

**Laporan Evaluasi Akhir Semester**  
**Internet of Things (R)**



**Oleh :**

Muhammad Ikraam Fahriono

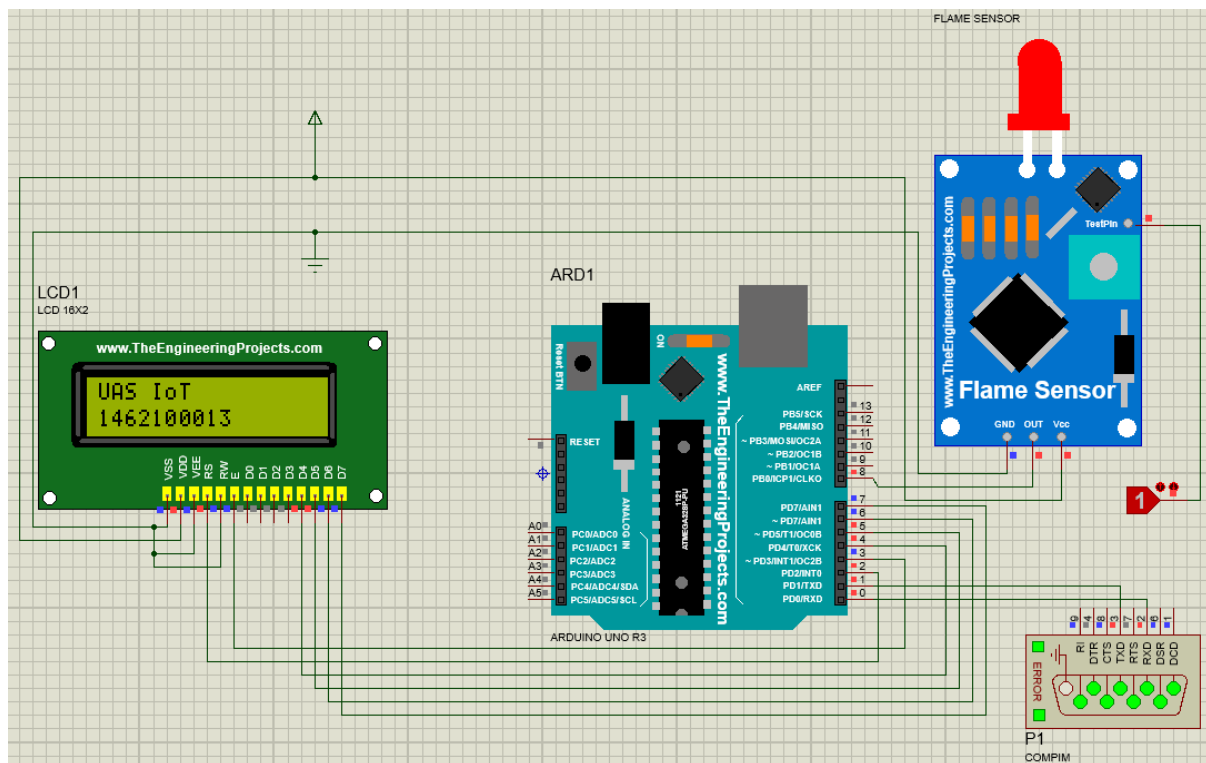
1462100013

**Dosen Pengampu:**

Ir. Elvianto Dwi Hartono, ST, MM, M.Kom, MT

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS 17 AGUSTUS 1945 SURABAYA**  
**2024**

## Desain Proteus



### Arduino UNO R3

Arduino UNO R3 adalah salah satu model papan mikrokontroler paling populer dalam keluarga Arduino. Papan ini menggunakan mikrokontroler ATmega328P dan berjalan pada tegangan operasional 5V. Dengan 14 pin digital I/O (di mana 6 dapat digunakan sebagai output PWM) dan 6 pin input analog, Arduino UNO R3 menyediakan banyak fleksibilitas untuk berbagai proyek. Papan ini juga dilengkapi dengan memori flash 32 KB, SRAM 2 KB, dan EEPROM 1 KB. Kecepatan clocknya adalah 16 MHz, yang cukup untuk sebagian besar aplikasi. Konektivitas melalui USB memungkinkan untuk memprogram dan mengontrol papan dengan mudah dari komputer, sementara jack daya eksternal memungkinkan papan untuk digunakan secara mandiri dalam proyek-proyek.

### LCD 16x2

LCD 16x2 adalah layar kristal cair yang mampu menampilkan 16 karakter pada dua baris. Layar ini sering digunakan dalam berbagai proyek elektronik untuk menampilkan informasi seperti status sistem, sensor readings, dan pesan lainnya. LCD 16x2 biasanya menggunakan driver HD44780, yang memungkinkan pengendalian yang mudah dengan hanya beberapa pin

dari mikrokontroler. Koneksi standar melibatkan penggunaan pin data dan kontrol (RS, RW, E) serta daya dan ground. Layar ini juga memiliki lampu latar yang dapat dikendalikan untuk visibilitas dalam berbagai kondisi pencahayaan. Kontras layar dapat disesuaikan menggunakan potensiometer yang terhubung ke pin V0.

### **Flame Sensor**

Flame Sensor adalah sensor yang digunakan untuk mendeteksi keberadaan api atau nyala. Sensor ini biasanya menggunakan fotodiode atau fototransistor untuk merespons cahaya dalam spektrum inframerah yang dipancarkan oleh api. Flame Sensor memiliki tiga pin: VCC (untuk daya), GND (untuk ground), dan DO (digital output). Ketika sensor mendeteksi api, pin DO akan memberikan sinyal digital yang dapat dibaca oleh mikrokontroler seperti Arduino. Sensor ini sering digunakan dalam sistem keamanan kebakaran, robot pemadam kebakaran, dan aplikasi lain yang memerlukan deteksi api.

### **COMPIM**

COMPIM (COM Port Interface Model) adalah komponen di Proteus yang digunakan untuk mensimulasikan komunikasi serial antara perangkat mikrokontroler dan port serial komputer. COMPIM memungkinkan pengguna untuk menghubungkan desain simulasi di Proteus dengan perangkat keras fisik melalui port serial komputer. Dengan mengkonfigurasi port dan baud rate yang sesuai, COMPIM dapat mengirim dan menerima data serial, memungkinkan pengguna untuk menguji dan mengembangkan perangkat lunak komunikasi serial dalam lingkungan simulasi. Ini sangat berguna untuk menguji protokol komunikasi serial dan debugging program sebelum diimplementasikan pada perangkat keras sebenarnya.

**SERIAL PORT (VSPD):**

**COMPIM: COM2**

**PYTHON: COM4**

## Source Code:

### Arduino IDE:

```
#include <LiquidCrystal.h> // Memasukkan library untuk mengendalikan LCD

#define FLAME_SENSOR_PIN 8 // Mendefinisikan pin 8 sebagai pin untuk Flame
Sensor

LiquidCrystal lcd(2, 3, 4, 5, 6, 7); // Menginisialisasi objek LCD dengan pin
yang terhubung ke Arduino

void setup() {
    Serial.begin(9600); // Memulai komunikasi serial dengan baud rate 9600
    lcd.begin(16, 2); // Menginisialisasi LCD dengan ukuran 16x2
    lcd.setCursor(0,0); // Menempatkan kursor pada posisi kolom 0 baris 0
    lcd.print("UAS IoT"); // Menampilkan teks "UAS IoT" pada baris pertama LCD
    lcd.setCursor(0,1); // Menempatkan kursor pada posisi kolom 0 baris 1
    lcd.print("1462100013"); // Menampilkan teks "1462100013" pada baris kedua
    LCD
    pinMode(FLAME_SENSOR_PIN, INPUT); // Mengatur pin Flame Sensor sebagai input
    delay(2000); // Menunggu selama 2 detik
}

void loop() {
    lcd.clear(); // Membersihkan layar LCD
    lcd.setCursor(0,0); // Menempatkan kursor pada posisi kolom 0 baris 0
    lcd.print("Status Sensor: "); // Menampilkan teks "Status Sensor: " pada
    baris pertama LCD
    lcd.setCursor(4,1); // Menempatkan kursor pada posisi kolom 4 baris 1
    int statusSensor = digitalRead(FLAME_SENSOR_PIN); // Membaca status dari
    Flame Sensor

    if (statusSensor == HIGH) { // Jika Flame Sensor mendeteksi api
        String json = (String){"\sensor_status\": \"true\"}"; // Membuat string
        JSON dengan status sensor true
        Serial.println(json); // Mengirim string JSON ke monitor serial
        lcd.print("MENYALA"); // Menampilkan teks "MENYALA" pada LCD
    } else { // Jika Flame Sensor tidak mendeteksi api
        String json = (String){"\sensor_status\": \"false\"}"; // Membuat string
        JSON dengan status sensor false
        Serial.println(json); // Mengirim string JSON ke monitor serial
        lcd.print("MATI"); // Menampilkan teks "MATI" pada LCD
    }

    delay(500); // Menunggu selama 0,5 detik sebelum iterasi berikutnya
}
```

## Python:

```
from fastapi import FastAPI # Mengimpor FastAPI untuk membuat aplikasi web
from fastapi.responses import HTMLResponse # Mengimpor HTMLResponse untuk
mengembalikan respon HTML

import random # Mengimpor modul random (walaupun tidak digunakan di sini)
import serial # Mengimpor modul serial untuk komunikasi serial
import json # Mengimpor modul json untuk memproses data JSON

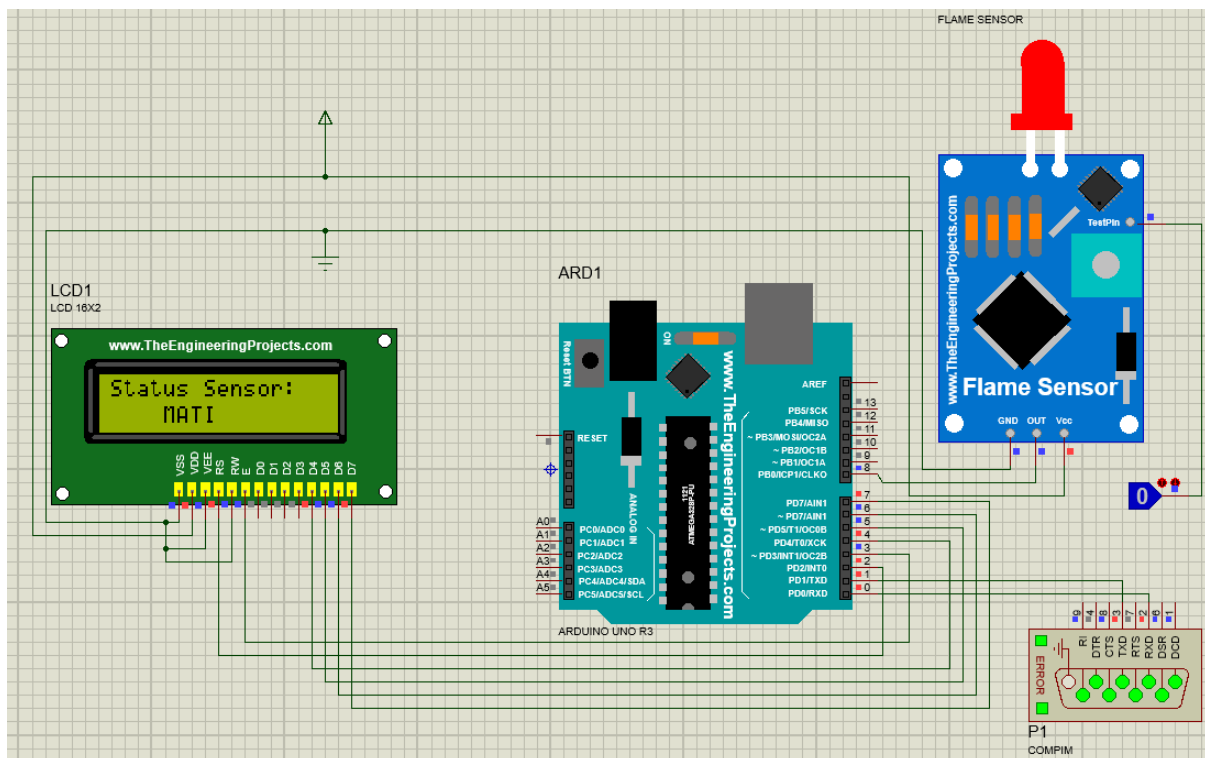
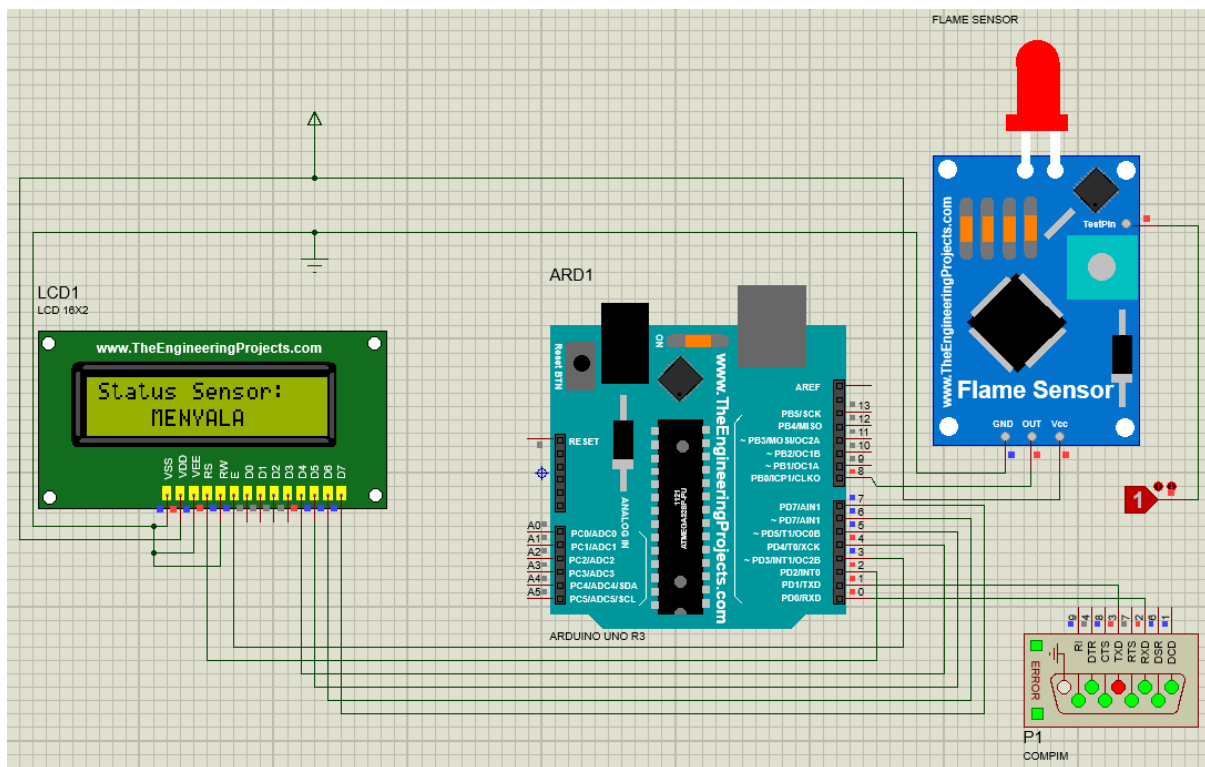
app = FastAPI() # Membuat instance dari aplikasi FastAPI

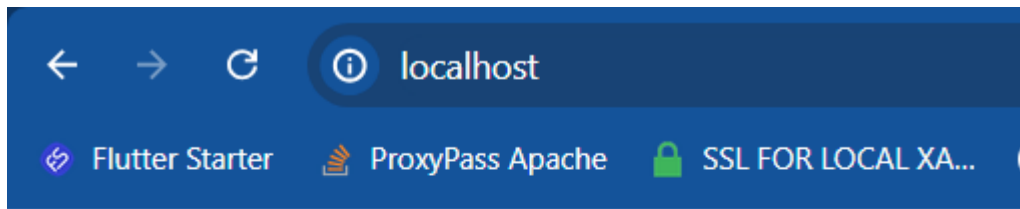
# Membuka port serial dalam loop hingga berhasil dibuka
while True:
    try:
        ser = serial.Serial('COM4', 9600) # Mencoba membuka port serial COM4
        dengan baud rate 9600
        print("Serial port opened successfully") # Menampilkan pesan jika
        port serial berhasil dibuka
        break # Keluar dari loop jika berhasil membuka port
    except:
        print("Error opening serial port") # Menampilkan pesan jika terjadi
        kesalahan saat membuka port serial
        pass # Melanjutkan loop untuk mencoba membuka kembali port

# Mendefinisikan endpoint root
@app.get("/", response_class=HTMLResponse)
async def read_root():
    with open("index.html") as f: # Membuka file index.html
        html_file = f.read() # Membaca isi file
        f.close() # Menutup file
        return HTMLResponse(content=html_file, status_code=200) #
        Mengembalikan isi file sebagai respon HTML dengan status kode 200

# Mendefinisikan endpoint untuk mendapatkan data dari serial
@app.get("/api/data")
async def get_data():
    if ser.in_waiting > 0: # Memeriksa apakah ada data yang menunggu di
        buffer serial
        data = ser.readline().decode('utf-8').strip() # Membaca satu baris
        data dari serial dan menghapus karakter whitespace
        status = json.loads(data) # Mengonversi data JSON menjadi dictionary
        Python
        ser.flushInput() # Membersihkan buffer input serial
        return status # Mengembalikan status sensor sebagai respon JSON
    else:
        return {'sensor_status': 'N/A'} # Mengembalikan respon jika tidak ada
        data yang tersedia
```

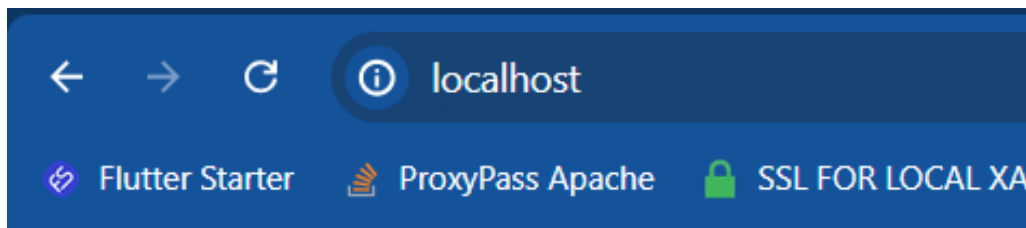
Hasil:





# Sensor Deteksi Api

Status sensor: MENYALA



# Sensor Deteksi Api

Status sensor: MATI

## Penutup

Menghubungkan Proteus dengan aplikasi web menggunakan COMPIM dan aplikasi Virtual Serial Port Driver memungkinkan simulasi dan pengujian proyek IoT secara efektif. Dalam Proteus, Arduino UNO R3 dihubungkan ke COMPIM yang dikonfigurasi menggunakan port COM virtual (misalnya, COM2). Aplikasi Virtual Serial Port Driver digunakan untuk membuat pasangan port COM virtual, seperti COM2 dan COM4, yang memungkinkan komunikasi antara Proteus dan sistem operasi. Python dengan modul `pyserial` digunakan untuk membaca data dari COM4, yang dikirim oleh Proteus melalui COM2. Data ini kemudian dapat diproses dan disajikan melalui aplikasi web menggunakan framework seperti FastAPI. Pendekatan ini memungkinkan pengembangan dan pengujian yang efisien dalam lingkungan simulasi sebelum implementasi pada perangkat keras sebenarnya.