



An-Najah National University

Faculty of Engineering and Information Technology

Computer Engineering Department

Hardware Graduation Project Report

OLISCAN

Automated Olive Sorting System

Students:

Abeer Rawajbeh

Mais Dwaikat

Supervisor:

Dr. Hikmat Darawsheh

Dedication

This project is dedicated to our respected families, who have been our first and greatest source of support throughout all stages of our academic journey. We sincerely appreciate their endless patience, continuous encouragement, and unwavering belief in us, which have been the true motivation behind our perseverance and determination despite all challenges.

We also dedicate this work to everyone who offered us support and assistance and had a positive impact on the completion and success of this project.

Acknowledgment

We would like to express our sincere gratitude to everyone who contributed to the successful completion of this graduation project. Special thanks are extended to our project supervisor, Dr. Hikmat Darawsheh, for his continuous guidance, patience, and valuable feedback throughout the project. His support and encouragement played an important role in achieving the objectives of this work. We also acknowledge the support of An-Najah National University, the Faculty of Engineering and Information Technology, and the Computer Engineering Department for providing the academic environment and resources necessary for completing this project. Finally, we would like to thank everyone who supported and encouraged us during the different stages of this project.

Disclaimer

This report was written by the student(s) at the Computer Engineering Department, Faculty of Engineering and Information Technology, 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 student(s). 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.

Contents

- Dedication** **1**

- Acknowledgment** **2**

- Disclaimer** **3**

- List of Figures** **7**

- List of Tables** **8**

- Abstract** **9**

- 1 Introduction** **10**
 - 1.1 Problem Statement 11
 - 1.2 Objectives 11
 - 1.2.1 Specific Objectives 12
 - 1.3 Scope of Work 12
 - 1.4 Significance of the Project 13
 - 1.5 Organization of the Report 13

- 2 Theoretical Background and Previous Work** **14**
 - 2.1 Theoretical Background 14
 - 2.1.1 Size Estimation 15
 - 2.2 Previous Work 15

- 3 Methodology** **16**

3.1	System Overview	16
3.2	System Workflow (Operational Sequence)	18
3.3	Preprocessing Stage (Leaf Removal and Entry Gates)	19
3.4	Feeding Control Stage (Servo Feeding Gate + DC Anti-Clog Arm)	20
3.5	Mechanical Transport Unit (Moving Cart Mechanism)	22
3.6	Inspection Station (Fixed Lighting + Camera on Cart)	23
3.7	Sorting Gate System (Servo Gate Under Cart)	25
3.8	Control and Communication System (Raspberry Pi – ESP32 – Arduino Integration)	27
3.9	Monitoring System (Bin Fullness + Weight Measurement)	30
3.9.1	Bin Fullness Detection (Ultrasonic Sensors)	30
3.9.2	Weight Measurement (Load Cells + Amplifiers)	31
3.10	User Interface and Interaction System	32
3.10.1	Main Display	32
3.10.2	Bin Displays	33
3.10.3	RFID Authentication	34
3.10.4	Keypad Input Module	35
3.10.5	Mobile Application	35
3.11	Power Supply and Electrical Design	36
3.11.1	12V Power Supply	37
3.11.2	5V Power Supply	38
3.11.3	3.3V Modules and Regulation	38
3.11.4	Relays and Electrical Isolation	38
3.11.5	DC–DC Buck Converter (Voltage Regulation)	39
4	Results and Analysis	40
4.1	Overview of Testing and Evaluation	40
4.2	Olive Classification Accuracy	40
4.3	System Response Time and Sorting Speed	41
4.4	Communication Reliability (Raspberry Pi ↔ ESP32)	42

4.5	Sensor Monitoring Results (Arduino Unit)	42
4.6	Summary of Results	43
5	Discussion	44
5.1	Discussion of Achieved Objectives	44
5.2	Strengths of the Proposed System	44
5.3	Limitations and Challenges	45
5.4	Suggested Improvements and Future Enhancements	45
6	Conclusion and Recommendations	46
6.1	Conclusion	46
6.2	Recommendations and Future Work	46
	References	47

List of Figures

- 3.1 Main control units used in OLI SCAN: Raspberry Pi for vision processing and ESP32 for actuation and wireless communication. 16
- 3.2 OLISCAN Wi-Fi router used to provide a stable local network for Raspberry Pi and ESP32 communication. 17
- 3.3 OLI SCAN final prototype overview. 19
- 3.4 Leaf removal and preprocessing stage. 20
- 3.5 DC motor used to drive the anti-clogging arm above the feeding gate. 21
- 3.6 Fan and DC motor mechanism for preprocessing and anti-jam operation. 22
- 3.7 Fixed LED lighting at the inspection station. 24
- 3.8 USB camera mounted inside the moving cart. 25
- 3.9 Servo motor used to control the feeding and sorting gate mechanism. 26
- 3.10 Servo motor used for the sorting gate mechanism. 27
- 3.11 OLISCAN network diagram showing Wi-Fi communication between Raspberry Pi and ESP32 through the OLISCAN router, and serial (RX/TX) connection between ESP32 and Arduino Mega. 28
- 3.12 Ultrasonic sensors used for bin fullness detection. 31
- 3.13 Weight measurement components: a digital scale setup and load cell sensors used for monitoring bin weight. 32
- 3.14 Main control panel display with keypad and RFID module. 33
- 3.15 Displays above bins showing counters and status. 34
- 3.16 RFID module used for authentication and access control. 35
- 3.17 Power supply and electrical distribution units. 37
- 3.18 Relay module used for electrical isolation and control of loads. 39

List of Tables

3.1	Main Hardware Components of OLI SCAN (Estimated)	17
3.2	Olive Classes and Output Bin Mapping (Estimated)	18
3.3	Computer Vision Calibration Parameters (Estimated)	23
3.4	Communication Links Between Controllers (Estimated)	29
3.5	Roles of Raspberry Pi, ESP32, and Arduino Mega (Estimated)	29
3.6	Interfaces Used in the OLI SCAN Prototype (Estimated)	29
3.7	System Operating States (Estimated)	36
4.1	Sorting Accuracy Results (Estimated)	41
4.2	System Timing Performance (Estimated)	41
4.3	Prototype Testing Checklist (Estimated)	42

Abstract

The olive industry is one of the most vital agricultural sectors, and maintaining the quality of olives before processing is essential to ensuring the production of high-quality olive oil. Traditional sorting methods rely mostly on manual labor, are time-consuming and prone to human error, often resulting in quality disparities. Therefore, this project aims to design and implement an automated olive sorting system that improves the accuracy, speed and efficiency of the classification process.

The importance of this project lies in its contribution to raising the quality of olive products, reducing human effort, and reducing loss. The system will sort olives in two main stages: first by size, then by color, with the ability to automatically identify and remove damaged olives and leaves.

The main objectives of the project are to develop an intelligent, reliable, and affordable machine capable of accurately classifying olives, improving production efficiency, and ensuring the maintenance of consistent quality standards.

The working methodology is based on building a physical system (hardware) integrated with sensors and camera modules. Sensors measure the size of each olive, while image processing techniques are used through the camera to detect color differences and identify damaged or unwanted materials. The system control logic will be implemented using a microcontroller to automate the sorting and removal process.

Although similar sorting systems exist for fruits and vegetables, the number of specialized olive sorting systems is small, especially in local markets. Hence, this project aims to bridge this gap by providing an automated and economical olive sorting solution that meets regional production needs.

Chapter 1

Introduction

The olive sector is regarded as one of the most significant sectors in agriculture in Palestine and in the Mediterranean region as a whole. Additionally, olive trees are not only considered a source of food but also a symbol of identity that has long been attributed to the people. Every olive season is considered a great occasion in which the members of a given family come together to harvest olives in order to get essential items such as olive oil and table olives in a bid to sustain the economy.

In addition, it should be noted that the significance of olives does not just lie within their local importance but that they possess vast applications within food industries, medical drugs, cosmetics, and even within medical applications. Hence, with this vast value being contributed by this agricultural product, it has also become vital that this industry be enhanced by adopting modern technological applications.

Olive oil production has different stages, with sorting being one of the most important processes before actual processing. The quality of olives would directly relate to the final quality of olive oil. The presence of bruised olives, foreign materials like leaves, or an irregular size and color would affect the taste, smell, and look of the produced oil. Therefore, ensuring accuracy in sorting is very important, where high-quality standards need to be fulfilled and loss in production should be at a minimum.

In conventional olive processing plants, sorting can be done manually. This process may be simple, but it consumes a lot of manual effort and time and tends to be operator-dependent. In manual sorting, some inaccuracies may arise owing to operator fatigue, operator inconsistencies, and a larger number of items being processed in a shorter period of time. This can result in quality fluctuation and increased waste levels, mostly during periods of high harvesting demand.

In order to overcome these shortcomings, the proposed project, OLI SCAN, intends to design and develop an automated system for the sorting of olives. The sorting of the olives will be undertaken in two primary stages, which include sorting the olives depending on their size,

followed by sorting depending on their color. The proposed design will also enable the detection of damaged olives as well as the removal of any leaves present in the olives.

Through the development of OLI SCAN, this project aims to assist olive producers by optimizing their efficiency, alleviating manual work, and ensuring quality, with a long-term goal of delivering a feasible and cheap automated sorting device that can satisfy regional demand and upgrade olive farming technology.

1.1 Problem Statement

The olive sorting process is an extremely vital stage in the olive oil manufacturing process, as the quality of the olives has a great impact on the quality of the olive oil produced. The olive sorting process is carried out manually in many local olive oil manufacturing plants, with much dependence on human personnel for sorting olives according to size, color, and the elimination of foreign particles like leaves, as well as sorting out rotten olives. The major drawback with this method is that it is time-consuming, laborious, and inefficient, especially during periods of high olive harvesting demands.

The manual sorting process can sometimes have inconsistencies. The precision of this sorting procedure can sometimes depend on the expertise of the human resources used for this task. Due to this, defective olives could be packed along with the good ones, or unwanted materials could not be completely eliminated. Such problems could sometimes result in issues with the consistency of the quality of the olives being produced.

Consequently, there is a great need to implement a computerized and efficient olive sorting device for easily and effectively categorizing olives. The key problem addressed in this project is that there is no inexpensive specialized olive sorting device that can easily and effectively categorize olives according to their sizes and colors and also distinguish damaged olives and leaves along with other unwanted items.

1.2 Objectives

The OLI SCAN project will design and implement an automated olive sorting system to increase the efficiency and precision of olive classification with a reduction in human effort, thus minimizing production losses. The system will be implemented on an integrated solution based on computer vision and embedded control.

1.2.1 Specific Objectives

- Design and construct a mechanical structure that is able to feed, move, and sort olives in a controlled continuous manner.
- Design a camera-based vision system on a Raspberry Pi to capture images of olives during the operation.
- To implement, using the Raspberry Pi, image processing and computer vision algorithms that will enable classification into classes of olives, either by color or by size.
- These cameras detect defective/damaged olives and unwanted materials, such as leaves, that need to be removed from the sorting line.
- To establish communication between the Raspberry Pi and the ESP32 to send sorting decisions wirelessly and reliably.
- In this case, the ESP32 Arduino will be programmed in such a way that actuators are controlled—motors/servos—and olives are guided to the correct output bins in accordance with received instructions.
- The objective of the current study was to evaluate performance in terms of accuracy, response time, sorting speed, and reliability under real operating conditions.

1.3 Scope of Work

This project scope maps to the development of a functional prototype for an automated olive sorting system. This is a system designed to inspect and sort olives automatically prior to their processing, as a way of improving accuracy in olive sorting and as a way of optimizing production efficiency. This system will combine a mechanical olive sorter and computer vision and control.

The Raspberry Pi captures images through the camera module; applies image processing and computer vision to inspect the olives; sorts them based on sorting criteria, including size and color, and defective ones in this project. The decision from the classification is further sent to ESP32, which acts as the primary control unit used for operating actuators—motors and servo mechanisms—and for WiFi/internet connectivity in the monitoring and interaction of the system.

The project also involves an Arduino board for support with complementing functionalities to improve the overall performance of the machine: driving additional sensors and measurements such as extra weight evaluation of olives, box fullness level detection, driving display units for giving real-time feedback and sorting results to the user.

This project scope will be limited to the construction and testing of a prototype suitable for small to medium-scale operation. This does not involve industrial-scale mechanical manufacturing, optimization in large production lines, or advanced chemical quality analysis of olives and olive oil. The focus shall be to provide a practical, cost-effective, reliable automated sorting system that can be applied in local and regional environments of olive production.

1.4 Significance of the Project

The importance of the OLI SCAN project is that it enables the modernization of the olive production industry through the implementation of an intelligent olive sorting system that uses automation. It is important to carry out proper olive sorting as far as possible before the olives undergo processing as a means of upholding high standards in quality.

But with the integration of computer vision with the Raspberry Pi, there is greater accuracy of inspection and classification with this proposed solution as opposed to existing methods used for sorting. Moreover, the ESP32 ensures flexibility is enhanced with the ability for wireless communication, actuation, as well as connection to the internet for monitoring interaction. Finally, the Arduino is also valuable with its ability for weight detection, full box detection, as well as real-time feedback display.

On the whole, this project will ensure that there are minimized wastes in production by identifying damaged olives and eliminating any foreign objects like leaves, among others. Additionally, this will greatly ensure optimized production processes and employee productivity. Since this system is cost-effective, olive farmers, especially small to medium-sized organizations, will greatly benefit from this project.

1.5 Organization of the Report

This report is composed of several chapters. Chapter One provides the introduction for the project, along with the problem statement, objectives, scope of work, and significance. Then, Chapter Two refers to the background and theory and reviews related previous work. Further, Chapter Three shows the proposed system methodology, which includes a description of the system structure, hardware components, and software implementation. The results are also presented, along with the performance evaluation of the developed prototype in Chapter Four. Chapter Five discusses the outcomes, limitations, and possible improvements. Lastly, the conclusions are mentioned in Chapter Six, in addition to recommendations and suggestions for future work.

Chapter 2

Theoretical Background and Previous Work

2.1 Theoretical Background

Automated sorting machines have found applications in agriculture because they can improve efficiency and minimize labor. The process of olive production also requires sorting since the condition of the olives will determine the final quality of the oil. The sort is considered vital in the olive production process since the condition of the olives will determine the final quality of the product. The factors that often decide the manner in which the olives will be sorted may include their size, color, defectiveness, as well as the presence of foreign objects like leaves.

Computer vision is a significant part of contemporary sorting technology. An inspection system based on cameras enables the vision system to “recognize” an object by identifying its visual features. In the context of the OLI SCAN project, the vision system’s Raspberry Pi performs image acquisition of olives in real-world conditions. Generally speaking, computer vision-based sorting processes involve image capture and pre-processing to filter images and compensate for illumination conditions. Segmentation to identify objects of interest (olives in this context) and decision making according to classification criteria are subsequent steps.

The analysis of colors is considered to be a significant feature in vision for olive classification. The colors of olives (green, brown, black) are normally related to the level of maturity and quality of olives. The image processing systems usually work in terms of color representations like RGB and HSV for distinguishing colors in a better manner. The analysis of HSV color represents illumination effects in terms of Value and color in terms of Hue and Saturation values, which makes it more preferable for systems working in variable lighting conditions. The system accordingly identifies the appropriately related class by analyzing the region of interest of color ranges and identifies the related bin for placing olives in it.

2.1.1 Size Estimation

There is also the possibility of size estimation using computer vision techniques. Based on measurements taken on the area or size in pixels detected in the olives, relative sizes can be estimated for classification into small and large olives. These measurements require a standardized distance for accurate pixel measurements that relate to actual sizes. Moreover, morphological operations can be necessary for eliminating noise in segmentation methods for accurate size estimation.

Embedded control systems play a crucial role in making the classification decision into a physical operation. For the proposed work, ESP32 is used as the control unit. The ESP32 takes the sorting signal from Raspberry Pi and controls the actuators, which are in charge of directing olives. ESP32 also enables WiFi as well as internet connectivity, and interaction with the system takes place via a web browser. This ensures that the classification information is provided instantaneously to the mechanical sorting system.

The Arduino board is employed for handling other system attributes that increase the reliability and usability of the system. Some of the attributes include checking the levels of fullness of the box, weighing olives if necessary, and using display screens for displaying the real-time results of the process. The auxiliary sensor abilities of the process increase the efficiency of the system and also avoid problems that may result from overflow and storage of olives.

2.2 Previous Work

Various automatic sorting systems have been created for fruits and vegetables. These systems include optical sensors, mechanical sorting, and computer vision. Many studies have demonstrated the high flexibility offered by a camera sorting system because it is able to sort objects with various shapes, colors, and defects. Commercial systems have been developed for sorting apples, tomatoes, and oranges. These systems employ image analysis methods for accurate determination of size, color, and defects.

Nevertheless, olive sorting systems are not very common and are mostly absent at local markets. The reason is primarily based on the fact that olive sorting involves the processing of small products at a faster rate and also taking into consideration the accuracy level when different products are classified separately. The colors of olives also differ slightly and demand appropriate light provision. Thus, a system such as OLI SCAN has its significance and contributes to olive sorting applications by offering low-cost easy-to-work-with Hardware Platforms (Raspberry Pi, ESP32, Arduino).

Chapter 3

Methodology

3.1 System Overview

In the following chapter of this manuscript, we describe the methodology and final implementation of how we carried out the project, known as OLI SCAN. The project is an automated system of sorting olives where efficiency is enhanced. The system is programmed to automate several functions from olive and impurity intake, followed by computer vision and finally sorting.

The OLI SCAN methodology follows an integrated architectural approach that involves mechanical designs, electronic control systems, sensors, and software systems. It is composed of numerous subsystems that perform unique functions as follows: preprocessing subsystem that involves removing leaves and controls feeding operations, computer vision subsystem, control for actuations, as well as monitoring and user interface systems.

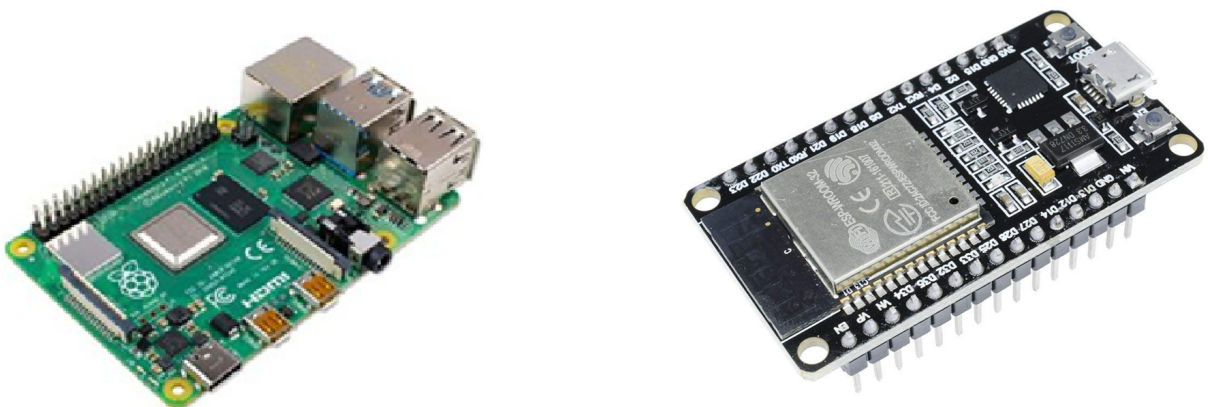


Figure 3.1: Main control units used in OLI SCAN: Raspberry Pi for vision processing and ESP32 for actuation and wireless communication.



Figure 3.2: OLISCAN Wi-Fi router used to provide a stable local network for Raspberry Pi and ESP32 communication.

Table 3.1: Main Hardware Components of OLI SCAN (Estimated)

Subsystem	Main Components	Function
Decision Unit	Raspberry Pi + USB Camera	Captures olive images and performs classification (size and color).
Actuation & Motion Control	ESP32 + Motors/Servos	Executes sorting actions and controls cart movement and gates.
Monitoring & Safety	Arduino Mega + Ultrasonic + Load Cells (HX711)	Detects bin fullness, measures weight, and triggers warnings.
User Interface	LCD/7-Segment + Keypad + RFID	Displays system status, counters, and provides access control.
Power System	12V Supply + 5V/3.3V Regulation + Buck Converter	Provides regulated voltages for controllers, sensors, and loads.

3.2 System Workflow (Operational Sequence)

The system works on the principle of sequential workflow that starts from the entry gate itself, where olives mixed with leaves enter into the system. During the preprocessing stage, the mechanical gates periodically get actuated and take the assistance of airflow developed by a fan to keep the leaves away.

After preprocessing, the olives proceed to the feeding section. This section includes a feeding gate driven by a servo that releases the olives in controlled amounts. A DC motor is mounted above the feeding gate, which has a mechanical arm to avoid congestion in olives by separating clustered ones and guiding them toward the opening for feeding.

The next process involves using a moving cart system to transport the olives to a destination. A camera and a lighting module are installed on this moving cart in order to capture images during this process. The image processing occurs on Raspberry Pi using the colors and sizes of the olives to classify them using a sorting process. The sorting process uses Raspberry Pi to transmit this message wirelessly using ESP32. Moving of a cart to a specified position using ESP32 occurs. Based on a specified weight and fullness of a bin using an Arduino module, a display and monitoring of weight using a mobile are implemented using this module.

Table 3.2: Olive Classes and Output Bin Mapping (Estimated)

Class	Bin ID	Description
Green Big	0	Green olives classified as large size.
Green Small	1	Green olives classified as small size.
Black Big	2	Black olives classified as large size.
Black Small	3	Black olives classified as small size.
Brown	4	Brown olives (maturity/defect class).



Figure 3.3: OLI SCAN final prototype overview.

3.3 Preprocessing Stage (Leaf Removal and Entry Gates)

The preprocessing section is in charge of purging the incoming olive stream from leaves and light weight foreign particles. Olives accompanied by leaves are received through the main gate. To control the incoming stream of olives and the accompanying leaves in the plant, three mechanical gates are used in the plant's entry section; the operation of the gate is periodic according to the time schedule.

At the same time, another fan works to provide the airflow that takes out the leaves. Leaves

have less weight compared to olives and can be moved by the airflow, allowing olives to be extracted based on their weights. This stage greatly enhances the precision of classification by ensuring that the inspection process takes place with the objects cleaned and few impurities. The gates and fan controls use the Arduino tool to set intervals based on the speed of the machine and the sorting sequence.

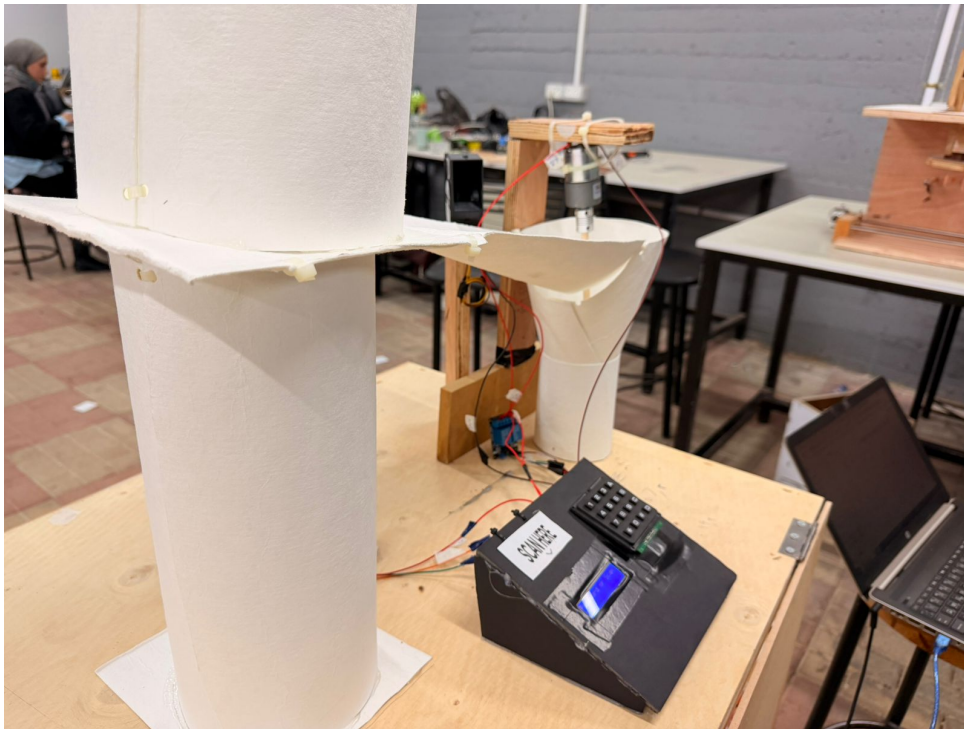


Figure 3.4: Leaf removal and preprocessing stage.

3.4 Feeding Control Stage (Servo Feeding Gate + DC Anti-Clog Arm)

Following this preprocessing operation, olives proceed to a feeding control stage. Its objective is to maintain a steady feed of olives to a cart inspection and sorting sequence. The feeding process involves two components that function in a concerted fashion in a cart inspection and sorting sequence:

Servo-driven feeding gate

A servo motor is used to control the opening and closing of a gate that leads to a feed region. For some period of time, it will allow exactly one olive to travel past it and then close again. This effectively excludes more than one olive from being present at once in the inspection zone.

DC motor with anti-clog arm (above the gate)

The DC motor is used to operate a mechanical arm which is placed on top of a feeding gate. Its role is to decongest a group of clustered olives near the entrance of the gate by separating them. It achieves this goal by pushing these olives toward the gate entrance to enable a servo feeding gate to function by releasing only one olive at a time.

The feeding timing, also called servo gate pulses, and the motion of the anti-clog arms are synchronized with the system cycle to allow continuous operation without blocking.



Figure 3.5: DC motor used to drive the anti-clogging arm above the feeding gate.

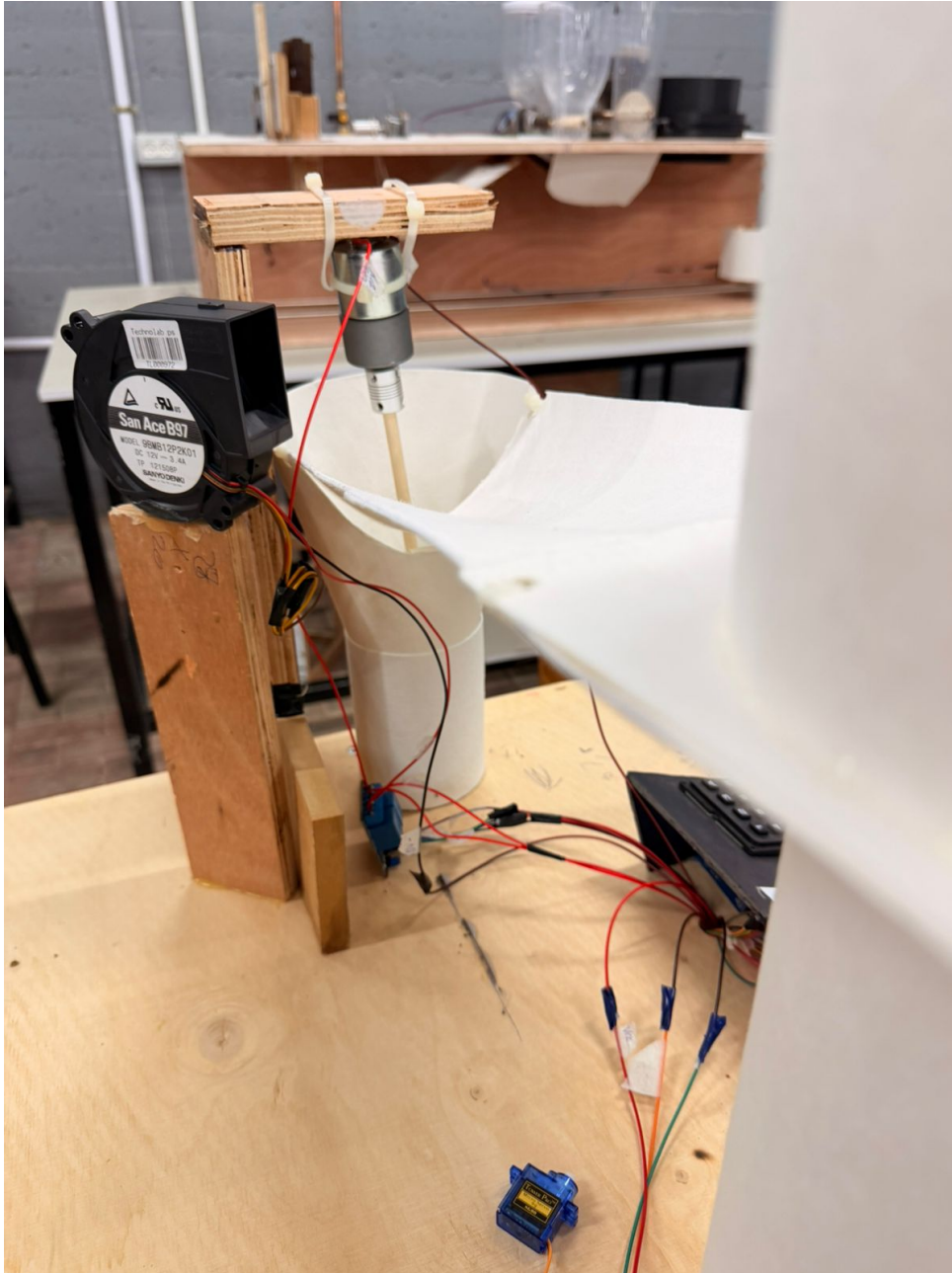


Figure 3.6: Fan and DC motor mechanism for preprocessing and anti-jam operation.

3.5 Mechanical Transport Unit (Moving Cart Mechanism)

Transport subsystem is responsible for a moving cart that carries olives along the system. It moves horizontally across the bins and it includes the camera module. The cart follows mainly two phases: (1) move to the inspection station and stops for classifying, and (2) move to the correct bin and release the olive. The movement of the cart is designed to be stable, having as little vibration as possible. Accurate positioning ensures correct alignment above bins before opening the gate.

3.6 Inspection Station (Fixed Lighting + Camera on Cart)

The dedicated inspection station is employed for the acquisition of images under a consistent lighting condition. The lighting system is rigidly positioned within the building and operated on a 12V power supply. Once the cart reaches the inspection station, it comes to rest directly below the lighting module. The camera on the cart captures images of the olive when the object of interest is still. Raspberry Pi will examine the image captured to classify the olive according to its color and dimensions, thus creating a decision for the bin.

Table 3.3: Computer Vision Calibration Parameters (Estimated)

Parameter	Value
Camera Resolution	1280×720 (approx.)
Frame Rate	25–30 FPS (approx.)
ROI Size	220×220 pixels (approx.)
Color Space	HSV
Green HSV Range	H:35–85, S:40–255, V:40–255 (approx.)
Black Detection	$V < 60$ (approx.)
Brown HSV Range	H:10–25, S:40–255, V:40–255 (approx.)
Size Threshold	200 pixels (approx.)
Filtering	Morphological operations + blur (approx.)



Figure 3.7: Fixed LED lighting at the inspection station.

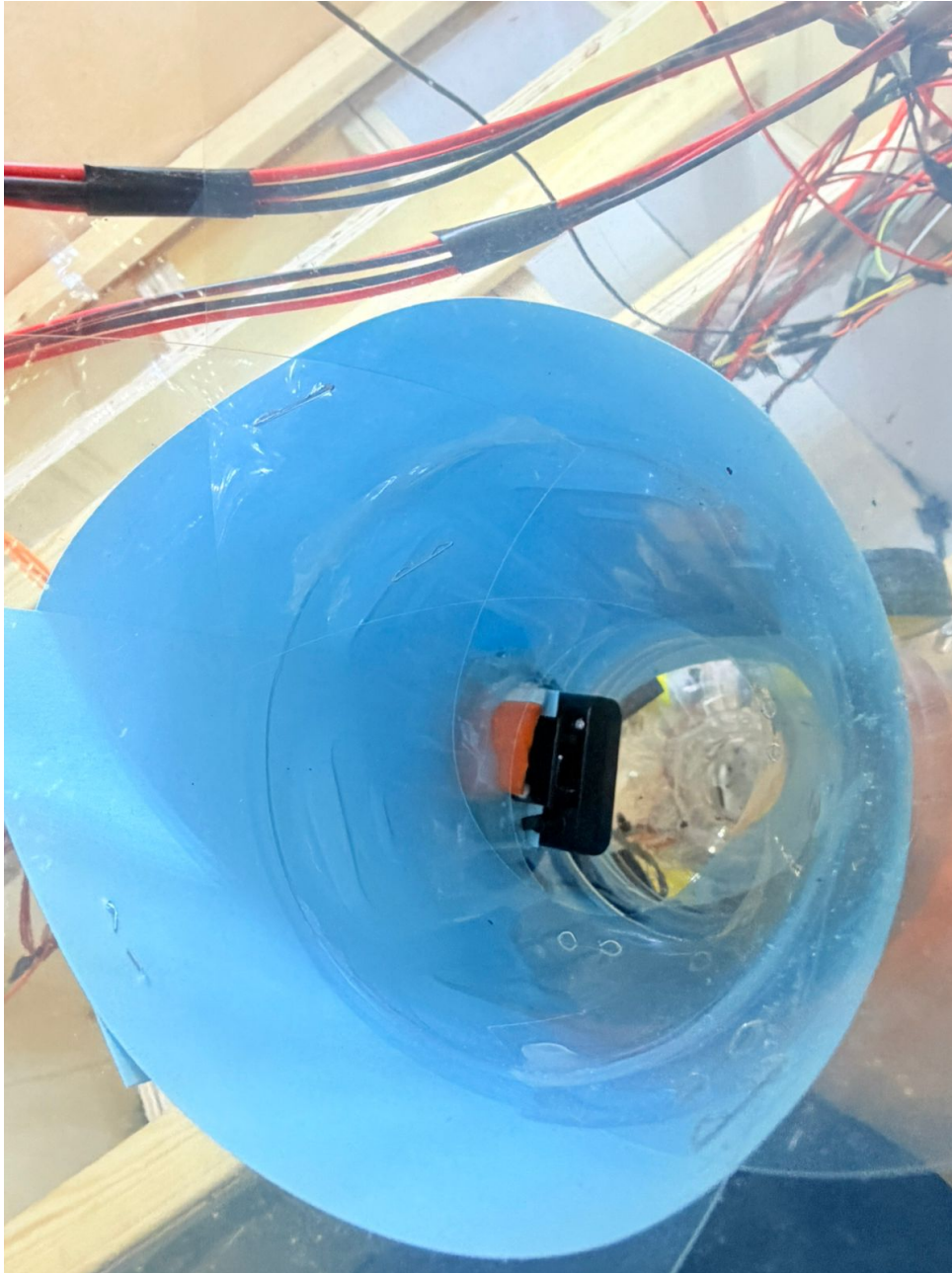


Figure 3.8: USB camera mounted inside the moving cart.

3.7 Sorting Gate System (Servo Gate Under Cart)

The final stage of the sort involves a servo gate placed underneath the cart. This servo gate then opens to deposit the olive into the desired bin, before being closed again in a repetitive motion.



Figure 3.9: Servo motor used to control the feeding and sorting gate mechanism.

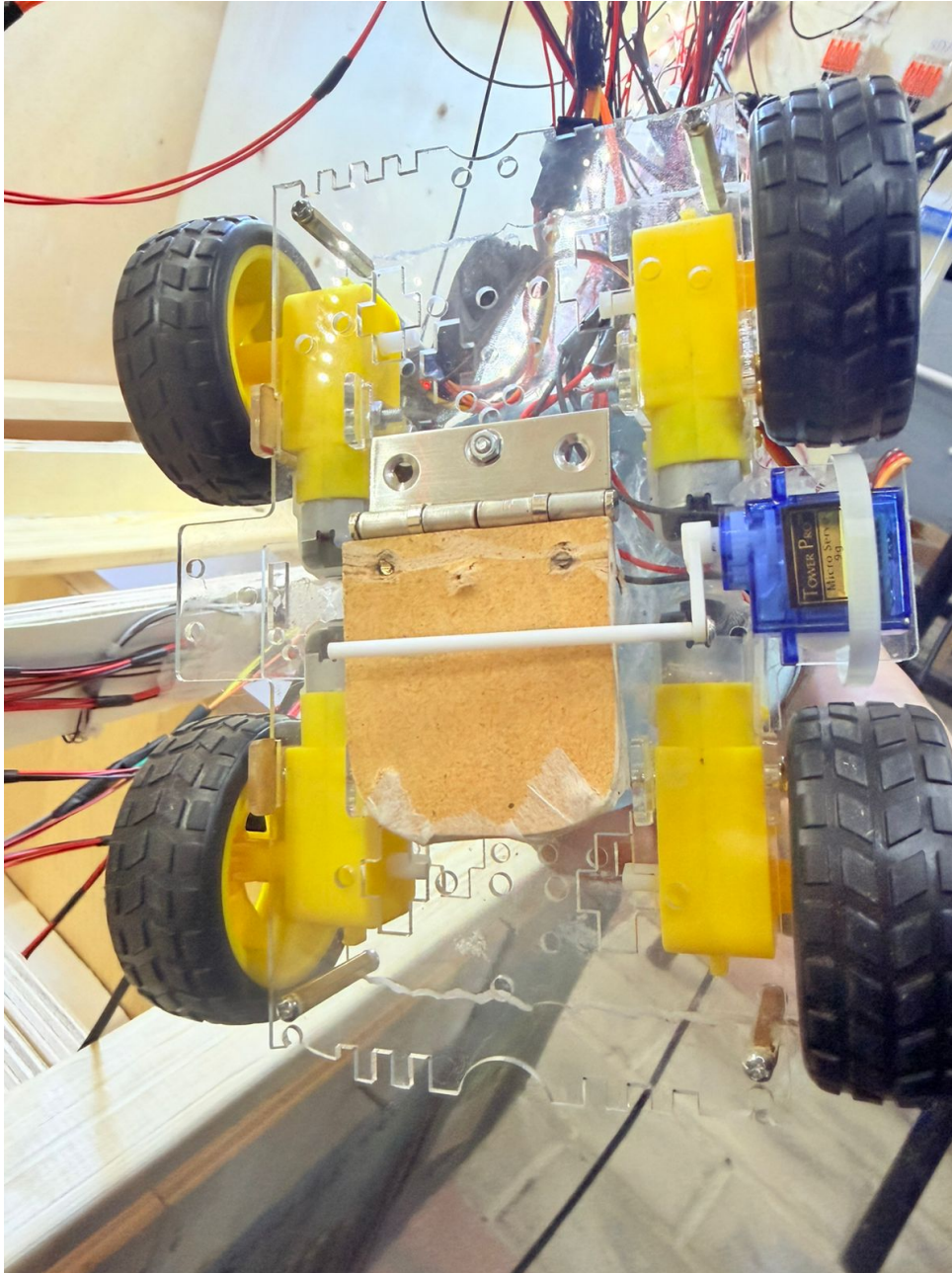


Figure 3.10: Servo motor used for the sorting gate mechanism.

3.8 Control and Communication System (Raspberry Pi – ESP32 – Arduino Integration)

The control system based on the OLI SCAN controller follows the concept of distributed control systems. It consists of three main components that process data: the Raspberry Pi, ESP32, and Arduino Mega components.

The Raspberry Pi will act as the decision-making unit. Its tasks will include the acquisition

of the olives' images as well as the processing of the corresponding images to classify the olives. Following the classification of the olives, the Raspberry Pi will send a sorting command corresponding to the chosen output bin. The command will be sent wirelessly to the ESP32 through WiFi communication.

ESP32 is also responsible for executing the sorting action. ESP32 manages all actuators involved in the movement of the cart and also manages a gate located under the cart after it receives a command from Raspberry Pi. The cart moves to an appropriate bin and drops an olive. It also enables communication services that allow for web-based communication.

Raspberry Pi and ESP32 are connected to the OLISCAN Wi-Fi router as station devices, also known as STA mode. This creates a stable network through which data exchange and system control take place. The Raspberry Pi communicates with ESP32 wirelessly through an HTTP and WebSocket protocol to transfer the results of the olive classification. This improves communication reliability and does not rely on an access point.

The functions of the Arduino Mega are the control of the auxiliary functions and interface systems. It controls the preprocessing elements via relay switching of the fan motors and DC motors. The feeding system elements to be controlled by the Arduino are the servo feeding gate and the anti-clog DC arm. Additionally, the Arduino has the functions of reading the sensors used in the system (load cell sensors and ultrasonic sensors), handling the RFID tag authentication systems and the keypads, as well as handling the display screens mounted on the system.

This distributed system design enhances the performance of the system by separating the processes of vision, actuation, and monitoring.

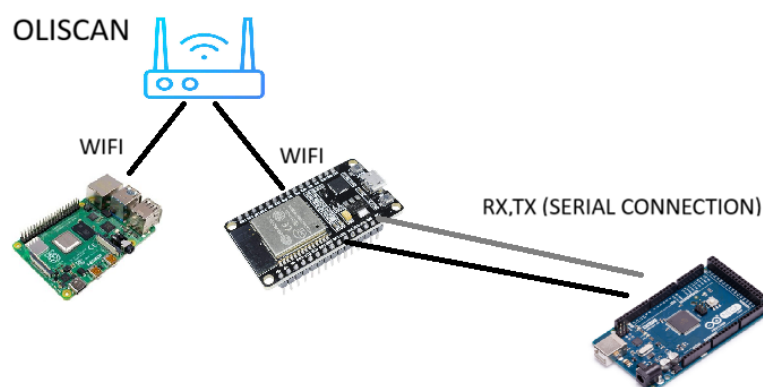


Figure 3.11: OLISCAN network diagram showing Wi-Fi communication between Raspberry Pi and ESP32 through the OLISCAN router, and serial (RX/TX) connection between ESP32 and Arduino Mega.

Table 3.4: Communication Links Between Controllers (Estimated)

From	To	Method	Purpose
Raspberry Pi	ESP32	WiFi via OLIS-CAN Router (HTTP/WebSocket)	Transmit sorting decision (target bin).
ESP32	Arduino Mega	UART (Serial)	Exchange status, counters, and enable signals (optional).
Arduino Mega	Sensors/Displays	GPIO / I2C / SPI	Monitoring (fullness/weight) and user feedback.

Table 3.5: Roles of Raspberry Pi, ESP32, and Arduino Mega (Estimated)

Controller	Main Responsibilities
Raspberry Pi	Image capture, preprocessing, olive classification (color and size), sending sorting commands.
ESP32	Receiving commands, controlling actuators (cart/gates), providing WiFi services and web interaction.
Arduino Mega	Sensor readings (fullness/weight), display updates (LCD/7-seg), RFID/keypad handling, relay control.

Table 3.6: Interfaces Used in the OLI SCAN Prototype (Estimated)

Interface	Used With	Purpose
WiFi	Raspberry Pi ↔ ESP32	Send sorting commands and status messages.
UART (Serial)	ESP32 ↔ Arduino Mega	Exchange counters/status and control signals.
I2C	LCD / Keypad modules	Simplify wiring and communication with peripherals.
SPI	RFID module / (optional sensors)	High-speed communication with RFID and modules.
GPIO/PWM	Servos / Relays / Sensors	Actuator control and reading sensor signals.

3.9 Monitoring System (Bin Fullness + Weight Measurement)

To guarantee the safe and uninterrupted functioning of the machine, the output bin of every scanning machine is given an added component of the OLI SCAN system, which is the monitoring component. These machines monitor their output bin fullness through the following primary sensing methods: bin fullness detection as well as weight measurement.

3.9.1 Bin Fullness Detection (Ultrasonic Sensors)

Each bin is installed with an ultrasonic device placed at the upper side of the bin. The device is supposed to measure the distance between its position and the surface level of the olives. As the surface is covered by more and more olives, the distance measured by the device will reduce accordingly. A threshold will define when the bin is full. Then, a warning will halt the process to avoid overflow.



Figure 3.12: Ultrasonic sensors used for bin fullness detection.

3.9.2 Weight Measurement (Load Cells + Amplifiers)

Each of the bins is further equipped with a load cell to record the accumulated weight of olives. The load cell has a low voltage level. It is then followed by an amplifier module attached to each of the load cells as a way of enhancing the digital display. The Arduino system is then used to read the information and calculate the weights.



Figure 3.13: Weight measurement components: a digital scale setup and load cell sensors used for monitoring bin weight.

This kind of monitoring system can offer both the physical security of stopping overflows from happening and the statistical information on production through the weighing of the bins.

3.10 User Interface and Interaction System

The user interface subsystem enables users to interact with OLI SCAN and achieve various outcomes such as sorting and warning. The prototype has various components to facilitate user interaction with the system.

3.10.1 Main Display

A main display has been provided on the control panel to display overall system information, including the mode and overall status of the machine, warning information, and overall counters.



Figure 3.14: Main control panel display with keypad and RFID module.

3.10.2 Bin Displays

Five additional displays are also provided above the five output bins. These displays contain specific information, like:

- the number of olives sorted into the bin,
- the weight of the bin based on the measurement of the load cell,
- fullness warning based on detection using an ultrasonic sensor.

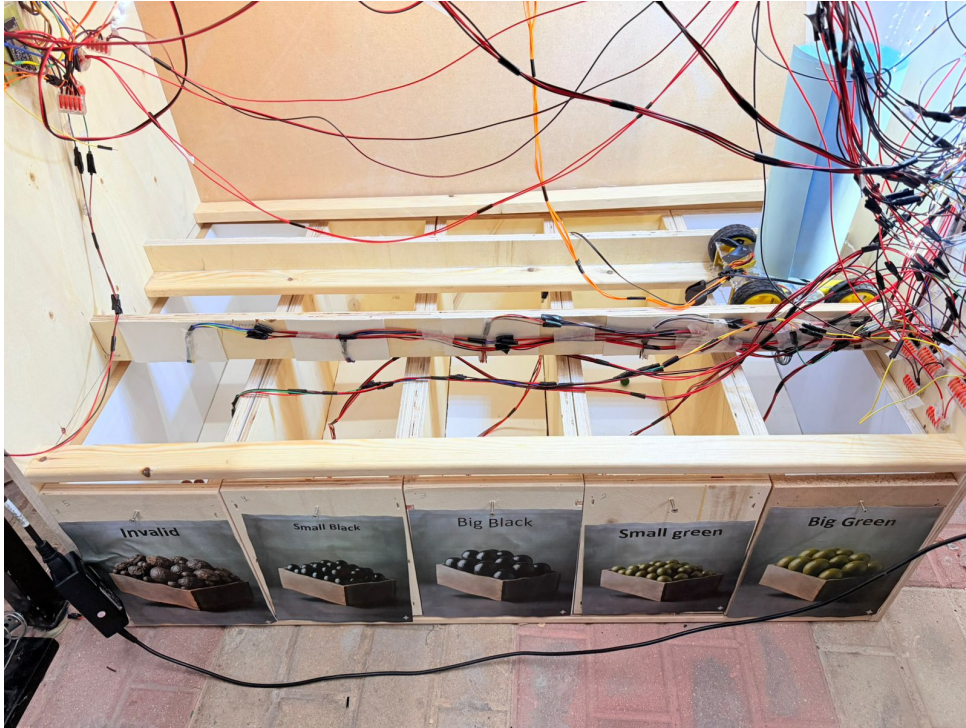


Figure 3.15: Displays above bins showing counters and status.

3.10.3 RFID Authentication

In terms of controlling the operation, the technology can be employed in identifying the users. The authorized users have the capacity to operate the machine by scanning the RFID card.

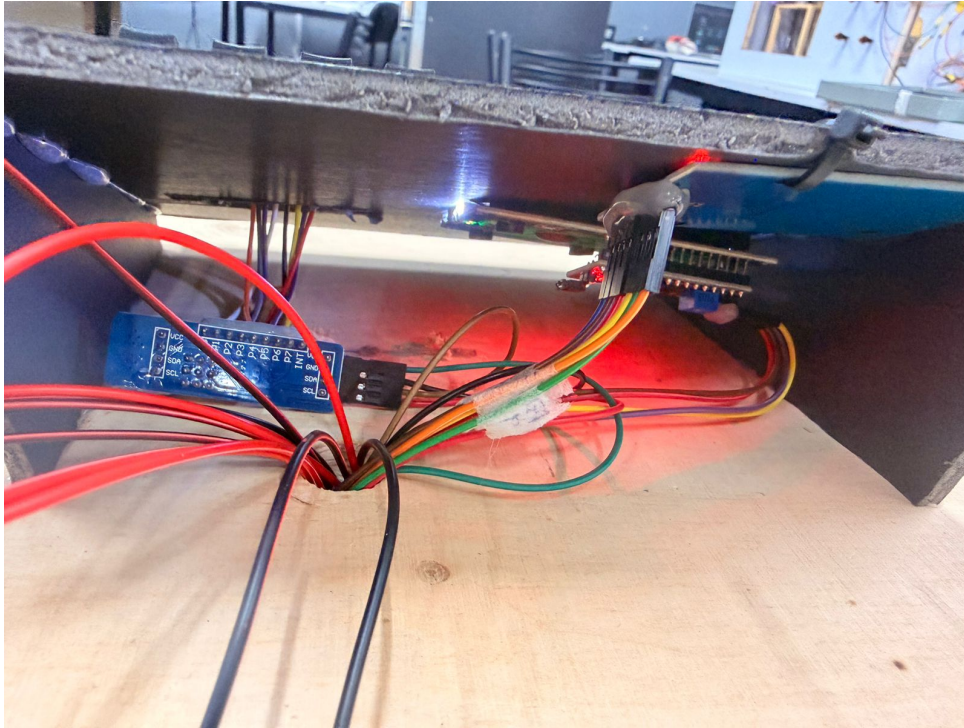


Figure 3.16: RFID module used for authentication and access control.

3.10.4 Keypad Input Module

A keypad can be used to input a password or control commands. Key functions of the keypad will be basic operations involved in setup, reset, and/or control depending on the interface.

3.10.5 Mobile Application

The system is further complemented by a mobile monitoring system. The monitoring system allows those who want to remotely observe the system and its activities through a mobile monitoring system.

Table 3.7: System Operating States (Estimated)

State	Description
Idle	System is powered, waiting for user authentication/start command.
Running	Normal operation: feeding, inspection, classification, sorting, monitoring.
Warning	A sensor triggers an alert (e.g., bin full) and the operator is notified.
Paused	Temporary stop due to safety condition or manual intervention.
Reset	Counters/state are cleared and system returns to ready mode.

3.11 Power Supply and Electrical Design

A multi-level power system is used in the OLI SCAN for the purpose of stable operation of different types of loads as well as electronic equipment. A 12V and 5V power converter are used.

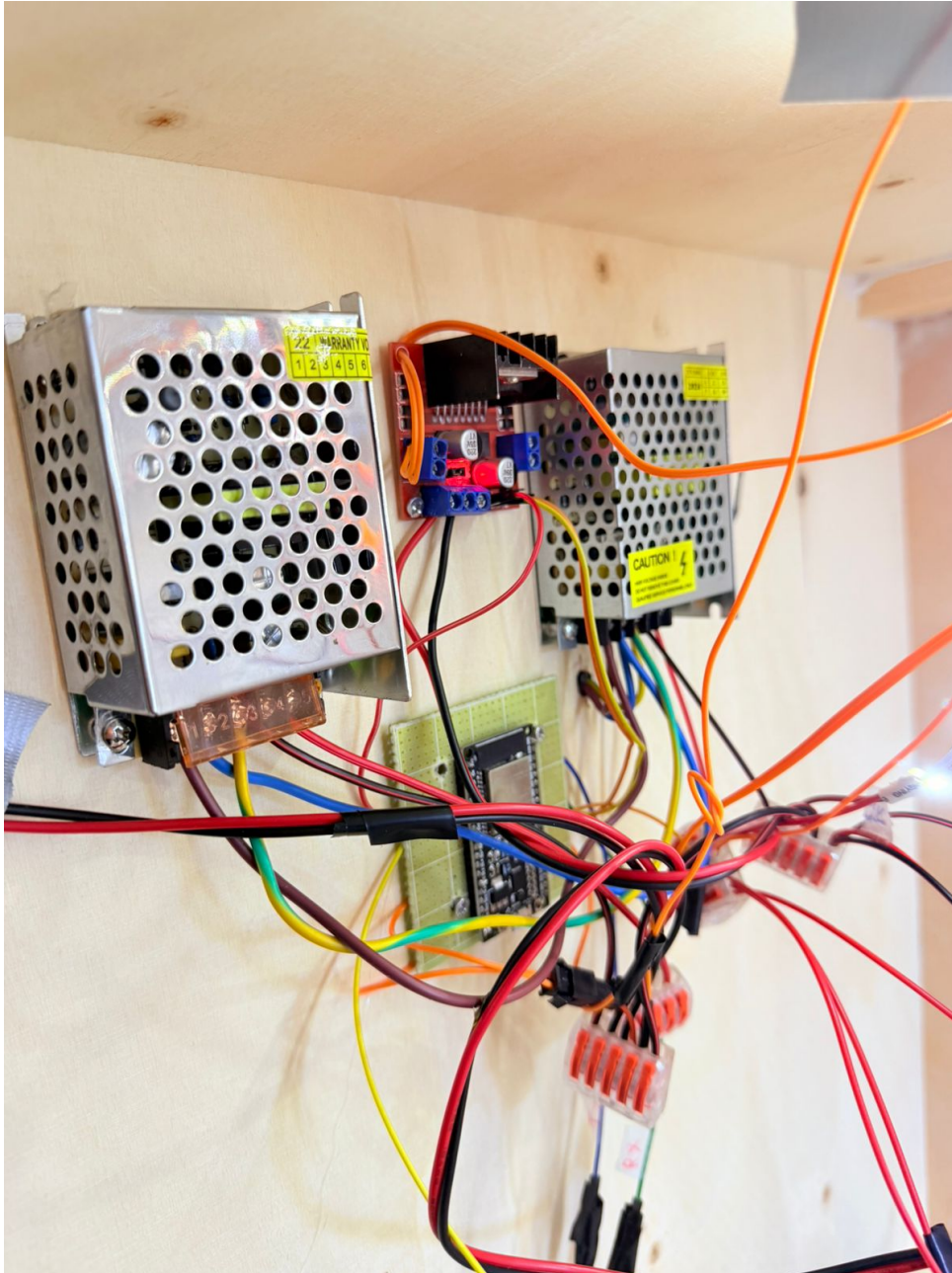


Figure 3.17: Power supply and electrical distribution units.

3.11.1 12V Power Supply

The 12V converter is used to power components with greater power ratings such as:

- the fixed lighting module of the inspection station,
- fan/blower system for leaves removal,
- DC motors for mechanical motion (when applicable).

3.11.2 5V Power Supply

The 5V converter can deliver power for control and sensor electronics such as:

- Arduino Mega board,
- sensors and amplifier modules,
- servo motors (if connected to regulated voltage supplies).

3.11.3 3.3V Modules and Regulation

Some of these modules even demand to run on 3.3V. Arduino is employed to deliver this voltage level if needed, while the idea of voltage regulation is taken to prevent damage to the components.

3.11.4 Relays and Electrical Isolation

Modules of relays are used where the control signal from the Arduino and the loads with large currents, such as the fan and the DC motors, are involved. This ensures the electric circuits are isolated from the microcontroller so that the microcontroller will not be affected in case of large currents.

Overall, the electrical system is responsible for the proper delivery of power, reducing voltage drop when the motor is running, and protecting other electronic devices.

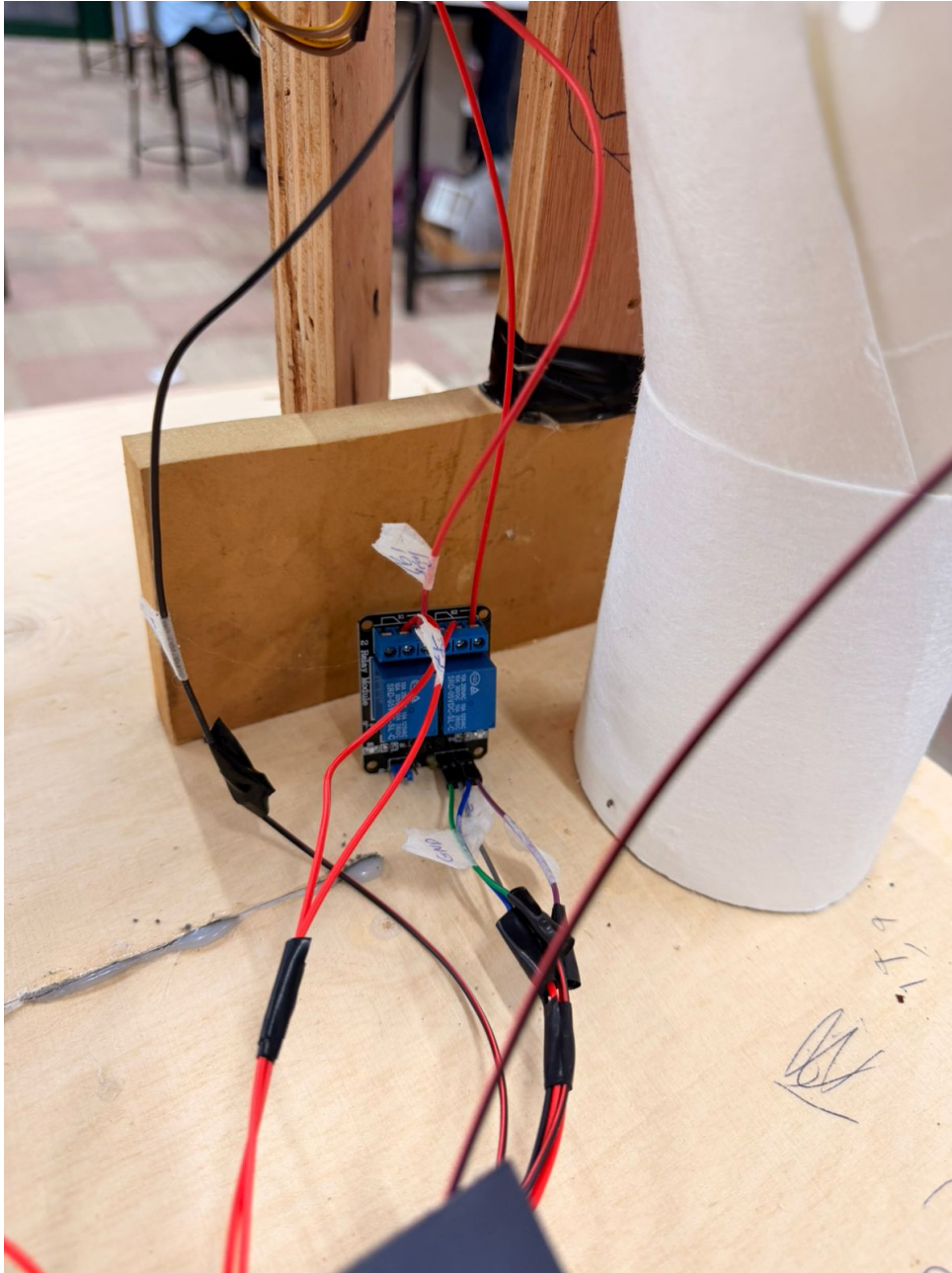


Figure 3.18: Relay module used for electrical isolation and control of loads.

3.11.5 DC–DC Buck Converter (Voltage Regulation)

There was an application of DC–DC buck converter (LM2596 Adjustable Converter) in the electrical part of this project to ensure that there were regulated low-voltage outputs for the control electronics involved. Since this project requires the use of a 12V voltage for the high-powered parts such as the lights and the motors, the buck converter was necessary to provide stabilizing low-voltage outputs such as 5V and 3.3V for the microcontrollers and sensor devices safely and reliably.

Chapter 4

Results and Analysis

4.1 Overview of Testing and Evaluation

This chapter presents the results obtained from the elaboration and tests of the OLI SCAN prototype. Verification is related to the performance of the overall system, which concerns olive inspection and classification on Raspberry Pi, actuator control, and connectivity across ESP32, with additional monitoring features developed on Arduino. Tests of the prototype were performed in real operation conditions using real olive samples with various sizes and colors in order to simulate as much as possible real production conditions.

The evaluation includes classification accuracy, system response time, sorting speed, communication reliability, and sensor performance (box fullness and weight measurements). User interface performance was also considered through the display system and mobile application monitoring.

4.2 Olive Classification Accuracy

The key functionality of OLI SCAN lies in correctly grading olives by appearance. During the testing, the Raspberry Pi camera captured the olives at the inspection zone and the computer vision algorithm ran on every frame to extract features related to color and size. Further, the classification obtained was translated into a sorting command.

It was observed that the system could differentiate among dominant olive color classes based on constant lighting conditions. Nevertheless, the accuracy in classification depended on various factors, which may include differences in lighting, presence/absence of shade, and distance between the olives and the camera.

Table 4.1: Sorting Accuracy Results (Estimated)

Class	Samples	Correct	Accuracy (%)
Green Big	50	47	94
Green Small	50	46	92
Black Big	50	48	96
Black Small	50	45	90
Brown	50	44	88
Overall	250	230	92

4.3 System Response Time and Sorting Speed

System response time is defined as the time difference between detection/classification of an olive and the mechanical sorting action for that olive. Several factors contribute to this delay, namely, camera capture of a frame, processing time taken on Raspberry Pi, delay in wireless transmission to ESP32, and movement time of the actuator.

Testing also showed that the system can continuously use the prototype with acceptable response time, enabling the system to accomplish sorting operations with minimal time wastage. The sorting speed was constrained primarily by the mechanical movement of the cart and the process of opening and closing the gate through the action of the servo motor. Optimization of mechanical and servo times can enhance speed.

Table 4.2: System Timing Performance (Estimated)

Metric	Value
Image capture + processing time (Raspberry Pi)	5–10 s (approx.)
WiFi transmission delay (Pi → ESP32)	0.05–0.2 s (approx.)
Cart movement to target bin	1.5–3.0 s (approx.)
Gate open/close action	0.3–0.7 s (approx.)
Total response time per olive	5.0–7.0 s (approx.)
Estimated throughput	15–30 olives/min (approx.)

Table 4.3: Prototype Testing Checklist (Estimated)

Test Case	Status	Notes
Leaf removal stage operation	Pass	Leaves reduced before inspection.
Single-olive feeding mechanism	Pass	Reduced clogging and improved inspection stability.
Image capture at inspection station	Pass	Fixed lighting improved consistency.
WiFi command transmission	Pass	Occasional delays depending on signal quality.
Cart movement to bins	Pass	Positioning stable during normal operation.
Gate dropping into correct bin	Pass	Requires synchronization with cart stop.
Bin fullness detection	Pass	Alerts generated when threshold reached.
Weight monitoring display	Pass	Weight values displayed for operator monitoring.

4.4 Communication Reliability (Raspberry Pi ↔ ESP32)

The correct sorting is dependent on reliable communication between Raspberry Pi and ESP32. During the functioning, the sorting decisions from Raspberry Pi were communicated to ESP32 over WiFi communication. ESP32 was able to receive commands and activate the motors and servo mechanisms accordingly.

The communication system is stable in most instances through the local wireless network connection, but there might be instances of delays depending on WiFi interference or the strength of the WiFi signals. To enhance the stability, the system can employ the use of acknowledgement messages.

4.5 Sensor Monitoring Results (Arduino Unit)

The features of sensor monitoring developed using Arduino increased the safety and usability of the system. Ultrasonic sensors proved useful in bin full levels, preventing overflows. More-

over, the load cell modules with amplifiers enabled accurate weight measurements useful in monitoring production processes.

The system showed (main screen and bin screens), and it successfully provided real-time counter information, alerts related to fullness, and operation alerts. This improved the understanding and operation of the system by the operator.

4.6 Summary of Results

In general, the OLI SCAN prototype was able to meet the main objectives in offering automated inspection and sorting of olives using the integrated system that incorporates Raspberry Pi vision processing and ESP32 actuator control and communication. This implies that the prototype was able to work effectively in a stable lighting and mechanical environment. The findings underscore the significance of stable illumination, camera calibration, and effective communication.

Chapter 5

Discussion

5.1 Discussion of Achieved Objectives

This paper examines the adequacy of the OLI SCAN project in addressing the principal goals. The prototype was successful in demonstrating the automated sorting process, right from the classification of the olives, through mechanical sorting, to the corresponding bins. The Raspberry Pi vision system had the capacity to identify the principal features such as the color of the olives, size, and successfully produced the necessary sorting outputs. The system was further successful in implementing the integrated approach among the Raspberry Pi, ESP32, and Arduino systems, such that the classification output was reflected in the physical process.

The results obtained confirm that the proposed design reduces the effort for manual sorting and provides better consistency in sorting. In particular, the performance indicates that OLI SCAN offers a functional solution capable of supporting local producers of olives with an automated, cost-effective sorting approach.

5.2 Strengths of the Proposed System

OLI SCAN System has a number of benefits over conventional methods of sorting. The first major advantage of this automated system for sorting is the application of a vision classification method that can easily adjust to any color and size of olives without the need for intensive machine grading equipment. Modularity is attained by combining Raspberry Pi, ESP32, and Arduino. Each of them carries out distinct tasks.

Another advantage is the monitoring function brought about by ultrasonic sensors, in addition to the load-cell modules. This enhances system safety by inhibiting overfilling, as well as allowing weight monitoring. Additionally, the multiple-screen display system is an advantage for user convenience, given that it enhances monitoring by displaying real-time status, in addi-

tion to having counters for each bin on the display. Additionally, the fact that it has a mobile app is an advantage.

5.3 Limitations and Challenges

Though the prototype worked successfully, some shortcomings were observed during its development. Lighting conditions have a significant influence on the vision-based classification. Change in lighting, shadows, or reflection might lead to incorrect colors, which could affect classification. Therefore, lighting conditions should be stable, along with camera calibration.

Another is related to the synchronization done mechanically via vision decision timing, involving gate movement. A perfect synchronization is needed for olive location, movement by the cart, or opening/closing motions by the servo motor. Otherwise, an incorrect bin will be assigned. Another is related to wireless communication, notably WiFi, which could experience delays due to possible interferences, but this can be remedied by retry/acknowledge approaches.

Further, the designed prototype is intended to be in the smaller to medium category, and to implement this system on an industrial level would demand better-designed mechanical mechanisms, faster-speed actuators, and advanced optimal solutions.

5.4 Suggested Improvements and Future Enhancements

Various areas could be improved to enhance the efficiency of the system. For the classification, better lighting control could be incorporated using fixed LED lights with a closed inspection chamber to mitigate illumination effects from the outside environment. Other processing modifications in the images could involve adaptive thresholds or machine learning classifiers.

On the mechanical part, the incorporation of position sensors or encoders may assist in synchronization to ensure the gate is opened at the right time. Reliability in communication may be ensured by the use of protocols related to acknowledgment messages and local command buffering.

Some additional improvements can be considered, including enhancing the functionalities on the mobile application, adding cloud storage facilities to sort the records, and improving remote maintenance. Lastly, industrial scaling may be done through using more robust materials, faster actuators, and multiple-lane sorting.

Chapter 6

Conclusion and Recommendations

6.1 Conclusion

This paper deals with the design and development of OLI SCAN, which is a computer-based automatic olive scanning system. This system is designed to make the scanning process more proficient. The system proposed here is the combination of computer vision techniques and control using Raspberry Pi, ESP32, and Arduino. Raspberry Pi will be used for scanning activities based on the image captured from the camera, ESP32 is involved in the scanning procedure using motors, servo motors, and provides the facility of connectivity, whereas Arduino will be used for other activities like weight calculation, full container, scanning using the RFID system, keyboard inputs, and multiple displays.

This prototype has been able to classify and sort the olives automatically according to their color and size and also supports practical monitoring features by using ultrasonic sensors and load cell modules. The system achieved its major goals: minimizing manual sorting, improving consistency, and being cost-effective to support local producers of olives. Overall, the results confirm that vision-based sorting combined with embedded control is a feasible and promising approach for olive classification and quality improvement.

6.2 Recommendations and Future Work

Developing based on OLI SCAN can lead to making several recommendations or improvements in the future:

- Enhance lighting control through the use of a closed inspection chamber and constant LED lighting to provide improved accuracy of classifications.
- Improve vision algorithms using advanced preprocessing and machine learning to better

cope with the shadows and color differences.

- This will make the system more reliable from a mechanical perspective by incorporating stronger materials for the structure, together with better alignment of the movable cart and gate system.
- To enhance sorting synchronization, position-detection sensors or encoders can be used to coordinate the movement of olives and the operation of the servo motors correctly.
- Improve the reliability of communication between Raspberry Pi and ESP32 using acknowledgments, buffering, and retry techniques.
- Add functionality to the mobile app for statistics, recording historic data, and system configuration.
- One possible direction to scale the system to handle more units produced would be to implement multiple sorting lanes and faster actuators more adapted to an industrial setting.

Such enhancements could potentially result in the system becoming more accurate, reliable, and applicable in olive processing factories.

References

Raspberry Pi Foundation, *Raspberry Pi Documentation*, accessed 2026.

Raspberry Pi Foundation, *Camera Software / libcamera Documentation*, accessed 2026.

OpenCV, *OpenCV Documentation*, accessed 2026.

OpenCV, *Color Spaces and HSV Conversion Documentation*, accessed 2026.

Espressif Systems, *ESP32 Technical Reference / Documentation*, accessed 2026.

Arduino, *Arduino Documentation*, accessed 2026.

Arduino, *Servo Library Documentation*, accessed 2026.

IEEE, *IEEE 802.11 Wireless LAN Standard*, accessed 2026.

IETF, *HTTP Semantics (RFC 9110)*, accessed 2026.

IETF, *The WebSocket Protocol (RFC 6455)*, accessed 2026.

ESPAsyncWebServer Project, *ESPAsyncWebServer Documentation*, accessed 2026.

AsyncTCP Project, *AsyncTCP Documentation*, accessed 2026.

Avia Semiconductor, *HX711 Datasheet*, accessed 2026.

Ultrasonic Sensor (HC-SR04) Datasheet/Documentation, accessed 2026.

NXP, *MFRC522 RFID Reader Documentation*, accessed 2026.

Miguel Balboa, *MFRC522 Arduino Library Documentation*, accessed 2026.

Avishay Orpaz, *TM1637Display Arduino Library Documentation*, accessed 2026.

LiquidCrystal_I2C Library Documentation, accessed 2026.

I2CKeyPad Library Documentation, accessed 2026.

Texas Instruments / Compatible Modules, *LM2596 DC–DC Buck Converter Documentation*, accessed 2026.

Relay Module Documentation (5V/12V, Active-Low/High Variants), accessed 2026.