

Project Title

Smart Water Fountain

PHASE-2:INNOVATION

RASPBERRY PI-3



Introduction:

Creating a smart water fountain IoT project using a Raspberry Pi 3 can be a fun and educational project. This project will allow you to remotely control and monitor a water fountain using the Raspberry Pi 3 and IoT technologies. Here's a step-by-step guide to get you started: Creating a smart water fountain IoT project using a Raspberry Pi 3 can be a fun and educational project. This project will allow you to remotely control and monitor a water fountain using the Raspberry Pi 3 and IoT technologies. Here's a step-by-step guide to get you started:

Components Needed:

1. Raspberry Pi 3
2. MicroSD card with Raspbian OS installed
3. Water pump
4. Water reservoir
5. Relay module
7. Tubing and nozzle for the fountain
8. Jumper wires

9. Power supply for the pump
10. IoT platform (e.g., ThingSpeak, AWS IoT, or Google Cloud IoT)

STEPS TO FOLLOW:

Step 1: Set Up Your Raspberry Pi

1. Install Raspbian OS on the microSD card and boot up your Raspberry Pi.
2. Connect the Raspberry Pi to your local Wi-Fi network using the built-in Wi-Fi or a USB Wi-Fi dongle.
3. Update and upgrade your system:

```
bash
```

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

Step 2: Assemble the Hardware

1. Connect the water pump to the relay module. The relay module will be used to control the pump.
2. Set up the water reservoir and fountain tubing. Make sure the water pump is submerged in the reservoir, and the nozzle is positioned where you want the water to flow.
3. Do the required connections to rpi ,to control pump with relayl.



Step 3: Write the Python Code

You'll need Python code to control the water pump. You can use libraries like RPi.GPIO for GPIO control and python for remote control

Here's a simplified example to get you started:

```
import RPi.GPIO as GPIO
```

```
import time
```

```
# Set the GPIO pin connected to the relay
```

```
relay_pin = 17 # Example: GPIO17
```

```
# Initialize GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setup(relay_pin, GPIO.OUT)

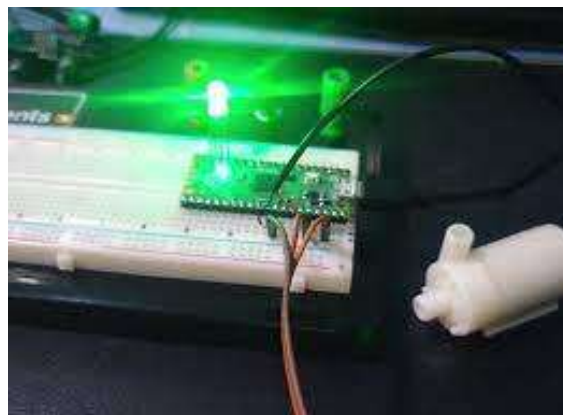
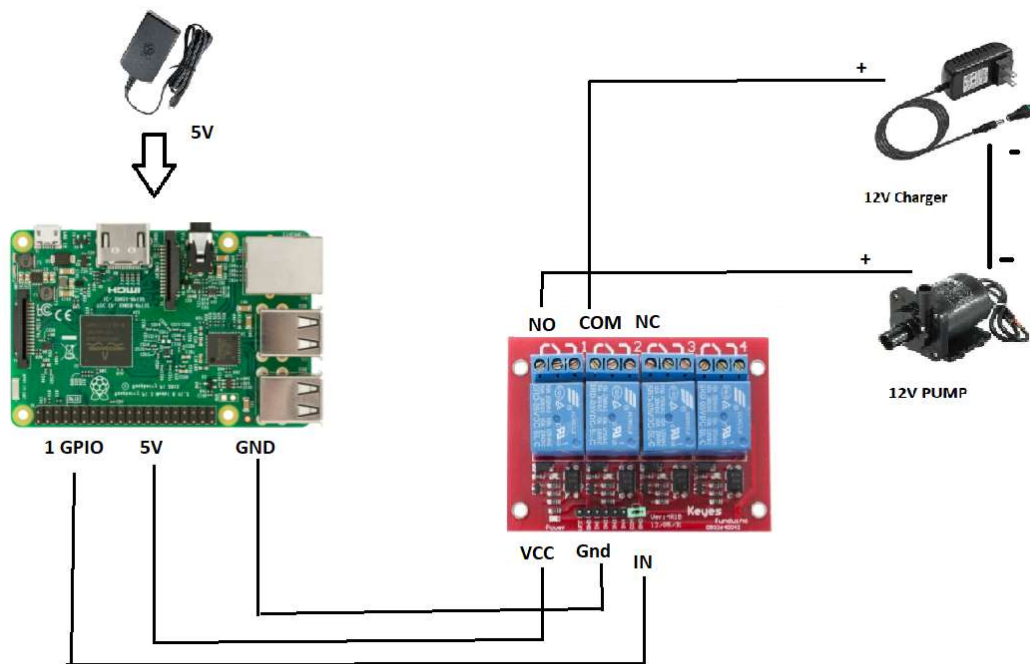
def turn_on_pump():
    GPIO.output(relay_pin, GPIO.HIGH)
    print("Pump is turned on.")

def turn_off_pump():
    GPIO.output(relay_pin, GPIO.LOW)
    print("Pump is turned off.")

try:
    while True:
        turn_on_pump()
        time.sleep(5) # Run the pump for 5 seconds
        turn_off_pump()
        time.sleep(10) # Wait for 10 seconds before running again

except KeyboardInterrupt:
    # Turn off the pump and cleanup GPIO on Ctrl+C
    turn_off_pump()
    GPIO.cleanup()
```

CIRCUIT DIAGRAM:



APPLICATIONS:

➤ Public Parks and Recreation Areas:

- Smart fountains in parks can provide convenient access to drinking water.
- They can include features like bottle-filling stations, which promote reusable water bottles and reduce plastic waste.

➤ Educational Institutions:

- Schools, colleges, and universities can use smart fountains to provide safe and clean drinking water to students and staff.

- These fountains can track water quality and usage, ensuring a healthy environment.

➤ Commercial Buildings:

- Office buildings and commercial complexes can install smart fountains for employees and visitors.

- They can integrate with access cards for touchless dispensing.