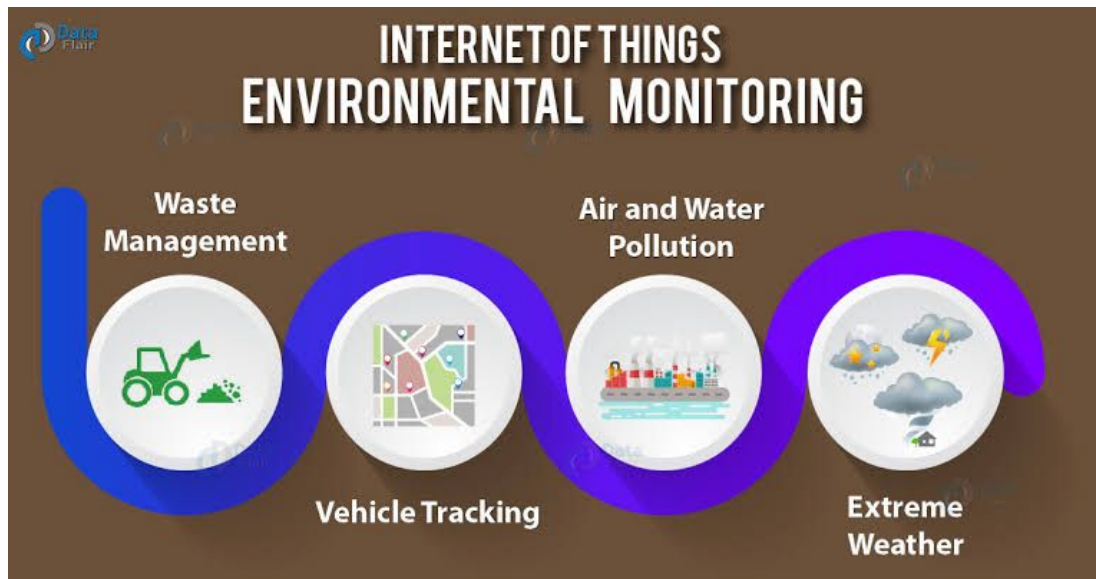


ENVIRONMENTAL MONITORING



IoT for environmental monitoring facilitates the development of wireless, remote environmental monitoring systems, which enable operations to remove much of the human interaction in system function, which reduces human labor, increases the range and frequency of sampling and monitoring, facilitates sophisticated on- ...

IoT Project in environmental monitoring :

Wireless Sensor Networks deployed on Urban Areas also for monitoring environmental weather status.

An economical pollution monitoring system also based on Internet of Things.

A monitoring system for indoor environment also using Internet of Things environment.

Importance of environmental monitoring :

IOT-based environmental monitoring is a critical aspect of modern sustainability. It allows businesses to better track energy consumption, monitor soil health and crop conditions, detect hazardous gases and conserve endangered species.

Components :

1. Arduino UNO
2. 10WATT1K
3. L293D
4. LDR
5. LED-RED, YELLOW
6. LM35 7. MOTOR
8. POT-HR
9. RELAY ULN2003A

Data collection :

● Air quality

Better Air Quality, Better Employee Performance

Air pollution is a global concern today.

Monitoring the quality of air in your office environment, especially in factories and mills,

helps take care of employee health and improves the productivity.

The network of data using the 'Internet of Things' model

is ideal for monitoring and optimizing indoor air quality and environment.

- Electricity monitoring

IoT-based environmental monitoring systems help in the early detection of power interruptions and help restore electricity.

Power demand monitoring and alarm notification functions identify issues, locate abnormalities and deal with it asap.

- Water leakage monitoring

IoT-based water management system is a great advantage to any workplace as it provides you with up-to-date and automatic data transmission regarding water loggers. This saves time on manual work while also providing data for analysis and monitoring the condition of the network over a period of time. This reduces emergencies or losses related to water leakage or scarcity.

Hardware components :

- 1) Raspberry Pi (or any compatible single-board computer)*
- 2) DHT11 sensor (for temperature and humidity)*
- 3) Internet connectivity (Wi-Fi/Ethernet)*

Software and services :

- 1) Python (for programming)
- 2) Adafruit DHT Library (for interfacing with the sensor)
- 3) IoT Cloud Service (e.g., Thingspeak, Adafruit IO, AWS IoT)

program :

```
import Adafruit_DHT
import requests

# Set your Thingspeak API key and channel ID
THINGSPEAK_API_KEY = 'YOUR_API_KEY'
THINGSPEAK_CHANNEL_ID = 'YOUR_CHANNEL_ID'

# Set sensor type (DHT11 or DHT22)
SENSOR_TYPE = Adafruit_DHT.DHT11

# GPIO pin connected to the sensor
GPIO_PIN = 4

# Read data from the sensor
humidity, temperature = Adafruit_DHT.read_retry
(SENSOR_TYPE, GPIO_PIN)

if humidity is not None and temperature is not None:

    # Send data to Thingspeak
    url = f'https://api.thingspeak.com/update?
api_key={THINGSPEAK_API_KEY}
&field1={temperature}&field2={humidity}'
    response = requests.get(url)
    if response.status_code == 200:
        print("Data sent to Thingspeak.")
    else:
        print("Failed to send data to Thingspeak.")
else:
    print("Failed to retrieve data from the sensor.")
```