

Praktikum LAB 15

Diperuntukkan untuk memenuhi salah satu tugas praktikum Mata Kuliah Aplikasi Mikrokontroler



Praktikum	: Aplikasi Mikrokontroler
Praktikum ke	: 15
Tanggal Praktikum	: Kamis, 28 Januari 2021
Tanggal Pengumpulan Laporan	: Senin, 1 Februari 2021
Nama dan NIM	: 1. Amir Husein (181344003)
Kelas	: 3-TNK
Instruktur	: 1. Ferry Satria, BSEE., M.T 2. Rahmawati Hasanah, S.ST., M.T

Politeknik Negeri Bandung
Tahun Ajaran 2020/2021

I. TUJUAN

- Mahasiswa dapat memahami prinsip penggunaan perintah-perintah pada pemrograman C Arduino
- Mahasiswa mampu melakukan koneksi antara program pada Arduino IDE dengan simulator Proteus
- Mahasiswa mampu mengonversi nilai tegangan menjadi bentuk diskrit untuk diolah pada mikrokontroler Arduino.
- Mahasiswa mengerti mengenai penggunaan berbagai sensor pada Arduino.

II. LANDASAN TEORI

1. Arduino

Arduino adalah sebuah minimum sistem mikrokontroler bersifat open-source yang banyak digunakan untuk membangun sebuah project elektronika. Platform Arduino berisi dua yaitu hardware berupa board dan sebuah software atau IDE (Integrated Development Environment) yang berjalan pada komputer, digunakan untuk menulis dan mengisikan program ke board Arduino.

Platform Arduino menjadi sangat populer bagi orang-orang yang ingin memulai belajar elektronika khususnya mikrokontroler, karena dengan menggunakan Arduino tidak lagi memerlukan hardware tambahan (sering disebut downloader) untuk mengisikan program kedalam board mikrokontroler, tapi hanya perlu kabel USB saja yang disambungkan dari komputer ke board Arduino. Bahasa pemrograman yang dituliskan pada Arduino IDE menggunakan bahasa pemrograman C++ yang telah disederhanakan, sehingga dapat lebih mudah dimengerti. Sebuah board Arduino didesain dengan standar bentuk board serta posisi dan susunan pin/port sehingga dapat lebih mudah digunakan dan diakses dengan perangkat lain.

2. Arduino IDE

IDE itu merupakan kependekan dari Integrated Development Environment, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (Sketch) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama Bootlader yang berfungsi sebagai penengah antara compiler Arduino dengan mikrokontroler.

Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan library C/C++ yang biasa disebut Wiring yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari software Processing yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino.

3. Array

Array merupakan salah satu tipe data pada pemrograman C di Arduino. Array merupakan susunan data yang terindeks, dengan indeks pertama diawali 0, serta data pada array bisa berupa int, string, bool, maupun lainnya. Data pada array bisa diinisialisasi langsung dengan menggunakan simbol kurung kurawal '{}', serta untuk melakukan pointing indeks dapat menggunakan kurung siku setelah variabel array itu sendiri. Seperti contoh terdapat array 'abjad' sebagai berikut yaitu `abjad[] = {a,c,g,j,n,d,a}`, maka untuk mendapatkan nilai dari g, dapat dipanggil dengan cara melakukan pointing dengan syntax `abjad[2]`.

4. Sensor

Sensor merupakan salah satu device elektronika yang sering ditemui dan digunakan pada berbagai macam produk elektronika yang memerlukan pengolahan input. Sensor ini berfungsi sebagai input bagi mikrokontroler, yang nanti datanya ini dapat diproses pada mikrokontroler dengan berbagai output yang telah diatur sedemikian rupa. Sensor pada dasarnya terbagi dua, yaitu sensor analog dan digital. Pada sensor analog, masukan input berupa nilai tegangan yang bervariasi, sedangkan pada sensor digital nilai input berupa dua level saja, yaitu HIGH dan LOW. Dibawah ini merupakan beberapa sensor yang biasa digunakan bersama dengan Arduino, seperti sensor jarak ultrasonik, sensor suhu, sensor cahaya (LDR), dan masih banyak lagi.



III. SOAL

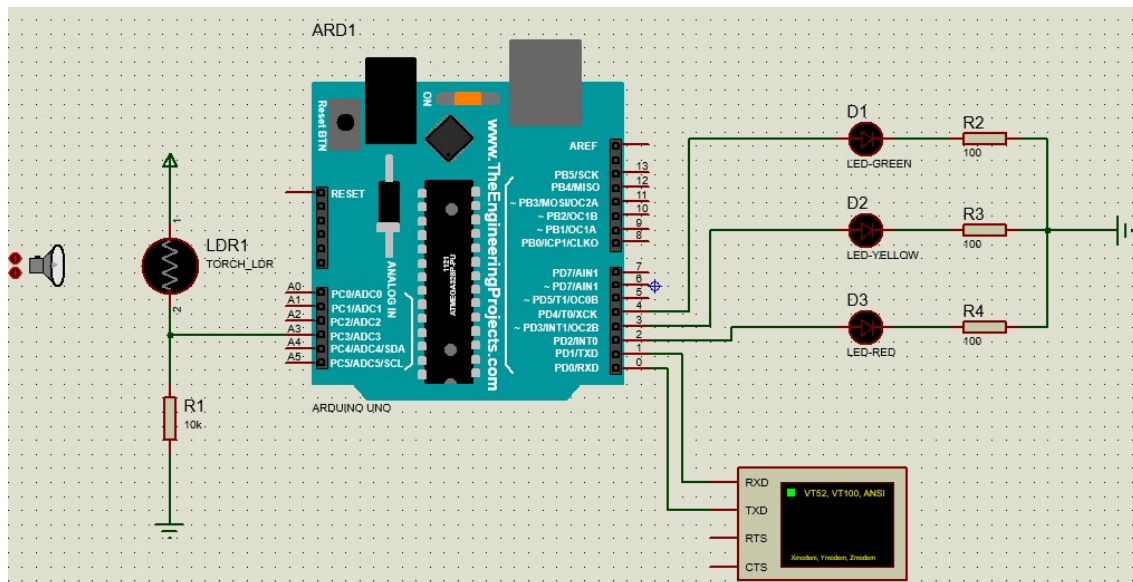
1. Buatlah rangkaian arduino yang terhubung dengan 1 buah sensor LDR dan 3 buah LED (kuning, hijau dan merah). Buat program agar ketika nilai LDR > 100 maka LED hijau menyala, nilai LDR > 300 maka LED kuning menyala, dan ketika nilai LDR > 500 maka LED merah menyala.
2. Buatlah simulasi alarm kebakaran sederhana, dengan rangkaian arduino yang terhubung dengan 1 buah sensor suhu LM35, 1 buah LCD dan 1 buah buzzer. Buatlah programnya agar nilai yang dideteksi oleh sensor suhu tampil di LCD dengan kalimat : "Suhu sekarang: C" dan ketika nilai suhu > 40 C maka buzzer menyala.
3. Buatlah simulasi sistem parkir sederhana, dengan rangkaian arduino yang terhubung dengan 1 buah sensor ultrasonik(sensor jarak), 1 buah LCD dan 1 buah buzzer. Buatlah programnya agar nilai yang dideteksi oleh sensor ultrasonik tampil di LCD dengan kalimat : "Jarak: cm" dan ketika nilai jarak < 30 cm maka buzzer bersuara.

IV. HASIL PROGRAM

1. Program :

```
const int ldr = A3;
const int led[] = {4,3,2};
void setup() {
    pinMode(ldr, INPUT);
    for(int i=0;i<3;i++){
        pinMode(led[i], OUTPUT);
    }
    Serial.begin(9600);
}
void loop() {
    int ldrIndex = analogRead(ldr);
    Serial.println(ldrIndex);
    delay(500);
    (ldrIndex > 100) ?
        ((ldrIndex > 300) ?
            ((ldrIndex > 500) ? ledConfig(Low,Low,High) :
                ledConfig(Low,High,Low)) :
                ledConfig(High,Low,Low)) :
                ledConfig(Low,Low,Low);
}
void ledConfig(bool ledStat1, bool ledStat2, bool ledStat3){
    digitalWrite(led[0], ledStat1);
    digitalWrite(led[1], ledStat2);
    digitalWrite(led[2], ledStat3);
}
```

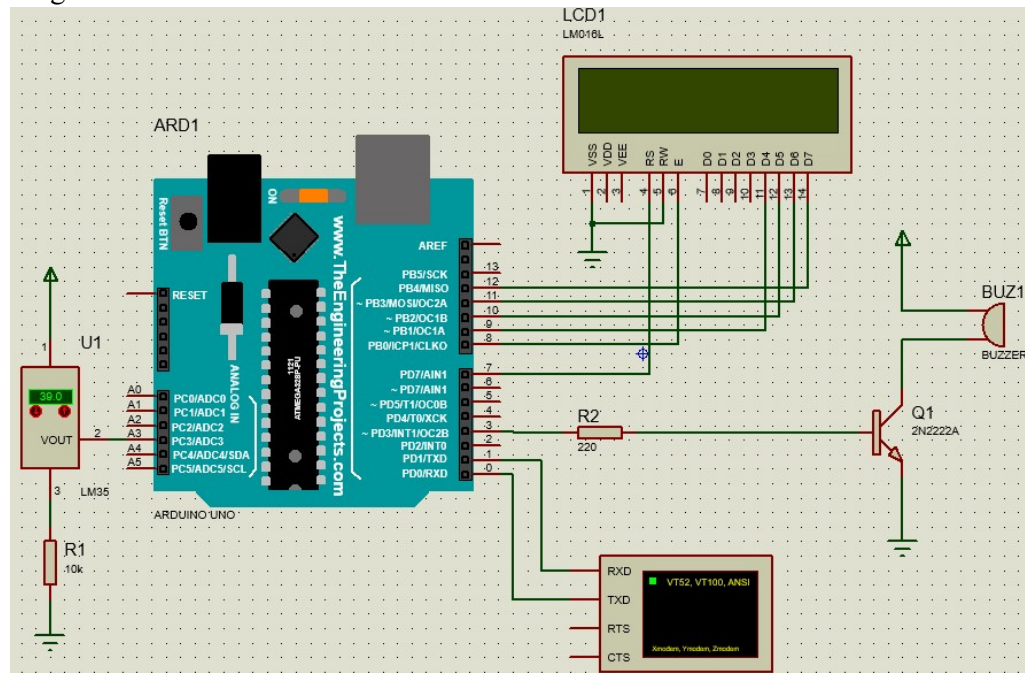
Rangkaian:



2. Program:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,8,9,10,11,12);
const int buz = 3;
const int lm = A3;
void setup() {
    lcd.begin(16,2);
    pinMode(buz, OUTPUT);
    pinMode(lm, INPUT);
    Serial.begin(9600);
}
void loop() {
    int suhu = analogRead(lm) * 0.46828125;
    lcd.setCursor(0,0);
    lcd.print("Suhu Sekarang:");
    lcd.setCursor(0,1);
    lcd.print(suhu);
    lcd.print(" C");
    delay(250);
    lcd.clear();
    (suhu > 40) ? digitalWrite(buz, HIGH) :
    digitalWrite(buz, LOW);
}
```

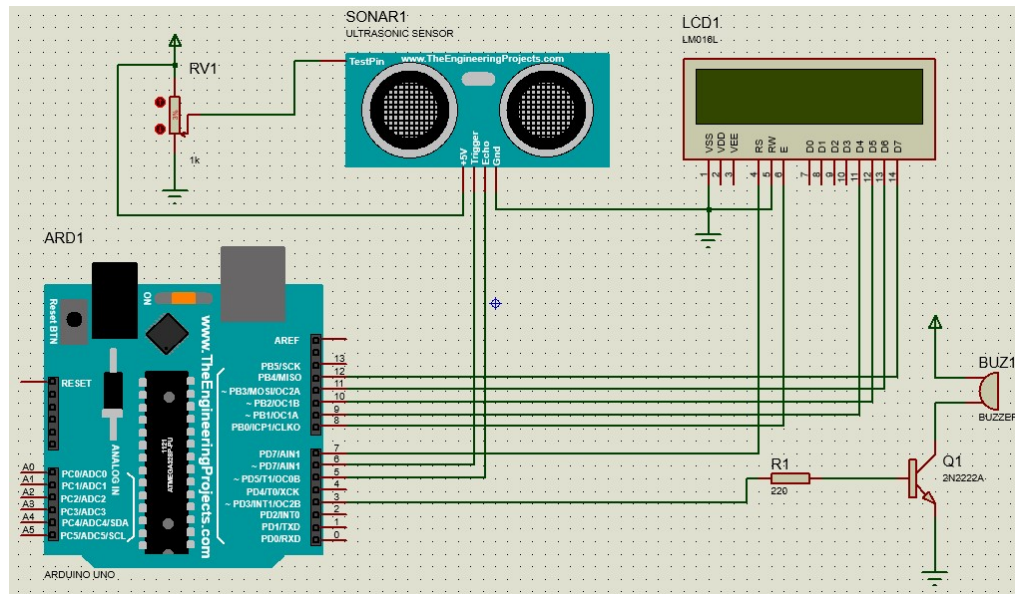
Rangkaian:



3. Program:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,8,9,10,11,12);
const int buz = 3;
const int trigPin = 6;
const int echoPin = 5;
void setup() {
  lcd.begin(16,2);
  pinMode(buz, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial.begin(9600);
}
void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Set trigPin HIGH selama 10uS
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  long duration = pulseIn(echoPin, HIGH);
  int jarak = duration*0.034/2;
  lcd.setCursor(0,0);
  lcd.print("Jarak:");
  lcd.setCursor(0,1);
  lcd.print(jarak);
  lcd.print(" cm");
  delay(250);
  lcd.clear();
  (jarak < 30) ? digitalWrite(buz, HIGH) :
  digitalWrite(buz, LOW);
}
```

Rangkaian:



V. ANALISIS

1. Pada program ini, pertama-tama adalah menginisialisasi pin untuk LDR dan LED yaitu A3 untuk LDR dan 4,3,dan 2 untuk LED. Pada fungsi **void setup()**, diatur bahwa pin LDR sebagai INPUT dan pin LED sebagai OUTPUT dengan memanfaatkan fungsi **pinMode()**.

Sebelum menuju ke fungsi **void loop()**, dibuat terlebih dahulu sebuah fungsi tambahan **ledConfig** dengan tiga buah parameter bertipe bool untuk status masing-masing LED dari LED pertama sampai terakhir. Kemudian pada fungsi **void loop()**, terdapat variabel **ldrIndex** yang berisi status INPUT dari pin LDR dengan memanfaatkan fungsi **analogRead()**, kemudian disertai delay selama 0.5 detik. Setelah itu dilakukan pengondisian dengan menggunakan ternary conditional, apabila **ldrIndex** > 100, jika tidak memenuhi maka dipanggil **ledConfig** dengan semua parameter LOW, lalu dilakukan pengecekan lagi **ldrIndex** > 300, jika tak memenuhi maka LED pertama akan HIGH, kemudian dilakukan lagi pengecekan apabila **ldrIndex** > 500, jika tidak memenuhi maka haya LED 2 yang HIGH, begitupun jika memenuhi, maka LED 3 bernilai HIGH.

2. Pada program ini, pertama-tama dilakukan impor library LCD, kemudian diinisialisasi variabel **lcd** dengan pin-pin berupa pin RS, E, D4, D5, D6, dan D7 secara berurutan. Kemudian dibuat variabel **buz** dengan nilai 3 yaitu pin nomor 3 arduino terhubung ke buzzer, begitupun variabel **lm** dengan nilai A3.

Pada fungsi **void setup()**, dipanggil sintaks **lcd.begin(16,2)**, karena yang digunakan pada praktikum ini adalah LCD 16x2, kemudian diatur bahwa pin buzzer akan bertindak sebagai OUTPUT dan pin sensor suhu **lm** sebagai INPUT. Pada fungsi **void loop()**, dibuat sebuah variabel baru bertipe integer bernama **suhu**, dengan nilai berupa hasil dari level tegangan **analogRead(lm)**, dikalikan dengan 0.4682. Nilai 0.4682 ini didapat dari nilai tegangan 500mV dibagi dengan 1023 (batas akhir konversi analog ke digital). Lalu set cursor LCD ke (0,0) dilanjutkan perintah **lcd.print("Suhu Sekarang:")** untuk dibaris pertama. Setelah itu set cursor ke baris selanjutnya untuk melakukan pencetakan nilai variabel **suhu** disertai huruf "C" dengan delay 250 ms dan perintah **lcd.clear()** untuk melakukan pembersihan pada LCD. Pada baris akhir, dibuat sebuah pengondisian apabila nilai suhu diatas 40, maka buzzer akan memiliki level HIGH sehingga berbunyi, bila tidak memenuhi, maka buzzer mati.

3. Pada program ini, pertama-tama dilakukan impor library LCD, kemudian diinisialisasi variabel **lcd** dengan pin-pin berupa pin RS, E, D4, D5, D6, dan D7 secara berurutan. Kemudian dibuat variabel **buz** dengan nilai 3 yaitu pin nomor 3 arduino terhubung ke buzzer, variabel **trigPin** dari sensor jarak ke pin nomor 6 dan **echoPin** ke nomor 5.

Pada fungsi **void setup()**, dipanggil sintaks **lcd.begin(16,2)**, karena yang digunakan pada praktikum ini adalah LCD 16x2, kemudian diatur bahwa pin buzzer akan bertindak sebagai OUTPUT, pin trigger sensor jarak sebagai OUTPUT, dan pin echo sebagai INPUT. Pada fungsi **void loop()**, pin **trigPin** akan di set ke HIGH selama 10us yang bertujuan untuk mengirim gelombang ultrasonic selama 10us yang kemudian di

set lagi ke LOW setelahnya. Gelombang ini tentu akan diterima oleh pin **echoPin** pada sensor jarak ultrasonic, yang kemudian akan dihitung berapa lama durasi dari dikirim sampai ke penerima yaitu **echoPin**. Nilai waktu ini disimpan pada variabel **duration** bertipe long dengan memanfaatkan fungsi **pulseIn**. Kemudian dibuat sebuah variabel **jarak** bertipe integer yang nilainya adalah **duration*(0.034/2)**. Untuk pencetakan pada layar LCD, sama seperti sebelumnya, hanya saja variabel terikatnya menjadi variabel **jarak**, serta teks pada baris pertama diubah menjadi “Jarak”, dan pada baris kedua menjadi “cm”. Setelah itu, langkah terakhir ialah membuat suatu pengondisian yang apabila nilai jarak < 30, maka buzzer akan memiliki level HIGH sehingga berbunyi, bila tidak memenuhi, maka buzzer mati.

VI. KESIMPULAN

Pada praktikum ini dapat diambil kesimpulan bahwa pada pemrograman Arduino menggunakan bahasa C, dapat digunakan beberapa perintah seperti halnya pemrograman C seperti biasa, yaitu seperti penggunaan IF, IF ELSE, FOR, dan WHILE LOOP. Selain itu juga, simulator Proteus dapat digunakan sebagai sarana dalam mengamati hasil compile program C Arduino tersebut, tanpa bersusah payah untuk menghubungkan perangkat Arduino langsung ke PC. Selain itu, penggunaan sensor pada Arduino dapat memudahkan dalam melakukan analisis dan pemrosesan data input untuk berbagai kebutuhan yang sangat beragam. Sensor juga terbagi menjadi dua, yaitu sensor analog dan digital dimana level tegangan yang diukur tentu berbeda. Pada sensor analog, nilai level tegangan bervariasi, sedangkan pada sensor digital level tegangannya hanya ada HIGH dan LOW.

VII. DAFTAR PUSTAKA

- Anonim. 2019. *Apa itu Arduino?*. Bluino. Diakses pada 11 Desember 2020. https://www.bluino.com/2019/09/apa-itu-arduino_13.html
- Sinau. 2016. *Mengenal Arduino Software (IDE)*. Sinau Arduino. Diakses pada 11 Desember 2020. <https://www.sinauarduino.com/artikel/mengenal-arduino-software-ide/>