**Tugas Pertemuan 5 LED**

**Source Code I2C:**

|  |
| --- |
| #include <Arduino.h>  #include <Wire.h>  int SDA\_pin = D2;  int SCL\_pin = D3;  void setup() {  Wire.begin(SDA\_pin, SCL\_pin);  Serial.begin(9600);  }  void loop() {  byte error, address;  int nDevices;  Serial.print("Scanning...");  nDevices = 0;  for(address = 1; address < 127; address++){  Wire.beginTransmission(address);  error = Wire.endTransmission();  if (error == 0){  Serial.print("I2C terbaca pada alamat 0x");  if (address < 16){  Serial.print("0");  }  Serial.print(address, HEX);  Serial.println("...!");  nDevices++;  } else if (error == 4){  Serial.print("Ada error yang tidak diketahui di alamat 0x");  if (address < 16){  Serial.print("0");  }  Serial.print(address, HEX);  }  }  if (nDevices == 0){  Serial.println("Tidak ada alamat I2C satupun yang terdeteksi");  } else {  Serial.print("Selesai");  }  delay(5000);  } |

**Penjelasan fungsi setiap blok kode:**

1. Header dan Deklarasi Pin

|  |
| --- |
| #include <Arduino.h>  #include <Wire.h>  int SDA\_pin = D2;  int SCL\_pin = D3; |

Penjelasan:

* #include <Arduino.h> dan #include <Wire.h>: Menyertakan pustaka Arduino dan pustaka Wire untuk komunikasi I2C.
* SDA\_pin dan SCL\_pin: Mendefinisikan pin SDA dan SCL yang akan digunakan untuk komunikasi I2C. Di sini digunakan pin D2 dan D3 (umumnya digunakan pada board ESP8266 atau sejenisnya).

1. Setup

|  |
| --- |
| void setup() {  Wire.begin(SDA\_pin, SCL\_pin);  Serial.begin(9600);  } |

Penjelasan:

* Wire.begin(SDA\_pin, SCL\_pin): Memulai komunikasi I2C dengan pin SDA dan SCL yang telah didefinisikan.
* Serial.begin(9600): Memulai komunikasi serial untuk mengirim data ke monitor serial dengan baud rate 9600 bps.

1. Loop

|  |
| --- |
| void loop() {  byte error, address;  int nDevices;  Serial.print("Scanning...");  nDevices = 0; |

Penjelasan:

* Variabel error menyimpan status dari Wire.endTransmission().
* Variabel address digunakan untuk iterasi dari alamat 1 hingga 126 (alamat I2C valid).
* nDevices adalah penghitung jumlah perangkat I2C yang ditemukan.
* Serial.print("Scanning..."): Menampilkan informasi bahwa proses scanning dimulai.

1. Pendeteksian Alamat I2C

|  |
| --- |
| for(address = 1; address < 127; address++){  Wire.beginTransmission(address);  error = Wire.endTransmission(); |

Penjelasan:

* Melakukan iterasi dari alamat I2C 1 sampai 126.
* Wire.beginTransmission(address) memulai transmisi ke alamat I2C tertentu.
* error = Wire.endTransmission() mengakhiri transmisi dan mengembalikan kode error.

1. Menangani Hasil Pemindaian

|  |
| --- |
| if (error == 0){  Serial.print("I2C terbaca pada alamat 0x");  if (address < 16){  Serial.print("0");  }  Serial.print(address, HEX);  Serial.println("...!");  nDevices++; |

Penjelasan:

* Jika tidak ada error (error == 0), berarti ada perangkat I2C di alamat tersebut.
* Mencetak alamat I2C dalam format heksadesimal.
* nDevices++ menambahkan jumlah perangkat yang ditemukan.

|  |
| --- |
| } else if (error == 4){  Serial.print("Ada error yang tidak diketahui di alamat 0x");  if (address < 16){  Serial.print("0");  }  Serial.print(address, HEX);  } |

Penjelasan:

* Jika tidak ada error (error == 0), berarti ada perangkat I2C di alamat tersebut.
* Menangani kasus error tidak diketahui (error == 4).

1. Menampilkan Hasil

|  |
| --- |
| if (nDevices == 0){  Serial.println("Tidak ada alamat I2C satupun yang terdeteksi");  } else {  Serial.print("Selesai");  } |

Penjelasan:

* Jika tidak ada perangkat I2C terdeteksi, tampilkan pesan bahwa tidak ada alamat I2C.
* Jika ada, tampilkan “Selesai” sebagai tanda akhir pemindaian.

1. Delay

|  |
| --- |
| delay(5000);  } |

Penjelasan:

* Menunggu 5 detik sebelum melakukan scanning ulang.

1. Kesimpulan

Program ini berfungsi sebagai pemindai perangkat I2C. Ia memeriksa setiap alamat dari 1 sampai 126 dan menampilkan alamat perangkat yang merespons. Ini sangat berguna saat kamu tidak yakin berapa alamat I2C dari sensor atau modul yang kamu gunakan.

**Source Code LCD:**

|  |
| --- |
| #include <Arduino.h>  #include <LiquidCrystal\_I2C.h>  int SDA\_pin = D2;  int SCL\_pin = D3;  LiquidCrystal\_I2C lcd(0x27, 16, 2); // 16 karakter 2 baris  void setup() {  Wire.begin(SDA\_pin, SCL\_pin);  lcd.init();  lcd.backlight();  lcd.setCursor(0, 0);  lcd.print("athifa♡♡♡");  lcd.setCursor(0, 1);  lcd.print("♡♡♡mas nuansa");  }  void loop() {  lcd.scrollDisplayLeft();  delay(200);  } |

**Penjelasan fungsi setiap blok kode:**

1. Library dan Inisialisasi

|  |
| --- |
| #include <Arduino.h>  #include <LiquidCrystal\_I2C.h> |

Penjelasan:

* #include <Arduino.h>:Menyertakan pustaka dasar Arduino.
* #include <LiquidCrystal\_I2C.h>: Menyertakan pustaka untuk mengendalikan LCD dengan antarmuka I2C.

1. Deklarasi Pin dan Objek LCD

|  |
| --- |
| int SDA\_pin = D2;  int SCL\_pin = D3;  LiquidCrystal\_I2C lcd(0x27, 16, 2); // 16 karakter, 2 baris |

Penjelasan:

* SDA\_pin dan SCL\_pin: Menentukan pin I2C (umumnya untuk ESP8266/NodeMCU).
* lcd(0x27, 16, 2): Membuat objek lcd dengan alamat I2C 0x27 dan ukuran layar 16 kolom x 2 baris.

1. Fungsi setup()

|  |
| --- |
| void setup() {  Wire.begin(SDA\_pin, SCL\_pin); // Memulai komunikasi I2C  lcd.init(); // Inisialisasi LCD  lcd.backlight(); // Menyalakan lampu latar LCD  lcd.setCursor(0, 0); // Posisi kursor di baris 0 kolom 0  lcd.print("athifa♡♡♡"); // Menampilkan teks di baris pertama  lcd.setCursor(0, 1); // Posisi kursor di baris 1 kolom 0  lcd.print("♡♡♡mas nuansa"); // Menampilkan teks di baris kedua  } |

Penjelasan:

* Baris ini menampilkan dua baris teks pada LCD saat program pertama kali dijalankan.
* Teks bersifat statis pada awalnya.

1. Fungsi `loop()`

|  |
| --- |
| void loop() {  lcd.scrollDisplayLeft(); // Menggeser seluruh tampilan LCD ke kiri  delay(200); // Menunggu 200 milidetik sebelum geser berikutnya  } |

Penjelasan:

* Membuat efek scroll horizontal ke kiri yang halus setiap 200 ms.
* Teks akan tampak bergerak dari kanan ke kiri secara terus-menerus.

1. Kesimpulan

Program ini akan menampilkan:

|  |
| --- |
| athifa♡♡♡  ♡♡♡mas nuansa |

Penjelasan:

dan akan menggeser seluruh tampilan ke kiri secara berkala, menciptakan efek teks berjalan (marquee) pada LCD.

**Alat rakit:**

