SMART UNDER-WATER DRONE USING RASPBERRY-PI FOR OCEAN CLEANUP, MARINE MONITORING, AND SAFETY APPLICATIONS

Project report in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering (Artificial Intelligence)

Submitted By

Souvagya Karmakar	Enrollment No. 12024002035140
Suchismita Basak	Enrollment No. 12024002035269
Sourav Debnath	Enrollment No. 12024002035222
Sourick Roy	Enrollment No. 12024002035256
Srijato Sarkar	Enrollment No. 12024002035258
Srinjay Konar	Enrollment No. 12024002035143
Subhankar Sahoo	Enrollment No. 12024002035226
Suchana Madhu	Enrollment No. 12024002035281
Suchana Pariyari	Enrollment No. 12024002035285
Sudip Shaw	Enrollment No. 12024002035063

Under the guidance of

Prof. Esha Choudhuri

Department of Computer Science and Engineering (Artificial Intelligence)

UNIVERSITY OF ENGINEERING AND MANAGEMENT, KOLKATA

University Area, Plot No. III – B/5, New Town, Action Area – III, Kolkata – 700160

CERTIFICATE

This is to certify that the project titled **Smart Under-water Drone using Raspberry-Pi for Ocean Cleanup , Marine Monitoring, and Safety Applications** submitted by **Souvagya Karmakar (University Registration No. 304202400901434 of 2024 - 2025), Suchismita Basak (University Registration No. 304202400901563 of 2024 - 2025), Sourav Debnath (University Registration No. 304202400901516 of 2024 - 2025), Sourick Roy (University Registration No. 304202400901550 of 2024 - 2025), Srijato Sarkar (University Registration No. 304202400901552 of 2024 - 2025), Srinjay Konar (University Registration No. 304202400901437 of 2024 - 2025), Subhankar Sahoo (University Registration No. 304202400901520 of 2024 - 2025), Suchana Madhu (University Registration No. 304202400901575 of 2024 - 2025), Suchana Pariary (University Registration No. 304202400901579 of 2024-2025), and Sudip Shaw (University Registration No. 304202400901357 of 2024–2025)** students of the University of Engineering and Management, Kolkata, in partial fulfillment of requirement for the degree of Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence), is a bonafide work carried out by them under the supervision and guidance of Prof. Esha Choudhuri during 3rd Semester of academic session of 2025 - 2026. The content of this report has not been submitted to any other university or institute. I am glad to inform that the work is entirely original and its performance is found to be quite satisfactory.

Signature of Supervisor
Department of CSE (AI)

ACKNOWLEDGEMENT

We would like to express our profound gratitude to Prof. Esha Choudhuri , Project Supervisor, Prof. (Dr.) Sudipta Sahana, Head of the Department of CSE (AI), and Prof. (Dr.) Sajal Dasgupta, Vice Chancellor, University of Engineering and Management, Kolkata for their contributions to the completion of my project titled **Smart Under-water Drone using Raspberry-Pi for Ocean Cleanup , Marine Monitoring, and Safety Applications**.

We would like to express special thanks to our project mentor for his/her time and the efforts he/she provided throughout the semester. Your useful advice and suggestions were really helpful to us during the project's completion. In this aspect, we are eternally grateful to you.

We want to acknowledge that this project was completed entirely by ourselves and not by someone else.

Last but not least, we would like to extend our warm regards to our family members and peers who have kept supporting us and always had faith in our work.

SOUVAGYA KARMAKAR

SUCHISMITA BASAK

SOURAV DEBNATH

SOURICK ROY

SRIJATO SARKAR

SRINJAY KONAR

SUBHANKAR SAHOO

SUCHANA MADHU

SUCHANA PARIYARI

SUDIP SHAW

TABLE OF CONTENTS

Content **Page No.**

CHAPTER – 6: CONCLUSION & FUTURE SCOPE

ABSTRACT

The Smart Under-water Drone project presents the design and development of a sensor-integrated underwater drone prototype aimed at real-time water quality monitoring and object detection. The system employs a Raspberry Pi 4 Model B as the central controller, interfaced with a suite of sensors including a pH sensor (via ADS1115 ADC), a DS18B20 waterproof temperature sensor, and a depth sensor to measure essential aquatic parameters. A servo-actuated gripper arm and four DC motors are incorporated to simulate propulsion and material collection. Visual monitoring is achieved through a USB webcam, enabling live image capture and basic object detection functions for environmental observation.

A Python-based Flask web dashboard provides real-time visualization of sensor data, allowing remote monitoring through any connected device. The prototype is powered by a rechargeable Type-C power bank, ensuring portability and ease of deployment. The modular system architecture allows future integration of pressure sensors and waterproof thrusters for enhanced underwater performance.

Preliminary experiments demonstrated stable sensor readings, consistent servo operation, and reliable data transmission between the Raspberry Pi and the web dashboard. The project establishes a practical proof-of-concept for deploying compact, Raspberry Pi-driven underwater systems for environmental data acquisition, pollution monitoring, and marine research applications.

CHAPTER – 1: INTRODUCTION

Water quality monitoring and underwater exploration are essential for understanding aquatic ecosystems, detecting pollutants, and maintaining environmental balance. Conventional systems used for such monitoring are often expensive, bulky, or limited in modular flexibility. To address these limitations, the Aqua Pi project proposes a compact underwater drone prototype that combines sensor-based monitoring and basic robotic functionality under the control of a Raspberry Pi 4 Model B.

The developed system integrates multiple environmental sensors, including a pH sensor (interfaced through the ADS1115 analog-to-digital converter), a DS18B20 waterproof temperature sensor, and a water depth sensor, to measure key physical and chemical parameters of the aquatic environment. A servo-actuated gripper arm is incorporated to simulate object collection or sample retrieval, while four DC motors provide directional thrust for prototype motion in shallow water environments.

Visual inspection and object detection capabilities are provided through a USB webcam, which transmits live video to the controlling Raspberry Pi. Data from all sensors are processed and displayed on a Flask-based web dashboard, enabling remote access and real-time observation through any device connected to the same network. Power for the system is supplied through a rechargeable Type-C power bank, ensuring operational flexibility and electrical safety during testing.

This prototype demonstrates scalable approach to designing small underwater monitoring systems using open-source hardware and software. By integrating sensing, actuation, and live data transmission into a single platform, Smart Under-water Drone serves as a foundation for developing future autonomous underwater vehicles capable of environmental monitoring, plastic waste detection, and marine research applications.

CHAPTER – 2: LITERATURE SURVEY

2.1 Advanced Underwater Drones

Lambertini et al. (2022) [1] presented a significant advancement in underwater drone architecture through their work on the *SUSHI DROP Project*, where a hybrid Unmanned Underwater Vehicle (UUV) named *Blucy* was developed for marine ecosystem mapping and sustainable fisheries monitoring. Their research emphasized non-invasive surveying methods using optical and acoustic sensors, coupled with real-time positioning and data synchronization systems to create accurate digital twins of marine environments. This approach enables continuous and georeferenced observation of underwater ecosystems, improving both environmental assessment and resource management efficiency.

Similarly, **Nguyen et al. (2023) [2]** explored the integration of artificial intelligence and robotic automation in underwater exploration, highlighting the use of deep learning algorithms for obstacle detection, motion control, and environmental mapping in complex marine conditions. Their findings underline how intelligent control systems can enhance underwater navigation precision and reduce human intervention in hazardous zones.

Furthermore, **Mantha et al. (2020) [3]** examined the role of digital twin technologies in marine and oceanic applications, providing insights into how virtual modeling and data-driven simulation can optimize the performance of underwater drones. The study also emphasized that digital twin-based predictive maintenance and simulation tools can help address operational challenges, including energy consumption and system reliability, in autonomous marine vehicles.

Together, these studies underline the growing intersection of autonomous underwater robotics, AI, and digital twin technologies — paving the way for sustainable and intelligent ocean monitoring systems.

2.2 Marine and Coastal Environmental Monitoring

Recent research has expanded the applications of unmanned aerial vehicles (UAVs) and underwater/remote-operated vehicles (UUVs/ROVs) to address persistent observational and monitoring gaps in marine and coastal environments. **Patrick Clifton Gray et al. (2022) [4]** argue that drones fill the “blind spot” between satellite remote-sensing and in-situ point measurements by enabling high spatial and temporal resolution observations of marine biological communities.

In another dimension, **Oleg Bukin et al. (2021) [5]** demonstrate that drones equipped with optical sensors and AI methods can be used for oil-pollution monitoring of the sea, showing how autonomous platforms contribute to environmental hazard detection.

Meanwhile, **Gabriela Escobar-Sánchez et al. (2022) [6]** examine both aerial and underwater drones for monitoring marine litter in shallow coastal waters, analyzing factors such as detection accuracy, water conditions, and cost-efficiency.

Collectively these works highlight the growing importance of autonomous unmanned systems in enhancing marine and coastal environmental monitoring — from biological oceanography to pollution tracking and debris detection — while also identifying technical, logistic and cost-related challenges for real-world deployment.

2.3 Ocean Cleanup

Recent advancements in artificial intelligence (AI), IoT, and edge computing have transformed the scope of marine ecosystem management and ocean sustainability. **Sultana et al. (2024) [7]** in their Springer publication *Artificial Intelligence for Sustainable Ocean Health* highlight how AI-driven techniques, such as deep learning and explainable AI (XAI), enhance data collection, monitoring, and forecasting of marine conditions. Their work emphasizes the use of satellite imaging, underwater sensors, and predictive models for applications like ocean pollution detection, overfishing control, and coral reef preservation. The study also stresses the importance of integrating federated learning and edge computing for real-time ocean monitoring and autonomous underwater decision-making.

Similarly, **Gupta et al. (2024) [8]** in *Environmental Sciences Proceedings* present how AI and robotic systems can be harnessed to detect and remove marine waste efficiently. Their findings illustrate that autonomous drones and robots equipped with vision-based models can identify and collect marine litter, contributing to cleaner and healthier oceans.

Complementing these approaches, **Sharma and Rathi (2025) [9]** explore the role of AI-based coastal drones for monitoring underwater biodiversity, emphasizing sustainable tourism and marine research. They highlight that edge AI processing on marine drones allows for faster image analysis and localized data-driven actions, reducing dependence on satellite connectivity.

Together, these studies establish AI and automation as critical enablers in achieving Sustainable Development Goal 14 (SDG-14: Life Below Water) by improving the precision, scalability, and responsiveness of ocean health management systems.

CHAPTER – 3: PROBLEM STATEMENT

3.1 Weakness of Existing Water Monitoring Solutions

The water quality monitoring systems we have today are mostly built for surface measurements or lab testing. They usually need people to collect samples by hand or keep the sensors fixed in one spot. Because they can’t move underwater, they miss how water conditions change at different depths or across various parts of a lake or river. This is a big issue since factors like temperature, pH, and chemical levels can differ a lot even within the same water body. On top of that, most of these systems can’t operate in real-time, so the process is slow and heavily depends on manual work. As a result, current surface-based systems can’t create a real-time underwater map of water quality, leaving a major gap in technology.

Key Points:

• Current technologies are mainly designed for surface use
• Lack of dynamic or underwater sensing features

• Changes in water parameters with depth go unmeasured
• Manual analysis causes time delays in data collection

3.2 No Integration of Sensor-Based Monitoring with Underwater Mobility

Most modern underwater drones focus on visual inspections since they're equipped with cameras, while most water monitoring systems rely on sensors but can't move around. There isn't a single affordable solution that combines underwater movement with multi-sensor environmental data collection. Without this integration, no system can fully represent underwater ecosystem behavior — cameras alone can't measure water conductivity, salinity, or acidity, and sensors alone can't show structural conditions or floating debris. For a complete and meaningful underwater environmental assessment, both visual and sensor data must be gathered at the same time using a mobile robotic platform.

Key Points:

• Camera-equipped drones don't have built-in sensors
• Sensor nodes can't move or navigate underwater
• No affordable all-in-one integrated system is available
• Without this integration, environmental data remains incomplete

3.3 No Modular and Scalable Design for Student Prototyping

Existing underwater systems are not built for modularity or experimentation. They are closed systems with fixed architecture and limited possibilities of adding sensors, components or a custom module. This is a significant disadvantage for academic researchers, because experimental robotics requires ongoing modification, reconfiguration and testing of new ideas. Students will want a platform in which parts can be added, replaced or reconfigured quickly. Without modularity, iterative development or comparative research is tremendously restricted. The barrier of not having a modular underwater platform slows innovation and limits the chance of new research at the student-level.

Key Points:

• Existing ROVs are not modular
• Academic projects need flexible upgrade options
• Lack of modularity blocks research innovation
• Prototype modification becomes difficult

CHAPTER – 4: PROPOSED SOLUTION

4.1 Integration of Multi-Sensor Monitoring System

This prototype integrates multiple water sensors such as pH sensor, temperature sensor (DS18B20) and depth sensor (HS-S37A) into one combined system. The Raspberry Pi collects continuous real-time data from these sensors and converts it into readable values using Python code and ADC (ADS1115) module. This integration allows underwater environmental parameters to be checked simultaneously, instead of using separate measurement tools. By combining all sensors in one platform, this system can read multiple water quality values at different points of depth and movement. This helps to understand how water quality changes with position under water. This data can later be stored and can also be used for future AI-based prediction or analysis studies.

Key Points:

• All sensors are interfaced with a single Raspberry Pi
• Real-time multi-parameter measurement is possible
• Allows measurement at different positions and depths
• Supports data collection for future machine learning

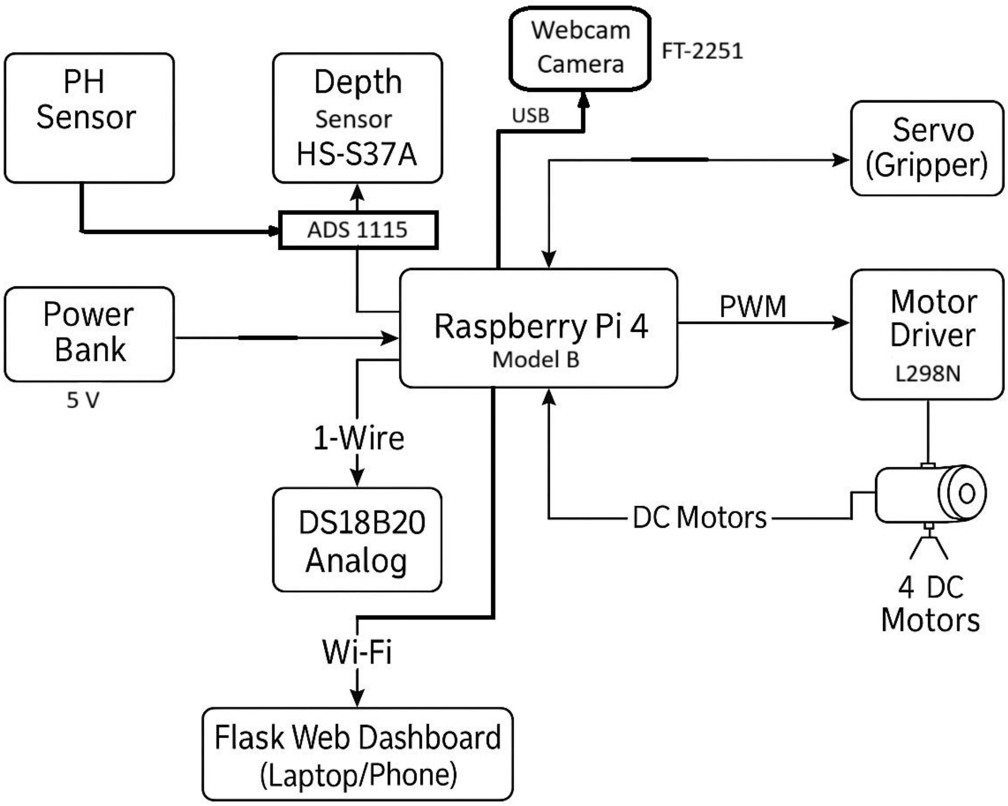

Figure 1: Workflow Diagram of Current Prototype

4.2 Real-Time Dashboard for Monitoring

The solution includes a web-based dashboard developed using Flask which shows all live data readings, including pH value, depth and water temperature. This dashboard can be opened in a laptop or mobile phone connected to the same network. The dashboard provides a simple user-friendly interface, so users do not need to look at raw Python output or terminal logs. Instead, they can monitor all sensor readings in a properly formatted interface. This approach helps in real-time decision making because the operator can check values instantly while operating the drone. In future, this dashboard can also display AI-based alerts such as “Plastic Detected” or “Drowning Detected”.

Key Points:

- Live data visualization in a Flask dashboard
- User-friendly monitoring without coding knowledge
- Can be accessed from any device on the network
- Future AI alerts can be added directly in dashboard

4.3 Properly Explained Flow Chart

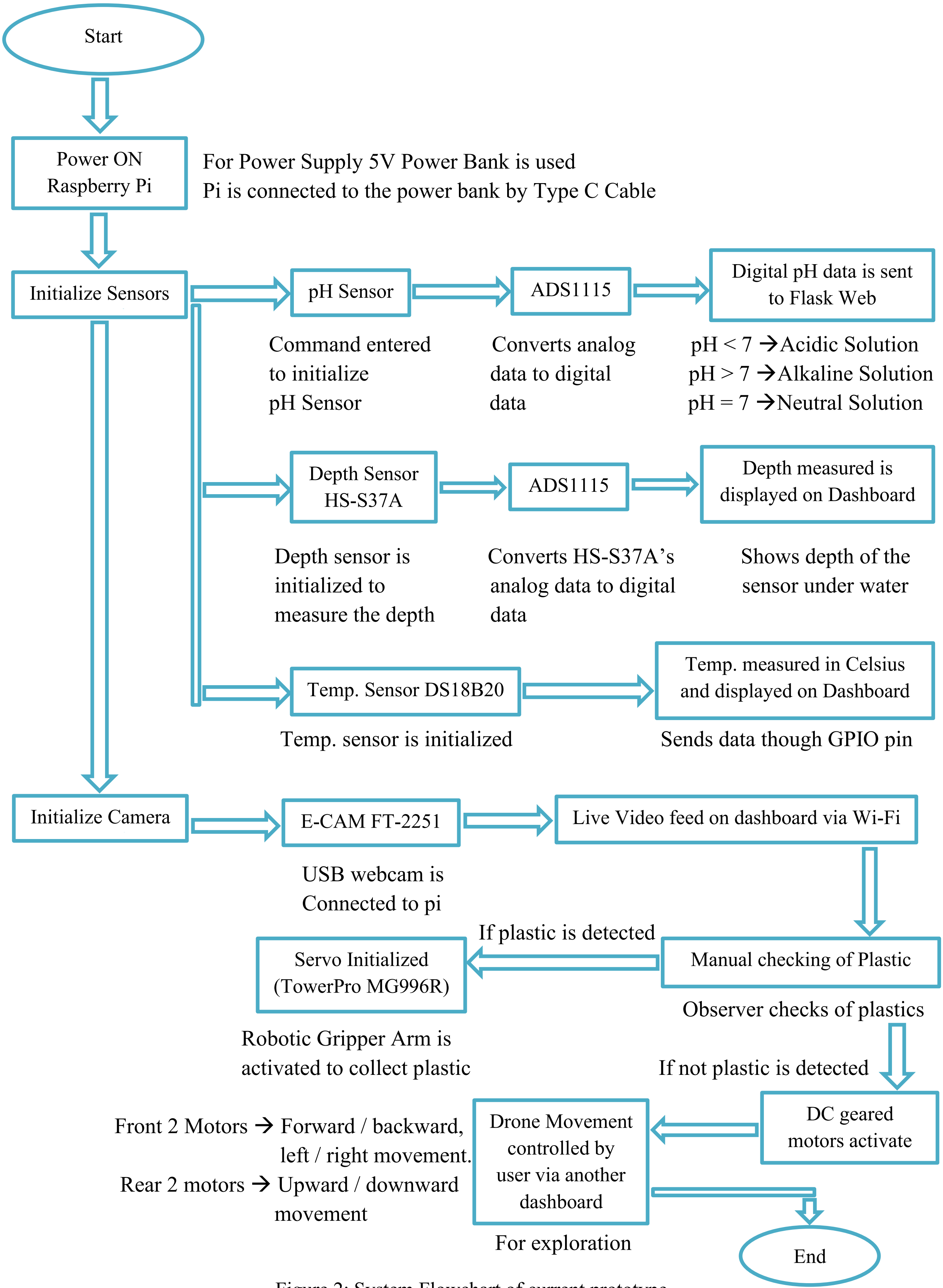

Figure 2: System Flowchart of current prototype

4.4 Base Platform for Future AI Integration

Although AI is not implemented in the first prototype due to budget and time limitation, the architecture of this system is designed in a way that supports future integration of AI models. Raspberry Pi can run lightweight AI detection models such as YOLOv5n or MobileNet which are suitable for real-time object detection in low-power devices. In future, the camera feed from the drone can be used to detect plastic waste, fish species or drowning behaviour in real time. This makes the system future-ready and highly scalable. The proposed solution therefore acts as a baseline platform that can be expanded, modified and improved in the coming academic semesters or final year project.

Key Points:

• Raspberry Pi supports lightweight AI models
• Camera feed can be used for plastic or drowning detection
• Prototype is designed for future expansion
• Makes this project future-proof and research-ready

In summary, this project proposes a low-cost, modular and extendable underwater drone platform that uses Raspberry Pi as the main processing unit to collect multiple water parameters in real time. The system integrates sensors for pH, temperature and depth measurement and displays all data through a user-friendly live dashboard interface. Although the current prototype is developed for demonstration on a cardboard model, the entire design is prepared in such a way that it can be upgraded later into a fully waterproof underwater version and can also support lightweight AI-based detection models. This makes the solution practical, scalable, and suitable for future expansion towards advanced underwater environmental monitoring and safety applications.

CHAPTER – 5 : EXPERIMENTAL SETUP AND RESULT ANALYSIS

5.1 Experimental Setup

5.1.1 Prototype Physical Assembly

The physical body of the prototype was prepared using a cardboard box structure. The reason for using cardboard in this phase was that this is still an initial development stage and a waterproof casing has not been made yet. The cardboard acted as the temporary external enclosure to hold and organize all components in fixed positions, so that proper placement, wiring path, spacing and heat flow could be studied in a controlled environment before shifting to an acrylic or waterproof body in the future. All components like Raspberry Pi, sensor modules, camera module, servo and motors were carefully placed inside the cardboard and fixed in positions to avoid loose movement. This helped the team to visualize the 3D internal layout and understand how components will be placed inside the real drone enclosure.

Key Points:

• Cardboard used only as temporary prototype enclosure
• Component positioning and spacing visualized in 3D form
• Helped study internal layout before waterproof body
• Gives clarity about physical arrangement inside future model

5.1.2 Electrical Wiring and Connectivity Interface

The next step was wiring all components systematically to the Raspberry Pi. Each sensor was connected to the ADC module (ADS1115) and from there into the GPIO pins of Raspberry Pi. The motor driver board was also connected to the correct pins for controlling DC motors. Care was taken to keep the wires organized to reduce cross interference and avoid risk of short circuit. The wiring was matched with the software GPIO configuration to ensure correct pin mapping. This stage confirmed that the electrical connections between Pi, sensors, ADC and motors were correct and functioning as expected.

Key Points:

- All components connected to Raspberry Pi GPIO
- Wiring done through ADC board for analog sensors
- Motor driver connected for motor control
- Proper pin mapping ensured software compatibility

5.1.3 Software Integration and Testing

Once the physical wiring was completed, the system was powered, and Python scripts were executed to ensure all connected devices were recognized by the Raspberry Pi. The objective here was to validate that the sensors responded to software commands, the motor driver received PWM signals correctly and the servo responded to rotation angles. This stage confirmed that both physical and digital integration was correct. The team validated that real-time values could be fetched continuously using simple Python loops.

Key Points:

- All sensors detected properly by Raspberry Pi
- Python scripts used for reading data values continuously
- Motor and servo tested for command response
- Ensured that software and hardware communication is correct

5.1.4 Dashboard Setup and Live Data Display

After confirming hardware communication, the Flask dashboard was developed to show values in real time on a web page. The Raspberry Pi acted as a mini-server, allowing real-time data values to be accessed on a browser using the IP address of the Pi. This made the monitoring process easier because now values could be viewed in a readable, clear format instead of raw console logs.

Key Points:

- Real-time monitoring via Flask dashboard
- Data displayed live in browser through Pi IP

• Eliminated need to view raw console logs
• Dashboard confirmed backend to interface link

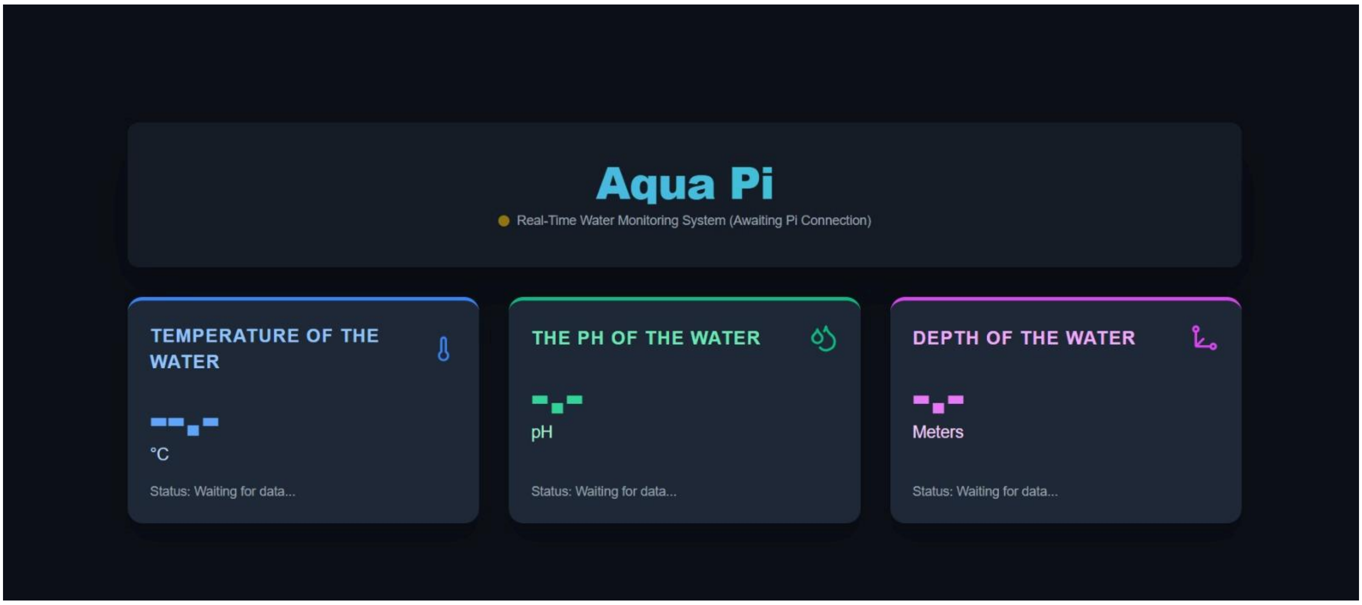

Image 1: Photograph of the dashboard

5.2 Result Analysis

5.2.1 Sensor Reading Behaviour

During the testing phase, the depth sensor produced the most stable and consistent readings when connected to the Raspberry Pi and monitored through the dashboard. This stability indicates that the depth sensor module has good linearity and does not require heavy calibration for basic functional testing in air. The DS18B20 waterproof temperature sensor also performed correctly and reacted to temperature variations when exposed to heat sources like hand warmth or room temperature shift. This proves that the temperature sensor can be used effectively for real-time data monitoring. On the other hand, the pH sensor exhibited fluctuating readings, which is a common behaviour in low-cost pH probes if calibration fluids or standard buffer solutions are not used. This means that the pH sensor needs proper calibration before deployment in real water. However, even with fluctuations, the sensor still demonstrated connectivity and response to the Raspberry Pi, which means future improvement is possible.

Key Points:

• Depth sensor most stable reading
• Temperature sensor responded to changes accurately
• pH sensor unstable, needs calibration
• Future calibration required for better accuracy

5.2.2 Dashboard Response Time

The real-time dashboard displayed sensor values instantly without any noticeable delay, proving that the system is able to perform live monitoring effectively. The dashboard continuously refreshed values, meaning that data changes were visible within seconds, which is important for any real-time environmental monitoring application. This behaviour indicates that the combination of Raspberry Pi and Flask is sufficient to provide communication between hardware and user interface without visual lag. The immediate reflection of values on the browser also shows that the Wi-Fi network and internal server logic is efficient enough for simple to medium complexity data display tasks. This proves that even at prototype stage, the system supports continuous real-time data flow, which is a strong foundation for future features including AI-based alerts and classification results from video feed.

Key Points:

• Dashboard updates instantly during tests
• No delay while reading live values
• Reliable communication between backend and UI
• Strong base for future real-time AI features

5.2.3 Motor and Servo Response

Motor testing showed that the DC motors responded correctly to PWM (Pulse Width Modulation) signals from the Raspberry Pi. The motor speed changed when different PWM values were applied, confirming that motor speed control can be implemented easily in future using this control method. Similarly, the servo motor was tested for angular motion, and it rotated correctly to the specific angles commanded through Python. This ensures that the servo can be used for small gripper movement mechanisms in future underwater operations. Although actual thrust underwater was not tested due to the cardboard enclosure being non-waterproof, the successful DC motor and servo response verifies that the system supports motion control, and only waterproof thrusters are needed in next stage to perform actual underwater monitoring.

Key Points:

• DC motors responded correctly to PWM control
• Speed control verified in testing
• Servo motor rotated as per input angle
• Proves robot movement control feasibility

5.2.4 Overall Evaluation of Prototype

Overall, the test results show that the prototype is partially successful at this stage. The main architecture including Raspberry Pi, sensors, camera, motor driver and dashboard interface is all working smoothly and communicating together. The only missing part is physical underwater testing, which will be implemented once a waterproof casing and underwater-capable motors/thrusters are added. The current setup has proven that the system can handle sensing, data processing and data display in real-time. This means the initial goal of building a base platform is successfully achieved. From this foundation, future teams or same team in next semester can upgrade the system into a fully functional underwater robot with extended AI integration and safety features.

Key Points:

• Core functional model confirmed working
• Underwater test remains pending
• System ready for future upgrading
• Base stage of research successfully completed

the experimental setup phase successfully validated the basic working architecture of the Smart Underwater Drone prototype using a cardboard enclosure to safely mount and test all components before moving towards a waterproof design. The sensors, Raspberry Pi, camera, motor driver, DC motors and servo motor were connected correctly and tested through Python scripts and a Flask-based live dashboard. During the result analysis, the depth and temperature sensor gave stable readings while the pH sensor showed fluctuations due to calibration requirements. The dashboard responded instantly to changes and the motors reacted correctly to PWM control. Although underwater testing was not performed yet, this stage proved that the foundation of the project is technically working and is now ready to be upgraded into a waterproof final model for real underwater trials in the next stage.

CHAPTER – 6: CONCLUSION & FUTURE SCOPE

6.1 Conclusion

This project successfully developed the first-stage prototype of a low-cost underwater drone concept using Raspberry Pi as the main processing unit. The system was built using a cardboard prototype model to demonstrate placement, connectivity, data processing, and real-time dashboard monitoring of underwater-relevant sensors. The temperature sensor and depth sensor were able to provide stable values, and the pH sensor showed fluctuations due to calibration requirement, indicating that improvement and fine-tuning is required. The dashboard displayed all live values without lag, proving that the communication between hardware and software is smooth. Although physical underwater testing could not be performed at this stage due to the cardboard enclosure, the system validated that the architecture is working and ready to be transformed into a fully waterproof functional underwater robot. Therefore, this prototype successfully proves the feasibility of developing a modular, low-budget underwater monitoring system that can further be enhanced with better hardware and AI features in the upcoming development stages.

6.2 Future Scope

6.2.1 Waterproof Body and Underwater Thrusters

In the next stage of development, the cardboard structure will be replaced by a fully waterproof body made from acrylic or polycarbonate casing material. This strong casing will protect the electronic components from water ingress, enabling the drone to operate safely underwater for longer durations. Along with this, proper underwater thruster motors will be integrated to support actual underwater movement in all directions inside a water body. These thrusters are specially designed to work inside water with higher torque and waterproof sealing. That will allow the drone to move forward, backward, up, down and rotate underwater, just like commercial ROVs. This will convert the prototype from a demonstration model to an actual functional underwater robot capable of underwater navigation.

Key Points:

• Replace cardboard with a waterproof casing
• Use underwater thruster motors for movement
• Allows long duration underwater operation
• Converts prototype into functioning ROV

6.2.2 AI Based Plastic Waste Detection

Once the underwater drone has a fully working video feed, lightweight AI models such as YOLO-nano or MobileNet SSD can be deployed inside the Raspberry Pi to identify plastic waste underwater in real time. This will help create an automation-based waste detection system which can assist lake authorities, NGOs, and environmental researchers to identify pollution zones without manual inspection. The AI model can also classify different waste types and track how much plastic is present in a particular location. This can help create pollution mapping and assist in cleaning strategy planning. With AI integration, the drone will no longer be a simple robotic tool but a smart intelligent system that understands and responds to its surroundings.

Key Points:

• Use deep learning for underwater plastic detection
• Helps identify pollution zones in real time
• Creates automatic detection without human guesswork
• Can support pollution mapping strategies

6.2.3 AI Based Drowning or Safety Alert System

Another important future scope is human safety. This drone can be upgraded to monitor swimming zones, river banks or lake edges to detect drowning behaviour. AI models can identify suspicious human movement patterns like sinking, irregular motion or non-responsive floating. When the system detects potential drowning, it can raise immediate alerts to authorities or lifeguards. This feature can save lives by offering early warning. Thus, this project not only supports environmental protection but can also contribute towards human life protection in future.

Key Points:

• Drone can monitor water bodies for safety
• AI can detect drowning patterns from camera feed
• Alerts can be sent for emergency response
• Supports safety in swimming and river zones

6.2.4 Autonomous Underwater Navigation

Another future improvement is to make the drone autonomous, where it can navigate underwater without manual control. Autonomous navigation can be achieved using ultrasonic sensors, IMU sensors, and AI-based path planning. This feature can allow the drone to survey large water areas automatically and collect continuous data without human operator control. This will improve efficiency and make the system more powerful and intelligent for large scale usage.

Key Points:

• Robot can navigate underwater independently
• Reduces manual control requirement
• Can cover larger water area for research
• Improves system intelligence and efficiency

APPENDIX –

A. Hardware Components Used:

COMPONENTS	MODEL VERSION	PURPOSE
Raspberry Pi	Raspberry Pi 4 Model B	Main Processing Unit
Camera Module	USB Webcam Camera	Underwater Visual Monitoring
pH Sensor	Generic pH Electrode	Measures acidity/alkalinity
Depth Sensor	HS-S37A	Measures Depth
ADC Module	ADS1115	Converts Analog Values to Digital
Motor Driver		
DC Motors	4 * DC 3-6V Geared Motor	Movement Demonstration
USB Power Battery	Power Bank 5V	Power Supply
Cardboard Prototype Body	Custom Build	Temporary Enclosure Phase

Table 1: Components used

B. Programming and Software Tools Used:

SOFTWARE/TOOLS	USAGE
Raspberry Pi OS	Operating System for Pi
Python 3	Main Programming Language
Flask Framework	Live Dashboard & Web Interface

GPIO & ADC Libraries	Hardware Control & Sensor Reading
Git Bash	Testing and Debugging Environment

Table 2: Programming & Software Tools Used

C. Python Libraries Utilized:

- Flask
- Time
- OS
- RPi.GPIO
- Board
- Adafruit_ads1*15

D. Testing Environment:

- Testin Conducted in Lab Environment
- Ambient Temperature: ~ 25^o
- Sensors tested in water and Buffer Solution
- Depth Sensor tested using a glass filled with water.

E. Model Image:

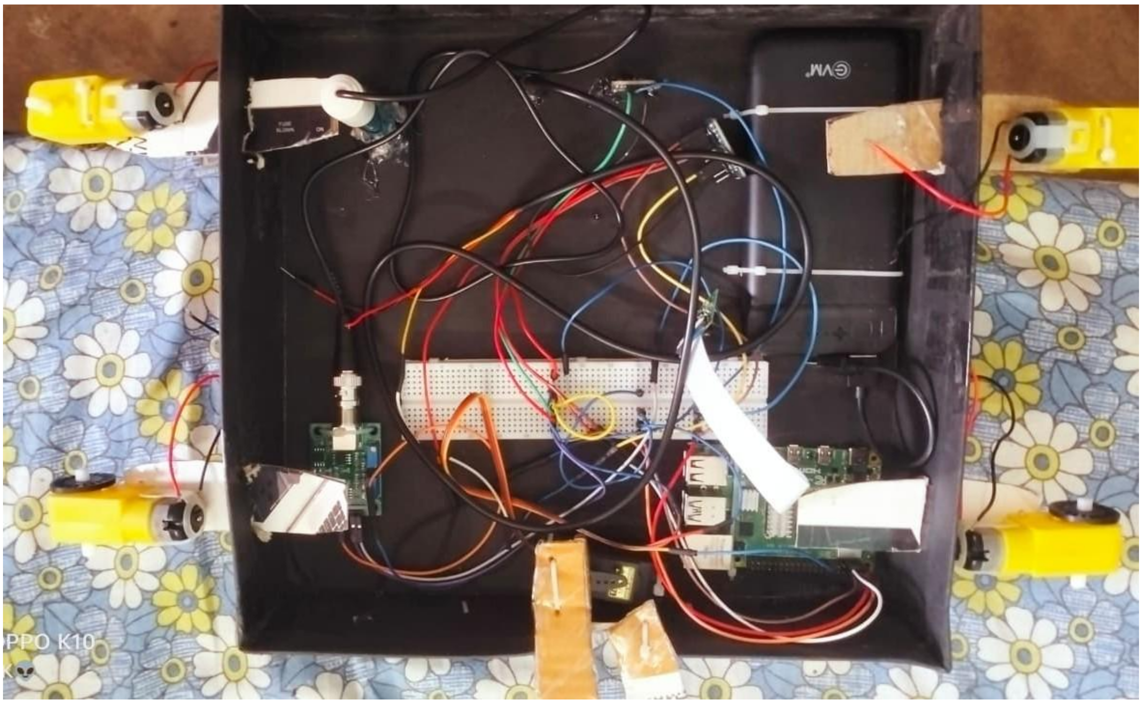

Image 2: Photograph of Current Prototype

F. Code Snippets:

depth_sensor.py

E: > 2nd Group Project > UnderwaterDrone > sensors > depth_sensor.py > ...

```python
from Adafruit_ADS1x15 import ADS1115
import time

adc = ADS1115()
GAIN = 1

def read_depth():
    value = adc.read_adc(1, gain=GAIN)
    voltage = value * (4.096 / 32767.0)
    depth_cm = (voltage / 3.3) * 100  # adjust calibration
    return depth_cm, voltage

if __name__ == "__main__":
    while True:
        depth, volt = read_depth()
        print(f"Voltage: {volt:.3f} V | Depth: {depth:.1f} cm")
        time.sleep(2)
```

Image 3: Photograph of the code for Depth Sensor

Welcome   test_servo.py

test_servo.py > ...

```python
from gpiozero import Servo
from time import sleep

# GPIO pin 17 (Pin 11)
# Custom pulse width range for full 180° movement
servo = Servo(17, min_pulse_width=0.0005, max_pulse_width=0.0025)

print("Testing full-range servo movement (Ctrl+C to stop)")
try:
    while True:
        print("Moving to 0°")
        servo.min()
        sleep(1)

        print("Moving to 90°")
        servo.mid()
        sleep(1)

        print("Moving to 180°")
        servo.max()
        sleep(1)

except KeyboardInterrupt:
    print("\nStopped.")
```

Image 4: Photograph of the code for Servo

```
depth_sensor.py 1    ph_sensor.py 1    temperature_ds18b20.py ×
E: > 2nd Group Project > UnderwaterDrone > sensors > temperature_ds18b20.py > ...
import os, glob, time

os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'

def read_temp_c():
    lines = open(device_file, 'r').readlines()
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines = open(device_file, 'r').readlines()
    temp_output = lines[1].split('t=')[-1]
    return float(temp_output) / 1000.0

if __name__ == "__main__":
    while True:
        print(f"Temperature: {read_temp_c():.2f} °C")
        time.sleep(2)
```

Image 5: Photograph of the code for the Temperature Sensor

BIBLIOGRAPHY –

[1] Lambertini, A., Menghini, M., Cimini, J., Odetti, A., Bruzzone, G., Bibuli, M., Mandanici, E., Vittuari, L., Castaldi, P., Caccia, M., & De Marchi, L. (2022). *Underwater drone architecture for marine digital twin: Lessons learned from SUSHI DROP project.* Sensors, 22(3), 744. https://doi.org/10.3390/s22030744

[2] Nguyen, T. T., Tran, D. M., & Vo, Q. N. (2023). *Artificial intelligence-based robotic systems for underwater exploration and environmental monitoring.* In Advances in Ocean Engineering and Technology (pp. 1–25). Elsevier. https://www.sciencedirect.com/science/article/pii/S2772671123001067

[3] Mantha, B. R., Ding, L., & Roberts, C. (2020). *Digital twins for built environment: An integrated framework for marine and offshore applications.* In Digital Twin Driven Smart Design (pp. 1–25). Academic Press. https://doi.org/10.1016/B978-0-12-819972-5.00001-X

[4] Gray, P. C., Larsen, G. D., & Johnston, D. W. (2022). *Drones address an observational blind spot for biological oceanography. Frontiers in Ecology and the Environment,* 20(7), 413-421. https://doi.org/10.1002/fee.2472

[5] Bukin, O., Proschenko, D., Korovetskiy, D., Chekhlenok, A., Yurchik, V., & Bukin, I. (2021). *Development of the artificial intelligence and optical sensing methods for oil pollution monitoring of the sea by drones. Applied Sciences,* 11(8), 3642. https://doi.org/10.3390/app11083642

[6] Escobar-Sánchez, G., Markfort, G., Berghald, M., Ritzenhofen, L., & Schernewski, G. (2022). *Aerial and underwater drones for marine litter monitoring in shallow coastal waters: Factors influencing item* *detection and cost-efficiency. Environmental Monitoring and Assessment,* 194, 863. https://doi.org/10.1007/s10661-022-10519-5

[7] Sultana, M., Bhattacharya, S., Adhikari, N., Sengupta, D., & De, D. (2024). *Artificial intelligence for sustainable ocean health.* In D. De, D. Sengupta, & T. A. Tran (Eds.), *Artificial Intelligence and Edge Computing for Sustainable Ocean Health* (pp. 3–32). Springer Nature. https://doi.org/10.1007/978-3-031-64642-3_1

[8] Gupta, A., & Tanwar, S. (2024). *Leveraging the power of AI for sustainable oceans. Environmental Sciences Proceedings, 22*(1), 45–57. https://www.mdpi.com/2076-3298/11/10/224

[9] Sharma, S., & Rathi, P. (2025). *Marine drone systems for coastal monitoring: AI-enabled approaches for sustainable tourism and biodiversity management. Transdisciplinary Journal of Environmental and Marine Studies, 4*(2), 1–18. https://www.transdisciplinaryjournal.com/uploads/archives/20250815212916_1.pdf