



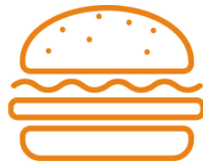
An-Najah National University

Faculty of Engineering & Information Technology

Computer Engineering Department

Presented in partial fulfilment of the requirements for Bachelor degree in
Computer Engineering

Graduation Project 2



BURGER STATION

Students:

Jana Khammash

Zain Abubaker

Supervisor:

Dr. Aladdin Masri

January 26, 2025

Acknowledgement

“

We would like to start by thanking Dr. Aladdin Masri, our supervisor, for all his help and support throughout this project. His guidance, advice, and feedback were very important in making our work better. We are really thankful for the time and effort he gave us. We also want to thank our friends for always being there for us. Your encouragement, help, and teamwork made this project much easier and more enjoyable. We truly appreciate your patience and support. A big thank you goes to our families for their love, care, and constant support. You believed in us from the start and helped us stay strong, even when things got tough. Your love and sacrifices gave us the motivation we needed to keep going. Finally, we are grateful to everyone who helped us with this project in any way. Whether it was through advice, assistance, or just being there for us, your support means a lot, and we truly appreciate it.

”

-Zain, Jana

Disclaimer

This report was written by Jana Khammash and Zain Abubaker at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Table of Contents

1	Abstract.....	7
2	Introduction.....	8
	2.1 General Background.....	8
	2.2 Problem.....	8
	2.3 Objectives.....	8
	2.4 Scope of the Project.....	9
	2.5 Importance.....	9
	2.6 Report Organization.....	9
3	Constraints, Standards, and Earlier Course Work.....	10
	3.1 Constraints.....	10
	3.1.1 Cost and Budget Constraint.....	10
	3.1.2 Project Structure Constraints.....	10
	3.1.3 Lack Of Mechanical knowledge.....	10
	3.1.4 Time Duration Constraint.....	10
	3.2 Standards.....	11
	3.2.1 Programming Language and IDE.....	11
	3.2.2 Application Development.....	11
	3.3 Earlier Course Work.....	11
4	Literature Review.....	12
5	Methodology.....	14
	5.1 System Architecture.....	14
	5.2 Hardware Components.....	17
	5.3 Software and Libraries.....	28
	5.3.1 Libraries used in Arduino code.....	28
	5.3.2 Libraries used in ESP32 code.....	29

5.4	Implementation.....	30
5.4.1	System Design.....	30
5.4.2	Machine Process of Work.....	32
5.4.3	Flow Chart.....	33
5.5	Final Project.....	34
5.6	Mobile Application.....	37
6	Results and Discussion.....	41
6.1	Results.....	41
6.2	Discussion.....	41
7	Conclusion and Recommendation.....	43
7.1	Conclusion.....	43
7.2	Recommendation.....	43
7.3	Future Work.....	44
8	References.....	45

List of Figures

Figure 4.1 Production lines Logistics Market.....	12
Figure 4.2 Product Lines Market Segmentation.....	13
Figure 5.1 Initial Design.....	14
Figure 5.2 Bread Structure.....	15
Figure 5.3 Topping Structure.....	15
Figure 5.4 Syringes Structure.....	15
Figure 5.5 Oven Structure.....	16
Figure 5.6 Burger Arm.....	16
Figure 5.7 Arduino Mega 2560.....	17
Figure 5.8 ESP-WROOM-32.....	18
Figure 5.9 Power Supply.....	18
Figure 5.10 Nema 17 Stepper Motor.....	19
Figure 5.11 L298N motor driver.....	19
Figure 5.12 J-5718HB2401 Stepper motor.....	20
Figure 5.13 YS-DIV268N Driver.....	20
Figure 5.14 Servo Motor.....	21
Figure 5.15 Keypad 1x4.....	21
Figure 5.16 RFID Reader.....	22
Figure 5.17 LCD 20x4.....	23
Figure 5.18 LCD Driver Module with 12C Interface.....	23
Figure 5.19 LDR Sensor.....	24
Figure 5.20 Laser.....	24
Figure 5.21 Ultrasonic Sensor.....	25
Figure 5.22 DC motor with gearbox.....	25
Figure 5.23 Single Channel Relay Module.....	26
Figure 5.24 Two Channel Relay Module.....	26
Figure 5.25 Jumper Wires.....	27
Figure 5.26 Standard Wires.....	27
Figure 5.27 Flow Chart.....	33
Figure 5.28 Final Project.....	34
Figure 5.29 Final Project.....	35
Figure 5.30 Mobile Application Dashboard.....	37
Figure 5.31 Mobile Application.....	37

1 Abstract

The goal of the Burger Station project is to create an automated burger production line. The system is controlled by an Arduino Mega, and integrated with sensors and mechanical components that will be discussed later in the report. The project involves creating a machine like a mini factory, making a mechanical assembly line so the burger moves on a conveyor belt step by step through a series of stations. The first step puts the bottom half of the burger bun on the packaging, while the raw meat is cooked until it's done. The machine adds sauces like ketchup and mayonnaise, it will also add toppings like tomatoes, lettuce, onions and cheese, which are the necessary components for any burger. Each ingredient is placed on the burger in an independent station. The machine will let the customer order using a keypad, they will be able to customize their burger order where they can pick their ingredients and preferences. A mobile application will be connected to it so people can order their burgers remotely. The system will include a payment feature using RFID so users can simply scan their card to pay. The project aims to provide a complete solution for assembling burgers from start to finish. It will make burger production more efficient, flexible and more precise. The project will cut down manual labor. And can be used in modern fast food restaurants. It addresses a growing trend towards automation in the food industry, which can help businesses meet high demand scenarios in a faster and more scalable way. While the concept of automating food production exists, like automated pizza machines, no project like Burger Station has been done before. Burger Station is unique in that it handles everything, from assembling to cooking and finalizing the burger. Burger Station's focus is on complete burger customization with the extra features we added and mentioned before. This project offers a comprehensive, automated solution that is not available on the market.

2 Introduction

2.1 General Background

As individuals, we often face inefficiencies in the fast-food industry, particularly during peak hours when preparing a customized burger can be time-consuming. Burger Station addresses this by introducing an automated burger production line. This innovative solution provides a quick, efficient, and fully automated process, producing high-quality customized burgers on demand. The integration of technology with food preparation revolutionizes how fast-food restaurants operate, offering convenience and precision while meeting high customer demand.

2.2 Problem

In fast-food restaurants, delays in making burgers are a common problem, especially during busy times. Customers often face long waits because the process of assembling and cooking burgers by hand takes time and can lead to mistakes. Traditional methods also make it harder to handle customized orders quickly, causing frustration for customers and lowering efficiency. Our project, Burger Station, solves these issues by automating the entire burger-making process. From cooking the patty to assembling and customizing the burger, the machine creates a smooth and quick experience for both customers and staff, reducing wait times and improving order accuracy.

2.3 Objectives

To address these challenges, Burger Station provides a complete automated solution for preparing and customizing burgers. The machine handles the entire process, from cooking the patty to assembling the burger and adding the chosen ingredients. Customers can easily customize their orders using a keypad or a mobile app, selecting their preferred toppings and sauces. The system also includes an RFID payment feature, allowing for quick transactions that will not need any cash or traditional card systems. This project aims to enhance efficiency in burger production by reducing manual work, minimizing errors, and speeding up the preparation process. By meeting the growing demand for fast and personalized food services, Burger Station is designed to improve customer satisfaction and support restaurants in handling high volumes of orders during busy periods.

2.4 Scope of the Project

The Burger Station project primarily targets fast-food restaurants and businesses looking to modernize their operations through automation. It also appeals to customers who value convenience, speed, and personalization in their dining experience. The project involves designing a fully automated burger production system that integrates mechanical components, sensors, and user-friendly interfaces. It serves a wide range of customers, from busy professionals to families, and is intended to operate efficiently in high-demand environments.

2.5 Importance

The development of Burger Station represents a significant advancement in the food service industry. By automating the entire burger-making process, this project minimizes human error, reduces labor costs, and enhances customer satisfaction through faster service and greater customization options. In a world where convenience and efficiency are paramount, Burger Station ensures that customers can enjoy freshly made, customized burgers without delays. It is particularly beneficial during peak hours, enabling restaurants to handle high order volumes while maintaining quality and precision. This innovation aligns with the growing trend of automation, offering a scalable solution for the future of fast food.

2.6 Report Organization

This report outlines the development of the Burger Station project, including:

- ❖ An introduction to the project and its objectives.
- ❖ A discussion of the constraints and challenges faced during the development process.
- ❖ A literature review exploring existing solutions in automated food production.
- ❖ The methodology used to design and implement the project.
- ❖ An analysis of outcomes, data gathered, and performance metrics.
- ❖ A discussion of limitations and areas for improvement.

Finally, the report concludes with recommendations, references, and resources that supported the project's success and development strategy.

3 Constraints, Standards, and Earlier Course Work

3.1 Constraints

3.1.1 Cost and Budget Constraint

Managing the project's budget while maintaining its functionality and durability was a challenge. The high cost of electronic components, coupled with repeated testing using ingredients and the damaged components during it, increased expenses. These costs restricted the addition of new features to stay in the budget.

3.1.2 Project Structure Constraints

We faced challenges in designing the structure of the project due to a lack of knowledge about measurements and the precise arrangement required for each component. Without a clear understanding of the optimal structure at the beginning, it consumed time and resources. Additionally, the frequent visits to the carpenter for minor modifications and additions in this repeated process delayed the progress. Furthermore, the availability of resources posed another challenge. The university's facilities and materials, as well as the carpenter we relied on for some components, were not consistently accessible.

3.1.3 Lack Of Mechanical knowledge

Our limited knowledge of mechanical systems made some parts of the project difficult to handle. Choosing the right materials was challenging because we risked picking materials that might not work well, which could cause the system to fail and put the whole project at risk. This slowed down our progress and made us rely more on experienced people. When we started building the project, we visited many shops to find the parts we needed, and sometimes we had to wait for the shops to get them in stock.

3.1.4 Time Duration Constraint

Although the project lasted the whole semester; however, we encountered several challenges that affected our timeline. The fall semester included multiple holidays, such as New Year's and other seasonal breaks, which reduced the number of working days. Additionally, city closures on certain days forced us to shift to online work, which slowed down progress on the physical aspects of the project. We also had to balance this project alongside other courses, final exams, and deadlines for other projects, which limited the time we could dedicate to it.

3.2 Standards

When building our burger station we followed a couple of standards and practices to achieve better functionality and ease of understanding the code.

3.2.1 Programming Language and IDE

The features and functionalities of the Burger Station were implemented using the Arduino IDE, a platform based on C/C++. The Arduino IDE was used to develop and upload code to the Arduino Mega, enabling the station's operations. Additionally, the ESP32 was programmed through the same platform to facilitate Wi-Fi communication with the application, leveraging Blynk libraries for seamless integration. [\[1\]](#)

3.2.2 Application Development

The Blynk platform enables prototyping, deployment, and remote management of connected devices. It connects hardware to the cloud and supports creating iOS, Android, and web apps. [\[2\]](#)

We developed a user-friendly mobile application that allows customers to remotely customize and place their burger orders for the Burger Station. This application integrates with the system using Wi-Fi communication, ensuring seamless interaction between the machine and the app. By leveraging a secure cloud platform, the application facilitates real-time order updates and ensures reliable data exchange, making the process convenient and efficient for users.

3.3 Earlier Course Work

The development and building of our production line, Burger Station, were significantly influenced by the knowledge and skills acquired from previous coursework. The computer engineering courses that we took played a huge role in helping us to build this hardware project. We took multiple courses that helped us understand the way that hardware components function, such as microcontrollers, microprocessors, and their labs. In these courses we learned more about the electronic components, such as stepper motors, keypads, LCDs, and many others; we also learned how they work and how to control them. Mainly, the microcontroller lab was very helpful because we did some experiments on Arduino that allowed us to know more about how it works. Also, the electronic and electrical courses we took were very useful for understanding the components we used and how they worked. In addition, we took an Arduino course with IEEE that was extremely helpful. We learned the basics of Arduino programming and how to connect them with the Arduino.

4 Literature Review

Food production lines have significantly improved since their early beginnings, evolving from simple manual operations into highly automated systems that use modern technology. These advancements have made food production faster, more efficient, and capable of handling complex tasks, such as customizing orders and ensuring high-quality standards. The shift towards automation, which gained momentum in the late 20th century, has allowed food production lines to process larger volumes while maintaining precision and consistency.

The demand for automated food production lines continues to grow, driven by the need for speed and customization in today's fast-paced world. The global food automation market has seen steady growth due to its ability to reduce labor costs, improve hygiene standards, and offer consistent product quality. Consumers increasingly expect fast and personalized options, and food production lines equipped with modern technologies have become a key solution to meet these expectations.

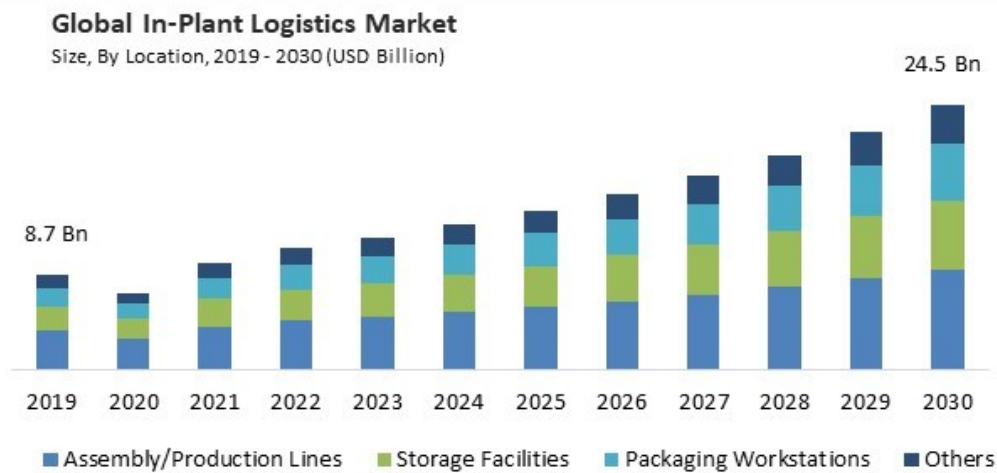


Figure 4.1 Production lines Logistics Market

Advanced technologies, such as the Internet of Things (IoT), Artificial Intelligence (AI), and cashless payment systems, are commonly used in modern food production lines. These technologies help streamline operations, manage inventory effectively, and improve customer satisfaction by enabling personalized and efficient food preparation. Additionally, the integration of sensors, stepper motors, and automated controls ensures accuracy and minimizes waste during the production process.

You can divide the production line market depending on the places that the production lines are placed in, these can give a general idea on how the production lines can provide an easy and fast customer experience especially in these environments. It's divided into corporate offices, hotels, restaurants, commercial places, and others. [3]

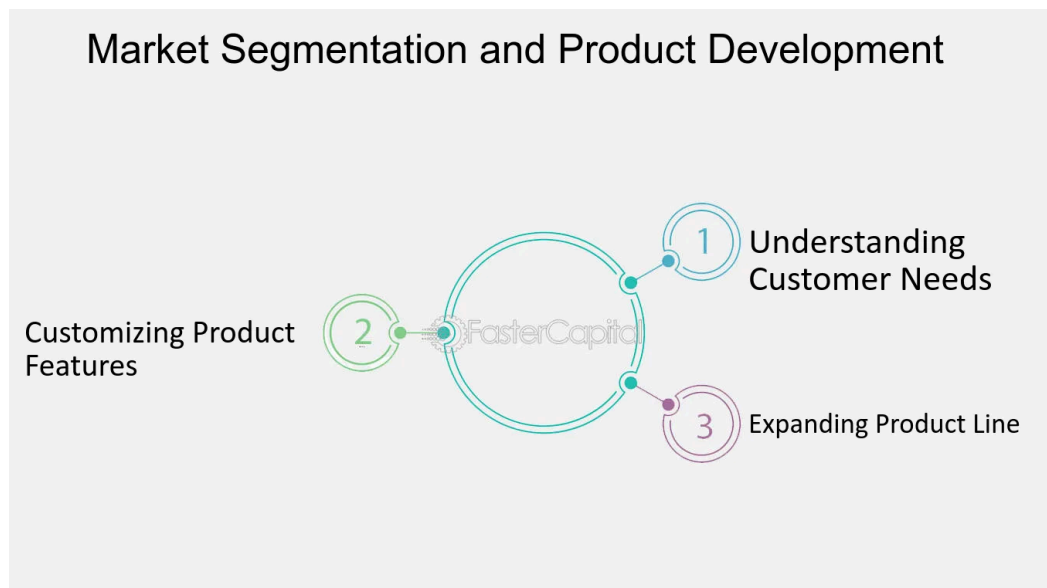


Figure 4.2 Product Lines Market Segmentation

Food production lines are versatile and can be customized for different environments, such as restaurants, catering businesses, and public spaces like event venues. These systems make it possible to handle varying levels of production while maintaining a seamless workflow and delivering high-quality food. However, incorporating advanced hardware and software increases costs and requires ongoing maintenance. Despite these challenges, there are many opportunities to further improve food production lines by adding more customization options, increasing energy efficiency, and expanding their capabilities to cater to a wider range of customer needs.

5 Methodology

After reviewing relevant studies, market research, and user feedback, it became clear that the burger production line is a viable and relevant project. This system aligns with modern automation trends and provides significant benefits to both businesses and customers by offering fast and customized meal preparation with minimal human intervention. The project aims to streamline the process of preparing burgers, ensuring efficiency, precision, and an enhanced user experience.

This chapter will cover the system architecture, hardware components and devices, software and libraries, implementation details, final design of the project, and integration with the mobile application.

5.1 System Architecture

In this section, we are going to delve deep into how we designed the architecture of the project. We designed a product line with a length of 2 meters, width 14.5 cm. At first we designed it without the top components so we can work easily, then we added the upper parts that involve the ingredients and connected it with the belt part. And finally added the oven part to the structure.

- **This was the initial design:**



Figure 5.1 Initial Design

- **Ingredients structure:**

We use corn flake Cereals Dispensers to the ingredients(Lettuce, onion, tomato and cheese).



Figure 5.2 Bread Structure



Figure 5.3 Toppings Structure

- **Syringes:**

We made 2 Syringes with 3D printing to move them and put in the ketchup and mayonnaise.

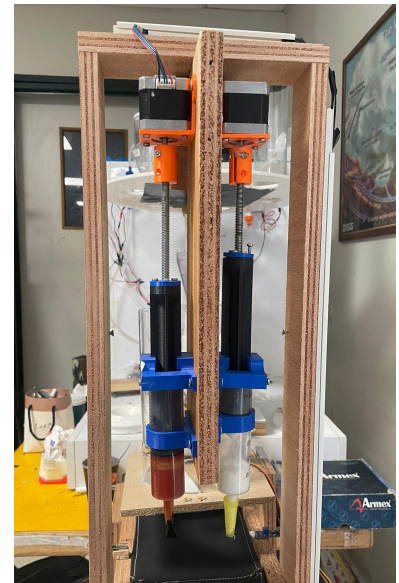


Figure 5.4 Syringes Structure

- **Oven structure:**

We used 2 heaters, one on the top and the other on the bottom, to cook the meat efficiently on both sides.



Figure 5.5 Oven structure

- **Burger arm:**

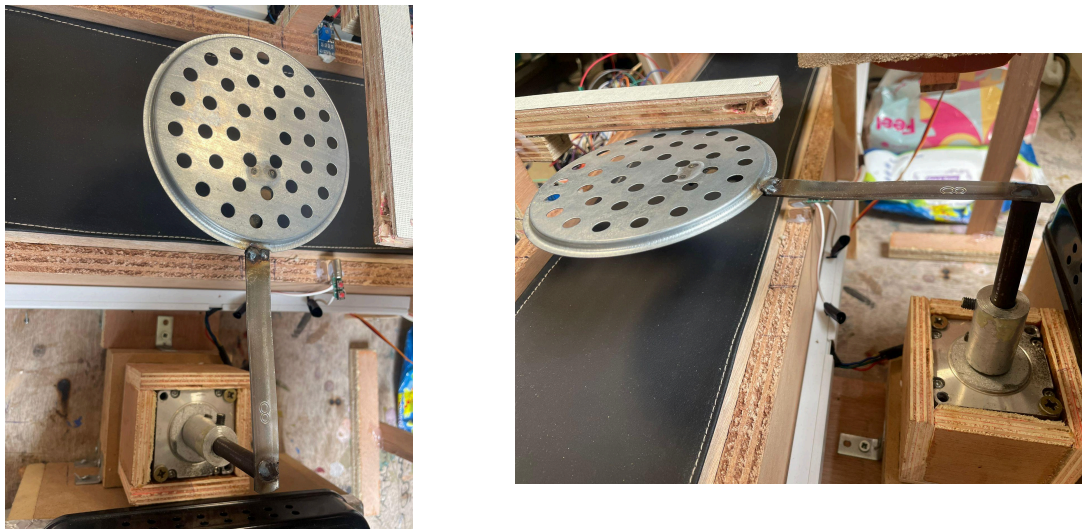


Figure 5.6 Burger arm

5.2 Hardware Components

- **Arduino Mega 2560:**

The Arduino Mega, based on the Atmega2560 microcontroller, was chosen for our project due to its extensive functionality compared to the standard Arduino board. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UART ports, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. Given the complexity of our project and the numerous hardware components involved, the Arduino Mega served as our central processor. Because it connects all the parts of the project together, and runs the code we wrote using C++ we developed. [\[4\]](#)

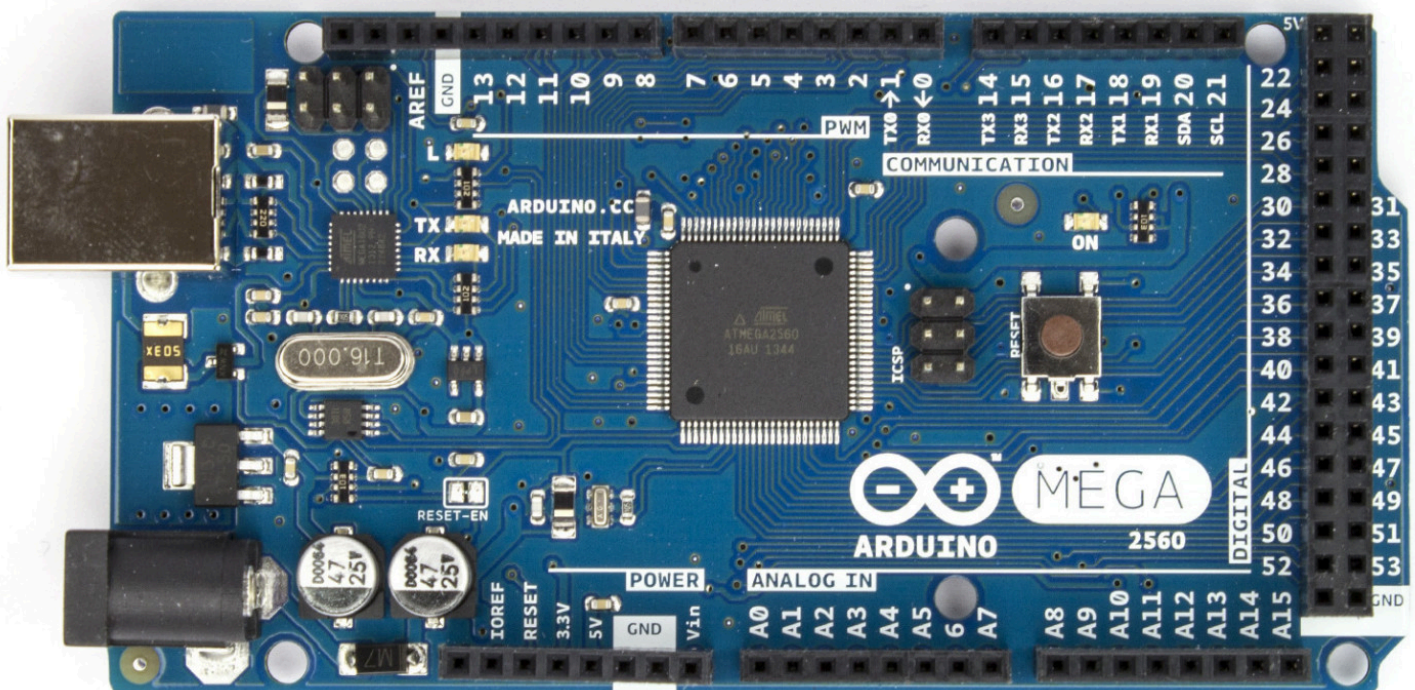


Figure 5.7 Arduino Mega 2560

- **ESP-WROOM-32:**

The ESP-WROOM-32 module supports 2.4 GHz ~ 2.5 GHz Wi-Fi (IEEE 802.11 b/g/n) and Bluetooth/BLE. It can act as a standalone MCU or provide wireless connectivity to host MCUs. In our project, it simplified IoT by connecting to the internet and enabling communication with our Blynk-based phone application. [5]

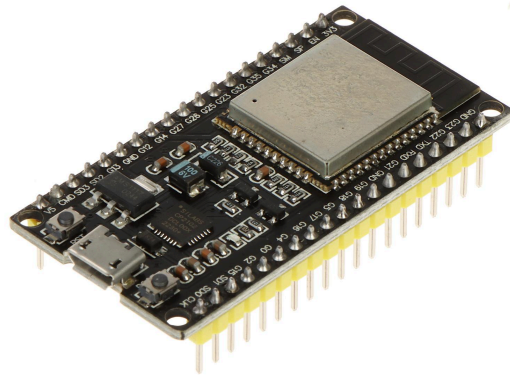


Figure 5.8 ESP-WROOM-32

- **Power Supply:**

We used a computer power supply to meet the voltage needs for our project because It can deliver the 5 volts required for various devices and the 12 volts needed for stepper motors. Furthermore, the power supply provides an adequate current output to fulfill the requirements of our project.



Figure 5.9 Power Supply

- **NEMA 17 42HS02 Stepper Motor:**

We used 6 Nema 17 stepper motors, two of them are used for the ketchup and mayonnaise sauce syringes, and the other four motors used for topping ingredients (tomatoes, lettuce, onions and cheese). [\[6\]](#)

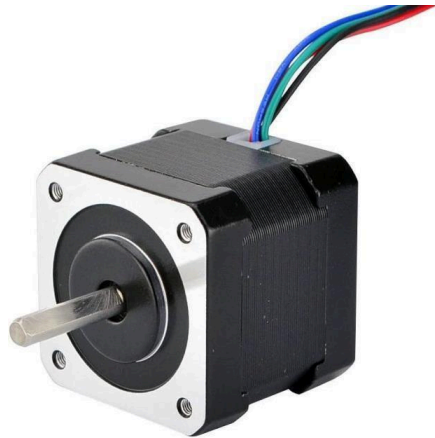


Figure 5.10 Nema 17 Stepper Motor

- **L298N motor driver:**

We used 6 H-bridges to drive the stepper motors, one for each motor. The dual H-bridges of the L298N module are capable of driving one of the stepper engine's electromagnet coils. The stepper motor's shaft can rotate precisely in small steps, either forward or backward, by activating its coils in a specific sequence. We connect the A+, A-, B+ and B- wires from the stepper motor with the L298N Driver. [\[7\]](#)

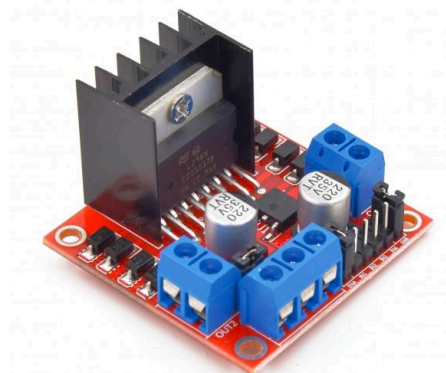


Figure 5.11 L298N motor driver

- **J-5718HB2401 Stepper motor:**

A Nema 23 stepper motor operates by energizing its coils in a precise sequence, causing the rotor to rotate in fixed steps (typically 1.8° per step). The direction and speed of rotation are controlled by the input pulse signals from a stepper motor driver. We used this motor to control the rotation of the burger arm, integrating it with a YS-DIV268N driver.

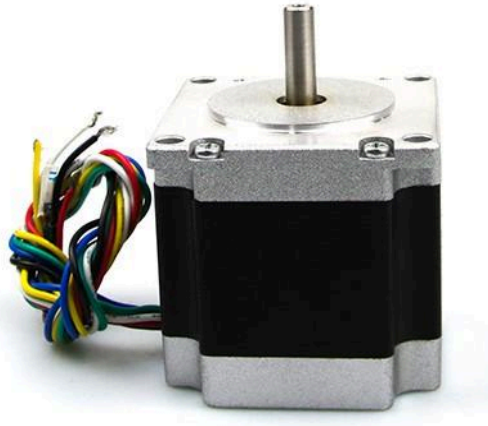


Figure 5.12 J-5718HB2401 Stepper motor

- **YS-DIV268N Driver:**

The HY DIV168N-3 driver is a versatile stepper motor driver that supports up to 5.6A of current and operates within a voltage range of 20V to 50V DC. It features microstepping capabilities, adjustable current settings, and protections against over-voltage, over-current, and short circuits, making it suitable for high-precision motion control applications. [8]



Figure 5.13 YS-DIV268N Driver

- **TowerPro TS90A Servo Motor:**

We used 5 servo motors, a lightweight, standard-sized servo known for its reliability and precision. They are low-power and cost-effective motors. They are commonly used in robotics and other projects that need precise rotational movement. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left. It is powered by a standard 4.8 to 6V power supply and receives signals from an Arduino or other microcontroller through its internal control circuitry. [\[9\]](#)



Figure 5.14 Servo Motor

- **Keypad 1x4:**

The 1×4 matrix keypad is an input device, used to provide input values in a project. It has 4 keys in total, which means it can provide 4 input values. We used it to enter values on the machine and to control it.

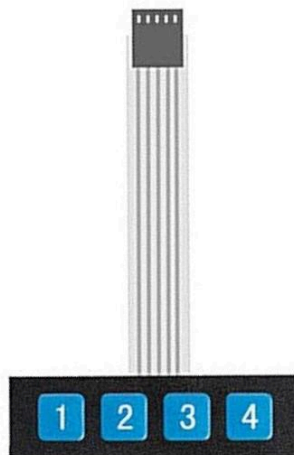


Figure 5.15 Keypad 1x4

- **RFID Reader:**

Radio Frequency Identification (RFID) is a wireless technology that consists of two main components: tags and readers. The reader uses one or more antennas to transmit radio waves and receive responses from the RFID tag. In our project, we used RFID to process customer payments by reading their card when purchasing.



Admin's tag:



Customer's card:

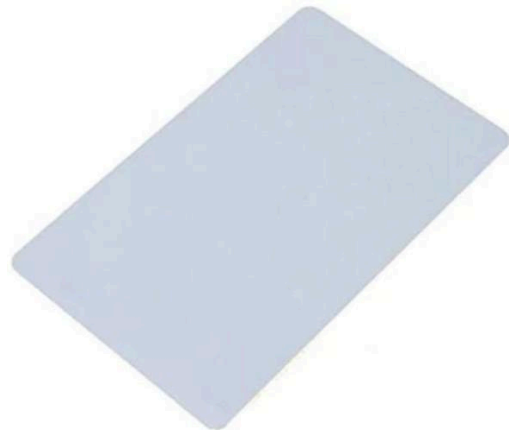


Figure 5.16 RFID Reader

- **LCD 20x4:**

The LCD 20x4 is a 20 characters by 4 lines liquid crystal display module. It's used for displaying output like text and simple graphics. We used it to display text such as asking the user what the ingredients he wants in his burger are and other instructions such as scanning the card request for the customer and printing the progress of his order that will be shown on the screen as a percentage.



Figure 5.17 LCD 20x4

- **LCD Driver Module with I2C Interface:**

A specialized electronic component designed to make it easier to integrate a liquid crystal display (LCD) into electronic systems is an LCD driver module with an I2C interface. It operates using four connections: GND, VCC, SCL, and SDA. The VCC is connected to a 5-volt power source, while the SCL and SDA pins from the driver are linked to the corresponding SCL and SDA pins on the Arduino.

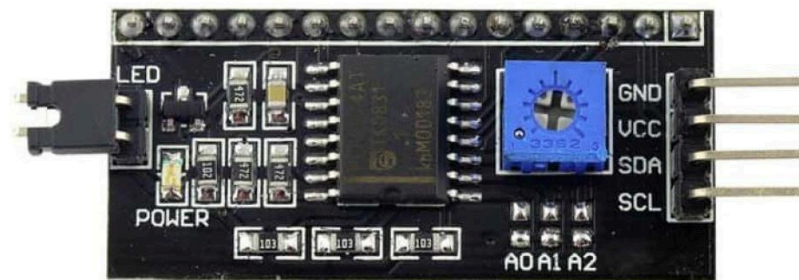


Figure 5.18 LCD Driver Module with I2C Interface

- **LDR sensor:**

A light-dependent resistor (LDR) sensor module is a tool used to measure variations in light intensity. When the amount of light striking the LDR's surface grows, its resistance lowers, and when the amount of light drops, it increases. The voltage across the LDR lowers as the light intensity rises because the LDR's resistance drops. The digital output changes states based on whether the light intensity falls or rises, depending on how the module is configured. We used this in our project to determine whether an object was reached in a particular location on several stations. We used 8 LDRs, one on each.



Figure 5.19 LDR Sensor

- **Laser:**

The laser module produces a coherent, concentrated light beam that is frequently utilized for precise targeting or sensing applications. It consists of a driver circuit for power control, a laser diode, and frequent optics for beam shaping or focusing.

We created tripwires, measured distances, and detected objects using the laser module in our project. We have constructed systems that are able to recognize when the beam is broken and initiate the intended action by merging the laser module with a light sensor.



Figure 5.20 Laser

- **Ultrasonic Sensor:**

An electronic component that uses ultrasonic sound waves (through air) to measure the distance of the target object and the reflected sound is converted into an electrical signal. We used it to track the amount of components left in the container containing the ingredients, so we thought of an ultrasonic sensor to detect the distance. The administrator receives this information, allowing them to restock the supplies as needed.

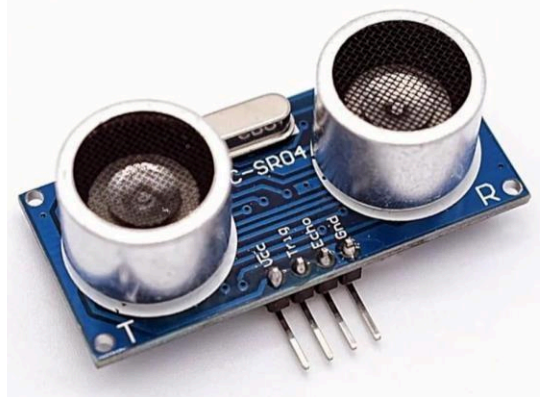


Figure 5.21 Ultrasonic Sensor

- **DC motor with gearbox:**

The DC motor was powered by a 12V power supply and controlled by a relay to regulate its direction. Its role was to drive the movement of the belt that required smooth and consistent rotation, making it an essential part of our machine system.



Figure 5.22 DC motor with gearbox

- **Single Channel Relay Module:**

A relay is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals. We used it to move the DC motor connected with the belt so that we can control it.

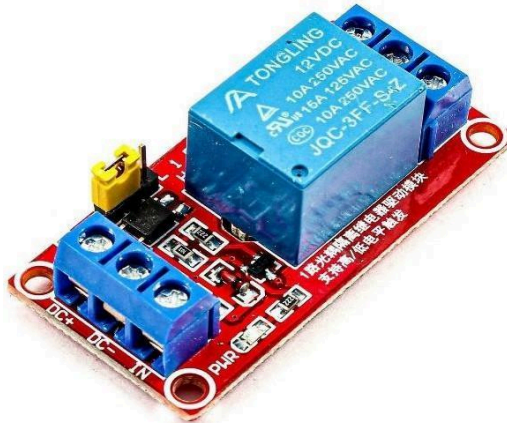


Figure 5.23 Single Channel Relay Module

- **Two Channel Relay Module:**

We used it to control the two heaters during the meat cooking stage, ensuring precise control over the cooking duration to achieve consistent and optimal results.

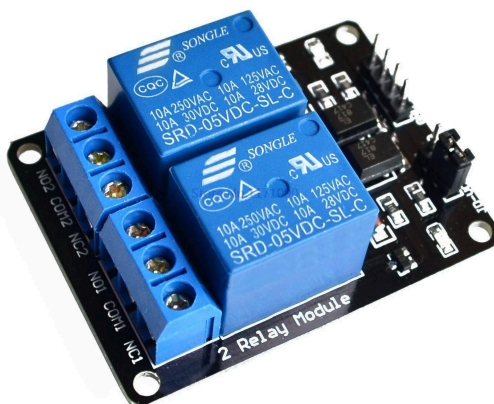


Figure 5.24 two Channel Relay Module

- **Wires:**

We used jumper wires and standard wires for the connections between components, male-to-male, female-to-female, and male-to-female.



Figure 5.25 Jumper Wires



Figure 5.26 Standard Wires

5.3 Software and Libraries

5.3.1 Libraries used in Arduino code

- **SPI.h:** A built-in library that provides functions for using the Serial Peripheral Interface (SPI) protocol. Which connects between microcontrollers and peripheral devices such as sensors, displays, and RFID modules.
- **Servo.h:** A library that provides an easy way to control servo motors using Arduino.
- **Keypad.h:** Matrix keypads can be linked using the keyboard library, we used it to read the input from the keypad buttons and read their values.
- **MFRC522.h:** A library is used to interface with MFRC522 RFID reader modules, enabling Arduino projects to read and write data to RFID tags and cards. It simplifies communication with the module over SPI, providing functions for authentication, data reading, and writing.
- **LiquidCrystal_I2C.h:** An Arduino library that interacts with LCD displays that use the I2C communication protocol. I2C communication makes connecting an LCD display to the Arduino board much easier with less pins.
- **Wire.h:** A library used in Arduino programming to facilitate communication between devices over the I²C (Inter-Integrated Circuit) protocol. It allows multiple devices, like sensors, displays, and microcontrollers, to communicate using just two wires: SDA (data line) and SCL (clock line).
- **Stepper.h:** A library in Arduino used to control stepper motors. Stepper motors are precise motors that divide a full rotation into multiple steps, allowing you to control their position, speed, and direction accurately.

5.3.2 Libraries used in ESP32 code

- **Wifi.h:** A library specifically designed for ESP32 and ESP8266 boards, enabling them to connect to Wi-Fi networks. It provides easy-to-use functions to handle Wi-Fi connections, manage access points, and work in both client and server modes.
- **BlynkSimpleEsp32.h:** A necessary component for developing IoT applications with the ESP32 microcontroller and the Blynk platform.

5.4 Implementation

5.4.1 System Design

We used the Arduino Mega as the main controller in our burger production line because it has multiple serial ports, making it ideal for connecting and managing all the components efficiently. When the machine is powered on, both the Arduino and ESP-32 start. The ESP connects to the internet and sends data about orders and preferences from the mobile app to the Arduino. The mobile app allows users to customize their burgers by selecting ingredients or placing orders, while admins can be notified about any ingredient that should be refilled.

Before starting the process, the LCD turns on and displays a welcome message to greet the user. It then guides the user through customizing their order by asking a series of questions about the ingredients they want in their burger. The user interacts with the system using the keypad to make their selections, such as choosing whether to add ketchup, mayo, lettuce, or other ingredients. Once the customization is complete, the LCD prompts the user to place their card on the RFID reader for payment. The RFID reader verifies the balance on the card, and if the payment is successful, the production line starts, creating the burger according to the user's selected preferences.

The RFID reader ensures payment is completed before the production line starts. It reads the user's card to verify sufficient balance and only proceeds with the order if payment is successful. This prevents unauthorized use of the system. The Arduino also controls the relay, which manages the conveyor belt's movement. When an LDR detects the package, the relay stops the belt to allow precise execution of the action at that station, then restarts it to move the package to the next stage.

The production line uses LDRs placed at various stages (A0 to A7) to detect the position of the burger package on the conveyor belt. A0, A2, and A7 handle essential steps in the process, such as placing the bottom bun, cooking and placing the meat, and placing the top bun, respectively. These actions are always performed regardless of customization. A1, A3, A4, A5, and A6 are responsible for customizable ingredients like ketchup, mayo, cheese, tomatoes, lettuce, and onions. The Arduino processes the user preferences whether it was selected via the keypad or received from the ESP and triggers the appropriate motors and servos based on the detected LDRs. For example, when the package reaches A1, the system checks if the user selected ketchup or mayo, and if so, it activates the corresponding syringe stepper motors to dispense the sauces.

The DC motor is carefully synchronized with the belt and LDRs to ensure smooth operation. For instance, the servo at A0 places the bottom bun, the syringes at A1 dispense sauces, and the servo at A7 places the top bun. The stepper motors handle solid ingredients like cheese, tomatoes, lettuce, and onions, moving the required number of steps to ensure accurate dispensing.

The meat oven is controlled by two relays set to specific timers to ensure the meat is fully cooked. A NEMA 23 motor moves the raw meat into the oven and retrieves it once it's cooked, placing it on the package. A servo motor is used to place the meat onto the arm controlled by the NEMA 23 motor, ensuring precise handling and alignment throughout the process, while the meat is being cooked, a timer is displayed on the LCD to show a countdown procedure so that the customer can see the completion of the process. Once the countdown is done, the LCD will display: "Your meat is done", Showing the user that the process is almost done and his burger will be presented in seconds. This coordinated system ensures the meat is cooked perfectly and placed accurately for assembly.

In addition, the LCD displays a percentage of completion at every step as the package progresses through the LDR stations. This allows the user to track the progress of their burger being prepared. At the end of the process, the LCD shows "100%" along with a message saying, "Your burger is ready. Enjoy your meal!" to notify the user that their order is complete.

The Arduino Mega's multiple serial ports allow it to handle all these components simultaneously, ensuring they work in harmony. This integration enables the production line to operate efficiently, creating a seamless and automated burger-making experience for users while providing a convenient interface through the mobile app for both customers and administrators.

5.4.2 Machine Process of Work

At the beginning, the production line is off, and all the ingredients are full and ready to be placed on the package. The package is placed on the conveyor belt and also ready to move through all of the stations to receive every ingredient.

When starting the power supply the LCD is on. It displays a welcoming message: “ Welcome to Burger Station, Press 1 to Start”. This message will wait for customers to start the process of ordering by clicking 1 on the keypad below the LCD.

For the customers (users), they can order from the production line through two methods:

- Order from machine:

Customers can order from the machine by placing their order using the keypad. They can order and customize their burger by a series of questions displayed on the LCD when they click 1, the questions include every ingredient that can be customized, whether they prefer it or not. The LCD will ask the customer: “ Do you want Ketchup?”, “Do you want Mayo”, and so on. The customer can click 1 if he prefers the ingredient, 2 if he doesn't prefer it. And these answers are shown on the LCD to guide the customer on what to press on the keypad. After the series of questions. The LCD will ask the customer to place their card in order to pay, once he approaches his card on the RFID, the sensor will detect the card and start the process according to what he chose.

- Order from mobile application:

Customers can also use a mobile application for an easy ordering method, this way is more suitable if the customer prefers a specific kind of burger, or if he prefers to use his mobile phone for ordering. On the mobile application, there are three options, a Classic burger that includes all of the ingredients, a Cheese burger, which only includes cheese, and also an option to customize your burger. For customization. all of the ingredients are shown and a switch button of “On and OFF” below them. Using the switch button, the customer can choose the ingredients he prefers and click on “Place my Order”. Before starting the machine, the application shows a message for the user to place his card after ordering. After placing the card, The machine starts normally according to what the customer ordered. In the application, notifications appear to admins in case any ingredient needs to be refilled with the name of the ingredient.

5.4.3 Flow Chart

The flow chart for both admin and customers

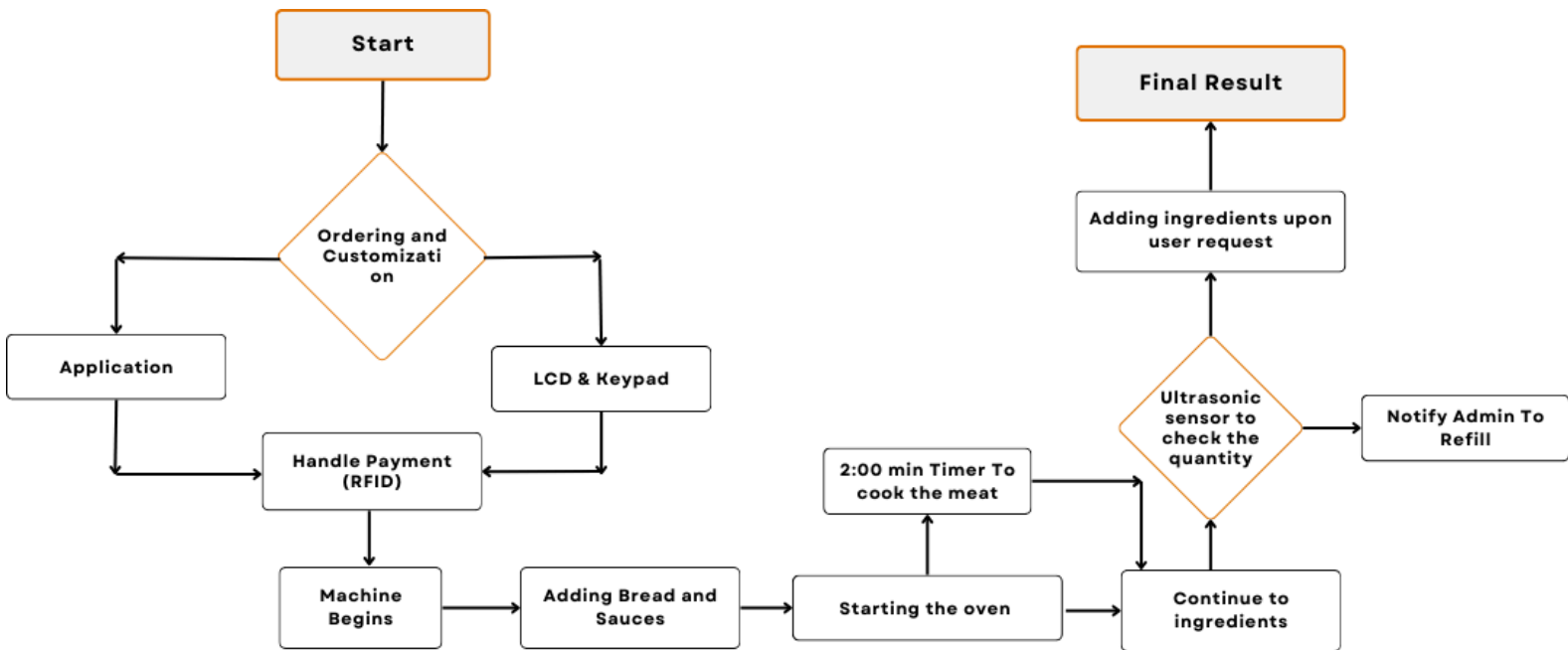


Figure 5.27 Flow Chart

5.5 Final Project

This picture shows the final project “Burger Station”, a smart production line which is used for preparing a customized burger to allow customers to have an easy and effective experience while ordering burgers at any time and any place, whatever the circumstances. It shows all the components we explained earlier in the report. Mainly, the oven with its components, the production line stations with the belt and the containers that carry all the ingredients.



Figure 5.28 Final Project



Figure 5.29 Final Project

5.6 Mobile Application

We developed a mobile application on Blynk with many features. The user can make its order and customize it by choosing the ingredients he wants.

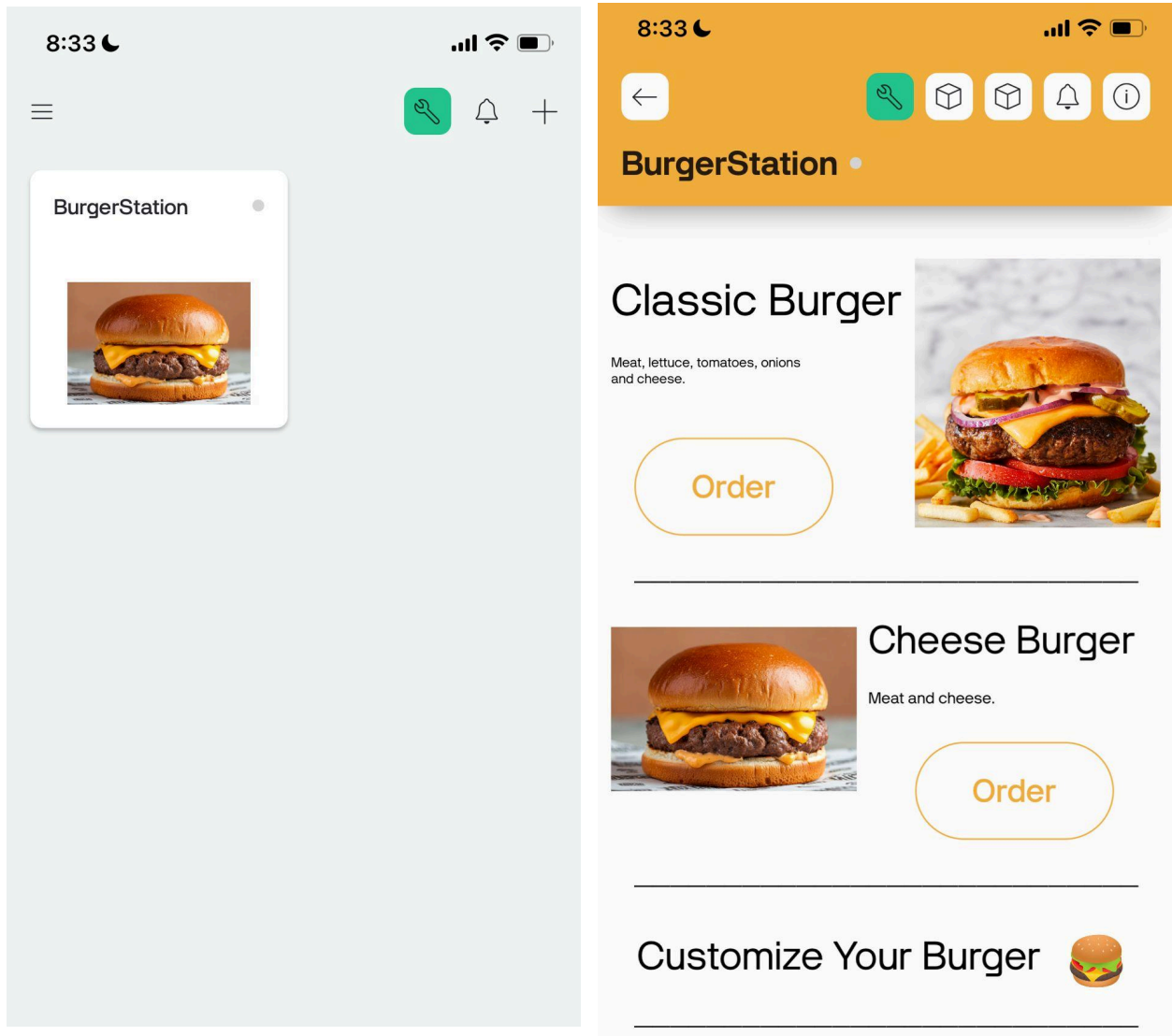


Figure 5.30 Mobile Application Dashboard

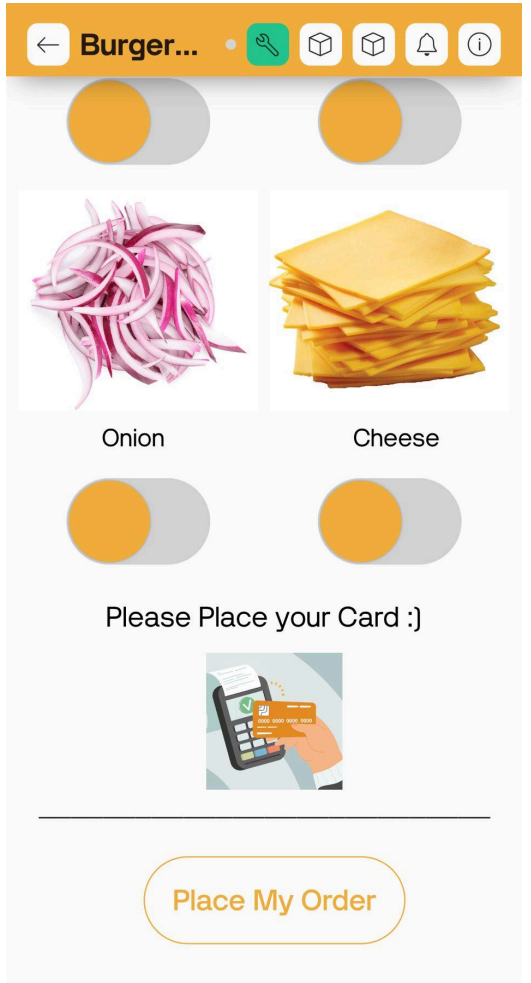
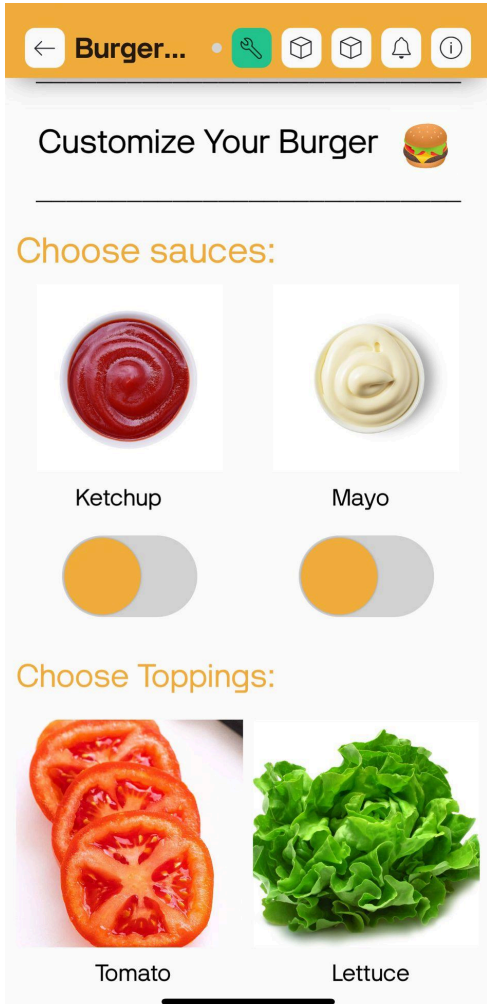


Figure 5.31 Mobile Application

6 Results and Discussion

The burger production line, designed to provide an efficient and seamless way to prepare customized burgers, using the latest technologies to deliver a unique customer experience. These technologies ensure that every step of the burger-making process is automated, accurate, and user-friendly, reducing human effort while maintaining consistency and quality.

6.1 Results

In the end, we successfully designed a smart burger production system capable of assembling customized burgers quickly, efficiently, and easy to use. The features implemented in the system provide a unique experience for the user, such as the precision of the conveyor belt and LDRs that detect the burger's position at every step, ensuring proper assembly. The use of a NEMA 23 motor and timed relays for cooking the meat guarantees that it is perfectly cooked and placed onto the burger without any manual intervention. Additionally, servo motors and stepper motors work together to dispense sauces and ingredients with high precision based on user preferences.

The ability to interact with the system through a keypad and LCD for order customization adds to the ease of use, displaying a step-by-step guide and showing a percentage of completion for better user engagement. Furthermore, the integration with a mobile application allows users to place orders remotely with cashless payment, reducing transaction times and improving accessibility. Security is also prioritized; the RFID reader ensures payment verification and admin access, protecting the system and its resources. These features combine to create a fast, reliable, and user-friendly production process, where customers can enjoy a personalized burger without unnecessary delays.

6.2 Discussion

The development of our burger production line shows how automation can make food preparation faster, easier, and more reliable. The system uses a conveyor belt and LDRs to track the burger package, stopping at each station to add ingredients like ketchup, mayo, lettuce, and cheese based on the customer's order. We also added a mobile app for placing orders and cashless payments, making it more convenient for users. The system displays progress on the LCD screen during the process, which keeps the customer informed and engaged.

While building the production line, we faced some challenges. One of the biggest was learning how to program and connect all the components, like the Arduino, ESP, motors, and sensors. Even though we had some background knowledge, we had to do more research and testing to make sure everything worked correctly. Another challenge was getting the parts we needed, which were sometimes expensive and hard to find. We also had a limited time to design and build the project, so we had to plan carefully and work efficiently to complete it.

This project also helped us think about how to make the system better in the future. For example, we could add more ingredient stations or new features, like a digital display for showing ads or extra order options. The system is designed to be flexible so it can grow and improve over time.

In the end, the burger production line is a practical and reliable solution for making burgers quickly and exactly how the customer wants them. It combines automation and ease of use to give customers a better experience while keeping the process efficient and simple.

7 Conclusion and Recommendation

7.1 Conclusion

In conclusion, our burger production line successfully demonstrates an innovative approach to automating the preparation of customized burgers. Designed for use in busy environments such as fast-food restaurants and public events, the system uses advanced automation techniques, including conveyor belts, stepper motors, servo motors, relays, and LDRs, to ensure precise and efficient preparation. By integrating modern technologies such as RFID for secure payments, mobile app connectivity for remote customization, and LCD displays for user engagement, the system provides a user-friendly experience. The design also prioritizes scalability and adaptability, allowing the system to be easily modified to meet changing demands or accommodate additional features in the future.

This project addresses the need for faster and more convenient food preparation methods while maintaining high quality and personalization. It serves as a solid foundation for future advancements in automated food production systems, making the process more efficient and accessible to customers.

7.2 Recommendation

Looking at the current progress of our burger production line, there are a few ways we can improve it in the future. Adding sensors or IoT technology to monitor ingredient levels in real-time would help manage inventory better and reduce downtime for refills. We could also offer more options for customization, like different sauces or toppings, and add more ingredient stations to make the system more flexible and improve customer satisfaction.

Making the system more energy-efficient by using eco-friendly parts, even if they cost a little more, would make it more sustainable in the long run.

We can also place the system in busy areas like food courts, events, and festivals, and use targeted marketing to highlight the convenience and customization options. This would help the system reach more people and become even more useful. These changes would improve the system's performance, customer experience, and market appeal, making it more successful and adaptable over time.

7.3 Future Work

There are several areas of future work for enhancing Burger Station. In order to improve our production line, we can:

- Expanding Ingredient Options: Add more ingredient stations for items like pickles and jalapeños to increase variety.
- Adding More Sauces: Include additional sauce dispensers for options like barbecue and mustard.
- Different Burger Types: Offer different burger types, such as mushroom, barbecue, or vegetarian options.
- Ingredient Cooling System: Install a cooling system to keep perishable ingredients, like meat and lettuce, fresh.
- Automated Cleaning System: Add an automated cleaning mechanism for dispensing areas and the conveyor belt.

8 References

- [1] Arduino IDE Documentation: <https://docs.arduino.cc/>.
- [2] Blynk Documentation: <https://docs.blynk.io/en>.
- [3] Global In-Plant Logistics Market: <https://www.kbvresearch.com/in-plant-logistics-market/>.
- [4] Arduino Mega Datasheet:
<https://docs.arduino.cc/resources/datasheets/A000067-datasheet.pdf>.
- [5] ESP-WROOM-32 Datasheet:
https://www.mouser.com/datasheet/2/891/esp-wroom-32_datasheet_en-1223836.pdf?srltid=AfmBOopbnjBTg-2dl2OJsrYaeRa_13jKsEu3HXNUDMdVgVJ1wSIQlaF
- [6] Stepper Motor Datasheet:
<https://pages.pbcllinear.com/rs/909-BFY-775/images/Data-Sheet-Stepper-Motor-Support.pdf>
- [7] L298N Motor Driver Datasheet:
<https://www.handson tec.com/dataspecs/L298N%20Motor%20Driver.pdf>
- [8] HY-DIV268N-5A Datasheet:
<https://www.sigmaelectronics.net/wp-content/uploads/2018/11/div268n-5a-datasheet.pdf>.
- [9] Servo Motor Datasheet:
http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf