

Workshop Aplikasi Mikroprosesor & Antarmuka

**PROGRAM STUDI
TEKNIK TELKOMUNIKASI**

Akuwan Saleh, MT

PENILAIAN

- ⇒ Laporan + Tugas + Presentasi = **60%**
 - eval-1(Lap.1-5) = 20%
 - eval-2(Lap.6-10) = 20%
 - eval-3(Lap.11-15 & (PPT+ presentasi)) = 20%
- ⇒ TPS = **40%**
 - eval-4 (TPS = Tugas Proyek Semester)

REFERENSI

- Rui Santos & Sara Santos, “ESP32 Web Server With Arduino IDE.pdf: Step By Step Project Guide”, <https://randomnerdtutorials.com/esp32-web-server-arduino-ide/>, juni 2020.
- Julien Bayle, “C Programming for Arduino”, Packt Publishing Ltd, Birmingham, May 2013
- Fabian Winkler, “Arduino/Processing Communication Workshop”, Fall, 2013.
- Jack Purdum, “Beginning C for Arduino, Learn C Programming for the Arduino and Compatible Microcontrollers”, Apress, 2012.
- John-David Warren, Josh Adams, and Harald Molle, “Arduino Robotics”, Springer, New York, 2011.
- Casey Reas and Ben Fry, “Getting Started with Processing”, O’Reilly Media, Inc., June 2010.
- Joshua Noble, “Programming Interactivity: A Designer’s Guide to Processing, Arduino, and openFrameworks”, O’Reilly Media, Inc., July 2009.
- Peter Hoddie, Lizzie Prader, “IoT Development for ESP32 and ESP8266 with JavaScript; A Practical Guide to XS and Moddable SDK”, Apress, Menlo Park, CA,USA, 2020.

MATERI

PENDAHULUAN

1. KOMUNIKASI MIKROKONTROLER DENGAN SOFTWARE PROCESSING
2. ANALOG INPUT DAN AUDIO PROCESSING
3. KONTROL MULTI LED MENGGUNAKAN ARDUINO DAN PROCESSING
4. PENCAMPUR WARNA VIRTUAL MENGGUNAKAN ARDUINO DAN PROCESSING
5. MONITORING SUHU DENGAN ARDUINO DAN PROCESSING
6. **MONITORING INTENSITAS CAHAYA DENGAN ARDUINO DAN PROCESSING**
7. KONTROL MOTOR DC MENGGUNAKAN ARDUINO DAN PROCESSING
8. APLIKASI SENSOR ULTRASONIC MENGGUNAKAN ARDUINO DAN PROCESSING

MATERI

9. KONTROL LAMPU AC 220 V BERBASIS ARDUINO DAN PROCESSING
10. MODUL WiFi ESP32 DENGAN ARDUINO IDE
11. KOMUNIKASI NIRKABEL MENGGUNAKAN MODUL RF 434 MHz DAN PROCESSING
12. ESP32 WEB SERVER UNTUK KONTROL LED DAN MENAMPILKAN GAMBAR
13. ANTARMUKA MODUL GPS DENGAN MIKROKONTROLER DAN PROCESSING
14. ESP32 WEB SERVER UNTUK PENGUKURAN SUHU DAN KELEMBABAN
15. KOMUNIKASI DATA BERBASIS BLUETOOTH DAN HP

6. MONITORING INTENSITAS CAHAYA DENGAN ARDUINO DAN PROCESSING

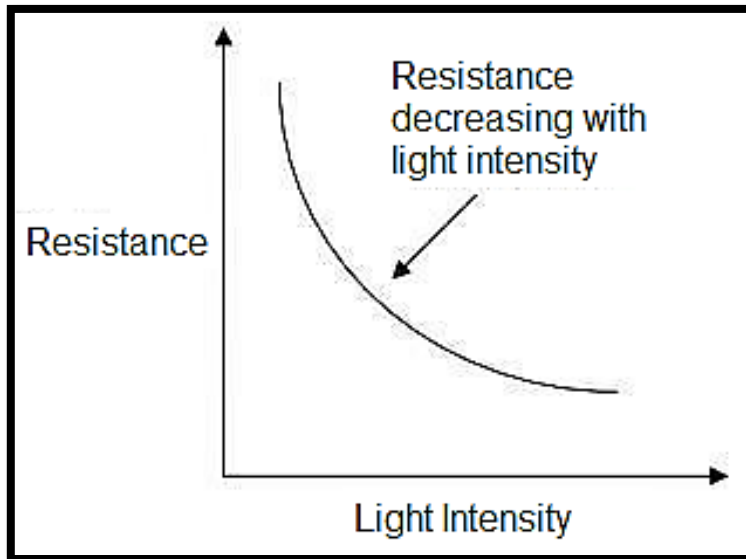
TUJUAN

- Membaca nilai Intensitas Cahaya menggunakan sensor LDR dan arduino
- Memantau perubahan nilai intensitas dan menampilkan dalam bentuk grafis menggunakan processing

DASAR TEORI

(Antarmuka LDR)

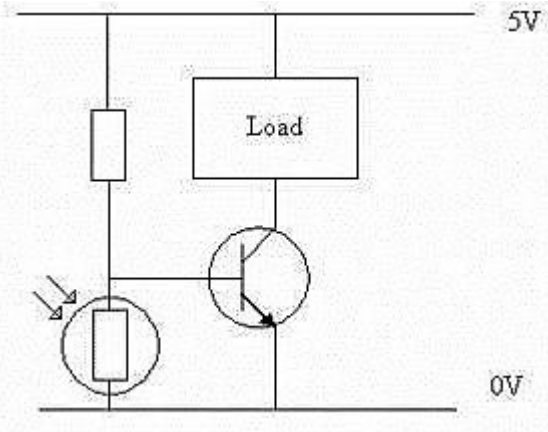
- Variasi resistansi LDR terhadap perubahan intensitas cahaya



- Jenis LDR yang paling umum memiliki resistansi yang turun dengan adanya peningkatan intensitas cahaya.

Contoh :

Kontrol Rangkaian Transistor menggunakan LDR



- Rangkaian ini menunjukkan cara sederhana untuk mengaktifkan sistem saat gelap. LDR bertindak sebagai variabel resistor dengan perubahan intensitas cahaya.
- Resistor LDR dan Resistor lainnya membentuk rangkaian '*Potential Divider*' sederhana, di mana titik pusat pembagi tegangan diumpankan ke Basis Transistor NPN.

- Bila tingkat cahaya turun, resistansi LDR akan meningkat. Karena resistensi ini meningkat dalam kaitannya dengan Resistor lain, yang memiliki nilai tahanan tetap, hal itu menyebabkan tegangan turun di LDR juga meningkat. Bila tegangan ini cukup besar (0.7V untuk transistor NPN), maka akan menyebabkan Transistor aktif/on.
- LDR sebagai sensor terhadap cahaya dan Arduino sebagai pemroses data dari LDR dan memberi perintah pada LED untuk menyala atau padam.

- Arduino akan membaca nilai perubahan dari sensor LDR melalui pin Analog. Nilai Analog LDR ini akan ditampilkan di serial monitor dalam tampilan nilai Desimal. Dari nilai ini akan dibuat batasan nilai kondisi Lampu menyala atau padam.

Contoh :

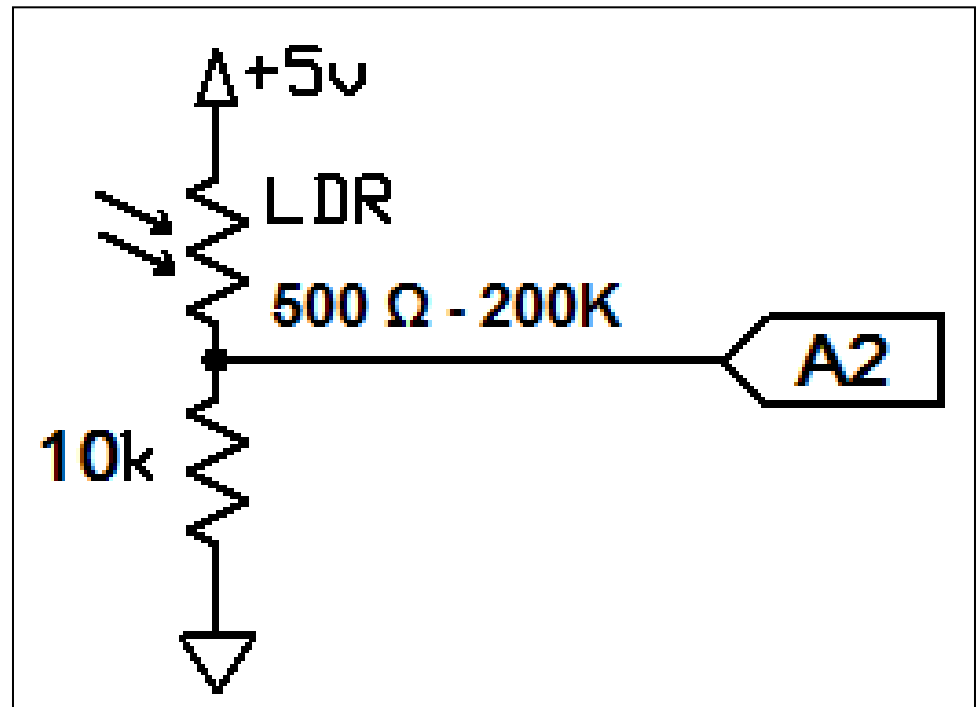


```
x = sensorValue;
if(x>475)
{
    digitalWrite(ledPins,HIGH);
    delay(1000);
}
if(x<475)
{
    digitalWrite(ledPins,LOW);
    delay(1000);
}
```

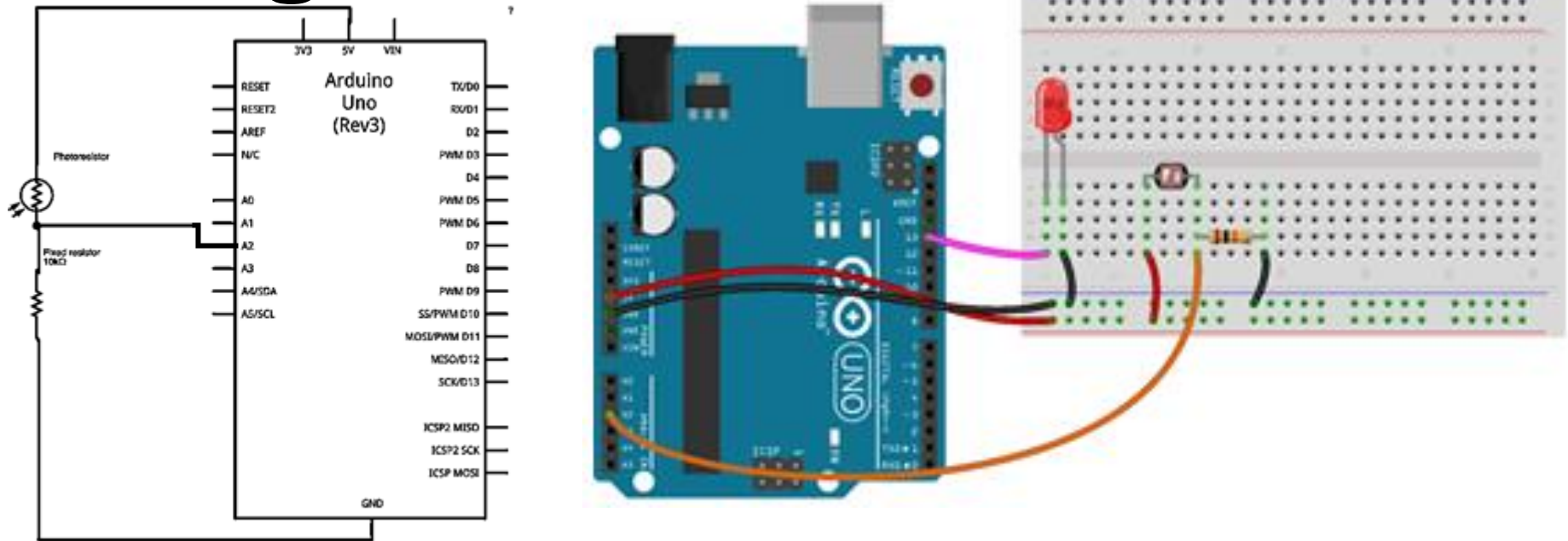
- Arduino mengubah data yang diterima ke dalam berbagai tingkat diskrit. Misalnya dari 0 sampai 1023.
- Untuk data tertentu maka 0 mewakili kegelapan maksimum dan 1023 mewakili paling terang sehingga cahaya diterima diubah menjadi salah satu nilai diskrit.
- Hasilnya LED saat setengah dari terang yaitu tingkat diskrit 512 maka output = 2,5 v.
- Processing mengolah nilai diskrit 0-1023 menjadi bentuk grafis, sehingga perubahan nilai intensitas cahaya dapat dipantau secara *realtime*.

Hardware :

- Arduino Uno Board
- 1x Breadboard
- 1x Sensor cahaya LDR
- 1x Resistor 10k
- Kabel jumper



Rangkaian:



- Hubungkan 5V dan GND dari Arduino ke Breadboard.
- Hubungkan kaki kiri LDR ke 5V.
- Hubungkan kaki kanan LDR ke pin A2 Arduino.
- Hubungkan kaki kiri resistor ke celah antara kaki kanan LDR dan GND arduino.
- Hubungkan kaki kanan resistor ke GND.
- Hubungkan LED dengan pin 13 Arduino.

PROGRAM-1:

Arduino Uno

```
int sensorPin = A2;  
int ledPins=13;  
int val = 0;  
void setup() {  
  Serial.begin(9600);  
  pinMode(ledPins,OUTPUT);  
}  
void loop() {  
  val = analogRead(sensorPin);  
  Serial.println(val);  
  digitalWrite(ledPins, HIGH);  
  delay(300);  
  digitalWrite(ledPins, LOW);  
  delay(300);  
}
```

PROGRAM-1:

Processing

```
import processing.serial.*;
Serial portSerial;
int dataSekarang;
int inByte;
int[] nilaiY;
int w;

void setup()
{
  size (700,400);
  w = width-10;
  strokeWeight(3);
  smooth();
  nilaiY = new int[w];
  //sesuaikan dengan nomer COM kita
  portSerial = new Serial(this, "COM4", 9600);
}

void draw()
{
  String inString = portSerial.readStringUntil('\n');
  if(inString != null){
    inString = trim(inString);
    inByte = int(inString);
    println(inByte);
    dataSekarang = int(map(inByte, 0, 1023, 0+10, height-50));
    dataSekarang = height-dataSekarang;
    background(0);
    for(int i = 1; i < w; i++) {
      nilaiY[i-1] = nilaiY[i];
    }
  }
}
```


PROGRAM-1:

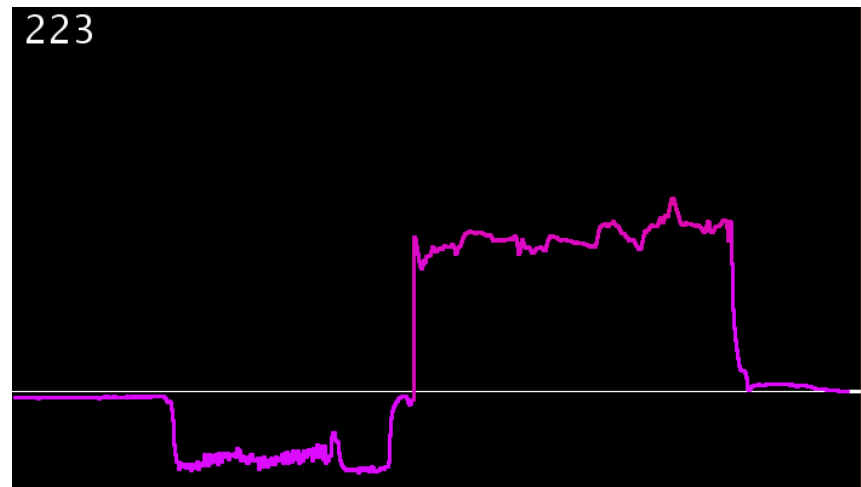
Processing

```
nilaiY[w-1] = dataSekarang;
```

```
textSize(32);  
text(inByte, 10, 30);
```

```
stroke(255);  
line(w, dataSekarang, width, dataSekarang);  
strokeWeight(1);  
line(0, dataSekarang, width, dataSekarang);  
strokeWeight(3);
```

```
for(int i=2; i<w; i++) {  
  strokeWeight(3);  
  stroke(220, 10, nilaiY[i]);  
  line(i, nilaiY[i-1], i, nilaiY[i]);  
}  
}  
}
```



DASAR TEORI

(Antarmuka Piezo Buzzer)

- Piezo/speaker untuk mendeteksi suara, sebagai sensor suara/sound.
- Memanfaatkan kemampuan prosesor untuk membaca sinyal analog melalui konverter ADC - analog ke digital.
- Piezo perangkat elektronik digunakan untuk memutar nada dan mendeteksi nada.
- Piezo membaca nilai antara 0 sampai 5volts, dan bukan hanya TINGGI atau LOW.

- Piezo memiliki polaritas, biasanya memiliki kabel merah dan hitam. Menghubungkan yang hitam ke ground dan yang merah ke pin Arduino.
- Dengan memberikan nilai frekuensi tertentu pada Piezo akan didapatkan sebuah Tone.

```
tone(piezoPin,500); // frekuensi 500Hz
```

- Perubahan volume yang tiba-tiba adalah elemen suara yang paling mudah dibaca, namun memproses gelombang suara dengan perangkat lunak memungkinkan untuk mendeteksi frekuensi atau ritme tertentu.

Hardware :

Arduino Uno Board

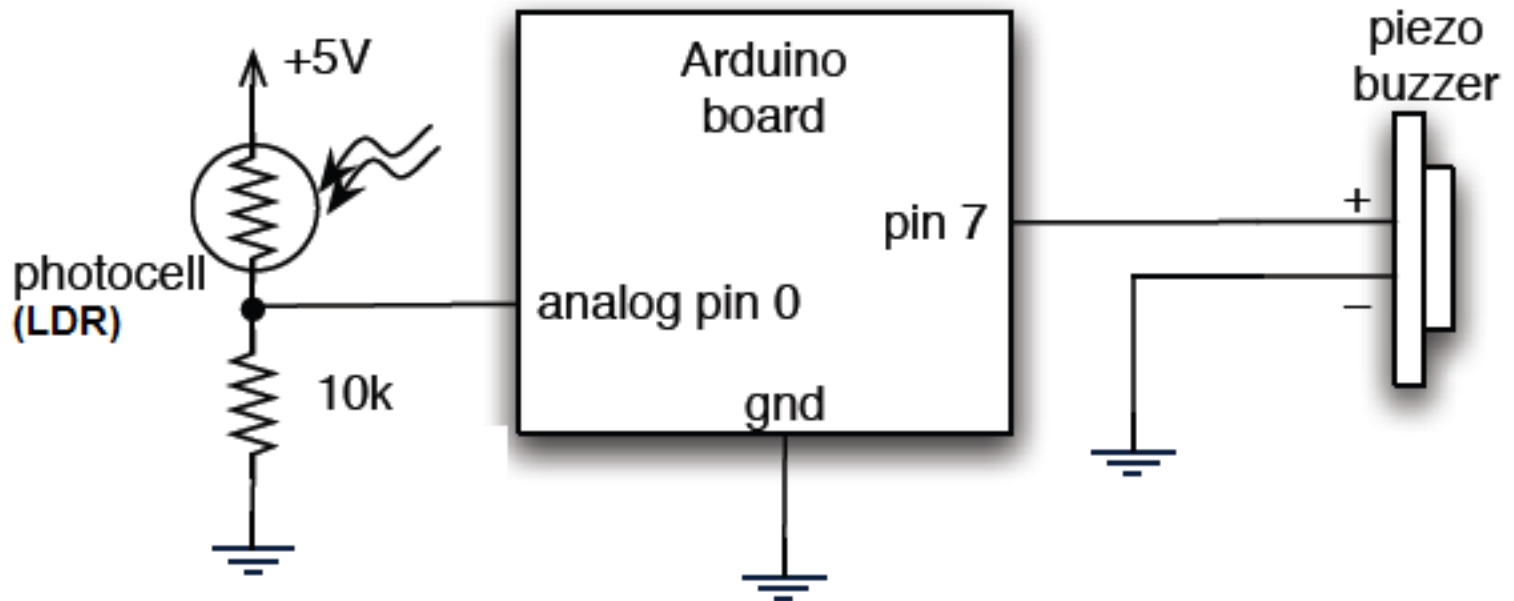
1x Sensor cahaya LDR

1x Resistor 10k

Kabel jumper

- Piezo buzzer (speaker)

Rangkaian 2:



PROGRAM-2:

Arduino Uno

```
int piezoPin = 7;  
int ldrPin = 0;  
int ldrValue = 0;  
void setup()  
{  
  Serial.begin(9600);  
}  
void loop()  
{  
  ldrValue = analogRead(ldrPin);  
  tone(piezoPin,500);          // 500Hz  
  delay(25);  
  noTone(piezoPin);  
  delay(ldrValue);  
  Serial.println(ldrValue);  
}
```

PROGRAM-2:

Processing

```
import processing.serial.*;
Serial port;
float val;
int x;
float easing = 0.05;
float easedVal;
void setup() {
    size(440, 440);
    frameRate(30);
    smooth();
    String arduinoPort = Serial.list()[0];
    port = new Serial(this, arduinoPort, 9600);
    background(0);
}
void draw() {
    if ( port.available() > 0) {
```

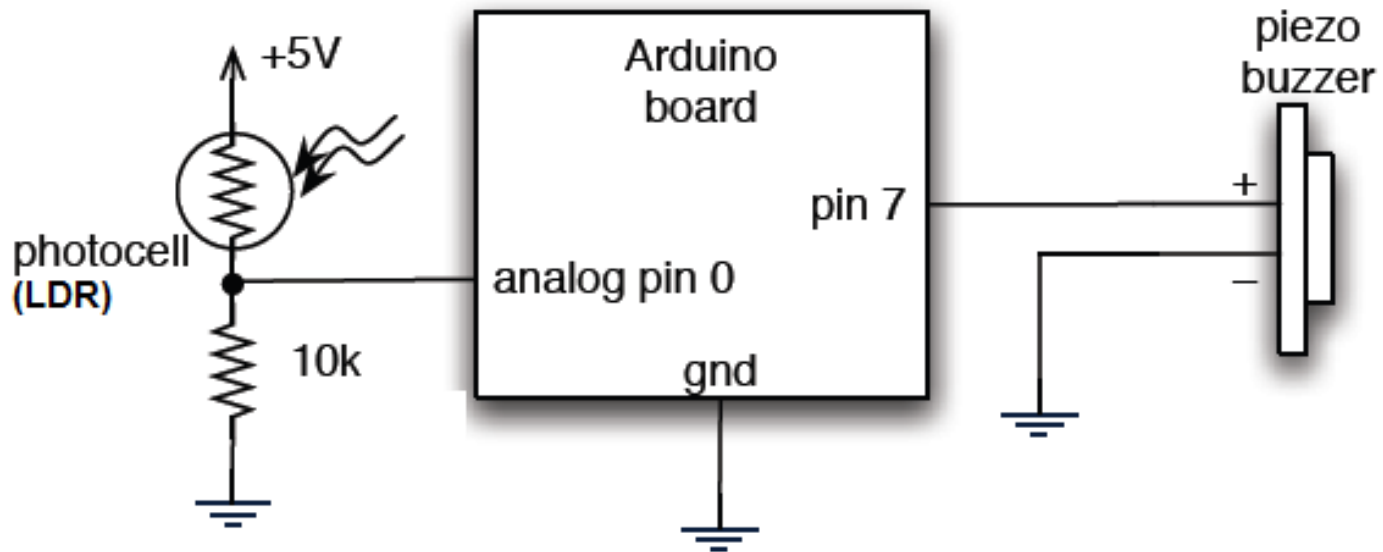
PROGRAM-2: (lanjutan)

Processing

```
    val = port.read();
    val = map(val, 0, 255, 0, height);
  }
float targetVal = val;
easedVal += (targetVal - easedVal) * easing;
stroke(0);
line(x, 0, x, height);
stroke(255);
line(x+1, 0, x+1, height);
line(x, 220, x, val);
line(x, 440, x, easedVal + 220);
x++;
    if (x > width) {
      x = 0;
    }
}
```

Latihan :

1. Buat aplikasi menggunakan rangkaian di bawah untuk alarm. Jika kondisi gelap (nilai data < 300) alarm akan berbunyi dengan frekuensi 1000Hz.



2. Buat aplikasi menggunakan rangkaian di bawah ini, untuk menampilkan nilai intensitas cahaya pada LED yang menyala redup - terang sesuai dengan grafik sinyal sebagai berikut.

