IOT-SMART WATER FOUNTAINS

PROJ_201029_TEAM_1

St. MOTHER THERESA ENGINEERING COLLEGE

APPROVED BY AICTE & AFFILIATED TO ANNA UNIVERSITY CHENNAI
THOOTHUKUDI, TAMIL NADU

NAME: K.KARPAKA PRIYA

ROLL NO:953021106031

Email id: Karpagapriya.ug.21.ec@mtec.ac.in

Title: "Smart Water Fountains: Innovative Solutions for Sustainable Hydration"

Abstract:

Smart water fountains represent a promising advancement in the realm of sustainable hydration and water management. These intelligent fixtures incorporate cutting-edge technology to provide convenient, efficient, and environmentally responsible access to clean drinking water. This abstract highlights the key aspects of smart water fountains, including their design, functionality, and the potential benefits they offer to individuals and communities. Smart water fountains leverage sensors, filtration systems, and data analytics to monitor water quality and usage in real-time. They are equipped with features such as touchless dispensing, bottle refilling, and user-friendly interfaces that promote accessibility and hygiene. These fountains also contribute to reducing single-use plastic bottle waste and conserving water resources by promoting the use of reusable containers.

In addition to their sustainability features, smart water fountains offer valuable insights into water consumption patterns, enabling informed decision-making for water infrastructure planning and resource management. This abstract explores the various applications of smart water fountains, from public spaces and educational institutions to businesses and urban planning, emphasizing their potential to enhance community well-being and environmental conservation.

As the world faces increasing challenges related to water scarcity and plastic pollution, smart water fountains represent a progressive solution that aligns with the principles of sustainability and technological innovation. This abstract provides an overview of their potential to revolutionize the way we access and consume water, promoting a more eco-conscious and convenient approach to hydration.

Objectives

The objectives of creating a smart water fountain are to leverage IoT (Internet of Things) technology to enhance the functionality, efficiency, and user experience of a traditional water fountain. Here are some key objectives when developing a smart water fountain:

1. Remote Monitoring and Control: Allow users to remotely monitor and control the water fountain through a web or mobile application. This enables convenient control and management of the fountain's operation from anywhere.
2. Efficient Water Management: Implement features such as water level sensors to ensure the fountain is not overfilled or runs dry. This helps in conserving water resources and maintaining the fountain's proper functioning.
3. Energy Efficiency: Use smart scheduling and sensors to turn on the fountain only when needed, reducing energy consumption. For instance, the fountain can turn on during specific hours or when motion is detected.
4. Water Quality Monitoring: Include sensors to monitor water quality, such as pH and turbidity sensors. This can help ensure that the water in the fountain is clean and safe.
5. Temperature Control: Implement temperature sensors and control mechanisms to maintain an optimal water temperature. This is crucial for certain types of fountains or aquatic features.
6. Security and Alerts: Set up alerts and notifications for unusual events, such as low water levels, water quality issues, or system malfunctions. Users can receive alerts through email or push notifications.
7. Data Logging and Analysis: Collect data from various sensors for analysis. This data can be used to understand usage patterns, detect trends, and make informed decisions about maintenance and improvements.
8. Customization and Personalization: Allow users to customize the fountain's operation, such as changing water flow patterns, lighting effects, and water display sequences.
9. Integration with Smart Home Systems: Make the smart water fountain compatible with popular smart home platforms like Amazon Alexa, Google Home, or Apple HomeKit. This enables voice control and integration with other smart devices in the home.
10. Educational and Aesthetic Value: Enhance the aesthetic appeal of the fountain by adding dynamic lighting, patterns, or water displays that can be programmed or synchronized with music. Additionally, the project can serve as an educational tool to learn about IoT and automation.
11. Weather Adaptation: Include weather sensors to adjust the fountain's operation based on environmental conditions. For example, the fountain can be turned off during rainy or windy weather.

12. Water Conservation: Promote water conservation by implementing features that minimize water waste, such as rainwater harvesting, recirculation systems, and water filtration.
13. Maintenance Alerts: Provide maintenance alerts to ensure the fountain is kept in good working condition. This includes alerts for pump maintenance, filter replacement, or cleaning schedules.
14. User Engagement: Engage users by offering features like interactive displays, QR code interactions, or social media sharing options.
15. Scalability and Upgradability: Design the system with the potential for future upgrades and expansions, allowing the addition of new sensors and features as needed.

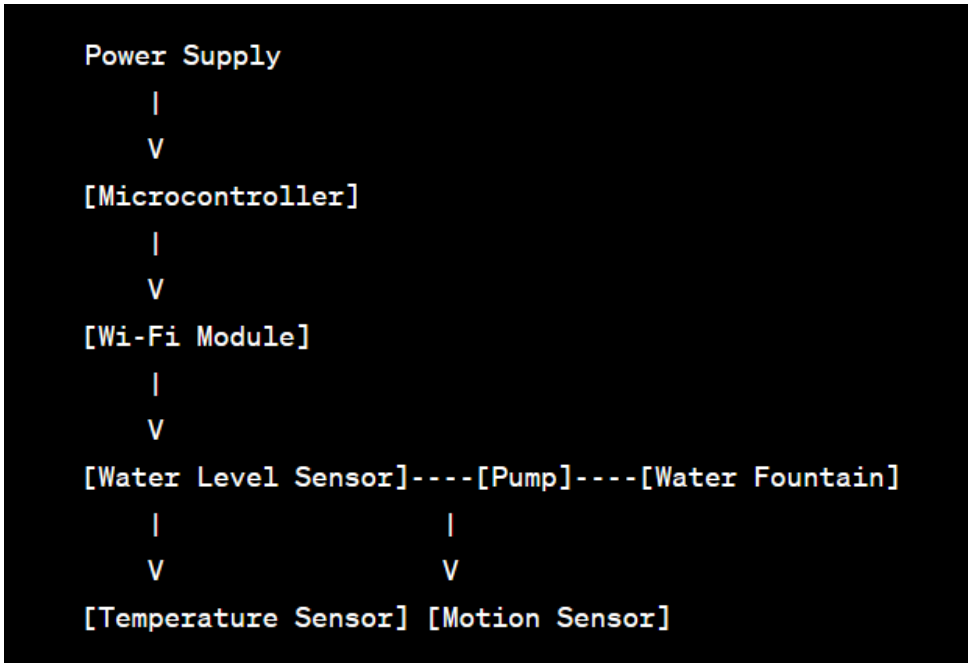
Components You'll Need:

1. Water Fountain: You can either purchase a pre-made water fountain or design your own.
2. Microcontroller: You'll need a microcontroller to control the fountain's operation and connect it to the internet. Raspberry Pi, Arduino, or ESP8266/ESP32 are popular choices.
3. Sensors: Depending on the features you want, consider adding sensors like water level sensors, temperature sensors, and motion sensors.
4. Relays or Motor Controllers: For controlling the pump and any other moving parts of the fountain.
5. Camera (Optional): If you want to include a video feed of the fountain, you can add a camera.
6. Power Supply: Ensure you have a stable power source for your devices.
7. Internet Connectivity: You'll need Wi-Fi or another internet connectivity method to connect your IoT device to the internet.

Project Steps:

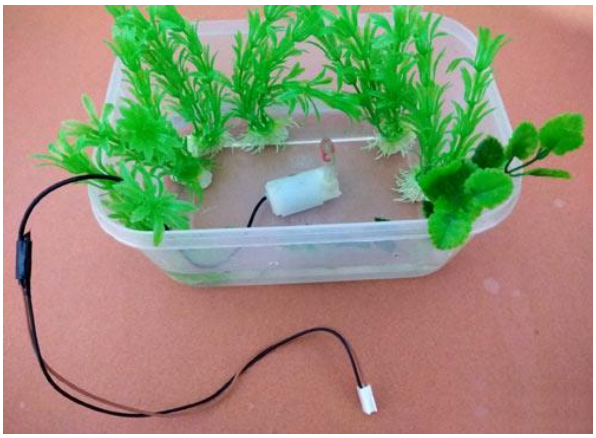
1. **Design the Fountain**: If you're building your own fountain, design it according to your preferences. Make sure you have a container for the water, a pump, and any decorative elements.
2. **Setup Microcontroller**: Depending on the chosen microcontroller, set it up with the necessary software and connect it to your local network.
3. **Sensor Integration**: If you're using sensors, connect them to the microcontroller. For example, a water level sensor can help you monitor the water level, and a temperature sensor can monitor the water temperature.
4. **Control Logic**: Write the code to control the fountain's operation based on the data from the sensors. For example, you can program the fountain to turn on when the water level is low or when motion is detected.
5. **Remote Access**: Implement remote access and control, allowing you to control the fountain from a web or mobile app. You can use platforms like Blynk, MQTT, or your own custom interface.
6. **Data Monitoring**: Set up data logging and monitoring. You can use cloud services like AWS IoT, Google Cloud IoT, or a Raspberry Pi as a server for storing and analyzing data.
7. **Notifications**: Configure alerts or notifications when certain conditions are met, like when the water level is critically low or the water temperature is too high.
8. **Optional Camera Feed**: If you're including a camera, set up a live video feed that you can access remotely.
9. **Enclosure**: Build or purchase an enclosure to protect the electronics from water and weather.
10. **Testing**: Test the system to ensure it works as expected. Make adjustments as necessary.
11. **Deployment**: Install the smart water fountain in your desired location.
12. **Maintenance**: Regularly check and maintain the system, making sure sensors are working correctly and the fountain is clean and functional.
13. **Scaling**: If desired, you can add more features like voice control, water quality monitoring, or automatic water filling.

Flow Chart

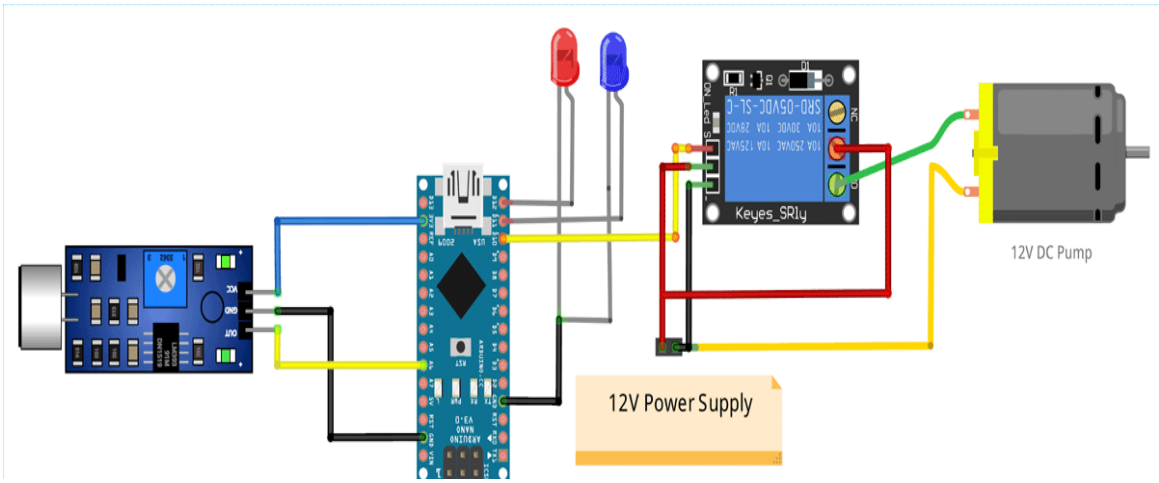


Material required

1. Arduino Nano
2. Sound sensor Module
3. 12V Relay Module
4. DC Pump
5. LEDs
6. Connecting wires
7. Vero board or Breadboard



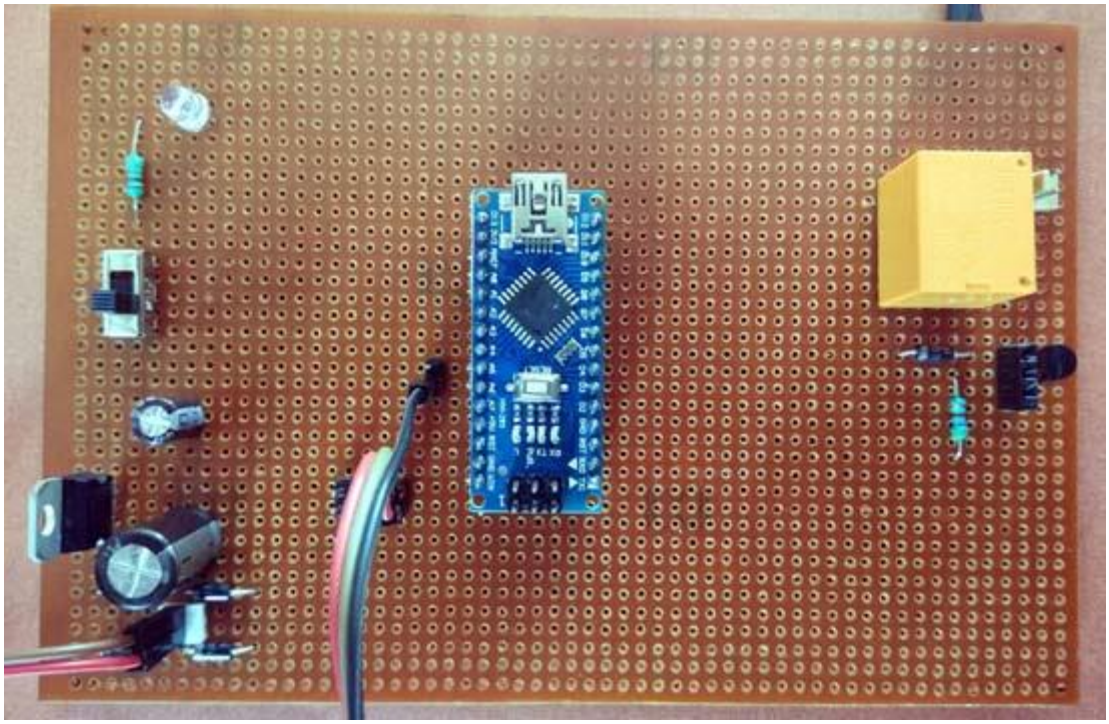
Musical Water Fountain Circuit Diagram



As shown in the above **musical fountain circuit diagram**, the sound sensor is powered with 3.3V supply of Arduino Nano and the output pin of the sound sensor module is connected to the analog input pin (A6) of Nano. You can use any of the analog pin, but make sure to change that in the program. The relay module and DC pump is powered by an external 12VDC power supply as shown in the figure. The input signal of relay module is connected to digital output pin D10 of Nano. For lighting effect I chose two different colours of LED and connected them to two digital output pins (D12, D11) of Nano.

Here the Pump is connected in such a way that when a HIGH pulse is given to the input of Relay module, the COM contact of the relay is get connected to the NO contact and the current gets a closed circuit path to flow across the pump to activate the water flow. Otherwise the pump will remain OFF. The HIGH/LOW pulses are generated from Arduino Nano depending on the sound input.

After soldering the complete circuit on perfboard, it will look like below:



Programming Arduino Nano for Dancing Fountain

The complete program of this **Arduino water fountain project** is given at the bottom of the page. But here I am just explaining that by parts for better understanding:

The first part of the program is to declare the necessary variables for assigning pin numbers that we are going to use in the next blocks of program. Then define a constant REF with a value which is the reference value of for the sound sensor module. The assigned value 700 is the bytes equivalent value of the output electrical signal of the sound sensor.

```
int sensor = A6;
int redled = 12;
int greenled = 11;
int pump = 10;

#define REF 700
```

In ***void setup* function** we have used *pinMode* function to assign the INPUT/OUTPUT data direction of the pins. Here sensor is taken as INPUT and all other devices are used as OUTPUT.

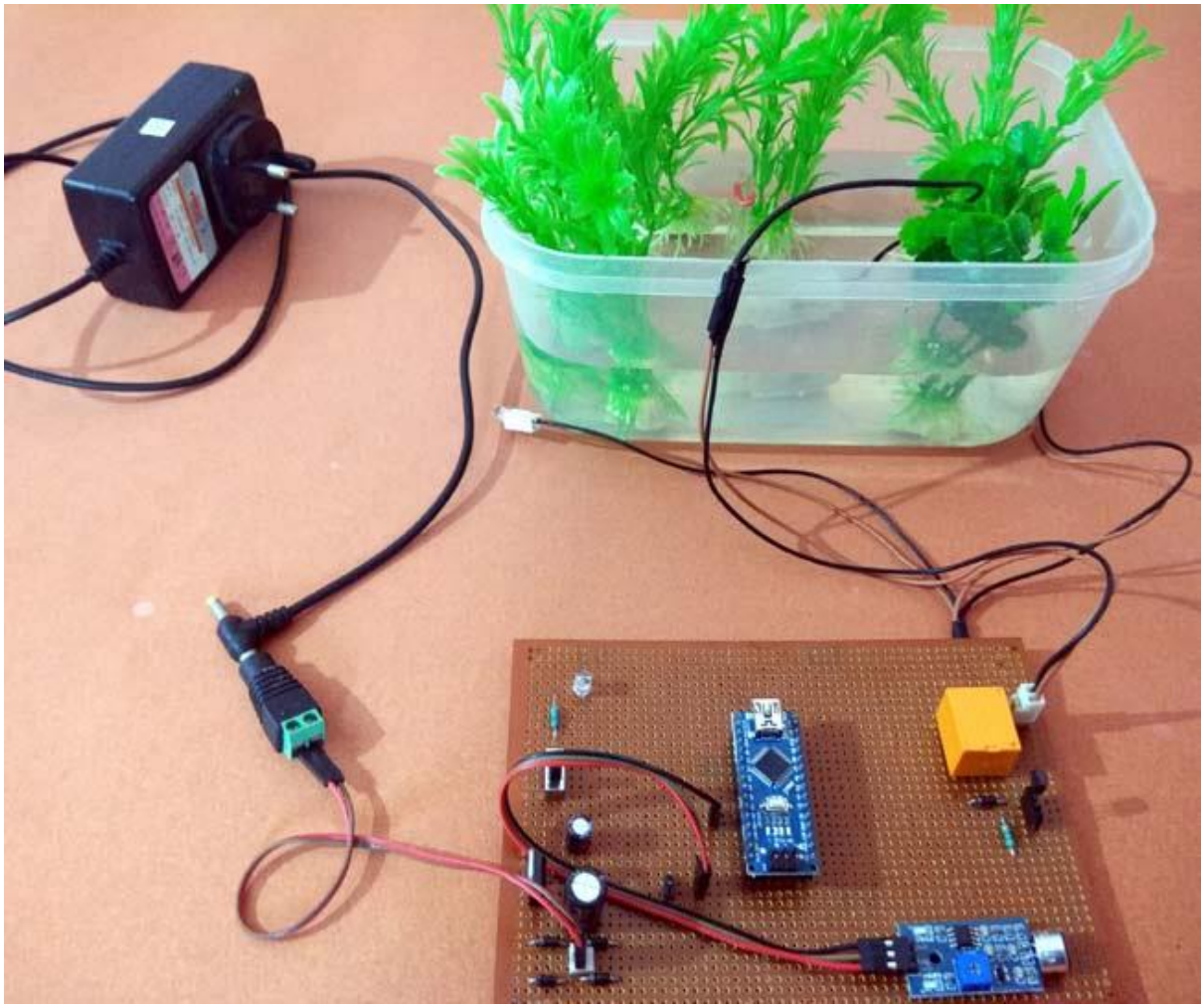
```
void setup()
{
  pinMode(sensor,INPUT);
  pinMode(redled,OUTPUT);
  pinMode(greenled,OUTPUT);
  pinMode(pump,OUTPUT);
}
```

Inside the infinite *loop*, *analogRead* function is called which readout the analog value input from the sensor pin and stores it in a variable *sensor_value*.

```
int sensor_value = analogRead (sensor);
```

In the final part an *if-else* loop is used to compare the input analog signal with the Reference value. If it is greater than the reference, then all the output pins are given HIGH output so that all the LEDs and Pump are activated, else everything remains OFF. Here we have also given a delay of 70 Milliseconds to distinct the ON/OFF time of the Relay.

```
if (sensor_value>REF)
 {
  digitalWrite(greenled,HIGH);
  digitalWrite(redled,HIGH);
  digitalWrite(pump,HIGH);
  delay(70);
 }
 else
 {
  digitalWrite(greenled,LOW);
  digitalWrite(redled,LOW);
  digitalWrite(pump,LOW);
  delay(70);
 }
```



This is how this Arduino controlled Water Fountain works, complete code with a working video is given below.

Simulation Code

```
int sensor = A6;

int redled = 12;

int greenled = 11;

int pump = 10;

#define REF 700

void setup()

{

  pinMode(sensor,INPUT);

  pinMode(redled,OUTPUT);

  pinMode(greenled,OUTPUT);

  pinMode(pump,OUTPUT);



}

void loop()

{



  int sensor_value = analogRead (sensor);



  if (sensor_value>REF)

  {

    digitalWrite(greenled,HIGH);
```

```
    digitalWrite(redled,HIGH);
    digitalWrite(pump,HIGH);
    delay(70);
   }
   else {
    digitalWrite(greenled,LOW);
    digitalWrite(redled,LOW);
    digitalWrite(pump,LOW);
    delay(70);
   }
}
```

Python Script code

```
import RPi.GPIO as GPIO
import time

# Define the GPIO pins
sensor_pin = 6
red_led_pin = 12
green_led_pin = 11
pump_pin = 10

# Set the reference value
REF = 700

# Configure GPIO settings
GPIO.setmode(GPIO.BCM)
GPIO.setup(sensor_pin, GPIO.IN)
GPIO.setup(red_led_pin, GPIO.OUT)
GPIO.setup(green_led_pin, GPIO.OUT)
GPIO.setup(pump_pin, GPIO.OUT)

try:
    while True:
```

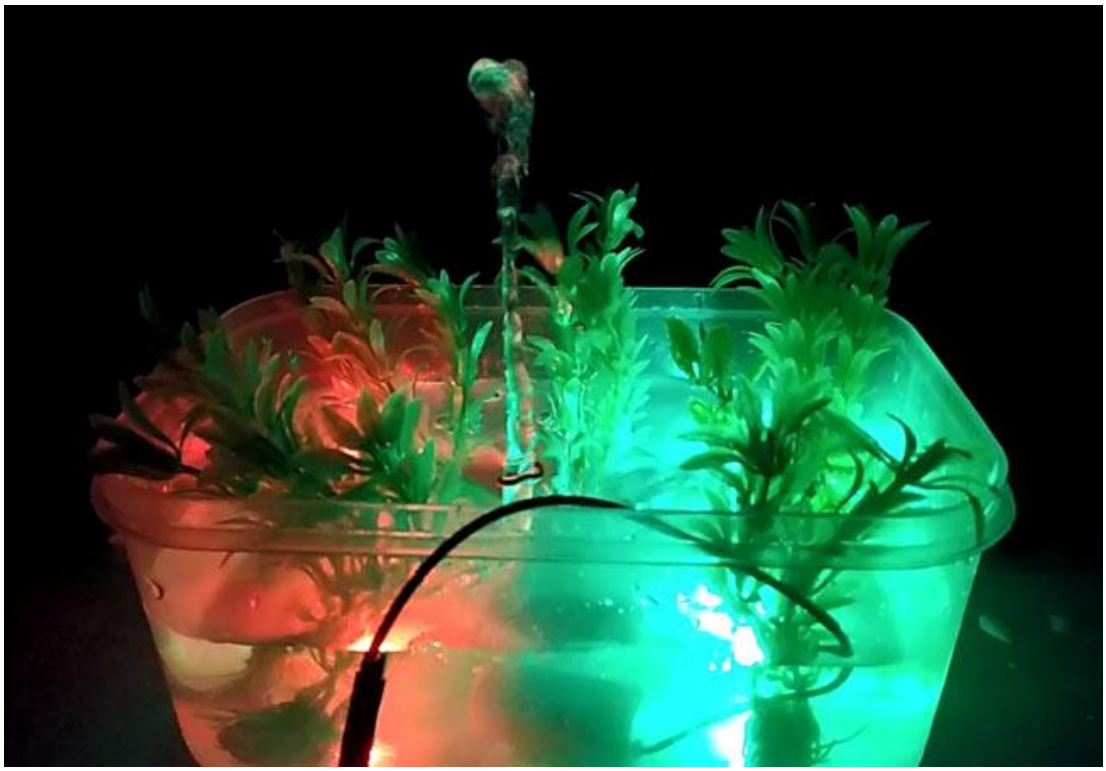
```
        sensor_value = analogRead(sensor_pin)  # Use an appropriate function to
read analog values (not GPIO input)


        if sensor_value > REF:
            GPIO.output(green_led_pin, GPIO.HIGH)
            GPIO.output(red_led_pin, GPIO.HIGH)
            GPIO.output(pump_pin, GPIO.HIGH)
            time.sleep(0.07)
        else:
            GPIO.output(green_led_pin, GPIO.LOW)
            GPIO.output(red_led_pin, GPIO.LOW)
            GPIO.output(pump_pin, GPIO.LOW)
            time.sleep(0.07)

except KeyboardInterrupt:
    pass

# Clean up the GPIO settings
GPIO.cleanup()
```

Output



Conclusion

In conclusion, a smart water fountain project aims to enhance the traditional concept of a water fountain by incorporating IoT technology. By enabling remote control, efficient water management, energy savings, water quality monitoring, and customization, a smart water fountain provides both practical benefits, such as water conservation and efficient maintenance, and aesthetic enhancements, including dynamic lighting and interactive displays. This convergence of technology and artistry transforms a simple water feature into a connected, intelligent, and engaging element that can be appreciated in various contexts, from home gardens to public spaces. Smart water fountains have emerged as a transformative solution for promoting sustainable hydration and responsible water management. These innovative fixtures blend cutting-edge technology with eco-conscious design to address pressing challenges related to water quality, plastic waste, and resource conservation.

By incorporating features like real-time water quality monitoring, touchless dispensing, and bottle refilling, smart water fountains offer convenient and hygienic access to clean drinking water in various settings, from public spaces to educational institutions and businesses. They not only reduce the reliance on single-use plastic bottles but also encourage the use of reusable containers, contributing to a significant reduction in plastic waste.

Moreover, smart water fountains serve as valuable data collection points, offering insights into water consumption patterns that can inform better decision-making for water infrastructure planning and resource allocation. This data-driven approach has the potential to revolutionize urban planning, ensuring more efficient and sustainable water management practices in our communities. As we confront the growing challenges of water scarcity and environmental sustainability, smart water fountains represent a tangible and effective solution. They align with the principles of environmental conservation and technological innovation, offering a way to promote responsible water usage while enhancing the quality of life in our cities. The adoption of smart water fountains not only ensures access to clean and safe drinking water but also paves the way for a more sustainable and eco-friendly future, where our daily actions align with the broader goal of preserving our planet's vital resources.