International Journal of Engineering Business and Social Science
Vol. 2 No. 01, September-October 2023, pages: 790-804
e-ISSN: 2980-4108, p-ISSN: 2980-4272
https://ijebss.ph/index.php/ijebss

Development of an SMS Notification System and Indoor Aquaponic Design

Leonardus Adityo Toto Pratomo[1], Jan Everhard Riwurohi[2]
[1.2] Universitas Budi Luhur, Indonesia
Email: 2211602178@student.budiluhur.ac.id, yan.everhard@budiluhur.ac.id

Keywords
Arduinouno, Sim900, Proteus, Arduinoide, Aquaponic

Abstract
Smart agricultural technology in household environments can be a breakthrough in meeting daily healthy food needs during the pandemic era independently. Proposing inexpensive microcontrollers can nurture vegetables by monitoring SMS notifications on temperature sensors, pH sensors, and water level sensors, along with household fish farming in limited spaces. The research method used Protues and Arduino IDE for prototyping, and the research design involved a literature review on aquaponics and the equipment and their functions. A total of 21 cited journals were selected. The research results include the requirements for growing vegetables using aquaponics and the design of a household horticultural system. The research developed software with three main sensors: pH sensor, temperature sensor, and water level sensor, all connected to an Arduino Uno and SIM900 for sending SMS notifications through the Proteus terminal monitor. The research is not yet perfect because it has not been implemented into the sensor and Arduino Uno and SIM900 device designs. However, the research found that the Arduino code is efficient and has been successfully simulated in Proteus..

1. Introduction

The evolving communication technology that can be integrated with modern agriculture through aquaponic methods, and the limited knowledge of urban farming (Marbun, 2022) make mobility an essential part of monitoring system development (A. Zandamela, 2017). Monitoring and control methods for aquaponics do not have to be present at all times and can be accessed through smartphones (Nina Rahayu, 2018). Economic and ecological feasibility is an innovation in aquaponic systems (Kreiss et al., 2023). Monitoring and proposing a prototype design using Proteus system and Arduino microcontrollers can be a solution for building an aquaponic monitoring system.

Arduino GSM and Arduino Uno are connected to water acidity (pH), room temperature sensors, and water level sensors. If they reach the researcher's predefined minimum thresholds, a message will be sent via short message service (SMS) using the telecontroller on Arduino GSM900 and Arduino Uno. The detected information is sent as a multi-object sensor (RumanaTasnim, 2020) with minimum thresholds set in the Arduino and Proteus programs.

2. Materials and Methods

Proposal for the design and development of indoor aquaponics implementing the research method of Proteus and Arduino prototypes. This method can provide ideas for both creators and users about how the system works or how it functions, as suggested by Gerald D. Everett and Raymond McLeod Jr (Gerald D. Everett, 2007), by identifying the magnitude and sources of development risks that can be reduced by:

1. Testing
2. Perform testing to mitigate identified risks
3. Determine when testing is completed
4. Manage testing as a standard project within the development project

Literature study

Literature studies are divided into 2, namely aquaponic science and microcontroller development,

1. Aquaponic :

Aquaponic is a modern agricultural science by utilizing fisheries and agriculture that is circulated with nutrients from fish aquariums filtered into the circulation of water systems in surrounding plants, several journals from literature reviews among (Pingle, 2020); (Utomo, 2020); (Yulianyahya, 2022)

2. Hardware and Software

pH sensor, a controlled nutrient solution system that can drain bases and acids to control the pH of the solution and store nutrients to control the EC of the solution temperature sensor, monitor and control indoor temperature. To develop heating, ventilation, and air conditioning (HVAC) systems useful for indoor environments, microclimate factors are controlled using several methods such as adaptive control of the outdoor climate

The motor water level sensor is also placed in the tank to drain water from the holding tank to the first tank which is the growbed. The sump tank consists of an ultrasonic sensor to determine the height of the water surface (Baydur, 2021) battery bank battery (to store electrical power), inverter (to convert direct current (DC) to alternative current (AC)), charge controller (to regulate battery charging and protect it from damage), installation using lithium batteries (Hati, 2021)

Peristaltic pumps and these systems, water and fish waste are pumped from fish tanks to plant growing containers. Fish waste is rich in ammonia. Bacteria that grow naturally in closed-loop systems convert ammonia into nitrites and then into nitrates. Nitrates serve as nutrients for plants. After absorption of nutrients by the roots of plants, clean, filtered water flows back into the fish tank. Aquaponic systems consume ninety percent less water than conventional methods due to their ability to reuse water resources

Buzzer The siphon outlet water flow is ON when water flows out of the siphon outlet and the bell will be turned off. But if the water does not flow, the buzzer will be ON. The sound of the bell tells people that water does not flow from the fish tank to the bed. The Servo Program begins by running a servo to rotate and feed the fish by dropping the fish pallet into the fish tank. Arduino Uno for data reception from sensor networks, sensors are controlled with the help of microcontrollers. Sensors like DHT11, DS18b20 and Ultrasonic are controlled with the help of NodeMCU and sensors like TDS and pH are controlled with the help of Arduino UNO microcontroller. (Velasco, 2020)

SIM 900 Gateway uses radio frequency or global system for mobile communication (GSM) or GPRS for communication. Using this architecture, users or farmers or farm managers can remotely monitor and control any action (such as watering) on the farm. The user is informed about the status of the system through the GSM module via SMS. Periodic monitoring of plants based on intelligent computer vision will result in early detection of plant diseases. Therefore, countermeasures to control the disease can be done as early as possible. Demonstrate methodologies adopted for crop health monitoring and disease detection in smart farms (ATTA, 2019, p. 7) (Murugan, Monitoring and controlling the desalination plant using IoT, 2023)

Proteus software and Arduino ideas are used to design and develop systems in the form of prototype prototypes and Arduino that can be developed according to microcontroller needs (N, 2022).

Needs Analysis

At this stage it aims to help researchers to analyze the needs of plant aquaculture with an aquaponic process system, which will facilitate the process of designing a microcontroller device monitoring system because by using this monitoring system, it can provide information in microcontroller devices that are being monitored and find out information about a device condition. The System Development Environment describes the software requirements used and hardware requirements in the development of the monitoring system (Yanes, 2020).

Design and manufacture of agricultural tools

Secure SMS device design using two pieces of hardware

The hardware modules used are Arduino UNO and SIM-900. Arduino UNO serves as the main device that executes and processes operating commands on the system. SIM-900 is used as a communication module in the form of SMS over GSM networks. The output of the overall implementation results and also that serves as a medium of input and output commands on the system used serial Arduino IDE monitor on laptops with baud rate (9600).

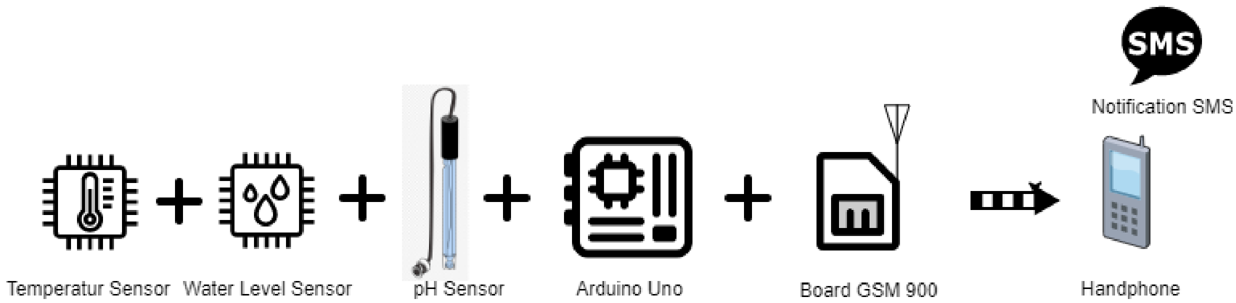


Figure 1 System input and output scheme (P, 2020)

Functional block diagram Figure 2 shows a block diagram of an aquaponic system using an Arduino microcontroller. The developed program is stored in the Arduino Uno microcontroller and is ready to run when needed. Circuit pH sensors, temperature sensors, water sensors, servos, LCD, GSM and peristaltic pumps are connected to Arduino boards. The program and hardware parts of an aquaponics system are combined together to perform usability (Murugan, Monitoring and controlling the desalination plant using IoT, 2023) (Zaini, 2020).

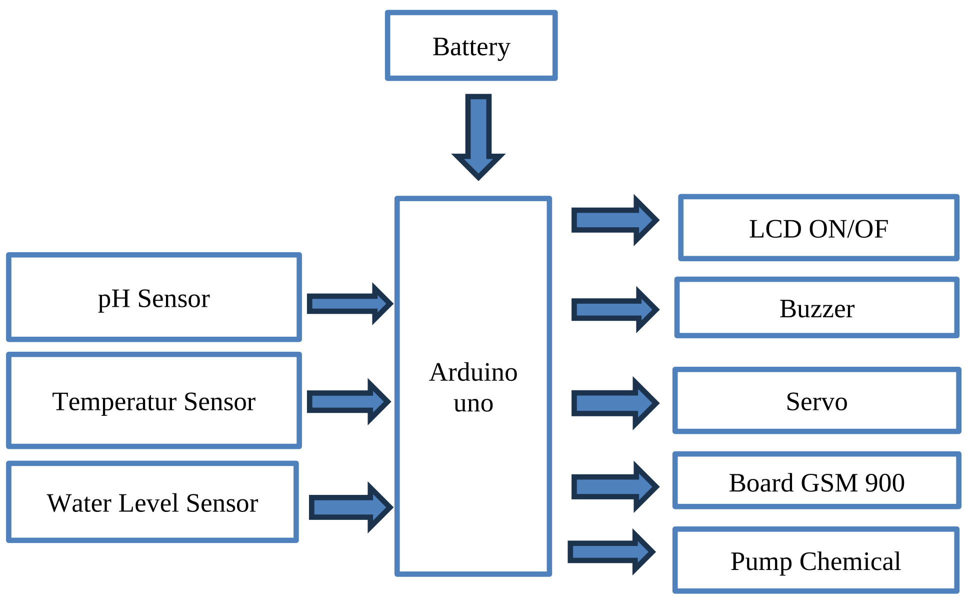


Figure 2 Aquaponics System Block Diagram

Aquaponic groove i.e. **Inputan** is a ph sensor, temperature sensor, water level sensor as input to Arduino uno as a microcontroller, battery based storage battery power small powered battery process Arduino uno is a device based on Arduino uno, **Output** board sim 900 as an sms sender connected to Arduino pump chemical if the ph decreases The buzzer component functions to provide a signal containing sound, If there is a minimum limit of an electric drive servo motor input to drive fish feeding every 12 hours.

The program starts by running a servo to rotate and feed the fish by dropping the fish pallet onto the fish tank. After that the pH, temperature and water sensors will start functioning. When the pH sensor detects the pH value of

the water outside the range, the LCD will display the sensor results and trigger the peristaltic pump to pump the solution chemicals into the water to maintain the pH value. While the water sensor is used to detect the flow of water entering the fish tank through the siphon outlet. If the flow of water through the siphon outlet is stopped, the buzzer will be triggered. Therefore, no water is pumped from the fish tank to the growing place. When pH, temperature and water sensors are out of range, a message will be sent to the mobile phone via GSM modem for notification. The servo is used to feed the fish automatically every 12 hours. (S. A. Z. Murad, 2017)
The series is divided into 4 main parts, namely:

1. Input

Consists of several sensors as input, namely PH sensors, water level sensors, and DS18B20 temperature sensors.

- The PH sensor functions to measure the PH value of the liquid/solution measured by a probe (measuring rod). The sensor will output a voltage that the Arduino reads in the form of an analog signal through pin A0. From the analog signal, Arduino will process it to get the PH value with the equation that has been determined in the program code.
- The Water Level sensor serves to measure the water level with a height scale according to the length of the container, the PH sensor consists of 3pins, namely positive (vcc) to be connected to the middle + or vcc of the Arduino, ground (gnd) to be connected to the negative or gnd path of the circuit, and the analog output pin connected to pin A1 of the Arduino. The output of the water level sensor is an analog voltage whose value depends on the measured water level on the sensor. The analog voltage is sent to the Arduino via pin A1 and converted to a scale of 0-100% water level level.
- The DS18B20 Temperature Sensor is a digital temperature sensor that uses the OneWire interface. This sensor is capable of measuring temperature in a wide temperature range with high precision. Consists of 3 pins, namely positive (vcc) to be connected to the + or vcc of the Arduino, ground (gnd) to be connected to the negative or gnd path of the circuit, and the output data pin to be connected to pin 5 of the Arduino. The output of the DS18B20 is a digital signal with a onewire interface. Arduino will read the digital data through pin 5 to get the temperature reading value from the sensor.

2. Processing

The Processing Algorithm / controller of the tool created is Arduino Uno. Arduino is a microcontroller that can be programmed with C language to function according to sensor reading needs or output output through motion, sound, light, and display actuators. In the circuit made Arduino is connected to the power supply as a circuit power source, connected to sensors to receive input, and finally connected to output devices in the form of LCD, GSM modules, pump motors, servos, and also buzzers. (Maryam Jawadwala, 2021)

3. Output

Output from the circuit made LCD i2c, GSM module, water pump motor, servo, and also buzzer.

- The i2c 16x2 LCD serves to display characters in 16 columns and 2 lines format. The communication interface uses i2c where only 2 SCL and SDA lines are needed to communicate with Arduino. The pins of this i2c LCD are positive (vcc) to be connected to the + or vcc of the Arduino, ground (gnd) to be connected to the negative or gnd path of the circuit, the SCL pin to be connected to the A4 pin of the Arduino, and the SDA pin to be connected to the A5 pin of the Arduino. The output of the Arduino will be converted to the LCD controller to control the display on the LCD according to the program running on the Arduino.
- GSM SIM900 module serves to send messages (SMS) over cellular networks. The communication between the SIM900 is Serial communication where it takes 2 pins to send data (TX) and pins to receive data (RX). To communicate with Arduino. The pins of this i2c LCD are positive (vcc) to be connected to the + or vcc of the Arduino, ground (gnd) to be connected to the negative or gnd path of the circuit, the TX pin to be connected to pin 3 of the Arduino, and the RX pin to be connected to pin 2 of the Arduino.
- The Water Pump Motor will be used to provide fluid supply to adjust the PH value according to plant needs. This water pump motor cannot be controlled directly from the Arduino output pin, so a mosfet driver is needed to control the turn and off of the motor according to the output of the Arduino. The water pump pin is connected to the mosfet output pin (drain) and the positive pin of the battery directly. While the pin for the source will be connected to the negative path (gnd) of the circuit and the gate as input control is connected to pin 6 arduino.
- The servo will later function as a valve controller for plant sprinklers. This servo is controlled with PWM voltage discharged from the Arduino via pin 9. This servo itself consists of 3 pins, namely positive (vcc) to be connected to the + or vcc of the Arduino, ground (gnd) to be connected to the negative or gnd path of the circuit, and the input pin connected to pin 9 of the Arduino.

- The buzzer serves to release a sound / warning when there is a situation that does not match the parameters specified in the Arduino program code. This buzzer consists of 2 pins, namely positive which will be connected to pin 4 of the Arduino and negative pin which is connected to the gnd path of the circuit.

4. Power Supply

The power supply in the circuit is sourced from a 2s lithium battery with a maximum voltage of 8.4 volts. Because the operating voltage requirement of Arduino and other modules is 5v, a 5v voltage regulator is needed. So that the battery circuit is connected to the Arduino VIN pin which is a voltage input line that will be offered to the internal voltage regulator first before being channeled to the microcontroller chip or other modules and components in the circuit.

Circuit simulation is made with proteus 8.15 simulation software. The circuit made is in the form of a series of Aquaponic control tools consisting of an Arduino mictocontroller on and several modules and components.

List of modules / components:

1. Arduino Uno
2. Modul GSM SIM900
3. LCD I2C 16x2
4. Sensor PH
5. Sensor Level Air
6. Sensor Suhu DS18B20
7. Water Pump dan Mosfet Driver
8. Servo
9. Buzzer
10. Battery Lithium

The overall cable design connection in the proteus simulation is shown in figure 3:

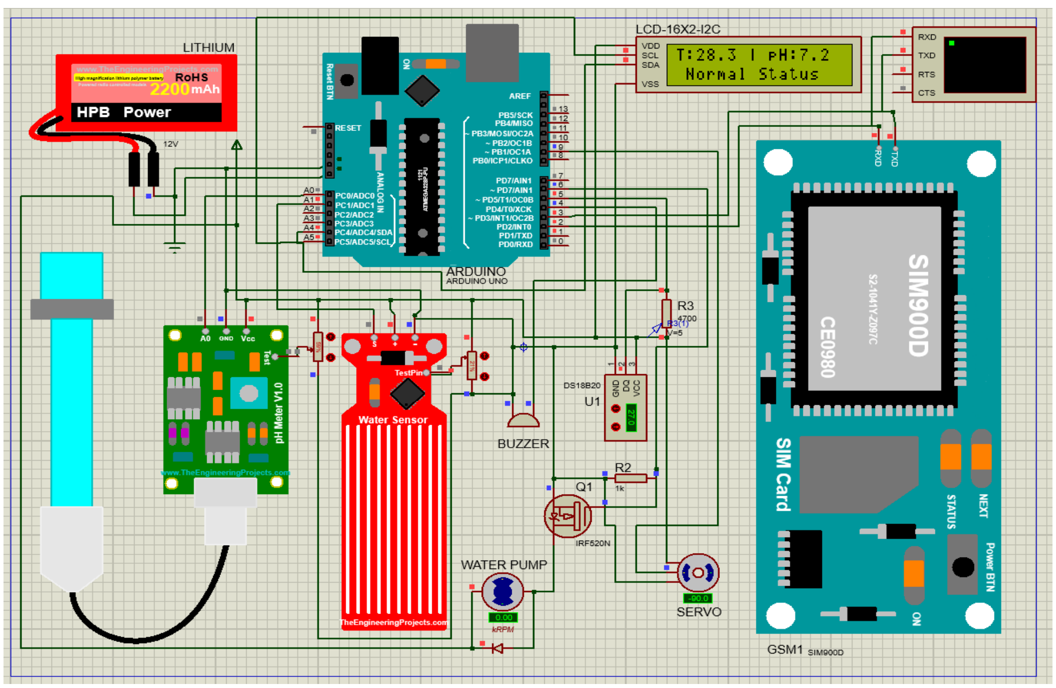


Figure 3 The wiring arrangement is as follows

Tool Name	Pin Name	Connect to tool/pin
Arduino Uno	VIN	Battery +
	GND	Battery -
	5v	Sensor and Modul
Modul GSM	VCC	VCC Arduino (Undisplayed)

	GND	GND (Undisplayed)
	TX	Pin 13 Arduino
	RX	Pin 11 Arduino
LCD I2C 16x2	VCC	VCC Arduino
	GND	GND
	SCL	Pin A4 Arduino
	SDA	Pin A5 Arduino
Sensor PH	VCC	VCC Arduino
	GND	GND
	AO	Pin A0 Arduino
Sensor Water Level	VCC	VCC Arduino
	GND	GND
	AO	Pin A1 Arduino
Sensor Suhu DS18B20	VCC	VCC Arduino
	GND	GND
	DQ (Data)	Pin 5 Arduino dan Resistor 4k7
Resistor 4k7	-	VCC Arduino
	-	Pin 5 Arduino dan Pin DQ DS18B20
	Echo	Pin 6 Arduino
Water Pump Motor	Pin +	Baterai + and Pin Katoda Dioda
	Pin -	Mosfet Drain and Pin Katoda Dioda
Mosfet 520N	Gate	Pin 6 Arduino
	Source	Battery - / GND
	Drain	Pin – Water Pump Motor
Servo	VCC	VCC Arduino
	GND	GND
	Signal In	Pin 9 Arduino
Buzzer	Pin +	Pin 4 Arduino
	Pin -	GND
Baterai Lithium	Pin +	Pin VIN Arduino
	Pin -	GND

Figure 4

3. Results and Discussions

Results on code plans in the system (Novita Dwi Susanti, 2021)

Explanation of the Arduino program code lines of the Aquaponics system created

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SoftwareSerial.h>
#include "DFRobot_RGBLCD1602.h"
#include <Servo.h>
```

Figure 5

In the figure 5 section of this code snippet, several libraries needed to communicate with the DS18B20 temperature sensor, a 16x2 RGB LCD, SoftwareSerial to communicate with the SIM900 module, and a library to control the servo are loaded.

```
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
SoftwareSerial SIM900(3, 2);
DFRobot_RGBLCD1602 lcd(16,2);
Servo myservo;
```

Figure 6

In figure 6 of this section, global objects and variables are declared. Such objects include DS18B20 temperature sensors, SIM900 modules, 16x2 RGB LCDs, and servos.

```
#define ONE_WIRE_BUS 5
#define pompa 6
#define buzzer 4
```

Figure 7

Figure 7 This section defines the arduino pin used for OneWire connection (temperature sensor) which is pin 5, the water pump (mosfet driver) which is pin 6, and the buzzer connected to pin 4.

```
unsigned long int avgValue;
float b;
int buf[10];
int temp;
int timer=0;
bool state=0;
uint8_t jam = 12;
unsigned long jeda;
```

Figure 8

Figure 8 Next is the declaration of variables to be used in the program, where each variable has a data type that corresponds to the type of value to be stored or used.

unsigned long int avgValue;: Declares the variable avgValue as a non-negative integer that can store an average value.float b;: Declares variable b as a fractional number (float).int buf[10];: Declares a buf array of length 10 that can store the value of integer.int temp;: Declares the temp variable as a number bulat.int timer=0;: Declares a timer variable and initializes it with a value of 0.bool state=0;: Declares a state variable as a boolean data type (true or false) and initializes it with a value of 0 (false). uint8_t hours = 12: Declares the hour variable as a nonnegative integer with the data type uint8_t and initializes it with the value 12 unsigned long pause. Declares the pause variable as a non-negative integer that can store a period of time in milliseconds.

```
ISR(TIMER0_COMPA_vect){
  timer++;
}
```

Figure 9

Figure 9 Next is part of the Interrupt Service Routine (ISR) used in Arduino.

The code will be run as a timer counter which will later be used for a timer to give water to plants that will trigger the servo to rotate valse every 12 hours to turn on the flow of water to plants.

An explanation for each line of Arduino program code in the setup function that will be executed only once at the beginning of the Arduino (tool) is turned on.

```
void setup(void) {
  TCCR0A=(1<<WGM01);
  OCR0A=0xF9;
  TIMSK0|=(1<<OCIE0A);
  sei();
  TCCR0B|=(1<<CS01);
  TCCR0B|=(1<<CS00);
```

Figure 10

Figure 10, In this section, some settings and configurations are used to set Timer/Counter 0 (TC0) on the microcontroller.

TCCR0A, OCR0A, TIMSK0, TCCR0B are registers used to configure TC0. WGM01 bit assignment (Waveform Generation Mode) and OCR0A value (Output Compare Register A) set TC0 operation mode as CTC (Clear Timer on Compare Match) and determine comparison value when TC0 reaches OCR0A.

The OCIE0A bit is set to activate the interrupt at the appropriate ratio. sei() is used to enable global interrupts. CS01 and CS00 bits are set on TCCR0B to set the TC0 prescaler and enable TC0.

```
Serial.begin(9600);
SIM900.begin(9600);
sensors.begin();
lcd.init();
```

Figure 11.

Figure 11 in this section, initializing serial communication via Serial.begin(9600) for communication with other devices via the serial port. SIM900.begin(9600) is used to initiate communication with the SIM900 module. sensors.begin() is used to initiate communication with the temperature sensor using the OneWire library. lcd.init() is used to initialize the LCD using the DFRobot_RGBLCD1602 library.

```
  lcd.setRGB(0, 0, 0);
  pinMode(pompa, OUTPUT);
  pinMode(buzzer, OUTPUT);
  digitalWrite(pompa, LOW);
  digitalWrite(buzzer, LOW);
  myservo.attach(9);
  myservo.write(0);
  jeda= jam * 3600000;
  lcd.clear();
}
```

Figure 12.

Figure 12, in this section, preliminary settings for the LCD, output pins for the pump and buzzer, and servo starting positioning are performed. lcd.setRGB(0, 0, 0) sets the color of the lcd. Set the value to 0 all because the one used is a single-color lcd. pinMode(pump, OUTPUT) and pinMode(buzzer, OUTPUT) set the pump and buzzer pins as output pins. digitalWrite(pump, LOW) and digitalWrite(buzzer, LOW) set the output of the pump and buzzer pins to LOW (off). Myservo.attach(9) associates the servo object with pin 9 of the Arduino. myservo.write(0) sets the initial position of the servo to an angle of 0 degrees.The pause variable is initialized with the result of multiplying the hour by 3600000. This is used as the desired interval time to control the servo of water flow later. lcd.clear() is used to clean the LCD display of previously displayed characters.

An explanation of the line of code in the 'loop()' function where the code in this 'loop()' will be executed endlessly repeatedly.

```
void loop(void) {

  if(timer>=jeda) {
    myservo.write(120);
    delay(1000);
    myservo.write(0);
    timer=0;
  }
```

Figure 13.

Figure 13, In this section, the program will check whether the 'timer' (a variable that continues to increase with the help of interrupts) has reached 'pause' (a certain time). If yes, then the servo will move from position 0 degrees to 120 degrees for 1 second, then return to position 0 degrees (drain water to the plant. After that, the 'timer' is set back to 0.

```
sensors.requestTemperatures();
float tempC = sensors.getTempCByIndex(0);
Serial.print("Temperature : ");
Serial.print(tempC);
```

Figure 14.

Figure 14, In this section, the temperature sensor is instructed to read the temperature and that temperature value is stored in the variable 'tempC'. Furthermore, this temperature value will be printed through serial communication.

```
int water = analogRead(A1);
Serial.print(" | Water value: ");
Serial.print(water);
```

Figure 15.

Figure 15, In this section, the reading of an analog value from pin A1 (which is connected to the water sensor) is performed using the 'analogRead()' function. The value will be printed through serial communication.

```
for(int i=0;i<10;i++) {
  buf[i]=analogRead(A0);
  delay(10);
}
```

Figure 16.

Figure 16, In this section, an analog value of pin A0 (connected to the pH sensor) is read 10 times using the 'analogRead' function. Each value read is stored in a 'buf[]' array, with a 10 millisecond lag between each read.

```
for(int i=0;i<9;i++) {
  for(int j=i+1;j<10;j++) {
    if(buf[i]>buf[j]) {
      temp=buf[i];
      buf[i]=buf[j];
      buf[j]=temp;
    }
  }
}
```

Figure 17.

Figure 17, In this section, ascending the values in the 'buf[]' array are sorted using the bubble sort method. This is done to retrieve the average value of the 6 largest values present on the array 'buf[]`.

```
avgValue=0;
for(int i=2;i<8;i++)
  avgValue+=buf[i];
```

Figure 18.

Figure 18, In this section, the sum of the 6 largest values of the array 'buf[]' (ranging from index 2 to index 7) is performed and the result of the sum is stored in the variable 'avgValue`.

```
float phValue=(float)avgValue*5.0/1024/6;
phValue=3.5*phValue;
Serial.print(" | pH:");
Serial.print(phValue,1);
Serial.println(" ");
```

Figure 19.

Figure 19, In this section, the average of the 6 largest values is converted into pH values using the specified formula. This pH value is printed through serial communication.

```
lcd.setCursor(0,0);
lcd.print("T:");
lcd.setCursor(2,0);
lcd.print("    ");
lcd.setCursor(2,0);
lcd.print(tempC,1);

lcd.setCursor(7,0);
lcd.print("|");

lcd.setCursor(9,0);
lcd.print("pH:");
lcd.setCursor(12,0);
lcd.print("   ");
lcd.setCursor(12,0);
lcd.print(phValue,1);
```

Figure 20.

Figure 20, In this section, the temperature and pH values will be displayed on the LCD using the 'DFRobot_RGBLCD1602' library. The LCD will display the information "T: [temperature value] | pH: [pH value]" in the first row with a predefined column position.

```
String pesan = "";
uint8_t stat=0, stp=0, stw=0, stt=0;
```

Figure 21.

Figure 21, In this section, the variable 'message' is used to store messages to be sent via SMS. The variables 'stat', 'stp', 'stw', and 'stt' are used as flags to determine the condition that causes the abnormal state.

```
if(phValue<7){
  stat=1;
  stp=1;
  analogWrite(pompa, 255);
  digitalWrite(buzzer, HIGH);
  delay(50);
  analogWrite(pompa, 0);
  digitalWrite(buzzer, LOW);
  delay(50);
  pesan += "The pH value is below threshold. ";
}
```

Figure 22.

Figure 22, In this section, it is checked whether the pH value ('phValue') is below 7. If yes, then 'stat' is set to 1 to indicate abnormal status, 'stp' is set to 1 to indicate low pH, and the water pump and buzzer will be activated momentarily. Additionally, the message "pH value below threshold." will be added to the variables' messages`.

```
if(water<50) {
  stat=1;
  stw=1;
  digitalWrite(buzzer, HIGH);
  delay(50);
  digitalWrite(buzzer, LOW);
  delay(50);
  pesan += "Water content below threshold. ";
}
```

Figure 23.

Figure 23, In this section, a check is made whether the reading value of the water sensor ('water') is below 50. If yes, then 'stat' is set to 1 to indicate abnormal status, 'stw' is set to 1 to indicate low water content, and the buzzer will be activated momentarily. Additionally, the message "Water Content below threshold." will be added into the 'message' variable`.

```
if(tempC>30) {
  stat=1;
  stt=1;
  digitalWrite(buzzer, HIGH);
  delay(50);
  digitalWrite(buzzer, LOW);
  delay(50);
  pesan += "Temperature over limit. ";
}
```

Figure 24.

Figure 24, In this section, checking whether the temperature value ('tempC') is above 30. If yes, then 'stat' is set to 1 to indicate abnormal status, 'stt' is set to 1 to indicate high temperature, and the buzzer will be activated momentarily. In addition, the message "Temperature exceeded limit." will be added into the variable 'message'.

```
if(stat==1) {
  lcd.setCursor(0,1);
  lcd.print("                ");

  if(stp==1){
    lcd.setCursor(0,1);
    lcd.print("PH!");
  }
  if(stw==1){
    lcd.setCursor(5,1);
    lcd.print("WTR!");
  }
  if(stt==1){
    lcd.setCursor(11,1);
    lcd.print("TEMP!");
  }

  SIM900.print("AT+CMGF=1\r");
  delay(100);
  pesan += "\n";
  SIM900.println("AT+CMGS=\"+6281284190595\"");
  delay(100);

  SIM900.println(pesan);
  delay(100);

  SIM900.println((char)26);
  delay(100);
  SIM900.println();
  delay(5000);
}
```

Figure 25.

Figure 25, In this section, it is checked whether the 'stat' is 1, which indicates an abnormal status. If yes, then the LCD will display a message according to the conditions that cause abnormal status of PH, water level, and temperature (warning status is "PH!", "WTR!", or "TEMP!").

Next, using the SIM900 module, the program will send an SMS message containing information of the abnormal condition to the specified phone number.

The message will end with ASCII character 26 (End of Text). There is also a delay before and after sending the message to ensure the message is delivered properly.

```
  else {
    lcd.setCursor(0,1);
    lcd.print(" Status Normal  ");
  }
}
```

Figure 26.

Figure 26, In this section, if there is no abnormal status (the 'stat' value is still 0), then the LCD will display a "Normal Status" message to indicate that the condition is normal.

Virtual Results

Monitor success testing is divided into 4 displays consisting of 3 sensors, and all three results

Virtual pH Monitor

pH Sensor testing is carried out to determine the feasibility of this sensor for research. For this reason, this circuit will be tested according to the needs of the program that the researchers designed, namely the minimum acidity limit distance that can be detected by the sensor is below 7 and the sensor will display on the Arduino IDE series monitor that "The pH Value is below threshold.". If less than that, the serial monitor will display a description of the minimum acidity limit of the object detected by the sensor. As in figure 27

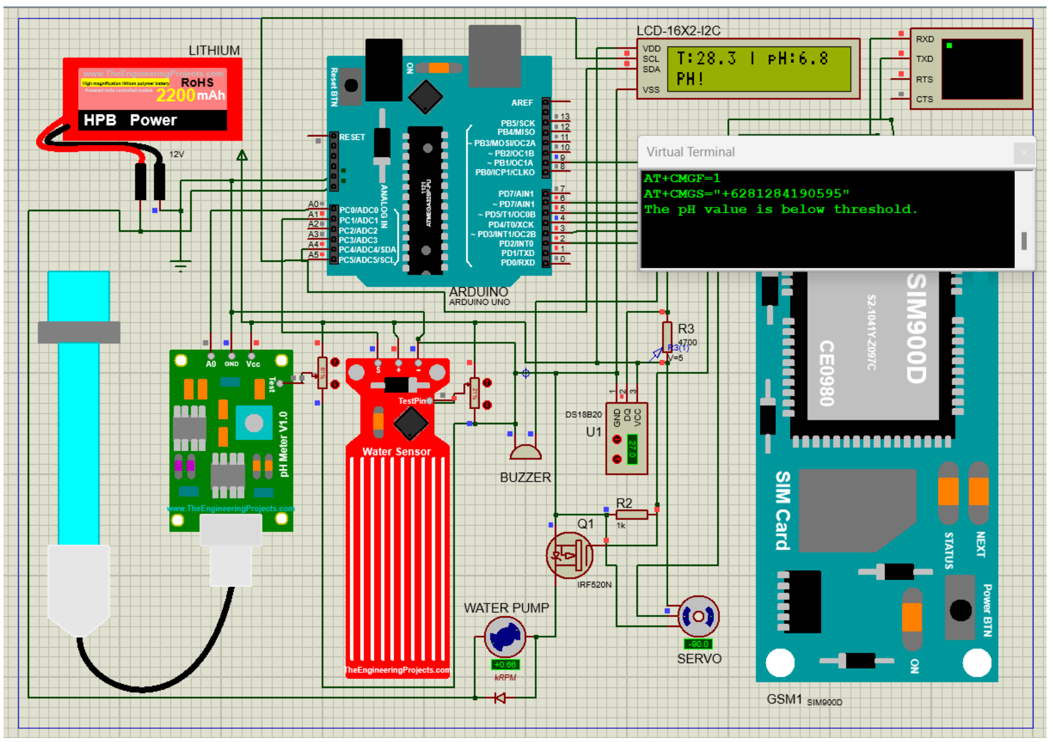


Figure 27 Virtual Monitor Water Level

From the test results of the water level sensor, it can be concluded that the system works where the output of all conditions works according to the actual conditions, namely the sensor can detect the minimum height and the serial displays the distance of the object with water level provisions with water content below threshold notifications as shown in figure 28 below.

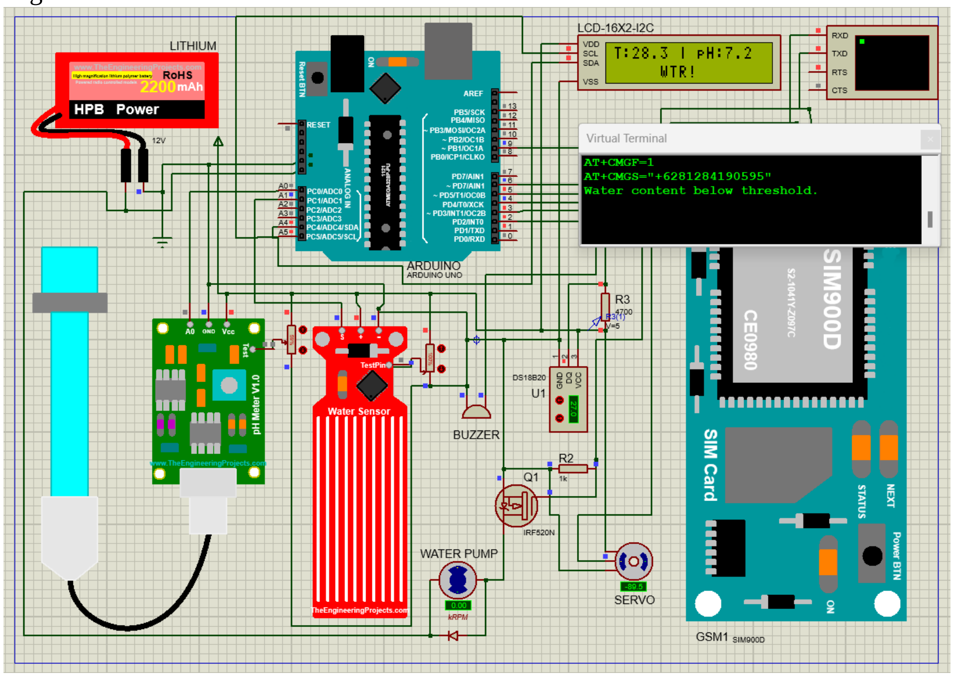
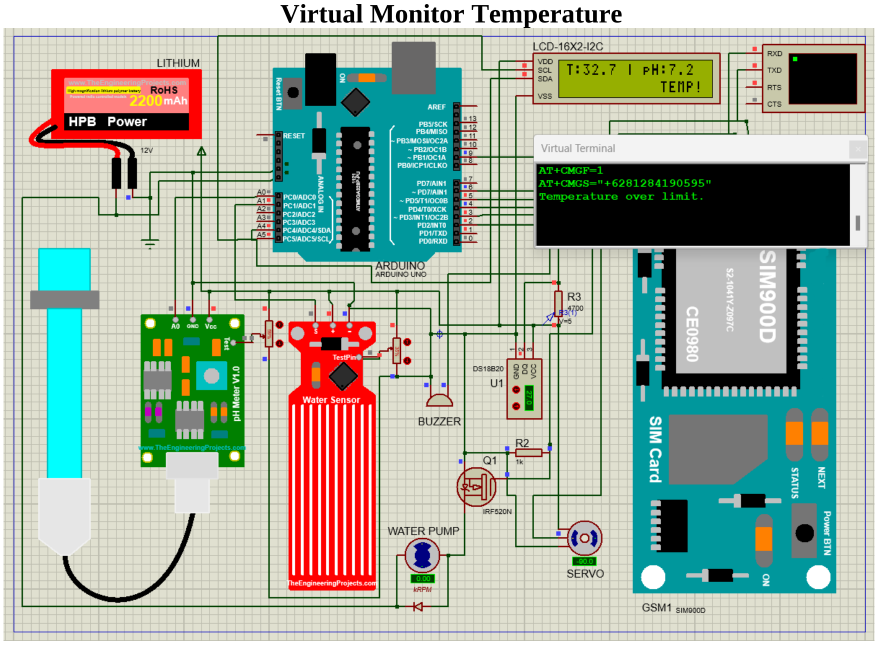


Figure 28
Virtual Monitor Temperature



Temperature sensor testing is done to determine room temperature, because it affects the fertility of plants with an aquaponics system. The maximum temperature limit that can be detected by the sensor is below 30 Celsius and the sensor will display on the Arduino IDE series monitor that "Temperature over limit.". If the maximum temperature feels hot above the specified one, the serial monitor will display a description of the maximum room temperature limit of the object detected by th.

The overall results as in 30, all notified into the monitoring of SMS testing on the proteus simulation virtual terminal

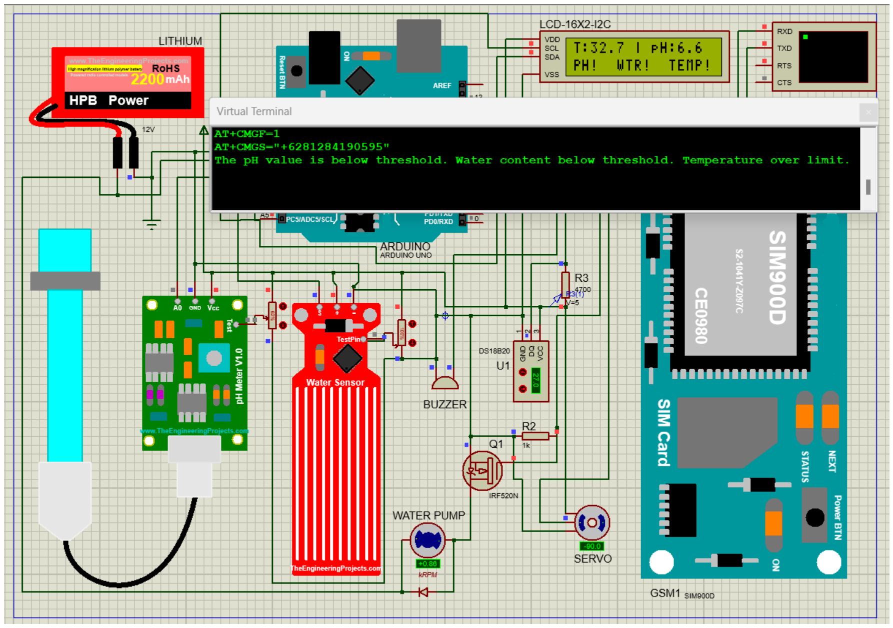


Figure 30

4. Conclusion

Based on the results of the discussion analysis in this study, several conclusions can be drawn as follows: Based on the tests that have been carried out on the analysis of how the Arduino Uno Microcontroller and Water level Sensor work, pH Sensor and Temperature Sensor in aquaponic design in proteus and Arduino ideas for the application of tool installation and in accordance with the initial purpose of the researcher, where this sensor can be used to detect pH levels, Water level and room temperature that have an output or output in the form of buzzers and SMS notifications, if the sensor detects objects at a certain level. The water level sensor has a distance range of 50 percent so that in this test it can run well with the water level in the container, the pH sensor has a range below 7 so that in this test it can run well if the pH in the water is at number 7. The temperature sensor has a range above 30 Celsius, so in this test it can run well if the temperature in the room is below 30 Celsius. This system cannot work if there is no electricity, and still does not include it in the installation of equipment, and the level of sensor range both indoor temperature sensors, underwater pH sensors and water level sensors in small reservoirs, and affect the conditions of different residential environments.

5. References

A. Zandamela, A. (2017). An Approach to Smart Home Security System Using Ardunio. *Electrical Engineering : An International Journal*, 1.

ATTA, M. J. (2019). DEVELOPMENT OF FISH POND MONITORING AND CONTROL SYSTEM. *FUTMinna Institutional Repository*, 2.

Baydur, C. (2021). ARM Based Smart Water Meter with LoRa for Soilless Agriculture Application. *European Journal of Science and Technology*, 1.

Gerald D. Everett, R. M. (2007). Software Testing Testing Across the Entire. *IEEE PRESS*, 45.

Hati, A. J. (2021). Smart Indoor Farms: Leveraging Technological Advancementsto Power a Sustainable Agricultural Revolution. *AgriEngineeringReviewSmart* , 1.

Kreiss, C. (2023). Ökologische Intensivierung der Aquakultur in Europa (GAIN). *Thünen- Institut für Seefischerei2023/21*, 1.

Marbun, N. (2022). HIDROPONIK DAN AQUAPONIK SEDERHANA: SOLUSI BUDIDAYA SAYUR DI LAHAN TERBATAS DALAM SKALA RUMAH TANGGA. *10.31604/jpm.v5i8.2834-2843*, 1.

Maryam Jawadwala, Y. P. (2021). Aquaponics for Agriculture using IOT. *International Journal of Engineering Research & Technology (IJERT)*, 3.

Murugan, T. M. (2023). Monitoring and controlling the desalination plant using IoT. *www.sciencedirect.com/journal/measurement-sensors*, 3.

Murugan, T. M. (2023). Monitoring and controlling the desalination plant using IoT. *ScienceDirect Measurement: Sensors*, 2.

N, K. K. (2022). STM32-based IoT Monitoring System for an Indoor Plant. *INTELLIGENT SYSTEMS AND APPLICATIONS IN* , 2.

Novita Dwi Susanti, D. S. (2021). Design and Implementation of Water Quality Monitoring System (Temperature, pH, TDS) in Aquaculture Using IoT at Low Cost. *6th International Conference of Food, Agriculture, and Natural Resource (IC-FANRES 2021)*, 3.

P, M. M. (2020). Secure SMS Using Pseudo-Random Bit Generator Based on Chaotic Map, and AES on Arduino UNO Board and SIM 900 Module. *IEEE Xplore*, 3.

Pingle, Y. (2020). Aquaponics for Agriculture using IOT. *International Journal of Engineering Research & Technology (IJERT)*, 1.

RumanaTasnim. (2020). Design of aSmart Biofloc Monitoring andControlling System using IoT. *Journal of Engineering Advancements*, 1.

S. A. Z. Murad, A. H. (2017). Design of aquaponics water monitoring system using Arduino microcontroller. *AIP Conference Proceedings*, 3.

Utomo, D. T. (2020). Perancangan Budidaya Pakcoy dan Lobster Menggunakan Sistem Aquaponic Berbasis Internet of Things . *JEECOM*, 1.

Velasco, J. (2020). Internet of things-based (IoT) inventory monitoring refrigerator using arduino sensor network. *Indonesian Journal of Electrical Engineering and Computer Science*, 1.

Yanes, A. R. (2020). Towards automated aquaponics: A review on monitoring, IoT, and smart systems. *Journal of Cleaner Production*, 10.

Yulianyahya, R. W. (2022). Optimalisasi Feses/Kotoran Dari Budidaya Ternak Ikan Sebagai Sumber Nutrisi Tanaman Dengan Sistem Aquaponik. *Pengmasku*, 1.

Zaini, M. (2020). PERANCANGAN SISTEM MONITORING TEGANGAN, ARUS DAN FREKUENSI PADA PEMBANGKIT LISTRIK TENAGA MIKROHIDRO BERBASIS IOT. *Jurnal Teknik Elektro* , 5.