ENERGY EFFICIENTRE STROOM LIGHT SYSTEM

INTRODUCTION:

Design and implement a smart lighting system for public restrooms that optimizes energy usage by using IoT sensors and controls to adjust lighting levels based on occupancy, natural light conditions, and user preferences.

COMPONENTS & KEYS

Occupancy Sensors:

Install motion sensors in restroom areas to detect user presence.

Utilize infrared or ultrasonic sensors to accurately detect motion and occupancy.

Light Sensors:

Deploy light sensors to measure natural light levels in the restroom.

Use data from light sensors to adjust artificial lighting accordingly.

IoT Hub/Platform:

Connect all sensors and lighting fixtures to a central IoT hub or platform for data collection and control.

Consider cloud-based platforms for remote monitoring and control.

Smart Lighting Fixtures:

Replace existing lighting fixtures with energy-efficient LED lights that can be dimmed or adjusted.

Select fixtures with IoT connectivity or retrofit existing fixtures with smart controllers.

Control Algorithms:

Develop algorithms that consider occupancy and natural light levels to adjust lighting intensity.

Implement control logic that gradually dims or brightens lights to avoid abrupt changes.

Energy Consumption Monitoring:

Implement energy meters to monitor and record energy usage by the lighting system.

Use this data for performance analysis and optimization.

PROJECT PHASE:

Planning & research

Define the specific requirements and objectives of the project.

Research energy-efficient lighting fixtures and IoT platforms.

Sensor Deployment and Connectivity:

Install occupancy and light sensors in restroom areas.

Ensure reliable connectivity to the IoT platform.

IoT Platform Development:

Set up the central IoT platform for data collection and control.

Develop software for data analysis and lighting control algorithms.

Fixture Installation or Retrofitting:

Replace existing fixtures with energy-efficient LED lights or retrofit them with smart controllers.

Control Interface Development:

Create a user interface (e.g., mobile app or control panels) for users to manually adjust lighting

.

Testing and Optimization:

Test the system in real-world restroom conditions.

Optimize lighting control algorithms based on sensor data and user feedback.

Energy Monitoring and Reporting:

Monitor and record energy consumption by the lighting system.

Generate reports on energy savings and system performance.

User Education and Training:

Educate restroom users on how to use the smart lighting system.

Provide training to facility staff on system maintenance and adjustments.

BENEFITES

Significant energy savings through optimized lighting control.

Enhanced user experience with comfortable and well-lit restrooms.

Reduced maintenance costs due to longer-lasting LED fixtures.

Environmental benefits from reduced energy consumption and lower carbon emissions.

CONCLUSION

An IoT-based energy-efficient restroom lighting system not only contributes to sustainability but also offers cost savings and improved user comfort. This project provides valuable insights into the integration of IoT technology in everyday facilities for resource conservation.

**Innovation for Energy-Efficient Restroom Light System**

The Energy-Efficient Restroom Light System is a cutting-edge technology designed to reduce energy consumption in public restrooms. This system leverages advanced sensors and lighting controls to optimize energy use while maintaining a comfortable and safe environment for users. This article provides a comprehensive overview of this innovative technology, including its design, implementation, and benefits.

To bring the innovative energy-efficient restroom light system to market, potential partnerships and collaborations can be explored with companies and organizations that share a similar vision for sustainable and eco-friendly solutions. Some potential partners include:

* Eco-friendly building and construction companies
* Sustainable energy companies and utilities
* Green technology startups and acceleration

### Challenges with Traditional Restroom Lighting

Traditional restroom lighting systems are often inefficient and wasteful, using up too much energy and raising electricity bills. Additionally, these systems require frequent maintenance and replacement, adding to the overall cost of operation.

### Our Innovative Solution

Our energy-efficient restroom lighting system is designed to address these challenges. By utilizing advanced LED technology and motion sensors, our system reduces energy consumption and minimizes waste. It also requires less maintenance and has a longer lifespan than traditional lighting systems.

### Benefits of Our System

Our energy-efficient restroom lighting system offers numerous benefits, including reduced energy costs, lower maintenance requirements, and longer lifespan. It also provides a more comfortable and convenient experience for restroom users, with automatic on/off functionality and adjustable brightness levels.

### Technical Specifications

Our energy-efficient restroom lighting system is designed to meet the highest technical standards. It features advanced LED technology, motion sensors, and adjustable brightness levels. It is also easy to install and compatible with a wide range of restroom configurations.

### Next Steps

If you are interested in learning more about our energy-efficient restroom lighting system and how it can benefit your organization, please contact us to schedule a consultation. We would be happy to answer any questions you may have and provide you with a customized quote.

### High Energy Consumption

Traditional restroom lighting systems consume a significant amount of energy, resulting in high electricity bills for businesses and organizations.



### Low Efficiency

Traditional lighting systems are often inefficient, with light being wasted in areas where it is not needed. This can lead to a lack of visibility in important areas such as mirrors and sinks.



Benefits of Our System:

### Energy Savings

Our energy-efficient lighting system can save up to 70% on energy costs compared to traditional restroom lighting.

### Improved Comfort

Our system provides a comfortable and well-lit environment for restroom users, improving their overall experience.

### Lower Maintenance Costs

Our long-lasting LED lights require less maintenance and replacement, reducing overall maintenance costs.

**Smart Rest Room On Automatic Room Lights using Arduino and PIR Sensor**

# Automatic Room Lights using Arduino and PIR Sensor

In this project, we will see the Automatic Room Lights using Arduino and PIR Sensor, where the lights in the room will automatically turn ON and OFF by detecting the presence of a human.

Such Automatic Room Lights can be implemented in your garages, staircases, bathrooms, etc. where we do not need continuous light but only when we are present.

Also, with the help of an automatic room light control system, you need not worry about electricity as the lights get automatically off when there is no person.

So, in this DIY project, we have implemented Automatic Room Lights using Arduino and PIR Sensor.

### Components Required for Automatic Room Lights using Arduino

* Arduino UNO
* PIR Sensor
* 5V Relay Module (Relay Board)
* LED
* 100Ω Resistor (1/4 Watt)
* Connecting Wires
* Breadboard
* Power Supply

**Steps to Create the IoT Water Overflow Control Project:**

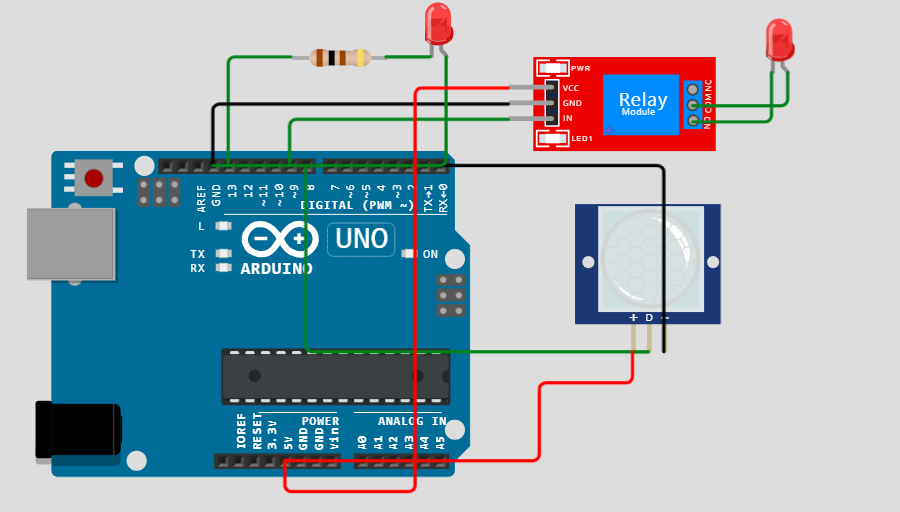
**Setup Hardware**

a. Connect the Relay module sensor to the microcontroller. Ensure you follow the sensor's datasheet and the microcontroller's pinout.

b. Connect the actuator to the microcontroller. The type of actuator will depend on your application

c. Connect any required power supplies to the microcontroller and actuator.

Refer below image

****

**Python program Here whe will use in raspberry pi**

You'll need a Python script to read data from the water level sensor, make decisions based on the water level, and control the actuator. Here's a simplified example using Raspberry Pi and RPi.GPIO:

Python

import time

import RPi.GPIO as GPIO

in1 = 9

sensor = 8

led = 13

t = 0

GPIO.setmode(GPIO.BCM)

GPIO.setup(in1, GPIO.OUT)

GPIO.setup(sensor, GPIO.IN)

GPIO.setup(led, GPIO.OUT)

GPIO.output(in1, GPIO.HIGH)

GPIO.output(led, GPIO.LOW)

while time.time() < 13:

GPIO.output(led, GPIO.HIGH)

time.sleep(0.05)

GPIO.output(led, GPIO.LOW)

time.sleep(0.05)

GPIO.output(led, GPIO.LOW)

while True:

GPIO.output(in1, GPIO.HIGH)

GPIO.output(led, GPIO.LOW)

if GPIO.input(sensor) == GPIO.HIGH:

t = int(time.time() \* 1000)

while time.time() \* 1000 < (t + 5000):

GPIO.output(in1, GPIO.LOW)

GPIO.output(led, GPIO.HIGH)

if time.time() \* 1000 > (t + 2300) and GPIO.input(sensor) == GPIO.HIGH:

t = int(time.time() \* 1000)

**4. Connect to the Internet (IoT):**

If you want to make it an IoT project, you'll need to add code for internet connectivity. This could be through Wi-Fi, Ethernet, or a cellular module, depending on your chosen microcontroller.

**5.Test and Debug:**

Test your setup with a controlled water source and verify that the system behaves as expected. Debug any issues that arise.

NOTE: Please note that this is a simplified example, and the actual implementation may vary based on your specific hardware and requirements. Additionally, consider safety measures, such as fail-safes and emergency shutdown procedures, when working with water control systems.

**Smart restroom on automatic room light sensor**

Creating a platform for water overflow control in a tank using MIT App Inventor involves several steps. MIT App Inventor is a user-friendly platform for creating Android apps, and it can be used to build a mobile app that monitors and controls water levels in a tank. Here's a basic outline of the steps involved.

**Design app interference**

• Create two labels (on off light and status).

• Add a button . Block-Based Coding:

• Use MIT App Inventor's block-based coding to program the app:

• When Screen1.Initialize:

• Initialize any necessary variables .

• When Button.Click:

• Toggle the state of the room free or valve (Open/Closed) .

• Update the status label accordingly .

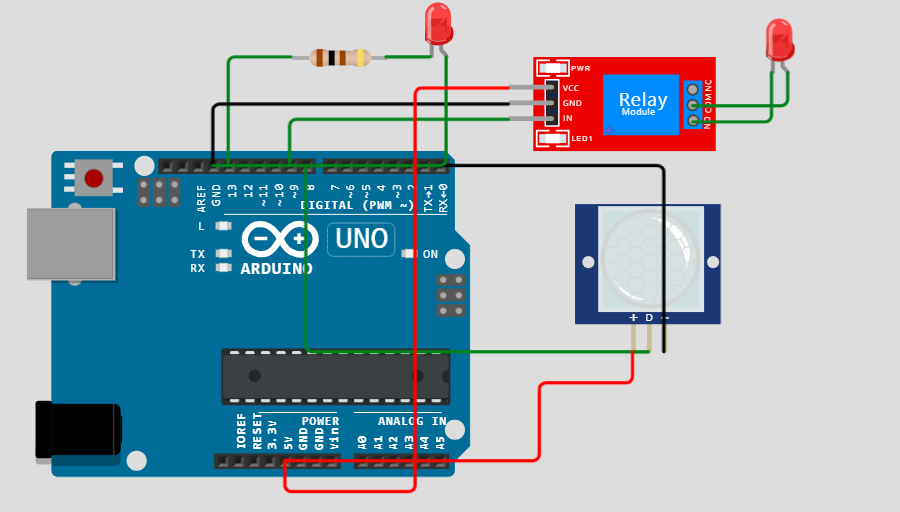
• Use a Timer component to periodically check the vacency(simulated for this example).

• Inside the Timer event handler, you can generate vacancy values as follows:

• Set a variable to a random number within a range representing the restroom.

• Display the vacency on the appropriate label

**Circuit design & program**



**Input program:**

Python

import time

import RPi.GPIO as GPIO

in1 = 9

sensor = 8

led = 13

t = 0

GPIO.setmode(GPIO.BCM)

GPIO.setup(in1, GPIO.OUT)

GPIO.setup(sensor, GPIO.IN)

GPIO.setup(led, GPIO.OUT)

GPIO.output(in1, GPIO.HIGH)

GPIO.output(led, GPIO.LOW)

while time.time() < 13:

GPIO.output(led, GPIO.HIGH)

time.sleep(0.05)

GPIO.output(led, GPIO.LOW)

time.sleep(0.05)

GPIO.output(led, GPIO.LOW)

while True:

GPIO.output(in1, GPIO.HIGH)

GPIO.output(led, GPIO.LOW)

if GPIO.input(sensor) == GPIO.HIGH:

t = int(time.time() \* 1000)

while time.time() \* 1000 < (t + 5000):

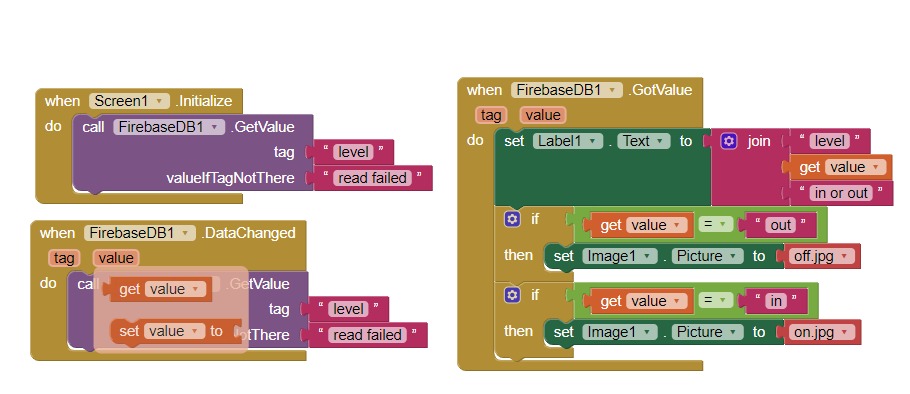
GPIO.output(in1, GPIO.LOW)

GPIO.output(led, GPIO.HIGH)

if time.time() \* 1000 > (t + 2300) and GPIO.input(sensor) == GPIO.HIGH:

t = int(time.time() \* 1000)

**Input program:**



Creating a platform for vaceny control in a restroom using MIT App Inventor involves several steps. MIT App Inventor is a user-friendly platform for creating Android apps, and it can be used to build a mobile app that monitors and controls vacancy in a restroom. Here's a basic outline of the steps involved:

1. Define the Requirements:

Clearly define the requirements and functionalities of your vacancy control system. You'll need to specify what features you want in your app, such as monitoring vacancy , sending alerts, and controlling restroom or valves.

2. Design the User Interface:

Use MIT App Inventor's drag-and-drop interface to design the user interface for your app. Create screens for monitoring vacancy , sending notification , and controlling the system.

3. Connect to Hardware:

- If you're connecting the app to hardware like sensors or actuators, you'll need to choose appropriate components and interface with them using Bluetooth, Wi-Fi, or other communication protocols

4. Coding in Blocks:

- Use the MIT App Inventor's visual programming language, which is block-based, to write the code for your app. This includes defining how the app interacts with the hardware, processes sensor data, and controls devices.

5. Set Up Alerts:

- Implement a notification system that sends alerts (push notifications, SMS, or email) to the user's device if the water level exceeds a predefined limit.

6. Testing and Debugging:

- Test the app thoroughly to ensure that it functions as expected. Debug any issues that may arise during testing

7. User Authentication (Optional):

- If necessary, implement user authentication to restrict access to the app and its control features

8. Documentation and Help:

- Create user documentation or help sections within the app to guide users on how to use it effectively

9. App Deployment:

- Once the app is ready, you can export it as an Android application and distribute it to your intended users. You can publish it on the Google Play Store or share the APK file directly.

10. Maintenance and Updates:

- Continue to maintain and update the app as needed, addressing any user feedback or bug reports. Remember that building an app to control physical systems requires careful consideration of safety and reliability.

It's essential to have a good understanding of the hardware you're connecting to and to thoroughly test the system before deploying it in a real-world environment. Additionally, consider the power source and connectivity options for your monitoring and control components