

LAPORAN PRAKTIKUM ARSITEKTUR DAN ORGANISASI KOMPUTER



Disusun Oleh :

Nama : Habib Hubaddilah
NIM :09030582226029
Prodi :Teknik Komputer
Dosen : Adi Hermansyah, S.Kom, M.T.

LAB. JARINGAN KOMPUTER DAN KOMUNIKASI DATA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
PALEMBANG 2023

Kendali 7 Lampu LED

- **Tujuan Praktikum:**

Praktikum ini bertujuan untuk menguji kemampuan pengendalian lampu LED dengan menggunakan Arduino dalam berbagai kondisi yang berbeda. Praktikan diharapkan dapat memahami konsep dasar pengendalian LED dan cara mengimplementasikannya dalam berbagai skenario.

- **Alat dan Bahan:**

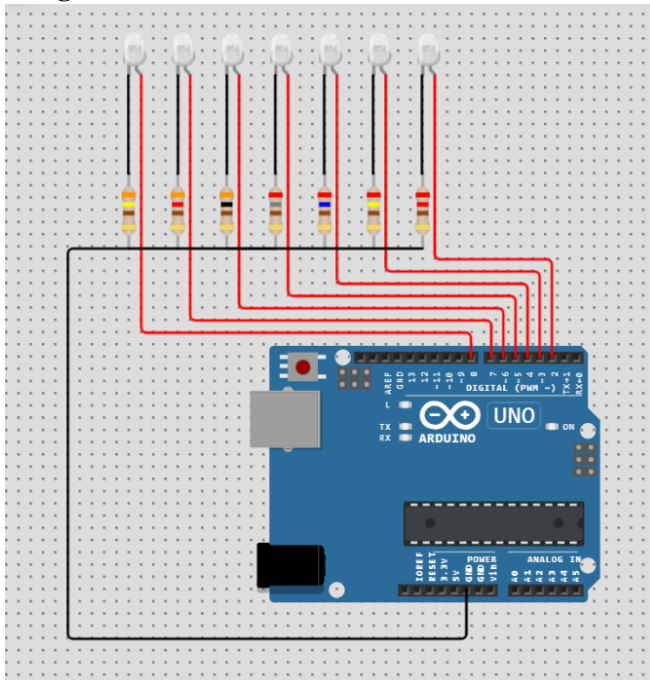
1. Papan Arduino Uno
2. 7 LED
3. 7 resistor

- **Pengaturan Percobaan:**

1. Menghubungkan 7 LED ke pin output pada papan Arduino, dengan menggunakan resistor untuk melindungi LED dari arus berlebih.
2. Memasang semua komponen sesuai dengan rangkaian yang diinginkan.
3. Membuat program Arduino untuk mengendalikan 7 LED sesuai dengan lima kondisi yang telah ditentukan.

- **Percobaan:**

Rangkaian:



link Wokwi : <https://wokwi.com/projects/378273252876703745>

Sketch.ino:

```

int ledPins[] = {2, 3, 4, 5, 6, 7, 8};

int numLeds = 7;
int delayTime = 1000;

void setup() {
  for (int i = 0; i < numLeds; i++) {
    pinMode(ledPins[i], OUTPUT);
  }
}

void loop() {
  //1. led akan hidup kelap kelip
  blinkled();
  // 2. Hidup Barengan
  allLedOn();
  // 3. Hidup dengan Delay yang Berbeda
  LedDelay();
  // 4. Hidup Bergantian dari Kiri ke Kanan dan Sebaliknya
  rightToleftLeds();
  // 5. Hidup bergantian dari redup ke terang
  fadeLeds();
}

void blinkled() {
  for (int i = 0; i < numLeds; i++) {
    digitalWrite(ledPins[i], HIGH);
  }
  delay(500);

  for (int i = 0; i < numLeds; i++) {
    digitalWrite(ledPins[i], LOW);
  }
  delay(500);
}

void allLedOn() {
  for (int i = 0; i < numLeds; i++) {
    digitalWrite(ledPins[i], HIGH);
  }
  delay(2000);
  for (int i = 0; i < numLeds; i++) {
    digitalWrite(ledPins[i], LOW);
  }
  delay(1000);
}

```

```

void LedDelay() {
    int delays[] = {500, 1000, 1500, 2000, 2500, 3000, 3500};
    for (int i = 0; i < numLeds; i++) {
        digitalWrite(ledPins[i], HIGH);
        delay(delays[i]);
        digitalWrite(ledPins[i], LOW);
    }
}

void rightToleftLeds(){
    for (int i = 0; i < numLeds; i++) {
        digitalWrite(ledPins[i], HIGH);
        delay(delayTime);
        digitalWrite(ledPins[i], LOW);
    }
    for (int i = numLeds - 1; i >= 0; i--) {
        digitalWrite(ledPins[i], HIGH);
        delay(delayTime);
        digitalWrite(ledPins[i], LOW);
    }
}

void fadeLeds() {
    for (int i = 0; i < 7; i++) {
        for (int brightness = 0; brightness <= 255; brightness += 5) {
            analogWrite(ledPins[i], brightness);
            delay(10);
        }
        for (int brightness = 255; brightness >= 0; brightness -= 5) {
            analogWrite(ledPins[i], brightness);
            delay(10);
        }
    }
}

```

- **Berikut Penjelasan 5 ketentuan Program 7 LED:**

1. Program Arduino yang dibuat akan membuat lampu LED berkedip secara bersamaan. Setiap LED akan menyala selama beberapa saat dan kemudian mati secara bergantian.

```
void blinkled() {
    for (int i = 0; i <
numLeds; i++) {
        digitalWrite(ledPins[i],
HIGH);
    }
    delay(500);

    for (int i = 0; i <
numLeds; i++) {
        digitalWrite(ledPins[i],
LOW);
    }
    delay(500);
}
```

'void blinkled()': Ini adalah deklarasi sebuah fungsi bernama blinkled. Fungsi ini tidak mengembalikan nilai (void), sehingga digunakan untuk mengendalikan LED dan tidak mengembalikan hasil apapun.

'for (int i = 0; i < numLeds; i++) { ... }': Ini adalah loop for yang digunakan untuk mengakses setiap LED dalam rangkaian. Variabel i digunakan sebagai penghitung yang dimulai dari 0 dan akan terus bertambah hingga mencapai jumlah LED yang disimpan dalam variabel **'numLeds'**.

'digitalWrite(ledPins[i], HIGH);': Pada iterasi tertentu, ini akan mengatur pin yang terhubung ke LED ke status tinggi (HIGH), yang akan menyala LED. ledPins adalah sebuah array yang menyimpan nomor pin untuk masing-masing LED.

'delay(500);': Setelah menghidupkan semua LED, ada jeda waktu selama 500 milidetik (setengah detik).

Kemudian, ada loop **'for'** lain yang akan mengatur semua LED kembali ke status rendah (LOW), yang akan mematikan LED.

'delay(500);': Setelah mematikan semua LED, ada jeda waktu lagi selama 500 milidetik.

2. Program akan mengontrol semua LED untuk menyala bersamaan. Semua LED akan menyala pada saat yang bersamaan dengan interval waktu tertentu.

```
void allLedOn() {
    for (int i = 0; i < numLeds; i++) {
        digitalWrite(ledPins[i], HIGH);
    }
    delay(2000);
    for (int i = 0; i < numLeds; i++) {
        digitalWrite(ledPins[i], LOW);
    }
    delay(1000);
}
```

'void allLedOn()': Ini adalah deklarasi sebuah fungsi bernama **'allLedOn'**. Fungsi ini tidak mengembalikan nilai (void), yang berarti fungsi ini digunakan untuk mengendalikan LED dan tidak mengembalikan hasil apapun.

'for (int i = 0; i < numLeds; i++) { ... }': Fungsi ini menggunakan loop **'for'** untuk mengakses setiap LED dalam rangkaian. Variabel i digunakan sebagai penghitung yang dimulai dari 0 dan akan terus bertambah hingga mencapai jumlah LED yang disimpan dalam variabel **'numLeds'**.

'digitalWrite(ledPins[i], HIGH);': Pada setiap iterasi, fungsi ini akan mengatur pin yang terhubung ke LED ke status tinggi (HIGH), yang akan menyala LED. ledPins adalah sebuah array yang menyimpan nomor pin untuk masing-masing LED.

'delay(2000);': Setelah menghidupkan semua LED, ada jeda waktu selama 2000 milidetik (2 detik).

Kemudian, ada loop **'for'** lain yang akan mengatur semua LED kembali ke status rendah (LOW), yang akan mematikan LED.

'delay(1000);': Setelah mematikan semua LED, ada jeda waktu selama 1000 milidetik (1 detik).

3. Program akan mengatur waktu delay yang berbeda untuk setiap LED, sehingga LED akan menyala dengan interval yang berbeda.

```
void LedDelay() {  
    int delays[] = {500, 1000,  
1500, 2000, 2500, 3000,  
3500};  
    for (int i = 0; i <  
numLeds; i++) {  
        digitalWrite(ledPins[i],  
HIGH);  
        delay(delays[i]);  
        digitalWrite(ledPins[i],  
LOW);  
    }  
}
```

'void LedDelay()': Ini adalah deklarasi sebuah fungsi bernama **'LedDelay'**. Fungsi ini tidak mengembalikan nilai (void), yang berarti digunakan untuk mengendalikan LED dan tidak mengembalikan hasil apapun.

'int delays[] = {500, 1000, 1500, 2000, 2500, 3000, 3500};': Fungsi ini mendefinisikan sebuah array bernama **'delays'**, yang berisi sejumlah angka yang mewakili jeda waktu (dalam milidetik) untuk setiap LED. Array ini berisi 7 elemen yang akan digunakan pada 7 LED berbeda dalam rangkaian.

'for (int i = 0; i < numLeds; i++) { ... }': Fungsi ini menggunakan loop **'for'** untuk mengakses setiap LED dalam rangkaian. Variabel **'i'** digunakan sebagai penghitung yang dimulai dari 0 dan akan terus bertambah hingga mencapai jumlah LED yang disimpan dalam variabel **'numLeds'**.

'digitalWrite(ledPins[i], HIGH);': Pada setiap iterasi, fungsi ini akan mengatur pin yang terhubung ke LED ke status tinggi (HIGH), yang akan menyala LED.

'delay(delays[i]);': Setelah menghidupkan LED, fungsi ini akan memanggil **'delay'** dengan argumen **'delays[i]'**, yang akan memberikan jeda waktu sesuai dengan nilai yang ada dalam array **'delays'** untuk LED tertentu.

'digitalWrite(ledPins[i], LOW);': Setelah jeda waktu, fungsi ini akan mengatur pin kembali ke status rendah (LOW), yang akan mematikan LED.

4. Program akan mengendalikan LED untuk menyala dari kiri ke kanan dan kemudian dari kanan ke kiri. Ini akan menciptakan efek pergerakan lampu dari satu sisi ke sisi lain.

```
void rightToLeftLeds(){
    for (int i = 0; i < numLeds; i++) {
        digitalWrite(ledPins[i], HIGH);
        delay(delayTime);
        digitalWrite(ledPins[i], LOW);
    }
    for (int i = numLeds - 1; i >= 0; i--) {
        digitalWrite(ledPins[i], HIGH);
        delay(delayTime);
        digitalWrite(ledPins[i], LOW);
    }
}
```

`void rightToLeftLeds()`: Ini adalah deklarasi sebuah fungsi bernama **`rightToLeftLeds`**. Fungsi ini tidak mengembalikan nilai (void), yang berarti digunakan untuk mengendalikan LED dan tidak mengembalikan hasil apapun.

`for (int i = 0; i < numLeds; i++) { ... }`: Fungsi ini menggunakan loop **`for`** untuk mengakses setiap LED dalam rangkaian dari kiri ke kanan. Variabel **`i`** digunakan sebagai penghitung yang dimulai dari 0 dan akan terus bertambah hingga mencapai jumlah LED yang disimpan dalam variabel **`numLeds`**.

`digitalWrite(ledPins[i], HIGH);`: Pada setiap iterasi, fungsi ini akan mengatur pin yang terhubung ke LED ke status tinggi (HIGH), yang akan menyala LED.

`delay(delayTime);`: Setelah menghidupkan LED, fungsi ini akan memanggil **`delay`** dengan argumen **`delayTime`**, yang akan memberikan jeda waktu tertentu sebelum mematikan LED.

`digitalWrite(ledPins[i], LOW);`: Setelah jeda waktu, fungsi ini akan mengatur pin kembali ke status rendah (LOW), yang akan mematikan LED.

Selanjutnya, ada loop **`for`** lain yang akan mengatur LED dari kanan ke kiri. Loop ini dimulai dari indeks **`numLeds - 1`** (indeks terakhir) dan akan terus berkurang hingga mencapai indeks 0.

Pada setiap iterasi loop kedua, LED yang sesuai akan dihidupkan dengan **`digitalWrite(ledPins[i], HIGH)`**, kemudian ada jeda waktu **`delay(delayTime)`**, dan terakhir, LED dimatikan dengan **`digitalWrite(ledPins[i], LOW)`**.

5. Program akan mengendalikan LED untuk perlahan-lahan menyala dari keadaan redup hingga terang satu per satu. Ini akan menciptakan efek "fade in" untuk setiap LED.

```

void fadeLeds() {
    for (int i = 0; i < 7; i++) {
        for (int brightness = 0; brightness <= 255;
            brightness += 5) {
            analogWrite(ledPins[i], brightness);
            delay(10);
        }
        for (int brightness = 255; brightness >= 0;
            brightness -= 5) {
            analogWrite(ledPins[i], brightness);
            delay(10);
        }
    }
}

```

1) `void fadeLeds()`: Ini adalah deklarasi sebuah fungsi bernama `fadeLeds`. Fungsi ini tidak mengembalikan nilai (void), yang berarti digunakan untuk mengendalikan LED dan tidak mengembalikan hasil apapun.

2) Fungsi ini memiliki dua loop bersarang:

a. Loop pertama menggunakan variabel `i` untuk mengakses setiap LED dalam rangkaian. Ini berjalan sebanyak 7 kali karena ada perulangan 7 LED (dalam rentang dari 0 hingga 6).

b. Di dalam loop pertama, ada dua perulangan lain:

- Perulangan pertama dengan variabel `brightness` yang mulai dari 0 dan bertambah sebanyak 5 setiap kali (maksimum 255). Ini menggunakan fungsi `analogWrite(ledPins[i], brightness)` untuk mengatur kecerahan LED dengan jeda waktu 10 milidetik (0.01 detik).
- Perulangan kedua dengan variabel `brightness` yang mulai dari 255 dan berkurang sebanyak 5 setiap kali hingga mencapai 0. Ini juga menggunakan `analogWrite` untuk mengurangi kecerahan LED dengan jeda waktu 10 milidetik.

- **Hasil dan Diskusi:**

Pada praktikum ini, Saya berhasil mengimplementasikan semua kondisi yang telah ditetapkan dengan bantuan Arduino. Kami memahami bagaimana mengontrol LED menggunakan Arduino dan bagaimana membuat efek visual yang berbeda dengan mengubah programnya.

- **Kesimpulan:**

Dalam praktikum ini, kami berhasil memahami dasar-dasar pengendalian LED dengan Arduino dan mengimplementasikan berbagai kondisi yang berbeda untuk 7 lampu LED. Praktikum ini membantu meningkatkan pemahaman kami tentang penggunaan Arduino dalam proyek elektronik sederhana.

link Wokwi : <https://wokwi.com/projects/378273252876703745>