

Teknologi IoT

Pendeteksi Hama



13321014_Asita Tambunan

D3 Teknologi Komputer
Institut Teknologi Del Fakultas Vokasi

A. Pendahuluan

Pada proyek ini kami membuat sistem Pendeteksi hama menggunakan sensor Pir. Dalam implementasinya, kami menggunakan Raspberry Pi sebagai perangkat mikrokontroler. Sensor Pir akan bekerja dengan mengeluarkan bunyi yang dimana bunyi tersebut akan berbunyi yang tergantung pada gerak hama .

B. Hasil dan Implementasi

1. Komponen

Ada beberapa komponen yang diperlukan untuk membangun sistem “Lampu Otomatis”, sebagai berikut:

- Breadboard
- Raspberry pi 4
- Sensor Pir
- Kabel Jumper

2. Code program pada raspberry pi 4

```
import RPi.GPIO as GPIO
import json
import time
from paho.mqtt import client as mqtt_client

broker = 'raspberrypi'
port = 1883
topic_publish = 'motion_detection_publish'
topic_subscribe = 'motion_detection_subscribe'
client_id = 'emqx'
username = 'user1'
password = 'user1'

pin_pir_sensor = 2 # GPIO pin for PIR sensor
pin_buzzer = 4 # GPIO pin for buzzer
```

```
GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(pin_pir_sensor, GPIO.IN)

GPIO.setup(pin_buzzer, GPIO.OUT)


def get_motion_status():
    if GPIO.input(pin_pir_sensor):
        return 1 # Motion detected
    else:
        return 0 # No motion detected


def control_buzzer(motion_status):
    if motion_status == 1:
        GPIO.output(pin_buzzer, GPIO.HIGH)
    else:
        GPIO.output(pin_buzzer, GPIO.LOW)


def connect_mqtt():
    def on_connect(client, userdata, flags, rc):
        if rc == 0:
            print("Connected to MQTT Broker!")
        else:
            print("Failed to connect, return code %d\n" % rc)

    client = mqtt_client.Client(client_id)
    client.username_pw_set(username, password)
    client.on_connect = on_connect
    client.connect(broker, port)
    return client


def publish_motion_status(client, motion_status):
    msg = json.dumps({
```

```
        "motion_status": motion_status
    })
    result = client.publish(topic_publish, msg)
    status = result.rc
    if status == 0:
        print(f"Motion Status: {motion_status}, published to topic {topic_publish}")
    else:
        print(f"Failed to send message to topic {topic_publish}")

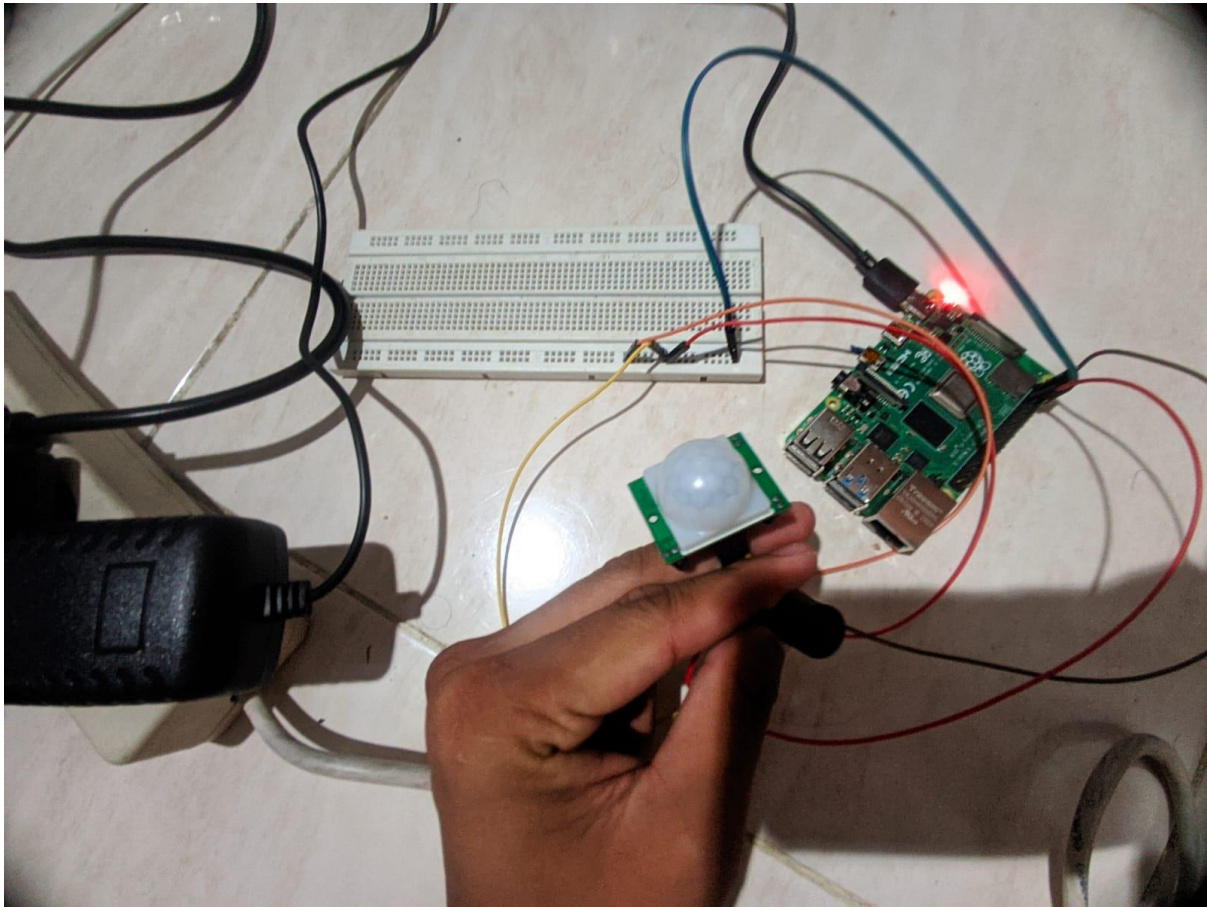
def on_message(client, userdata, msg):
    print(f"Received {str(msg.payload)} from topic {msg.topic}")

def publish_subscribe(client):
    client.on_message = on_message
    client.subscribe(topic_subscribe)
    while True:
        motion_status = get_motion_status()
        control_buzzer(motion_status)
        print(f"Motion Status: {motion_status}")
        publish_motion_status(client, motion_status)
        time.sleep(2)

def run():
    client = connect_mqtt()
    client.loop_start()
    publish_subscribe(client)

if __name__ == '__main__':
    run()
```

3. Rangkaian



Instalasi EMQX

Dalam proyek pendeteksi hama dengan sensor PIR, peran utama EMQX sebagai broker MQTT sangat penting. EMQ X memastikan komunikasi efisien antara sensor PIR dan perangkat lain, memungkinkan respons yang cepat terhadap keberadaan hama. Keunggulan EMQ X meliputi integrasi yang lancar, kemampuan pertumbuhan horizontal, dan keamanan data yang solid. Dengan infrastruktur yang handal, EMQ X mendukung keberhasilan proyek dalam menciptakan ekosistem yang efisien dan responsif terhadap tantangan pertanian.

```
kelompok16@raspberrypi:~ $ curl -s https://assets.emqx.com/scripts/install-emqx-deb.sh | sudo bash
```

```
kelompok16@raspberrypi:~ $ sudo apt-get install emqx
```

```
kelompok16@raspberrypi:~ $ sudo systemctl start emqx
```

Instalasi Paho MQTT

Instalasi Paho MQTT pada proyek pendeteksi hama menggunakan sensor PIR memberikan kemampuan untuk komunikasi efisien antar perangkat dengan menggunakan protokol MQTT. Fungsionalitas Paho MQTT, seperti model publikasi-langganan, manajemen pesan, dan

dukungan untuk berbagai bahasa pemrograman, memungkinkan pertukaran informasi deteksi hama yang cepat dan terpercaya. Dengan sifatnya yang sederhana dan mudah digunakan, Paho MQTT untuk pengembangan sistem yang responsif dan terintegrasi dalam proyek ini.

```
kelompok16@raspberrypi:~ $ sudo apt-get update
```

```
kelompok16@raspberrypi:~ $ sudo apt upgrade
```

```
kelompok16@raspberrypi:~ $ sudo pip3 install paho-mqtt
```

```
kelompok16@raspberrypi:~ $ sudo apt-get install python3-pip
```

Instalasi Node red

Instalasi Node-RED pada proyek pendeteksi hama menggunakan sensor PIR membawa manfaat signifikan. Node-RED memberikan antarmuka visual yang intuitif untuk mengelola aliran data dari sensor PIR, memfasilitasi integrasi dengan broker MQTT, dan memungkinkan pengaturan aksi otomatisasi. Dengan antarmuka yang mudah digunakan dan alat monitoring yang efisien, Node-RED menjadi alat yang berharga untuk merancang solusi yang responsif dan efisien dalam proyek ini.

```
kelompok16@raspberrypi:~ $ bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)
```

```
kelompok16@raspberrypi:~ $ node-red-start
```

Instalasi Influxdb

Instalasi InfluxDB pada proyek pendeteksi hama dengan sensor PIR berperan kunci dalam menyimpan, mengelola, dan menganalisis data waktu nyata. Fungsi utamanya mencakup penyimpanan data historis, kemampuan query fleksibel, visualisasi data, integrasi dengan Node-RED, skalabilitas tinggi, dan pemantauan data. Dengan InfluxDB, proyek dapat efisien menyimpan dan menganalisis data deteksi hama, memberikan wawasan yang diperlukan untuk pengambilan keputusan dan tindakan pencegahan yang efektif.

```
kelompok16@raspberrypi:~ $ wget -qO- https://repos.influxdata.com/influxdb.key | sudo apt-key add -  
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).  
OK
```

```
kelompok16@raspberrypi:~ $ echo "deb https://repos.influxdata.com/debian stretch stable" | sudo tee /etc/apt/sources.list.d/influxdb.list  
deb https://repos.influxdata.com/debian stretch stable
```

```
kelompok16@raspberrypi:~ $ sudo apt install influxdb
```

```
kelompok16@raspberrypi:~ $ sudo systemctl unmask influxdb
kelompok16@raspberrypi:~ $ sudo systemctl enable influxdb
Synchronizing state of influxdb.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable influxdb
kelompok16@raspberrypi:~ $ sudo systemctl start influxdb
```

```
kelompok16@raspberrypi:~ $ sudo apt install influxdb-client
```

```
kelompok16@raspberrypi:~ $ influx
Connected to http://localhost:8086 version 1.6.7-rc0
InfluxDB shell version: 1.6.7-rc0
> CREATE USER kelompok16 WITH PASSWORD 'kelompok16' WITH ALL PRIVILEGES
> exit
```

```
[http]

auth-enabled = true
pprof-enabled = true
pprof-auth-enabled = true
ping-auth-enabled = true
```

```
kelompok16@raspberrypi:~ $ sudo nano /etc/influxdb/influxdb.conf
kelompok16@raspberrypi:~ $ sudo systemctl restart influxdb
kelompok16@raspberrypi:~ $ influx -username kelompok16 -password kelompok16
Connected to http://localhost:8086 version 1.6.7-rc0
InfluxDB shell version: 1.6.7-rc0
> CREATE DATABASE sensor
> exit
```

Instalasi Grafana

Instalasi Grafana pada proyek pendeteksi hama dengan sensor PIR berfungsi untuk memberikan visualisasi data yang interaktif dan informatif. Grafana memungkinkan pemantauan real-time, dan integrasi yang baik dengan InfluxDB dan Node-RED. Dengan fitur customization, notifikasi, dan peringatan, Grafana meningkatkan pemahaman terhadap keberadaan hama, mendukung pengambilan keputusan yang lebih efektif dalam proyek ini.

```
kelompok16@raspberrypi:~ $ sudo apt-get install software-properties-common
```

```
kelompok16@raspberrypi:~ $ sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

```
kelompok16@raspberrypi:~ $ sudo wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
```

```
kelompok16@raspberrypi:~ $ sudo apt-get install grafana
```

```
kelompok16@raspberrypi:~ $ sudo systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /lib/systemd/system/grafana-server.service.
kelompok16@raspberrypi:~ $ sudo systemctl start grafana-server
```

Keberhasilan Proyek

Emqx

WebSocket Client

Default

+

Connection

Host

raspberrypi

Port

8083

Path

/mqtt

Client ID

emqx_NTYxMJ

Username

user1

Password

Keepalive

60

Clean Start

true

Session Expiry Interval

0

s

TLS

false

ProtocolVersion

MQTT 5

Disconnect

Connect

Subscription

Topic

motion_detection_publish

QoS

0

Subscribe

Topic	QoS	Time	Actions
motion_detection_publish	0	2023-12-18 23:25:58	<div>Cancel</div>

Publish

Topic

testtopic/1

Payload

{"msg": "hello"}

QoS

0

Retain

Publish

Received

Topic	QoS	Payload	Time
motion_detection	0	{"motion_status": 1}	2023-12-18 23:26:37

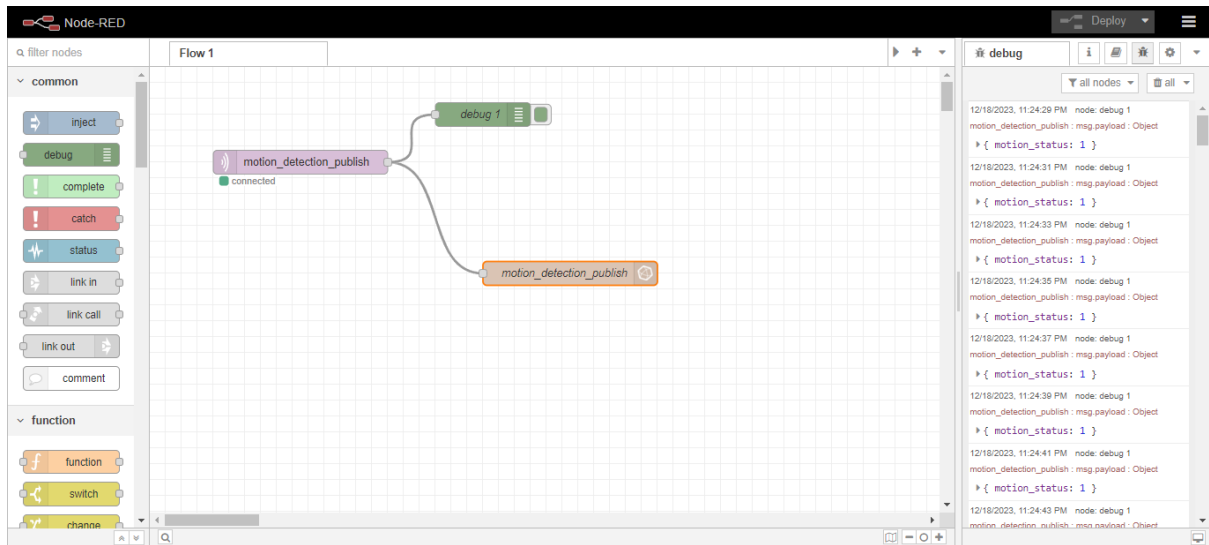
Published

Topic	QoS	Payload	Time
No Data			

Influxdb

```
name: measurements
name
----
topic/satu
> show measurements
name: measurements
name
----
motion_detection_publish
topic/satu
> select * from "motion_detection_publish"
name: motion_detection_publish
time                motion_status
----                -
1702913726590742425 1
1702913728591050615 1
```


Node Red



Grafana

