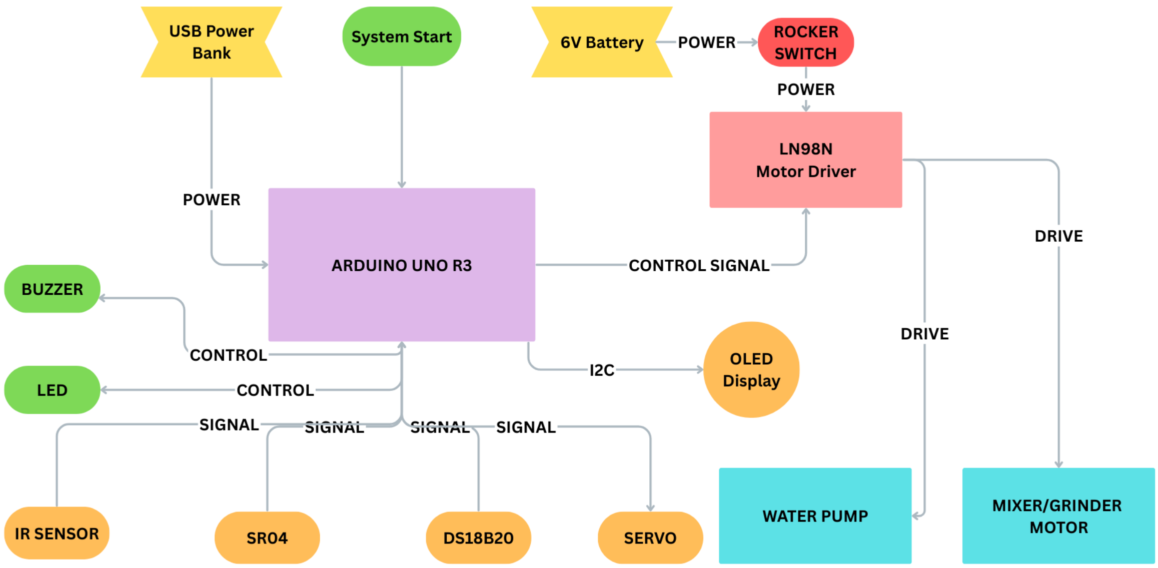
The Smart Coffee Machine uses lots of different modules and sensors integrating with Arduino Uno R3 to automate the coffee making process. A list of components and their role is shown in the table below:

#	Component	Module/Sensor	Function / Role
1	Microcontroller	Arduino Uno R3	Main controller to handle logic, process sensor output/input and provice signals.
2	Temperature Sensor	LM35 / DS18B20	Monitors water temperature
3	Water Level Sensor	YL-69 / HC-SR04	Detects if enough water is present or not
4	Water Pump	DC 3–6V Mini Pump	Pumps water into the grinding/mixing pot
5	Switch	Rocker Switch KCD11	Manually turns on/off 6v battery connection
6	Motor Driver	L298N Module	Drives DC motor and Water Pump helps with power distribution and regulation
7	Display	OLED (SSD1306)	Shows user process prompts and sequence
8	Buzzer	Passive Buzzer Module	Provides audible alert once brew is done
9	Mixer Motor	3–12V Yellow DC Motor	Mixes and Grinds ingredients
10	Power Supply	6V Battery + USB Power Bank	6V supply to Motor and Pump, Power Bank powers Arduino using USB
11	Mug Detection Sensor	IR Detection Sensor	Detects the presence of a mug to start the process

Other Components:

- Breadboard – For multiple wire connections.
- Jumper Wires – To connect and extend connection.
- Water Pipe – Pipe through which water flows
- Anti-Cutter Blades – Used for grinding and mixing beans

3.2 Block Diagram:



3.3 Interfacing Challenges:

Interfacing Challenges and Solutions

- **Power Isolation Issues:** Motors and Arduino sharing power leads to unstable power flow.
Solution: Seperate power for Arduino (Power Bank) and motors/pumps (6V battery).
- **Signal Compatibility:** Sensors require different voltage.
Solution: Provide 5v from Arduino to sensors operating within 5v.
- **Current Draw by Motors:** Pump and Motor exceed Arduino's current capacity.
Solution: Offload current to Pump and Motor using L298N
- **False Mug Detection:** IR sensors may falsely detect other objects as mug.
Solution: Add verification delay of 5 seconds so user can remove any object within some seconds if put mistakenly.

4 Communication Protocols

This section discusses the communication protocols used and why they were chosen.

4.1 Protocol Selection:

The smart coffee machine uses I2C, GPIO and PWM communication protocols. An overview is given

- **I2C:** Used for the OLED Display (SSD1306). It uses two wires to establish serial protocol via SDA & SCL which is suitable for short distance communication with peripherals having low-speed.
- **GPIO (Digital I/O):** Used for IR Sensor, Motor Driver (L298N), Buzzer, and Switch. GPIO pins handle the High/Low digital logic to control these devices.
- **PWM:** Used to regulate the motor and pump speed. PWM helps to simulate analog output by rapidly switching digital pins.
- **OneWire:** Used in the DS18B20 Temperature Sensor. A single wire is used for communication protocol which is made to read temperature sensors efficiently.

Note: No wireless protocols such as UART, Bluetooth or Wifi have been implemented as the system is for offline use.

4.2 Protocol Justification:

- **I2C:** Chosen for the OLED display because it only uses 2 pins for communication. Arduino's A4 is used for SDA and A5 for SCL with the OLED display to communicate. This simplifies the peripheral communication.
- **GPIO:** Ideal and adequate for basic components such as IR sensor and switch which can operate using simple High and Low logic. GPIO's are easy to configure and they are also suitable for real time usage.
- **PWM:** Used to precisely control the motor, pump and servo's position. Arduino's analogWrite() function is used to enable the PWM on digital pins that require it.
- **OneWire:** Chosen for DS18B20 sensor in order to simplify temperature monitoring using only one data pin.

5 Implementation Plan

This section provides the methodology and expected outcome of the Smart Coffee Machine.

5.1 Methodology:

A pseudocode of the actual code has been given below **Protocol Configuration (Pseudocode):**

```
BEGIN setup
    START Serial Communication at 9600 bps
    // Initialize I2C protocol
    CONNECT to OLED display using I2C pins (SDA, SCL)
    // Set up GPIO pins
    SET pinMode for IR Sensor as INPUT
    SET pinMode for Buzzer as OUTPUT
    SET pinMode for Motor Driver pins as OUTPUT
    SET pinMode for Water Pump as OUTPUT
    SET pinMode for Switch as INPUT
    SET pinMode for Temperature Sensor as INPUT
    SET pinMode for Water Level Sensor as INPUT
    // Set up PWM pins (for motors)
    SET motor speed pin as PWM OUTPUT
    DISPLAY "System Ready" on OLED
END setup
```

Loop Function (Pseudocode):

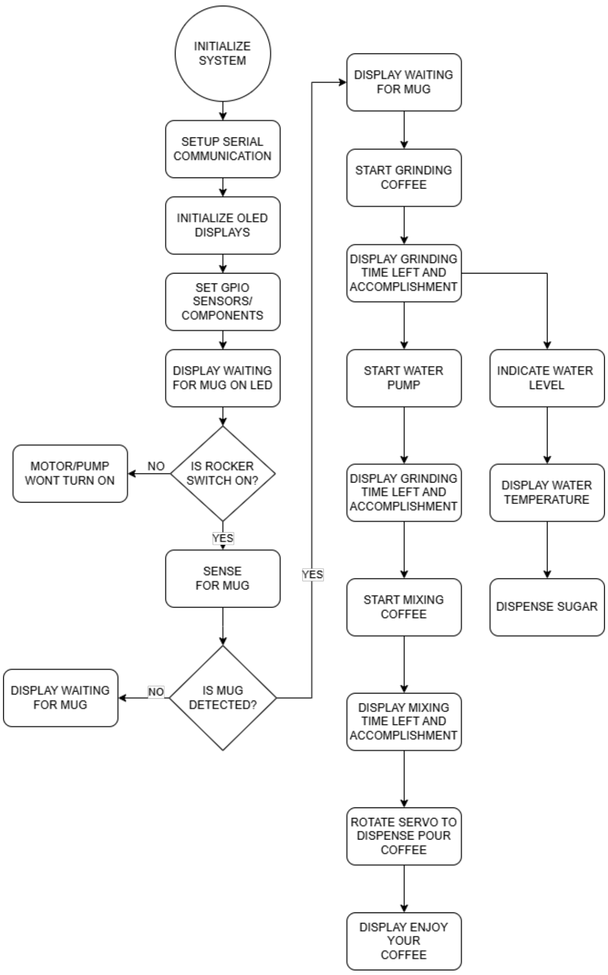
```
BEGIN loop
    IF switch is ON THEN
        IF IR sensor detects mug THEN
            DISPLAY "Mug Detected"
            IF water level is sufficient AND temperature is normal THEN
                TURN ON water pump
                WAIT 3 seconds
                TURN OFF water pump
                START heater until temperature reaches ideal range
                DISPLAY "Brewing..."
                START motor for grinding/mixing
                WAIT 5 seconds
                STOP motor
                BUZZER ON for 1 second
                DISPLAY "Coffee Ready"
            ELSE
                DISPLAY "Check Water or Temp"
                BUZZER ON for warning
        ELSE
            DISPLAY "Place Mug"
    ELSE
        DISPLAY "System OFF"
END loop
```

2nd Arduino (Pseudocode):

```
BEGIN setup
  START Serial Communication at 9600 bps
  SET pin modes for LEDs and button
  ATTACH servo motor and set initial angle
  INITIALIZE temperature sensor
```

```
  INITIALIZE OLED display
END setup
BEGIN loop
  READ water level from analog sensor
  DISPLAY water level on Serial Monitor
  IF water level is low THEN TURN ON red LED
  ELSE IF medium THEN TURN ON yellow LED
  ELSE TURN ON blue LED
  ENDIF
  IF button is pressed THEN ROTATE servo to open valve
  ELSE ROTATE servo to closed position
  ENDIF
  READ temperature from DS18B20 sensor
  DISPLAY temperature on Serial Monitor and OLED screen
  SHORT delay to prevent flickering
END loop
```

5.2 Flowchart:



6 Future Work and Potential Applications

This section discusses the functionalities that could be added further to extend the ease of usage.

6.1 Future Improvements

- Integrate a touchscreen UI for beverage selection and custom control.
- Add volumetric flow sensor for precise brew volume.
- Develop a mobile app for complete remote control.

6.2 Applications

- Home automation systems integrated with smart-home platform.
- Office smart kitchens requiring unattended brewing.

7 Literature and Technical Sources

- **DIY Arduino Project: Controlling a Water Pump with TA6586 Chip!**
  `https://youtube.com/shorts/WwMWVLKI9cA?si=sFR88NS9l7I8ZsHl`