TOUCHLESS WATER DISPENSER

`

MINI PROJECT LAB

Submitted by

PRAVIN K	**REG.NO: 22TH0277**
TRAVIDRAJ A	**REG.NO: 22TE0060**
VISHWA V	**REG.NO: 22TH0316**
NAVEEN MATHIMARAN T	**REG.NO: 22TH0267**

In partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Of

PONDICHERRY UNIVERSITY

DEPARTMENT OF INFORMATION TECHNOLOGY

MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

KALITHEERTHALKUPPAM, PUDUCHERRY- 605 107.

MAY - 2025

MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that the Mini Project Lab Work titled "Touchless Water Dispenser" is a bonafide work done by **PRAVIN K [Reg. No. 22TH0277], TRAVIDRAJ A [Reg. No. 22TE0060], VISHWA V [Reg. No. 22TH0316] and NAVEEN MATHIMARAN T [Reg. No. 22TH0267]** in partial fulfillment for the award of the degree of Bachelor of Technology in Information Technology of the Pondicherry University during the academic year 2024-25.

Staff in charge **Head of the Department**

(Mrs.J. PRABAVADHI) **(Dr. P. SIVAKUMAR)**

Submitted for the University Examination held on.............................

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our Chairman & Managing Director **Shri. M. Dhanasekaran,** our Secretary **Dr. K. Narayanasamy,** our Treasurer Shri. **D. Rajarajan** and our Joint Secretary Shri **S. Velayudham** of **Sri Manakula Vinayaga Educational Trust, Puducherry** for providing the necessary facilities to successfully complete our mini project and report works.

We express our sincere thanks to our beloved Principal **Dr. S. Malarkkan** for having provided the necessary facilities and encouragement for the successful completion of this project work.

We express our sincere thanks to **Dr. P. Sivakumar**, **Head of the Department**, Department of Information Technology**,** for his support in making necessary arrangements for the conduction of the mini project and also for guiding us to execute our project successfully.

We express our sincere thanks to **Dr. P. Sivakumar**, Professor, Department of Information Technology for he consistent reviews which motivated us in completing a project.

We thank all our department faculty members, non-teaching staff, and my friends for helping us to complete the document successfully on time. We would like to express our eternal gratitude to our parents for the sacrifices they made for educating and preparing us for our future and their everlasting love and support.

We thank the Almighty for blessing us with such wonderful people and for being with us always.

PRAVIN K
TRAVIDRAJ A
VISHWA V
NAVEEN MATHIMARAN T

ABSTRACT

In today's world, the integration of automation and hygiene practices has become increasingly essential, particularly in domains involving public health, resource management, and contactless operation. One such area of concern is the conventional water dispensing systems that often require manual handling, posing risks of contamination and inefficiency. To address this challenge, this project presents the development of a **Touchless Water Dispensing System with Smart Level Detection**, an intelligent solution designed using an Arduino microcontroller, IR sensor, LiDAR-based distance sensor (VL53L0X), and motor driver modules to automate water dispensing based on container size and fill level detection.

The system operates through a seamless interaction of multiple hardware components, orchestrated to deliver a hands-free and precise water dispensing experience. The working principle involves the detection of a container placed beneath the dispenser using an IR sensor. Once a container is detected, the system initiates a vertical motor mechanism that adjusts based on the time the container remains under detection, indirectly determining the height of the container. By calculating the time taken during vertical movement and knowing the constant motor speed, the effective height of the container is computed. This helps determine how much water the container can safely hold without overflow.

Upon measuring the container height, the system switches into a **pumping phase**, utilizing the VL53L0X Time-of-Flight sensor to continuously monitor the water level inside the container. The pump is activated via an L298N motor driver and continues to operate until the detected water level matches the predefined threshold (target fill level), calculated earlier based on the container height. Once the container is filled to the desired level, the pump is automatically turned off, ensuring no spillage and complete automation.

SUSTAINABLE DEVELOPMENT GOALS (SDGs) MAPPING

Title : TOUCHLESS WATER DISPENSER

SDG Goal-3 (Good Health and Well-Being)

SDG Goal-6 (Clean Water and Sanitation)

SDG Goal-12 (Responsible Consumption and Production)

SDG 3 – Good Health and Well-being

This touchless water dispenser system improves public hygiene by eliminating the need for direct contact, thereby reducing the transmission of germs and viruses. It is especially beneficial in hospitals, schools, and public areas where maintaining sanitation is critical. By offering a safe and contact-free method of accessing water, the project contributes significantly to health and well-being in both urban and rural communities.

SDG 6 – Clean Water and Sanitation

The system supports clean water initiatives by ensuring water flows only when needed, based on sensor detection. It prevents spillage and wastage, making water use more efficient and sustainable. This aligns with the goal of providing access to clean and affordable water, especially in regions facing water scarcity. The project encourages mindful consumption while promoting hygiene and sanitation in daily life.

SDG 12 – Responsible Consumption and Production

By using intelligent automation and sensors, this system promotes the efficient use of natural resources, particularly water. It ensures water is dispensed only as required, reducing unnecessary consumption. This project reflects sustainable innovation in everyday utilities, minimizing environmental impact and fostering responsible behavior. It highlights how smart technology can support a circular economy and reduce overuse of essential resources like water.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1. OVERVIEW

In the wake of increasing health awareness and the demand for automation, touchless systems have gained significant importance. Water dispensers, commonly found in public places, households, and offices, often require manual interaction, increasing the risk of contamination. They can also lead to water wastage due to human error during filling. Our project, the Touchless Smart Water Dispenser, aims to solve these problems by providing a fully automated, hygienic, and efficient solution for dispensing water without any physical contact.

The system uses an Arduino Uno as the control unit, paired with an IR sensor, a VL53L0X LiDAR sensor, and a DC water pump controlled through an L298N motor driver. When a container is placed under the dispenser, the IR sensor detects its presence and initiates the process. A short motor movement helps measure the container’s height using a time-based calculation. From this, the system determines the target water fill level, adapting automatically to different container sizes.

Once the measurement is complete, the VL53L0X LiDAR sensor starts monitoring the water level inside the container in real time. As water is pumped, the sensor continuously checks the distance from the top of the container to the water surface. When the water reaches the target level, the system automatically stops the pump, preventing both overflow and underfilling. A status LED indicates when the water is being dispensed.

This solution is ideal for environments that require hygiene and precision, such as hospitals, public facilities, smart homes, schools, and offices. It helps eliminate the risk of disease transmission through shared touchpoints and also promotes water conservation by dispensing only the required amount.

In conclusion, the Touchless Smart Water Dispenser is a compact and innovative solution designed to improve public health standards and resource efficiency. It represents a step forward in creating smart, sustainable, and user-friendly systems for everyday us.

1.2. EXISTING SYSTEM

Before developing our Touchless Smart Water Dispenser, we explored various existing solutions and technologies designed to automate water dispensing and improve hygiene in fluid handling systems. While several commercial and DIY projects address aspects of touchless dispensing, most lack full adaptability to varying container sizes or real-time water level control. Below are some existing systems that relate to our project:

Automatic Water Dispensers with IR Sensors

These are the most common types of touchless dispensers. They use **infrared (IR) proximity sensors** to detect the presence of hands or containers and trigger a **predefined duration** of water flow. However, these systems:

- Do not measure the size of the container.
- Lack accuracy in water quantity dispensed.
- Cannot stop automatically based on fill level, leading to **overflows or underfills**.

Limitation: Fixed time-based control without real-time feedback or container adaptability.

Ultrasonic Sensor-Based Dispensers

Some DIY and industrial systems use **ultrasonic sensors** to detect the water level inside a container. These offer **non-contact distance sensing**, but:

- Their accuracy reduces when dealing with narrow openings or reflective surfaces like water.
- They often require complex calibration.
- Most implementations still rely on **manual input** to select container size or fill level.

Limitation: Not fully automated or adaptive to unknown containers.

Smart Bottle-Filling Stations

Advanced public water stations, like those found in airports and offices, have **sensor-activated dispensers** that detect bottles and fill them automatically. Some include displays to show water quantity and filter status.

Limitation: These systems are expensive, fixed installations, and generally assume **standard bottle sizes**. They are not portable or customizable for various environments or DIY use.

DIY Arduino-Based Water Level Controllers

Hobbyists and students have created basic **Arduino water level control systems** for tanks and bottles using float sensors or basic time logic. These systems:

- Typically control motor pumps based on water level.
- May use LEDs to indicate level or relay modules to switch motors.

Limitation: Most are **not touchless**, don't use real-time sensors like LiDAR, and aren't optimized for adaptive container filling.

1.2.1. Comparison with our project

Feature	**Existing Systems**	**Proposed Project**
Touchless Operation	Yes (some)	✅ Yes
Automatic Container Size Detection	❌ No	✅ Yes (via motor timing method)
Real-Time Water Level Monitoring	Partially (ultrasonic)	✅ Yes (via LiDAR sensor)
Adaptive Filling	❌ No	✅ Yes
Low-Cost and Portable	❌ Mostly commercial/large	✅ Yes (DIY and compact)

1.3. PROPOSED SYSTEM

To meet the demands for hygiene, automation, and efficient water usage, the proposed smart water dispenser system integrates intelligent components and sensors to deliver a fully touchless, adaptive, and reliable solution.

Automated container detection

The system uses an **Infrared (IR) sensor** to automatically detect the presence of a container under the dispenser. Once a container is identified, the system initiates the measurement and filling process without requiring any manual interaction.

Smart height measurement

To adapt to different container sizes, the system employs a **motor-driven measurement method**. By calculating the time it takes for the motor to guide the container and using a known motor speed, the system estimates the container's height and determines the target water level to fill.

Real-time water level monitoring

A **VL53L0X LiDAR distance sensor** is used to continuously monitor the rising water level in real time. This enables the system to precisely stop water flow once the desired level is reached, avoiding overflows or underfills.

Automatic pump control

The water pump is controlled using an **L298N motor driver**, which is activated only when necessary. The system turns off the pump automatically when the water reaches the calculated target level, ensuring accurate and efficient dispensing.

Touchless operation for hygiene

By eliminating the need for physical contact, the system reduces the risk of germ transmission, making it ideal for public use in hospitals, schools, and offices.

Visual status indication

An onboard **status LED** provides a visual indicator of the system's state—whether it is dispensing water, idle, or has completed the fill process.

Low-cost and portable design

The system uses **affordable, readily available components** and is compact in design, making it easy to deploy in various locations without high installation costs.

Energy Efficient And Safe

The system operates only when needed, conserving energy. Components are powered and managed efficiently to avoid overheating or unnecessary power consumption.

Future expandability

The modular design allows for future upgrades such as **IoT integration**, **mobile app control**, and **remote monitoring**, paving the way for smart home or smart campus deployments

Key Features

- **Automatic Container Detection** using an IR sensor.
- **Container Height Measurement** through controlled motion and time calculation.
- **Real-Time Water Level Monitoring** using a laser-based VL53L0X LiDAR sensor.
- **Controlled Water Pumping** via a motor driver
  (L298N), ensuring no overflow and minimal wastage.
- **Touch-Free Operation**, enhancing hygiene and usability in public or shared spaces.

Workflow

1. When a container is placed, the IR sensor detects its presence.
2. A motor-
   driven platform moves briefly to estimate the container's height using the travel time.
3. The system calculates the **target water fill level**.

4. The pump activates and fills water while continuously measuring the real-time water level using the LiDAR sensor.
5. Once the water reaches the target level, the system stops the pump and signals completion.

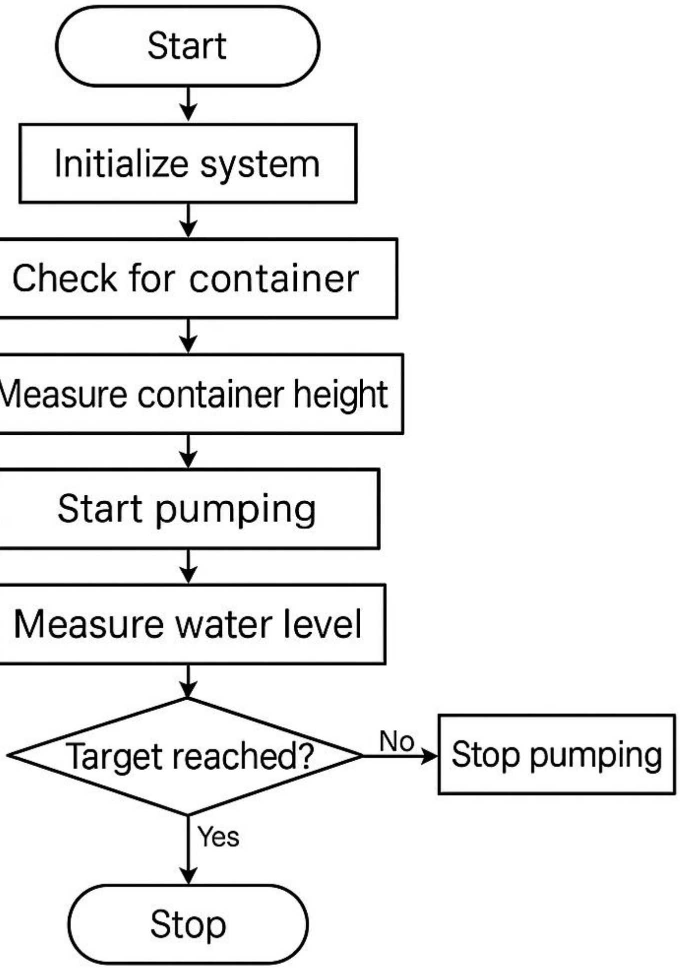


FIG 1.3: FLOW CHART

1.4. ARCHITECTURE DIAGRAM

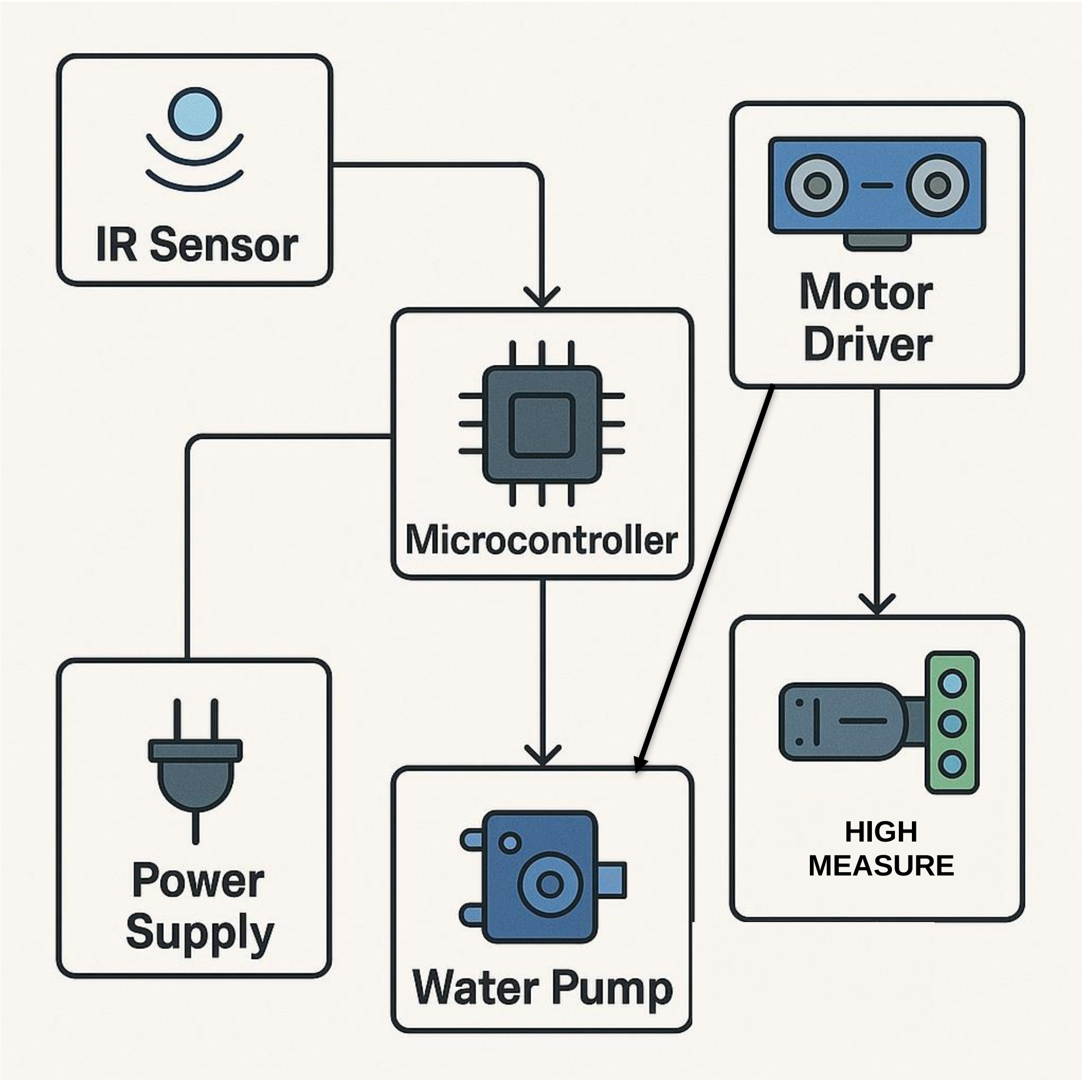


Fig 1.4: ARCHITECTURE DIAGRAM

This touchless water dispenser system automatically detects a container and fills it with water based on available space. An IR sensor detects the presence of a container, triggering the microcontroller to activate a motor that helps measure container height. A VL53L0X distance sensor continuously monitors the water level to calculate the remaining space. The microcontroller uses this data to control a water pump through a motor driver (L298N). Water flows until the measured level matches the calculated target. A status LED indicates pump activity. The entire system is powered by a regulated power supply, ensuring safe and reliable operation.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1. SOFTWARE REQUIREMENTS

To ensure the smooth operation, precise measurement, and automation of the Touchless Smart Water Dispenser, the following software tools and libraries are essential:

Arduino IDE

The **Arduino Integrated Development Environment (IDE)** is the primary platform used to write, compile, and upload code to the Arduino microcontroller. It offers a user-friendly interface and supports various libraries for sensor and module integration.

- **Version Recommended:** 1.8.19 or latest
- **Operating System Compatibility:** Windows, macOS, Linux

Added arduino libraries

To interface external components like the distance sensor and motor driver, specific libraries are required:

- **Adafruit VL53L0X Library:**
  Used for communicating with the VL53L0X LiDAR distance sensor. Enables precise distance readings via I2C communication.
- **Wire.h Library:**
  Built-in Arduino library for I2C communication, necessary for VL53L0X operation.

Serial monitor tool and Usb drivers

The Serial Monitor within the Arduino IDE is used for debugging and real-time observation of sensor data and system states. It helps validate operation stages such as container detection, fill level estimation, and pump control.

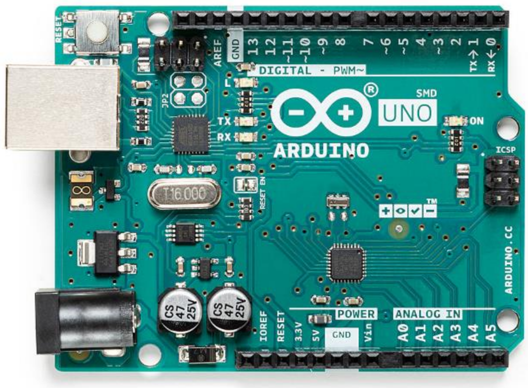
Required for establishing communication between the computer and the Arduino board. Drivers vary based on the board type (e.g., CH340 for clones).

2.2. HARDWARE REQUIREMENTS

To build a fully functional and reliable Touchless Smart Water Dispenser, the following hardware components are required. These components work in tandem to ensure precise operation, sensor interaction, and system control.

Microcontroller

- **Arduino Uno (or compatible board)**
  The central unit of the system that processes inputs from sensors, calculates the required fill level, and controls the water pump and motors. It uses the Arduino IDE for programming and real-time monitoring.

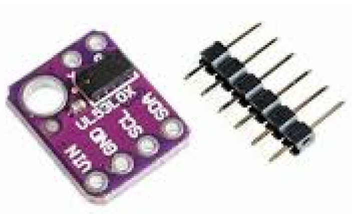


Infrared (ir) sensor

- **IR Sensor (e.g., KY-033 or similar)**
  Used to detect the presence of a container under the dispenser. It triggers the system to start the water filling process when the container is detected.

Li-DAR distance sensor

- **Adafruit VL53L0X LiDAR Distance Sensor**
  This sensor provides highly accurate distance measurements, allowing the system to continuously monitor the water level in the container. It ensures that the system stops pumping once the desired water level is reached.



Dc motor and motor driver module

- **DC Motor**
  Used to move or position the container for measuring its height. The motor will be controlled based on timing and system requirements.

- **L298N Motor Driver**
  A dual H-Bridge motor driver that controls the direction and speed of the DC motor, ensuring precise motor control and container movement.

Water pump

- **12V DC Water Pump**
  The pump is responsible for moving water into the container. It is controlled by the L298N motor driver to activate and deactivate based on the target water level.

Power supply

- **12V DC Power Adapter**
  Provides the necessary voltage for the Arduino, water pump, and motor. The power supply must be capable of supporting the combined power requirements of the system components.

Led indicator

- **LED (5mm or similar)**
  Used to indicate the status of the dispenser, such as when water is being dispensed or when the process is complete. This visual feedback helps the user know when the system is active or idle.

Resistors and jumper wires

- **Standard Resistors and Jumper Wires**
  Required for proper connections between components like the sensors, motor, and microcontroller.

Breadboard or pcb

- **Breadboard or Custom PCB**
  Used for assembling the components and connecting them to the microcontroller. For permanent setups, a custom printed circuit board (PCB) may be used for a more stable and professional design.

CHAPTER 3

DESIGN

3.1. CLASS DIAGRAM

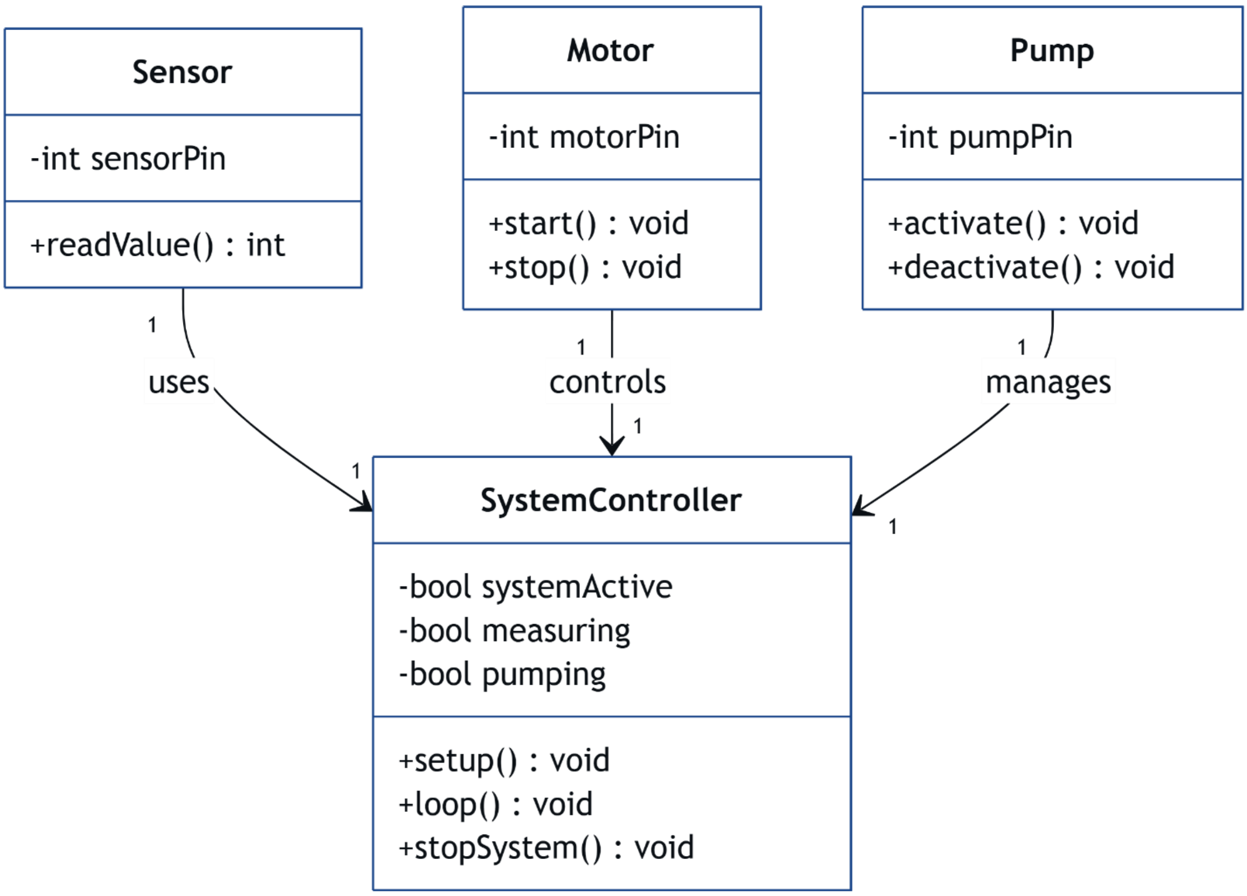


FIG 3.1: CLASS DIAGRAM

The UML class diagram represents the architecture of a touchless water dispenser system. The SystemController class is central, coordinating the entire process by using the Sensor to detect and measure, controlling the Motor to adjust for container positioning, and managing the Pump to dispense water. Each class has specific attributes and methods that reflect its hardware functionality, such as starting/stopping the motor or reading sensor values. The diagram highlights clear relationships and responsibilities, ensuring a modular and organized design for smooth operation of the system.

3.2. SEQUENCE DIAGRAM

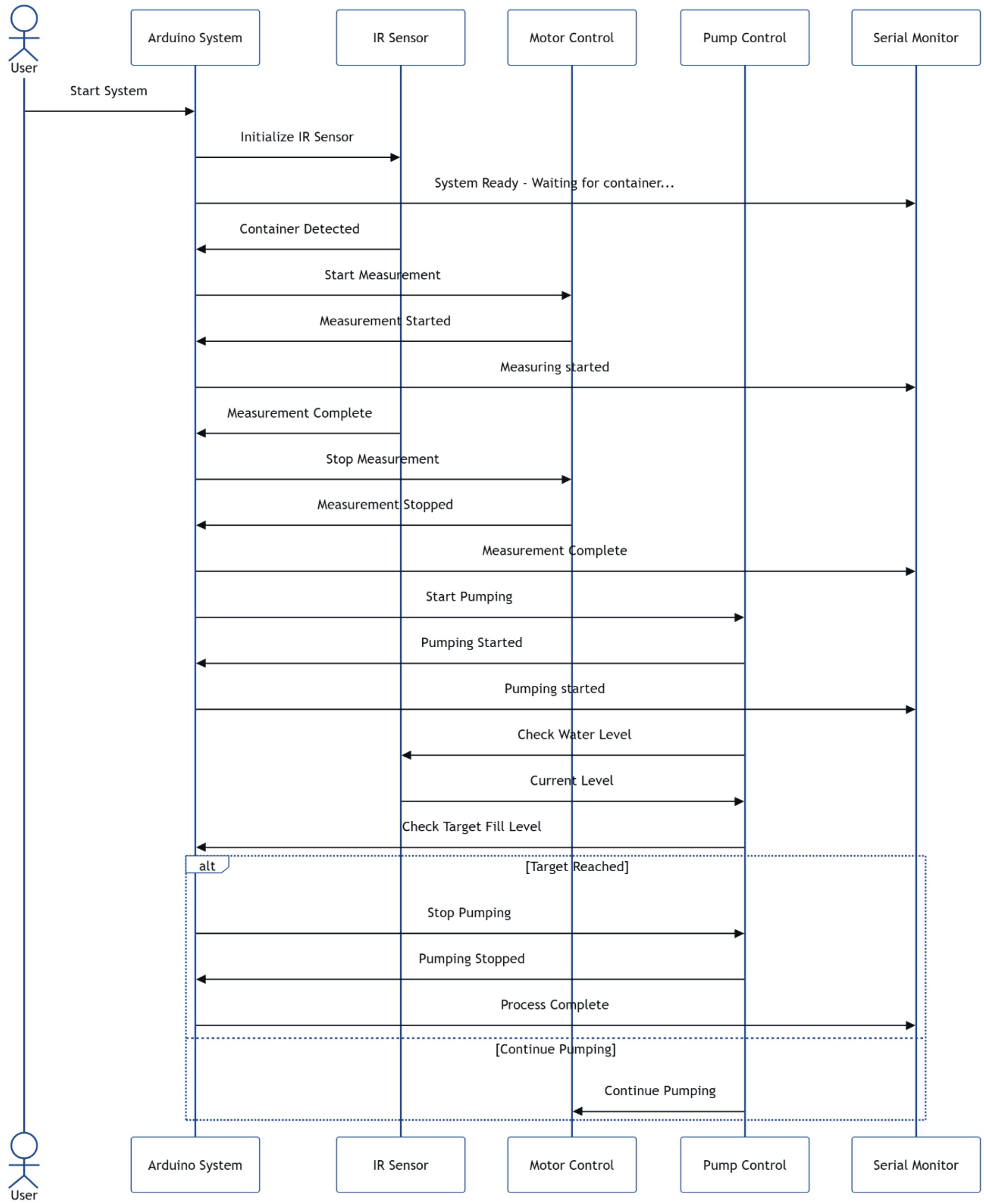


FIG 3.2: SEQUENCE DIAGRAM

The sequence diagram illustrates the interactions in the touchless water dispenser system. It begins with the Arduino system initializing components, particularly the IR sensor. When a container is detected, the motor starts moving to measure its height. Once measurement is complete, the pump is activated to start filling water. The system continuously checks the water level using a distance sensor and compares it to the target fill level. If the target is reached, the pump stops and the system completes the process. This diagram effectively visualizes the real-time sequence of operations and communication among system components.

3.3. ACTIVITY DIAGRAM

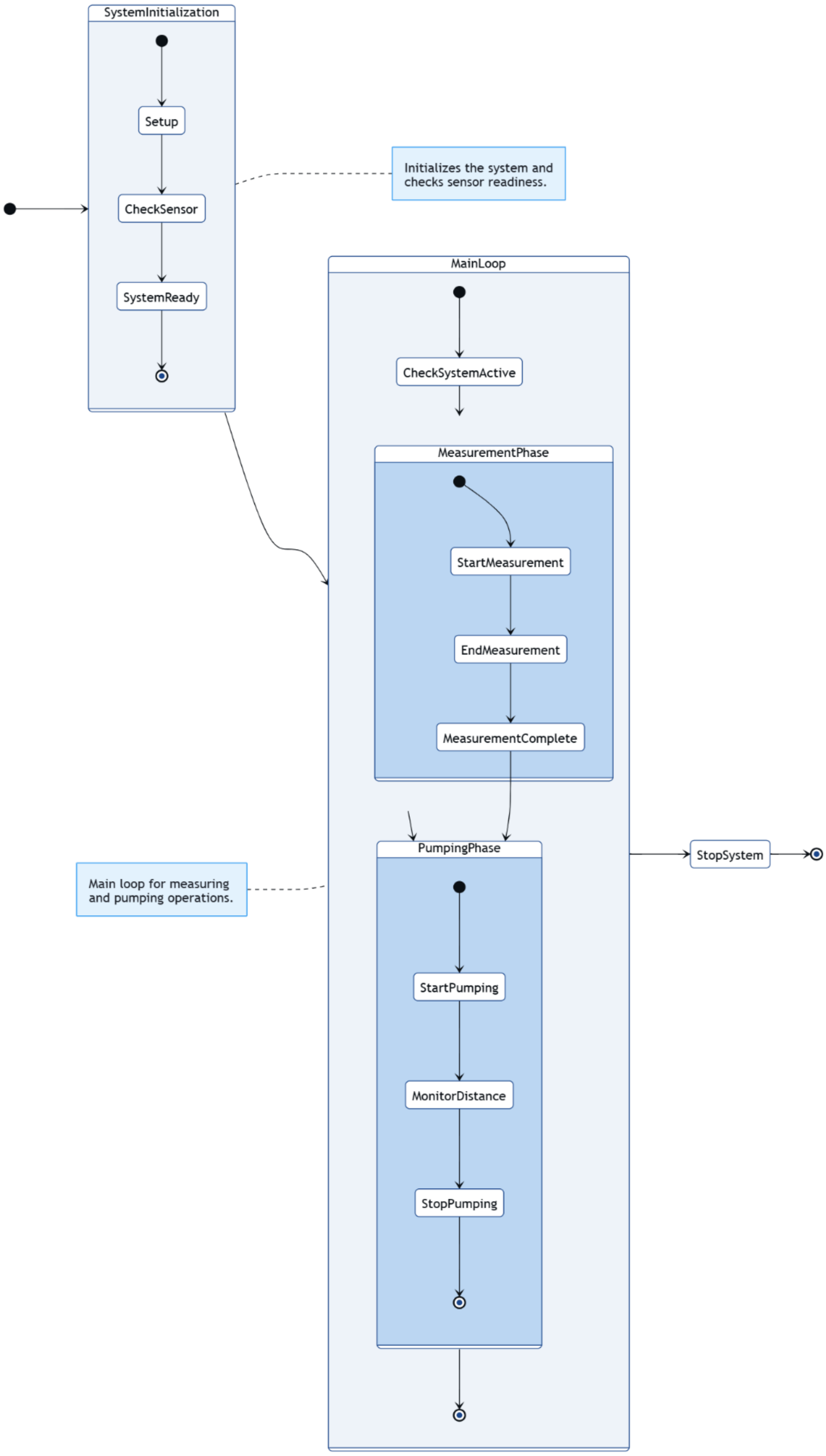


FIG 3.3: ACTIVITY DIAGRAM

This activity diagram illustrates the operational flow of the touchless water dispenser. The process begins with system initialization, including setup and sensor checks. Once the system is ready, it enters a continuous main loop. In the measurement phase, the container's height is measured using a motor, and the system calculates the target fill level. It then transitions into the pumping phase, where water is dispensed while monitoring the distance with a sensor. Once the target level is reached, pumping stops, and the system concludes its operation. This diagram effectively outlines the logical steps and decision points in the system's workflow.

CHAPTER 4

CODING

This Arduino code powers a touchless water dispenser that detects a container using an IR sensor, measures its height, and fills it accurately using a LiDAR sensor and motor-pump system, ensuring hands-free and efficient water dispensing.

4.1. Program

```
#include <Wire.h>

#include "Adafruit_VL53L0X.h"

#define IR_SENSOR 2            // IR sensor OUT pin

#define MOTOR_INT1 3           // L298N IN1

#define MOTOR_INT2 4           // L298N IN2

#define MOTOR_ENA 5            // L298N ENA (PWM)

#define PUMP_INT3 6            // L298N IN3 (Pump)

#define PUMP_INT4 7            // L298N IN4 (Pump)

#define WATER_FLOW_PIN 13      // Status LED

#define MOTOR_SPEED 1.667      // Speed in cm/sec

#define FRAME_SIZE 37          // Container Frame Size in cm

Adafruit_VL53L0X lox = Adafruit_VL53L0X();

unsigned long startTime = 0;

unsigned long endTime = 0;

bool measuring = false;

bool pumping = false;

bool systemActive = true;

int distance;
```

```
float remainingSpace;
float targetFillLevel = 0;
void setup() {
  pinMode(IR_SENSOR, INPUT);
  pinMode(MOTOR_INT1, OUTPUT);
  pinMode(MOTOR_INT2, OUTPUT);
  pinMode(MOTOR_ENA, OUTPUT);
  pinMode(PUMP_INT3, OUTPUT);
  pinMode(PUMP_INT4, OUTPUT);
  pinMode(WATER_FLOW_PIN, OUTPUT);
  Serial.begin(9600);
  while (!Serial) { delay(1); }
  if (!lox.begin()) {
    Serial.println(F("Failed to boot VL53L0X"));
    while (1);
  }
  Serial.println(F("System Ready - Waiting for container..."));
}
void loop() {
  if (!systemActive) return;
  int irStatus = digitalRead(IR_SENSOR);
  // Measurement phase
  if (irStatus == LOW && !measuring) {
    startTime = millis();
    measuring = true;
```

```
  Serial.println("Container detected - measuring started");
  digitalWrite(MOTOR_INT1, LOW);
  digitalWrite(MOTOR_INT2, HIGH);
  // digitalWrite(MOTOR_INT1, HIGH);
  // digitalWrite(MOTOR_INT2, LOW);
  analogWrite(MOTOR_ENA, 255);
}
// End measurement
if (irStatus == HIGH && measuring) {
  measuring = false;
  digitalWrite(MOTOR_INT1, LOW);
  digitalWrite(MOTOR_INT2, LOW);
  analogWrite(MOTOR_ENA, 0);
  delay(100);
  endTime = millis();
  float containerHeight = MOTOR_SPEED * (endTime - startTime) / 1000.0;
  targetFillLevel = FRAME_SIZE - containerHeight + 2;
  Serial.println("\n--- Measurement Complete ---");
  Serial.print("Container hight: ");
  Serial.println(containerHeight);
  Serial.print("Target Fill Level: ");
  Serial.print(targetFillLevel);
  Serial.println(" cm");
  pumping = true;
  Serial.println("Pumping started (LiDAR values below)");
```

```
  }
  // Pumping phase with continuous LiDAR output
  if (pumping) {
    VL53L0X_RangingMeasurementData_t measure;
    lox.rangingTest(&measure, false);
    if (measure.RangeStatus != 4) {
      distance = measure.RangeMilliMeter / 10;
      Serial.print("Current Level: ");
      Serial.print(distance);
      Serial.println(" cm");
      if (distance <= targetFillLevel) {
        stopSystem();
      } else {
        digitalWrite(PUMP_INT3, LOW);
        digitalWrite(PUMP_INT4, HIGH);
        digitalWrite(WATER_FLOW_PIN, HIGH);
      }
    }
  }
  delay(500);
}
void stopSystem() {
  digitalWrite(PUMP_INT3, LOW);
  digitalWrite(PUMP_INT4, LOW);
  digitalWrite(WATER_FLOW_PIN, LOW);
```

```
pumping = false;

systemActive = false;

Serial.println("\n--- Process Complete ---");

Serial.println("Target reached - System stopped");

Serial.println("Reset Arduino to restart");

  }
```

4.2. EXPLANATION FOR CODING

Library Inclusion and Sensor Initialization

The code begins by including the necessary libraries: Wire.h for I2C communication and Adafruit_VL53L0X.h for interfacing with the VL53L0X LiDAR distance sensor. An object lox of the Adafruit_VL53L0X class is initialized to control the sensor.

Pin Definitions and Constants

A set of macros define the digital pins connected to various hardware components:

- IR_SENSOR is connected to an IR sensor to detect the presence of a container.
- MOTOR_INT1, MOTOR_INT2, and MOTOR_ENA control the DC motor for container movement.
- PUMP_INT3 and PUMP_INT4 control the water pump via an H-bridge.
- WATER_FLOW_PIN is used to turn on an LED indicator during water flow.
- MOTOR_SPEED is set to 1.667 cm/sec, representing the movement speed of the container platform.
- FRAME_SIZE is the maximum height of the container enclosure, set at 37 cm.

Variable Declarations

Variables are initialized to track timing (startTime, endTime), system states (measuring, pumping, systemActive), and values such as distance, remainingSpace, and targetFillLevel.

Setup Function

In the setup() function:

- All input/output pins are configured.
- Serial communication is started for debugging and monitoring.
- The VL53L0X sensor is initialized. If it fails to start, the system halts with an error message.
- A startup message is printed to indicate the system is ready.

Main Loop Function

The loop() function continuously checks for system activity and controls the sequence of operations.

Container Detection and Measurement Start

- If the IR sensor detects a container (LOW signal), the system records the start time, sets the measuring flag to true, and activates the motor to move the container.

End of Measurement

- When the container is removed (HIGH signal), the system stops the motor and calculates the container's height using the elapsed time and motor speed.
- The targetFillLevel is calculated by subtracting the container height from the FRAME_SIZE, with a slight adjustment.
- The pumping flag is set to true, indicating that the water-filling phase should begin.

Water Pumping and Monitoring

If the pumping flag is true:

- The system continuously reads distance data from the VL53L0X sensor to monitor the water level inside the container.
- If the measured distance (from sensor to water surface) is **less than or equal to** the targetFillLevel, the pump is stopped.
- If the water level is still below the target, the pump is activated, and the status LED is turned on.

System Stop Function

The stopSystem() function is called when the desired water level is reached:

- The pump and LED are turned off.
- The system flags are updated to prevent further operation unless the Arduino is reset.
- A message is printed to indicate successful operation and shutdown.

CHAPTER 5

IMPLEMENTATION

The implementation phase involves translating the design and planned features of the touchless water dispenser into a functional prototype. This phase covers hardware integration, software development, and system calibration to ensure accurate detection, measurement, and water dispensing. The implementation process was executed in several key steps:

Hardware Assembly

The first step involved assembling all physical components. The following key modules were connected and configured:

- **IR Sensor** was installed at the container detection zone to sense when a container is placed.
- **VL53L0X LiDAR Sensor** was mounted above the container area to continuously measure the water level inside the container.
- **L298N Motor Driver** was connected to control both the conveyor motor and the water pump motor.
- **DC Motor** was aligned with a small conveyor or sliding mechanism to detect the height of the container based on time and speed.
- **Water Pump** was placed in line with the water source and controlled via the motor driver.
- **LED Indicator** was added to represent water flow status.

All components were connected to the **Arduino Uno**, with proper pin assignments and power supply considerations.

Software Programming

Using the Arduino IDE, the system logic was implemented in C/C++. The code handles real-time sensor inputs, distance calculations, and motor control:

- **IR Detection**: When a container is detected (IR sensor goes LOW), the system starts the measurement phase.

- **Container Height Measurement**: The time duration of conveyor motor movement is used with a pre-defined speed constant to calculate the container's height.
- **Water Level Target Calculation**: The difference between the frame size and container height determines how much water is needed.
- **Distance Monitoring**: The LiDAR sensor continuously checks the current water level.
- **Pump Control**: The pump turns on until the measured level matches the target level, then turns off automatically.
- **System Halt**: Once filled, the system stops and displays a completion message via the Serial Monitor.

Testing and Calibration

- **Container Detection Calibration**: Adjusted IR sensitivity and tested detection timing.
- **Motor Speed Calibration**: Calibrated motor speed (1.667 cm/sec) by measuring the actual distance moved in given time.
- **LiDAR Accuracy Testing**: Calibrated distance values and ensured stable readings in different lighting environments.
- **Water Flow Timing**: Tuned the pump control loop for optimal fill time and precision.

Safety Measures and Logic Locking

- The system includes a safety shutdown if the container is removed mid-operation.
- The pump will not activate unless a container is present and height has been properly measured.
- Once the target level is reached, the system locks itself to avoid unintended reactivation.

Final Integration and Demonstration

After successful testing and debugging, all modules were mounted in a compact layout, wires were neatly organized, and the system was enclosed safely. A live demonstration was performed to show the following flow:

1. Container detection
2. Automatic measurement
3. Water dispensing

4. Automatic stop

The implementation successfully proved that the system could operate hands-free and intelligently manage water dispensing based on container size.

CHAPTER 6

TESTING

Testing in a **Touchless Water Dispenser System** involves several stages to ensure that the hardware components, control logic, and system functionality work together reliably and efficiently.

5.1. UNIT TESTING

Category: Module-Level Testing

- Verified individual components like IR sensor, VL53L0X distance sensor, L298N motor driver, and water pump.
- Used the Arduino Serial Monitor to debug and confirm sensor readings, motor direction control, and LED status separately.
- Ensured PWM signals control the motor speed correctly and each pin behaves as expected.

5.2. INTEGRATION TESTING

Category: Multi-Component Communication Testing

- Confirmed that IR sensor detection triggers the measurement phase accurately.
- Ensured the distance measured by LiDAR is used to calculate the target fill level.
- Validated that water pump activation corresponds properly with distance checks and container height.

5.3. SYSTEM TESTING

Category: Full System Validation

- Simulated a real-world scenario where a container is placed, filled to the required level, and removed.
- Ensured the entire flow—detection, measurement, pump activation, and system stop—runs seamlessly.
- Checked for unintended activations and tested reset functionality by restarting the Arduino.

5.4. PERFORMANCE TESTING

Category: Load and Timing Evaluation

- Tested how quickly the system responds to the container placement and removal.
- Measured time accuracy for motor-driven height measurement.
- Evaluated consistency of LiDAR readings under different lighting or ambient conditions.

5.5. RELIABILITY TESTING

Category: Long-Term and Error Resilience Testing

- Ran the dispenser through multiple cycles to ensure it operates consistently over time.
- Introduced delays and varied timings to check system recovery and restart behavior.
- Verified behavior in case of missing or misaligned containers.

5.6. SECURITY TESTING

Category: Safety and Protection Testing

- Ensured that the water pump shuts off reliably to prevent overflow or flooding.
- Validated safety cutoff logic when the container is removed mid-process.
- Confirmed all components shut down correctly on system stop.

5.7. FUNCTIONALITY TESTING

Category: Feature Validation

- Verified core functionalities: container detection, distance measurement, motor control, and pump operation.
- Checked LED status indication during pump activity.
- Ensured accurate water level filling as per the calculated target.

CHAPTER 7

OUTPUT SCREENS

```
System Ready - Waiting for container...
Container detected - measuring started

--- Measurement Complete ---
Container hight: 6.83
Target Fill Level: 32.17 cm
Pumping started (LiDAR values below)
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 38 cm
Current Level: 37 cm
Current Level: 37 cm
Current Level: 36 cm
Current Level: 36 cm
Current Level: 35 cm
Current Level: 34 cm
Current Level: 33 cm
Current Level: 32 cm

--- Process Complete ---
Target reached - System stopped
Reset Arduino to restart
```

Fig 7: OUTPUT SCREEN

CHAPTER 8

CONCLUSION

The Touchless Water Dispenser project was designed and developed to address the need for a hygienic, efficient, and intelligent water dispensing system, especially in environments where minimal contact is crucial. Through the integration of sensors and automation, the system provides a seamless solution for detecting containers, measuring their height, and dispensing the exact amount of water required—without any human touch.

By utilizing an **IR sensor**, the system successfully detects the presence of a container. The **DC motor** and timing mechanism allow for the accurate estimation of container height, while the **VL53L0X LiDAR sensor** continuously monitors the water level in real-time. Based on these inputs, the system calculates the target fill level and controls the **water pump** accordingly through the **L298N motor driver**. An LED status indicator enhances usability by showing when water is actively being dispensed.

Throughout the implementation, special attention was given to ensuring safety, accuracy, and ease of use. The system includes automatic shutdown features once the target water level is reached, and it prevents accidental activation when no container is detected. Calibration and testing ensured that the sensors worked reliably under various conditions, and the system performed consistently across multiple trials.

In conclusion, this project demonstrates the successful integration of hardware and software to build an intelligent, touchless, and automated water dispensing system. It is a cost-effective, scalable solution that can be deployed in public places such as offices, hospitals, schools, and homes to promote hygiene and conserve water. The project lays the foundation for future enhancements like IoT connectivity, mobile app integration, and multi-container adaptability, making it a relevant and future-ready innovation.

REFERENCES

[1] Adafruit Industries, “VL53L0X Time-of-Flight Distance Sensor,” Adafruit. [Online]. Available: https://www.adafruit.com/product/3317. [Accessed: May 6, 2025].

[2] Arduino, “Arduino Uno Rev3,” Arduino. [Online]. Available: https://store.arduino.cc/products/arduino-uno-rev3. [Accessed: May 6, 2025].

[3] STMicroelectronics, “VL53L0X Datasheet,” STMicroelectronics. [Online]. Available: https://www.st.com/en/imaging-and-photonics-solutions/vl53l0x.html. [Accessed: May 6, 2025].

[4] Electronics Hub, “L298N Motor Driver Module – Working and Interfacing,” Electronics Hub. [Online]. Available: https://www.electronicshub.org/l298n-motor-driver/. [Accessed: May 6, 2025].

[5] Circuit Digest, “Automatic Water Dispenser using Arduino,” Circuit Digest. [Online]. Available: https://circuitdigest.com/microcontroller-projects/automatic-water-dispenser-using-arduino. [Accessed: May 6, 2025].

[6] Scribbr, “IEEE Website Citation | Guide with Examples,” Scribbr. [Online]. Available: https://www.scribbr.com/ieee/ieee-website-citation/. [Accessed: May 6, 2025].

[7] IEEE, “IEEE Reference Style Guide,” IEEE Author Center. [Online]. Available: https://journals.ieeeauthorcenter.ieee.org/wp-content/uploads/sites/7/IEEE_Reference_Guide.pdf. [Accessed: May 6, 2025].