An-Najah National University

Faculty of Engineering & Information Technology

Presented in partial fulfillment of the requirements for

Bachelor degree in Computer Engineering

Graduation Project 2

MilkShake Machine

Students:

Eba Khalil

Samah Sheeha

Supervisor:
Dr.Abdallah
Rashed

September 7 , 2024

Acknowledgment

We extend our heartfelt thanks to Dr. Abdallah Rashed, our supervisor, for his invaluable guidance and support in enhancing our project. We also wish to express our deep gratitude to our family, friends, and everyone else who contributed their encouragement and assistance throughout this journey.

Disclaimer Statement

This report was prepared by Eba Khalil and Samah Sheeha, students in the Computer Engineering Department at the Faculty of Engineering, An-Najah National University. The report remains unchanged, except for editorial adjustments, and may contain language or content errors. The views, outcomes, and recommendations presented are entirely those of the students. An-Najah National University assumes no responsibility or liability for any consequences arising from the use of this report for purposes other than its intended one

Contents

List of Figures

Abstract

The milkshake machine is a fully autonomous system that handles everything from selecting the drink to receiving it. Since the machine operates independently, there’s no need for a worker to prepare milkshakes, saving both money and effort. It’s also easy to use and safe, making it ideal for schools, universities, and institutions as a fresh and healthy alternative to coffee machines.

Our machine is connected to a mobile application, allowing clients to choose their desired milkshake. They can select from predefined options or customize their drink using a keypad and LCD or the mobile app. After choosing, the client pays via RFID, and the machine starts the process by measuring the ingredients, blending them with milk, and pouring the milkshake into the selected cup size (two sizes available). The machine then cleans the blender and prepares for the next order.

In our project, we used an Arduino Mega microcontroller and an ESP32 module to manage interactions with the blender, pumps, valves, and various motors (stepper and DC). We used several ultrasonic sensors to ensure everything operates accurately. An IR sensor detects the required ingredient, activating the corresponding motor. We also used a laser with an LDR module for multiple tasks: one to lower the cup, another to open the blender valve to dispense the milkshake, and a third to cover the cup.

Chapter 1

Introduction

1.1 Statement of the problem

The Milkshake Machine, unlike traditional coffee makers, aims to provide consumers with fresh and healthy beverages. It addresses several issues found in other drink dispensers. A key problem is that customers often lack control over the exact amount of each ingredient added to their drinks. Additionally, most machines do not support ordering through mobile apps, nor do they offer real-time updates on the ingredients used or the remaining quantities available.

1.2 Objectives of the work

The goal of this project is to develop a modern milkshake dispenser that meets the needs of today's consumers. We are introducing a mobile app that allows users to easily order and pay while giving them greater control over the customization of their drinks. The machine will provide real-time updates on sensor readings and the remaining quantities of each ingredient, ensuring a smooth experience for both customers and administrators. Our aim is to resolve the limitations of existing drink dispensers and cater to the growing demand for personalized, tech-friendly experiences in the food and beverage industry.

1.3 Significance of our work

As demand grows for convenient, customizable beverages, people are increasingly seeking control over the ingredients in their drinks. Additionally, mobile ordering for food and drinks is becoming more popular. Our milkshake machine is designed to meet these needs. The work we're doing has the potential to revolutionize how drink dispensers operate, offering a unique and practical solution that aligns with modern consumer preferences. By enabling users to tailor their drinks to their tastes, our machine could significantly improve the beverage experience for everyone.

1.4 Organization of the report

This report is structured into several key sections. The introduction offers an overview of the milkshake project and its objectives. The second section defines the scope and limitations of the work. The third section details the methodology and procedures used throughout the project. The fourth section presents the results and findings, including any challenges faced and the solutions implemented. The fifth section discusses the significance and potential impact of the project. Finally, the conclusion summarizes the main points and provides recommendations for future improvements. Appendices are included to provide additional data and information relevant to the project.

Chapter 2

Constraints, Standards/ Codes and Earlier course work

2.1 Constraints

While designing and constructing our machine, we encountered several constraints:

1. Power Supply: Our machine had several electrically powered components, including a blender that required 220 volts, motors, pumps, and valves that needed 12 volts, and an LCD that required 5 volts. By using a power supply unit in combination with an Arduino to generate 5 volts and 12 volts, we were able to address these power requirements. For the 220 volts, we utilized standard household or lab electricity.

2. Stepper Motor: Initially, we tried to rotate the disk with the components using a previously utilized Nema stepper motor, but this attempt was unsuccessful. We welded the stepper motor to the disk; however, this process damaged the stepper's coils due to the heat, which also caused issues with the YS-DIV268N driver. After extensive testing, we had to replace the damaged stepper motor with a new one to complete the task.

3. Wiring: The wires tended to become disconnected during movement, so we resolved this by securing them with welding and then applying silicone for added stability.

4. Mixer Challenge: Another challenge we faced was pour the mixture into the cup. We tried using valves and pumps, but the mixture kept getting stuck and clogging. As a solution, we designed a new system involving a faucet connected to a motor that could control its opening and closing, which effectively resolved the issue.

2.2 Standards and Codes

We developed the software components of the milkshake machine using the Arduino IDE, which is written in C++ and includes essential libraries such as Keypad.h, LiquidCrystalI2C.h, wire.h, HX711_ADC, and Servo.h. The user interface was created using App Inventor. Throughout the development process, we adhered to standards and guidelines relevant to the hardware components to ensure seamless integration and reliable performance.

2.3 Earlier coursework

The work we did in our earlier courses was essential in helping us develop the milkshake machine. The electronics course provided us with a solid foundation in essential electronics concepts, which proved invaluable when constructing and troubleshooting the hardware components, such as sensors, motors, valves, and pumps.

The course on microcontrollers using PIC controllers was particularly beneficial. It equipped us with extensive knowledge of programming microcontrollers, which was essential for building our machine using the Arduino Mega. We learned how to connect various components and utilize tools like I2C and PWM, which enabled us to write the core code that powers the machine.

The networks course, with its focus on communication protocols and networking fundamentals, allowed us to establish the machine's networking infrastructure. This was critical for developing a solution that enabled users to control the machine remotely via a mobile app and integrate the ESP module.

Finally, the Critical Thinking course was instrumental in guiding our approach to the project. It provided us with the ability to think methodically and make informed decisions, which was key to the project's success. The course helped us hone our critical thinking skills, enabling us to identify potential issues, analyze them, and devise effective solutions. These skills were especially important as we faced design and power challenges throughout the project.

Chapter 3

Literature Review

The advancement of autonomous drink vending systems and technical innovation has led to the creation of our hardware graduation project: an autonomous milkshake dispensing system that surpasses conventional drink machines. This project represents a comprehensive reimagining of the beverage experience, encompassing the entire process from order placement to drink delivery. By eliminating the need for human intervention, this fully autonomous system not only reduces costs and labor but also emphasizes ease of use and safety. The potential impact of this innovation is particularly significant in educational institutions, where it offers a customized, health-focused alternative to traditional coffee machines.

Earlier innovations have paved the way for similar advancements. For instance, Barron Noel, Mulder Luke, and Hieltjes Ben developed the "Bartini automated drink dispensing system," a cutting-edge solution for the hospitality industry. This system allows users to automatically pour mixed drinks through a simple interface. Using a graphic user interface (GUI), users can choose from preset drink recipes or customize their own based on the ingredients available in the machine. The entire process—valve control, motor operation, and the GUI—is managed by a single embedded processor system (Barron et al., 2016).

Similarly, an "IoT-based Smart Automatic Juice Vending Machine" was developed by student Sarvesh Pandey and Assistant Professor Mrs. Manjiri M. Gogate at SLRTCE, Maharashtra University, India. This project focused on providing a smart, coin-operated juice vending machine that offers various kinds of fresh juice, utilizing IoT to enhance the user experience (Pandey & Gogate, 2017).

Another great example is the smart coffee vending machine created by Kwangsoo Kim, Dong-Hwan Park, Hyochan Bang, and Geonsoo Hong, which automatically adjusts the amount of coffee, sugar, and creamer based on the customer’s taste preferences. This study highlighted the potential for sensor and actuator networks to improve the customizability of vending machines (Kim et al., 2014).

In our milkshake project, customers can select from a menu of predefined or customized drinks using a keypad or smartphone application, and then pay using RFID technology. To ensure precision and integration, we utilize various sensors, such as ultrasonic and IR sensors, to measure ingredient quantities and confirm cup presence and placement, ensuring the seamless delivery of the desired beverage. Similarly, the "IoT-based Smart Automatic Juice Vending Machine" employed juice level sensors and solenoid valves for dispensing, with data wirelessly transmitted to the cloud via the ESP-8266 module (Pandey & Gogate, 2017).

While our project shares common ground with these previous studies, it introduces a unique approach by emphasizing the use of raw ingredients to create healthier milkshakes, distinguishing it from traditional vending machine offerings. This focus on health-conscious customization represents a significant step forward in the field of vending machine innovation.

Chapter 4

Methodology

In this chapter, we'll take a closer look at the hardware parts used to create the milkshake machine, explaining how they all connect and work together in the overall design. We'll also go over how the system functions and walk through the development and integration of the software and mobile app to ensure everything runs smoothly for the user.

4.1 Hardware Components

4.1.1 Microcontrollers

Arduino Mega 2560

The Arduino Mega 2560 is a powerful microcontroller that's perfect for our milkshake machine project. It's built on the ATmega2560 chip and comes packed with useful features, including 54 digital input/output pins (with 15 for PWM output), 16 analog inputs, 4 serial ports, and a 16 MHz crystal oscillator. It also has a USB connector, power jack, ICSP header, and a reset button. With all these capabilities, it's well-suited to handle the various devices and components we'll be connecting in our project.

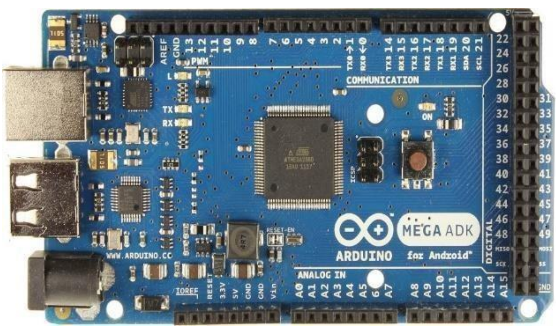


Figure 4.1: Arduino Mega 2560

ESP32-DevKitC ESP32-WROOM-32U Core Board

The ESP32-WROOM-32 is a powerful module that combines Wi-Fi, Bluetooth, and Bluetooth Low Energy (LE) capabilities. It's highly adaptable and can be used for everything from simple, energy-efficient sensor networks to more complex applications like voice recognition, music streaming, and MP3 playback. In our project, the ESP32 module plays a crucial role by connecting the system to Wi-Fi, enabling the operation of the mobile application that controls the milkshake machine.

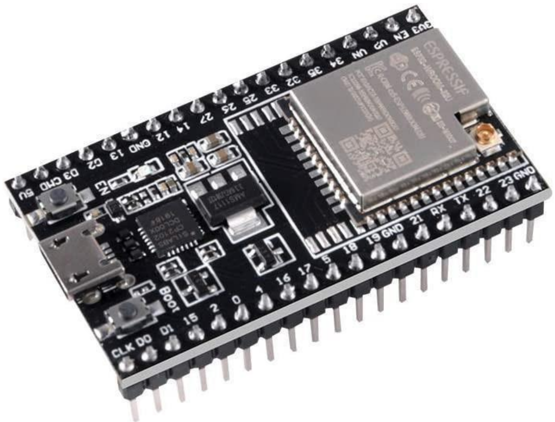


Figure 4.2: ESP32-DevKitC ESP32-WROOM-32U Core Board

4.1.2 Motors and drivers

J-5718HB2401 Stepper motor and YS-DIV268N driver

The J-5718HB2401 is a stepper motor designed for precise control, allowing for discrete rotations of its shaft in response to electrical current pulses. Despite their potential size, stepper motors like this one are capable of operating on low current levels, making them efficient for various applications. In our milkshake machine project, this stepper motor, with its ability to handle a weight load of up to 20 kg, is tasked with rotating the disk that holds the containers of ingredients.

To drive the stepper motor, we employed a 12A power source alongside the YS-DIV268N driver. The motor coils were connected to the A and B pins of the driver, while the control pins were connected to the corresponding Arduino pins, with the Arduino's ground pin linked to the driver's negative pins. The DC+ and DC- pins were powered by the 12A power supply.

We developed a program utilizing the Arduino’s pulse width modulation (PWM) pins to control the motor’s movement. The enable pin for each motor is consistently active, while the direction pin is adjusted based on the desired rotation—initially moving clockwise, then reversing to counter-clockwise after each order to return to its original position. The number of pulses can be modified to control the degree of movement, and the delay between pulses can be adjusted to control the motor's speed.

Figure 4.3: J-5718HB2401 Stepper motor

Figure 4.4: YS-DIV268N driver

DC motor and H-Bridge

- An electrical device that transforms electrical energy into mechanical energy. It is made up of a stationary field magnet that produces a magnetic field and a spinning armature that rotates. The armature receives an electric current, which creates a torque that turns the motor.
- We reused a computer disk's DC motor for our project because the 3D-printed cup dispensing element needed a DC motor to be driven. We used an H-Bridge to control how the motor operated. The voltage and current levels of the Arduino are too low to directly control the DC motor, regardless of its ability it can generate a PWM signal. So, between the Arduino and the DC motor, we incorporated a hardware driver called the H-Bridge.
- In our design, the H-Bridge served two purposes. First, it increased the voltage and current levels of the Arduino PWM signal, enabling speed control. In order to offer directional control, it also switched the power supply's pole after receiving the control signal from the Arduino.

Figure 4.5: DC motor

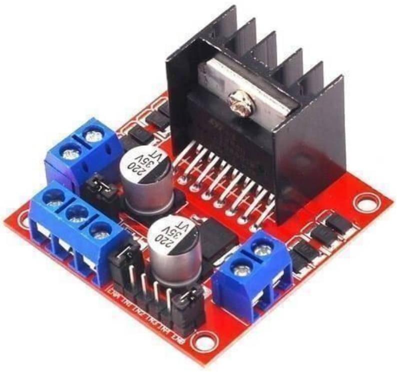


Figure 4.6: H-Bridge

4.1.3 Sensors

IR Sensor Module

An electrical sensor known as an infrared sensor picks up infrared radiation from its surroundings. Based on variations in the infrared radiation that objects emit, the sensor is made to check the temperature of objects or detect motion. Many different applications, including remote temperature sensing, motion detection, and security systems, frequently employ infrared sensors.

In our project, the IR sensor module was applied to the conveyor belt in three different places. The goal was to find out whether cups were present at the points of cup dispensing, drink pouring, and cup stopping at the end of the conveyor belt.

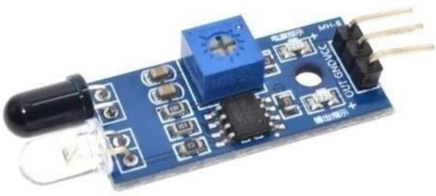


Figure 4.7: IR Sensor Module

Ultrasonic Sensor

An ultrasonic sensor measures how far away an object is by using sound waves that are too high-pitched for humans to hear. It sends out sound waves, which hit the object and bounce back. By timing how long it takes for the sound waves to return, the sensor can calculate the distance to the object.

was used to measure the distance to calculate the number ofcomponents still present in the fruit container and the pot that holds the juice and water. The administrator sees this information so they may replenish the materials as needed.

Figure 4.8: Ultrasonic Sensor

LDR module

In our project, we used the LDR module in combination with a laser to detect specific actions. The LDR and laser were aligned to detect when cups are lowered, when the mixer faucet opens to pour the milkshake, and when the cup is covered. This setup allowed for precise control and automation of these processes.

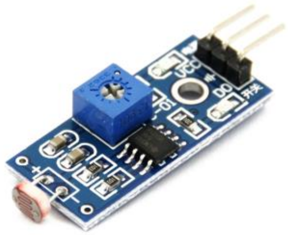


Figure 4.9: LDR module

4.1.4 Input/Output Devices

LCD and I2C

Liquid Crystal Displays (LCDs) are electronic displays commonly used in various applications due to their ability to clearly present information. A 20x2 LCD, which can display 20 characters per row across two rows, is a specific type that’s particularly useful in scenarios requiring the display of considerable data, such as in consumer electronics, medical devices, and industrial systems.
In our milkshake project, we integrated a 20x2 LCD as an output device to provide clients with relevant information and instructions, enhancing the user experience. The LCD displays prompts and questions that guide the user, who then inputs responses via a keypad. The system processes these inputs and presents the corresponding results on the LCD, creating a smooth and intuitive interaction.
To simplify the connection between the microcontroller and the LCD, we utilized an I2C Serial Interface Adapter. This small module uses the I2C communication protocol to convert the parallel signals from the LCD into serial signals, allowing the display to be controlled with just two I/O pins from the microcontroller. This not only reduces the number of wires needed but also allows multiple devices to share the same I2C bus, making it an efficient solution for space-constrained environments.

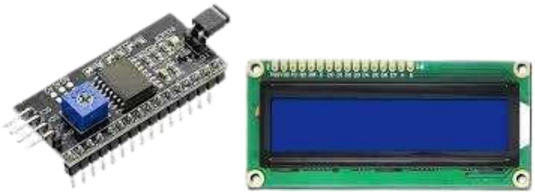


Figure 4.10: 20*2 LCD and I2C

RFID

Radio Frequency Identification (RFID) technology works with two key parts: tags and readers. The reader, which has one or more antennas, sends out radio waves and picks up signals from RFID tags in the area. These tags can either be passive or active. Passive tags don't need a battery—they get their power from the reader's signal. Active tags, on the other hand, have their own internal battery to power them. Both types of tags send their unique ID and other information back to the reader.

In our milkshake project, we implemented RFID as an authorization method. Customers can conveniently place orders by scanning their RFID cards, which the system uses to link their unique ID with their chosen drink. This seamless integration simplifies the payment process, making it efficient and user-friendly.

Figure 4.11: RFID

Keypad

A keypad is a set of buttons arranged in a matrix, with each button identified by its row and column coordinates. In microcontroller-based systems, especially those developed with the Arduino platform, keypads are commonly used as input devices.
In our project, the keypad serves as an input device that allows customers to select their preferred beverage. Instructions are clearly and simply displayed on the accompanying LCD screen. The customer can then confirm their choice by entering the information on the keypad.

Figure 4.12: Keypad

Coupling

The Aluminum Flexible Shaft Coupling is a lightweight yet robust component known for its superior strength. In our milkshake project, we utilize this coupling to securely connect the stepper motors to the ingredients, ensuring smooth and efficient operation.

.

Figure 4.13: Coupling

4.1.5 Power Devices

Power Supply

We chose to use a computer power supply since it can offer the required 5 volts for numerous devices and 12 volts for pumps and stepper motors in order to meet the voltage specifications for our project. The power supply also provides a sufficient current output to fulfill the needs of our project.

Figure 4.14: Power Supply

4.1.6 Other Devices

3D Cup dispensing

In our machine, we incorporated a 3D-printed cup dispenser that serves as a separator, ensuring that a single cup drops down while maintaining the rest of the stack securely in place.

Figure 4.15: Cup Dispensing Piece

Conveyor Belt

In our milkshake project, a conveyor system is used to smoothly transport the cups from the dispensing area to the pouring station, ensuring that each cup is perfectly positioned for filling before the customer can pick it up and enjoy their drink.

Figure 4.16: Conveyor Belt

Relay

Relays are electronic switches that enable low voltage and current signals to control higher voltage and current loads, we used 5-volt relays that can be easily managed by the Arduino microcontroller. These relays typically come on a small circuit board equipped with an LED indicator and screw terminals, allowing for seamless connection between the control signals and the high-power components in our system.

Figure 4.17: Relay Module

Valve

In our milkshake project, we use a valve to control the amount of milk. The valve opens and closes to regulate the milk flow accurately, ensuring the desired quantity for each order. This integration helps in maintaining consistency and quality in the milkshake preparation.

Figure 4.18: Valve

Mixer (Blender)

we utilized a blender, an electric device operating at 220 volts, specifically designed for blending ingredients. The blender is connected to a relay controlled by the Arduino, which regulates its operation, ensuring the blender runs precisely when needed to prepare the perfect milkshake.

Figure 4.19: Mixer(Blender)

Ingredients Container

This plastic component serves as a container for the milkshake ingredients. It is equipped with a rotating blade mechanism attached to a stepper motor. As the stepper motor moves, the ingredients are gradually released from the container, falling directly into the blender for smooth mixing and preparation of the milkshake.

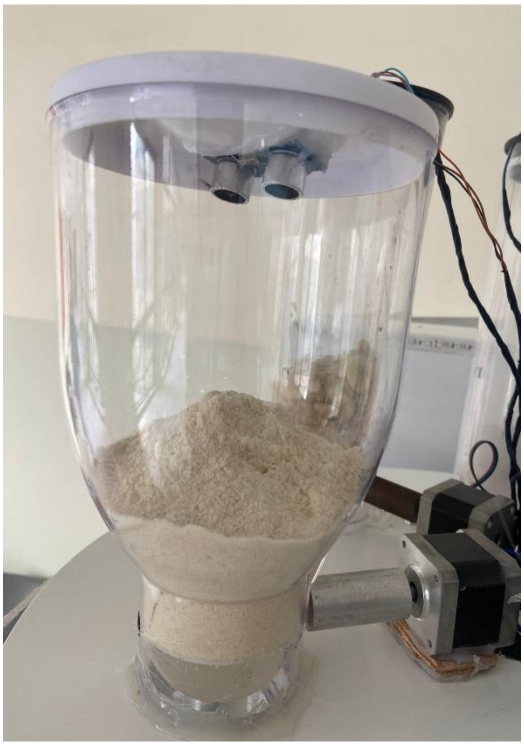


Figure 4.20: Ingredients Container

Pumps

In the cleaning mode, we utilized a pump to deliver water into the blender, ensuring it is thoroughly cleaned after preparing each cup of milkshake, maintaining hygiene and readiness for the next order.

Figure 4.21: Pump

A4988 Stepper Motor Driver

We used a set of A4988 stepper motor drivers, with each driver dedicated to controlling a specific stepper motor. These drivers provide precise control over the motor's speed and direction. Additionally, they offer built-in overcurrent protection and are easily interfaced with our Arduino system, ensuring smooth operation and efficient motor management throughout the milkshake preparation process.

Figure 4.22: Stepper Motor Driver

In our milkshake project, we used a laser placed opposite an LDR sensor. The laser beam is interrupted when a cup is lowered, when the mixer faucet is opened to pour the milkshake, or when the cup is covered. This precise detection ensures a smooth workflow, making the laser essential for the system's reliability and efficiency.

Figure 4.23: Laser

Peltier module

In our project, we used a Peltier module for cooling the milk. This device works on the principle of the Peltier effect, where heat is absorbed on one side and dissipated on the other when an electric current passes through it. The Peltier module is an efficient and compact solution for maintaining the milk at the desired temperature, ensuring product quality throughout the process.

Figure 4.24: Peltier module

Fan

we used a fan to help cool the milk. The fan works by circulating air around the cooling area, ensuring even temperature distribution and aiding the cooling process. This simple yet effective component enhances the efficiency of the cooling system, maintaining the milk at the optimal temperature for the best quality.

Figure 4.25: Fan

heat sink

In our project, we used a heat sink to assist in cooling the milk. The heat sink absorbs and dissipates excess heat from the cooling system, helping to maintain a stable temperature. Its design, with metal fins, maximizes surface area for better heat dissipation, making it a crucial component for efficient cooling and ensuring the milk stays at the ideal temperature.

Figure 4.26: Heat Sick

Breadboard

We utilized a breadboard to connect various components, including the 5V and 12V power supplies, ensuring seamless connections to the electronics that required these specific voltages.

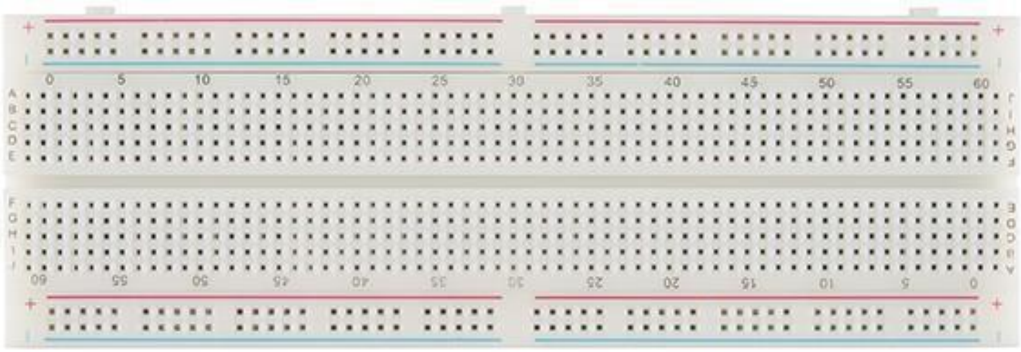


Figure 4.27: Breadboard

Wires

We utilized three types of wires—male-to-male, female-to-female, and male-to-female—to handle different connection requirements throughout the system.

Figure 4.28: Wires

4.2 Software Implementation

When placing an order, the user starts by specifying the number of cups they want, choosing the cup size (small or medium), selecting the flavor (Cerelec, Oreo, or Banana), and determining the preferred sugar level. Once these details are entered, the user presses the "Start" button to initiate the process.

Ingredient Dispensing:

The machine’s system begins rotating and stops at each ingredient station according to the user’s selection.
Ingredients are dispensed into the blender in precise quantities based on the user's choices.
Mixing:

Once all ingredients are added, the blender is activated to mix the ingredients thoroughly.
Simultaneously, a cup is positioned on the conveyor belt, which moves the cup to the dispensing station.
Pouring and Sealing:

The conveyor belt continues to transport the cup to the pouring point, where the milkshake is dispensed into the cup.
After the pouring process, the conveyor moves the cup to the sealing station where a lid is placed on the cup to ensure freshness.
Cleaning:

Following the completion of the milkshake preparation, the machine automatically enters cleaning mode.
The pump starts dispensing water into the blender, and the blender is activated to clean any residue from the mixing process.
Once the cleaning cycle is complete, the cup is removed, and the machine is readied for the next order.
Preparation for Next Order:

The cup is removed from the conveyor, and the system resets to be ready for the next order.
This description ensures that each step of the milkshake preparation process is clear and user-friendly.

4.2.1 Flow Chart

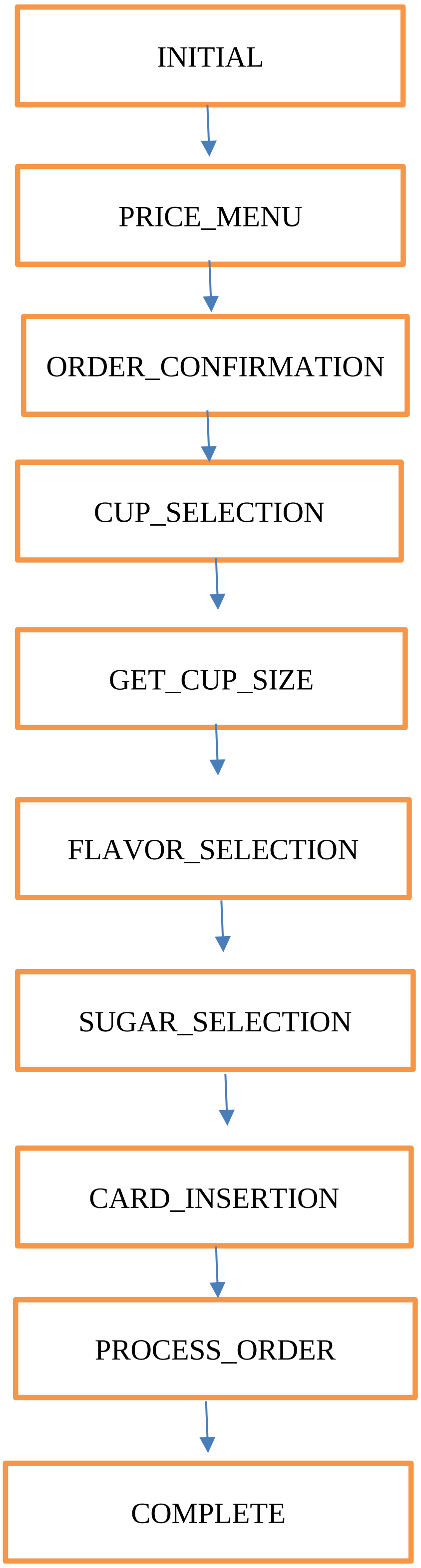


Figure 4.29: flow chart

4.3 Hardware Implementation

4.3.1 Rotating Disk and Ingredients Containers

Our setup features a circular wooden disk that holds the ingredient containers. This disk rotates, positioning each container directly above the blender so the ingredients can fall seamlessly into it. Each container is equipped with a ball featuring blades, which is connected to a stepper motor via a coupling. When the motor rotates, the ball spins as well, releasing the ingredient directly into the blender.

4.3.2 Input-Output Unit

User interaction is facilitated through input/output components. An LCD displays the available drink options and system status, while the keypad allows the user to make selections. Additionally, an RFID system is integrated for payment, enabling users to conveniently pay for their drinks.

4.3.3 Controlling Unit

The Arduino handles all machine operations in the control section. The ESP32, connected to a mobile app, manages the ultrasonic sensors and communicates with the Arduino. This allows users to place orders through the app, with system supervision managed by the administrator.

4.3.4 Monitoring Unit

Our system's monitoring is achieved through various sensors. Ultrasonic sensors track the levels of each ingredient and milk, while infrared sensors detect the presence of the cup. The infrared sensor also ensures that the cup reaches the correct pouring and capping positions.

4.3.5 Mixing Unit

The blender, controlled by the Arduino, is responsible for mixing the ingredients to create the desired drink.

4.3.6 Pumping Unit

A pump is used to direct water into the blender, while a valve is employed to regulate the flow of milk, ensuring no leaks during the process.

4.4 Mobile Application

We created a straightforward mobile app using App Inventor that allows users to easily order their drinks. We then uploaded its code to an ESP32 board to ensure smooth functionality.

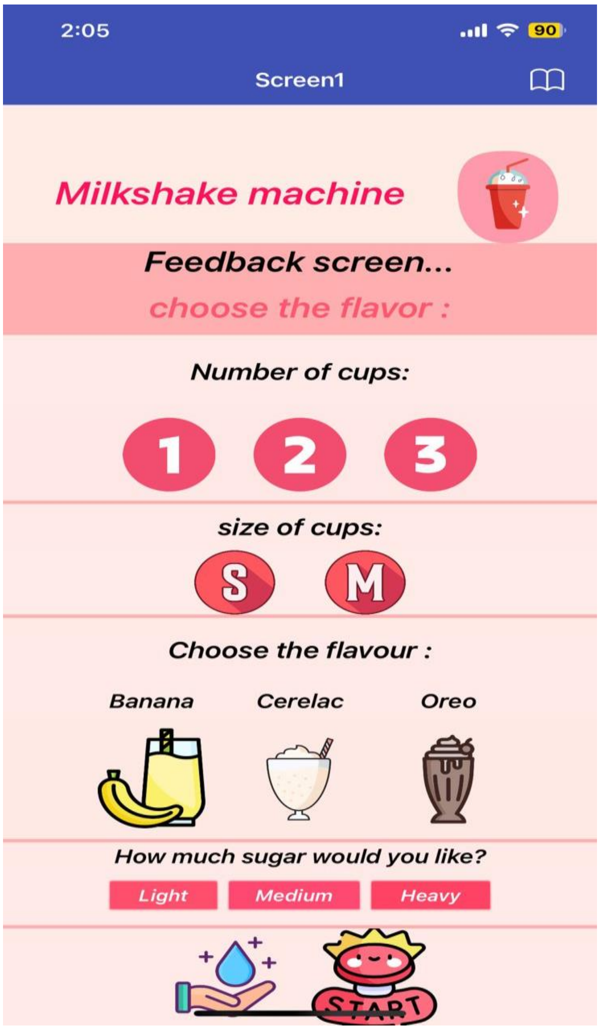
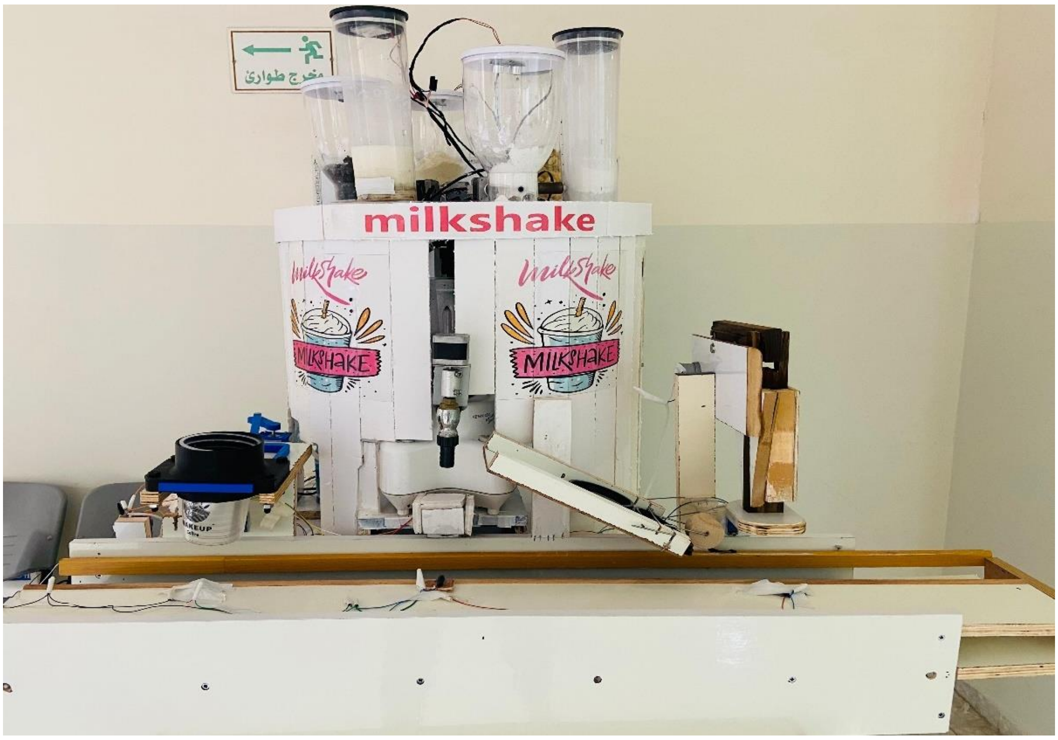


Figure 4.30: Mobile Application

4.5 Full System images



lkshake
MilkShake

MILKSHa
WAKEUP
Coffee

Chapter

Results and Discussion

As our project wrapped up, we developed an autonomous and smart milkshake machine that met our goals perfectly. The admin can easily track ingredients and their containers via a mobile app, while users can place orders effortlessly.

During our road trip, we encountered several challenges and addressed them effectively:

- ESP32 Upload Issue: We struggled with uploading code to the ESP32, which initially resisted our efforts. After extensive troubleshooting, we found that booting the device during the upload resolved the issue. We also addressed serial communication problems between the Arduino Mega and ESP32 by using a voltage divider circuit with 1kΩ and 2.2kΩ resistors to match voltage levels.

- Stepper Motor Challenge: Initially, the disk or roller wouldn't rotate with our stepper motor. We then switched to the J-5718HB2401 stepper motor, which proved to be the optimal choice after testing.

- Powering Components: Managing power for various components was tricky due to differing voltage requirements (220V and 12V). We used a 12V power supply and the Arduino for 5V, while household and laboratory electricity handled the 220V components.

By overcoming these hurdles, we ensured our smart milkshake machine operated smoothly and achieved our intended goals.

Chapter

Conclusions and Recommendation

6.1 Summary

We developed the Milkshake Machine, an innovative automated system designed to simplify the process of ordering milkshakes for customers. The machine features both a smartphone application and a remote-ordering keypad for seamless interaction.

One of the standout features of our machine is its ability to handle multiple ordering options effortlessly. Customers can place orders for multiple cups of their chosen flavor in a single transaction. The machine also offers customizable sugar levels, allowing users to choose between light, medium, or heavy sweetness. Additionally, the machine ensures convenience with its automatic cleaning mode, which activates after each milkshake is made, maintaining hygiene effortlessly.

The Milkshake Machine supports payment via RFID cards, enhancing the payment process's ease and security. Users can select from three types of drinks and choose between small or large cup sizes. The machine also includes a cup-covering feature to ensure drinks are securely sealed.

During development, we encountered and resolved several challenges. Powering the different components of the machine posed a challenge, as some required 12 volts while others needed 220 volts. We addressed this by repurposing power supply and arduino to provide both 5V and 12V, while regular household and laboratory electricity was used for the 220V components. Additionally, we replaced an old stepper motor that was unable to rotate the disk with a more reliable alternative.

In conclusion, the Milkshake Machine is a sophisticated and user-friendly device that streamlines the process of ordering milkshakes. Its advanced features, including customizable options, efficient cleaning, and secure payment methods, offer a distinctive and enjoyable experience for users.

6.2 Recommendations

Recommendations:

1. Handle the Arduino board with care to ensure its longevity and proper functioning.

2. Avoid powering sensors and other devices directly from the Arduino board. Instead, use a separate, reliable power source.

3. Since wires can be prone to damage, make sure to solder them rather than relying on simple connections for greater durability.

4. Test each component thoroughly before integrating it into the project to ensure everything works as intended.

6.3 What we have learned

1. Learn the procedures for operating equipment such as pumps, blenders, and valves, as well as handling sensors like ultrasonic and infrared, along with DC and stepper motors.

2. Follow guidelines for integrating and connecting high-voltage sensors and devices with the Arduino.

3. Utilize the Wi-Fi features of the ESP32 by connecting it to the Arduino for enhanced functionality.

6.4 Future Work

- Develop a system that automatically refills ingredients, milk, and water to streamline operations.

- Expand the variety of ingredients and milkshake flavors available for a broader selection.

- Introduce a feature to monitor and generate reports on drink orders for better tracking and analysis.

References

Barron, N., Mulder, L., & Hieltjes, B. (2016). *Bartini Drink Dispensing System [automated drink dispensing system]*.

Kim, K., Park, D.-H., Bang, H., Hong, G., & Jin, S. (2014). Smart coffee vending machine using sensor and actuator networks. *2014 IEEE International Conference on Consumer Electronics (ICCE)*, 71–72. https://doi.org/10.1109/ICCE.2014.6775913

Pandey, S., & Gogate, M. M. M. (2017). Iot based smart automatic juice vending machine. *International Journal of Engineering & Technology*, *5*(2), 171–176.

Ratnasri, N., & Sharmilan, T. (2021). Vending Machine Technologies: A Review Article. *International Journal of Sciences: Basic and Applied Research (IJSBAR)*, *58*(2), 160–166.