

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

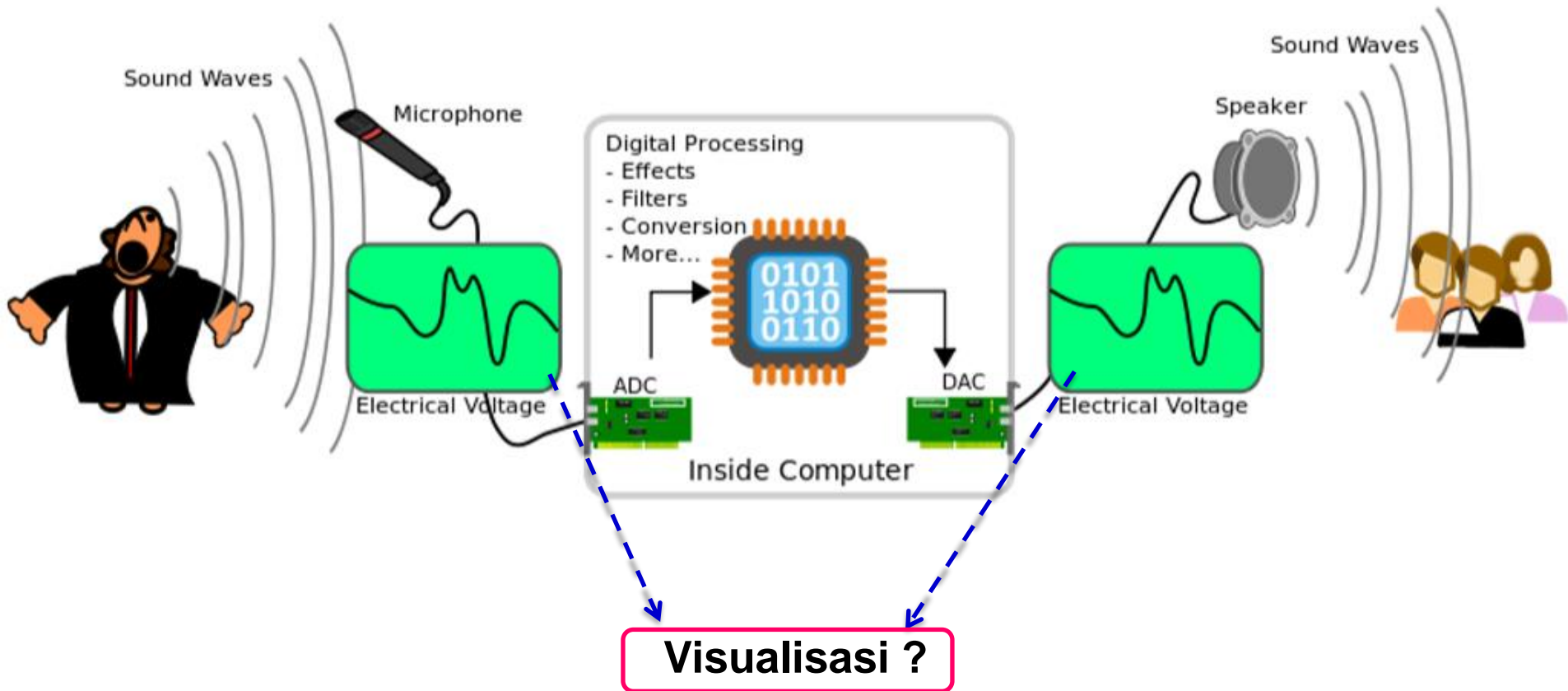
2. ANALOG INPUT DAN AUDIO PROCESSING

TUJUAN

- Membaca data yang masuk pada pin Analog Input Arduino.
- Memvisualisasikan data dari pin Analog Input dalam bentuk grafik.
- Memvisualisasikan data file audio dalam bentuk grafik.

DASAR TEORI

Relasi ADC dan DAC



DASAR TEORI

➤ ANALOG INPUT DENGAN PROCESSING

- **Visualisasi** merupakan salah satu proses yang penting dalam memaknai suatu data.
- Data yang **divisualkan** dalam berbagai bentuk grafik yang sesuai (**grafik garis, batang, pie, dsb.**) dapat mempermudah pengguna dalam menganalisis data.
- Dalam kaitannya dengan **Arduino**, pengguna dapat **memvisualisasikan** data tersebut menggunakan **Processing**.

DASAR TEORI

- Menggunakan sebuah **Arduino Uno** dan **potensiometer** sebagai sensor.
- **Potensiometer** digunakan untuk memodelkan penggunaan semua jenis sensor analog.
- Baca sensor atau perangkat input dan kirim nilainya sebagai byte melalui protokol serial ke PC → Processing.
- **Potensiometer** untuk mengatur obyek pada Processing melalui Arduino

DASAR TEORI

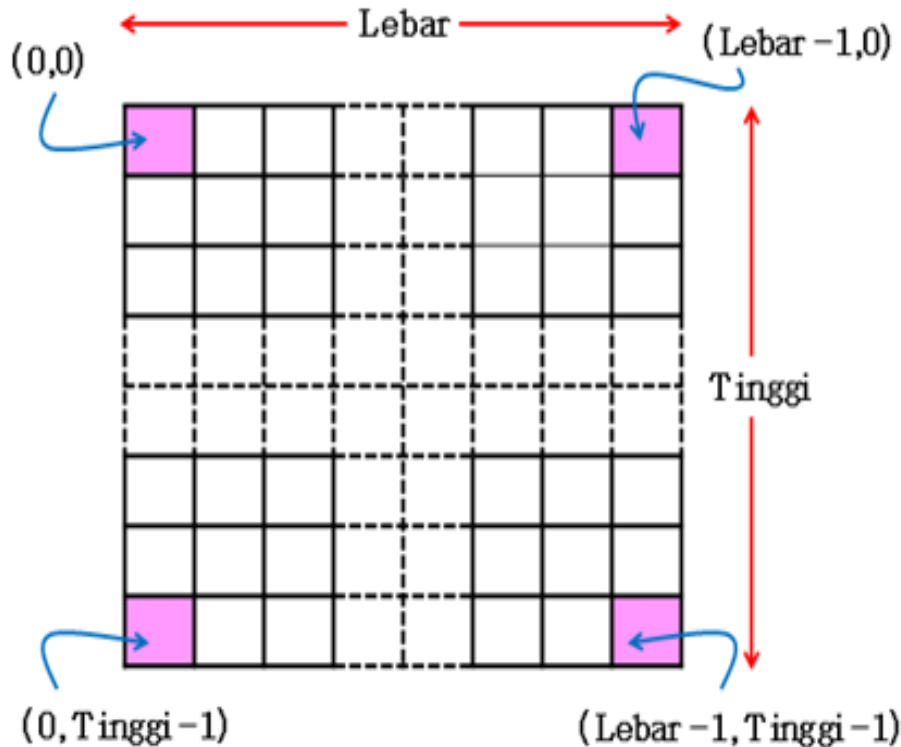
➤ KONSEP PIKSEL

- Sebuah obyek dapat dipadukan dengan obyek yang lain.
- Selanjutnya, obyek - obyek ini dapat dibuatkan animasi. Lebih jauh lagi, obyek - obyek ini dapat berinteraksi dengan lingkungan di luar komputer.
- Menggambar pada layar komputer sama halnya dengan menggambar pada kertas. Harus memulai suatu obyek dari suatu sudut atau titik tertentu pada bidang kerja.

DASAR TEORI

- Pada layar komputer, elemen-elemen terkecil pembentuk bidang kerja dikenal sebagai piksel. Piksel ini dapat diilustrasikan seperti grid kertas kotak.
- Posisi dari tiap-tiap kotak ditentukan oleh koordinat x dan y-nya. Pada Processing, koordinat-x adalah jarak dari sisi kiri jendela display (**Display Window**), dan koordinat-y adalah jarak dari sisi atas jendela display.

DASAR TEORI



Pada Gambar tersebut, piksel $(0,0)$ diawali dari sisi kiri-atas, dan akan berakhir pada sisi kanan-bawah. Indeks dimulai dari 0, bukan dari 1. layar berukuran 200×300 piksel, maka sudut kiri-atas memiliki koordinat $(0,0)$, dan sudut kanan-bawah memiliki koordinat (199×299) .

Pada ukuran 200×300 piksel, nilai 200 menyatakan Lebar layar, sedangkan nilai 300 menyatakan Tinggi layar.

DASAR TEORI

➤ PLOTING DATA ARDUINO MENGGUNAKAN PROCESSING

Gelombang Sinus

- Membangkitkan sebuah gelombang sinus pada Arduino dan memvisualisasikannya pada Processing.
- Gelombang sinus memiliki nilai yang dapat bergerak dari -1 sampai 1 dengan nilai 0 sebagai sumbu tengah. Karena nilai yang akan dikirim memiliki nilai dari 0 sampai 1023 (**ingat**, resolusi ADC pada Arduino Uno 10-bit ($2^{10} = 1024$)).

DASAR TEORI

➤ PLOTING DATA ARDUINO MENGGUNAKAN PROCESSING

Gelombang Sinus

- Dengan menggeser agar semua komponen sehingga memiliki nilai positif. Sumbu utama (nilai 0) akan digeser menjadi 511 (hampir setengah dari nilai maksimal), dan selanjutnya gelombang akan bergerak dari 0 sampai mendekati 1023.

PROGRAM: *(Gelombang Sinus)*

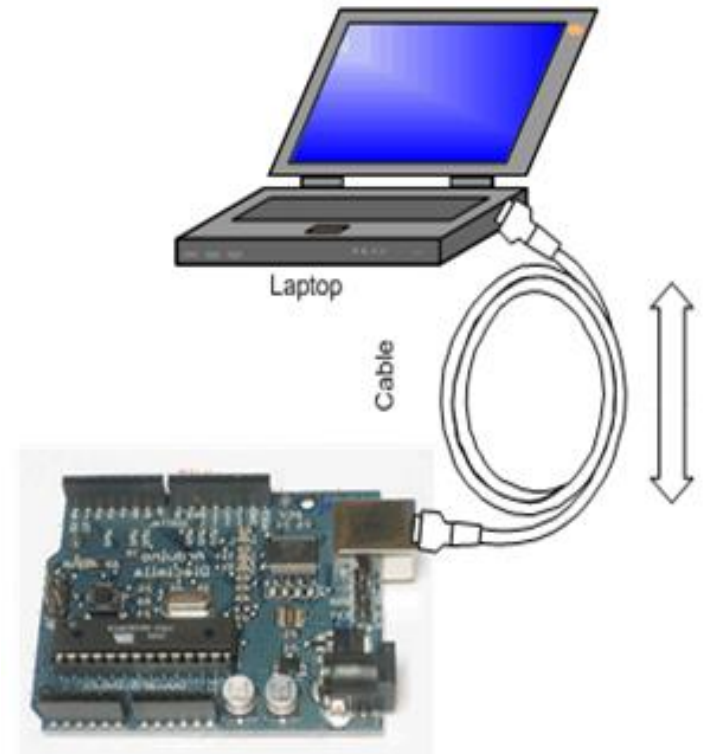
Arduino Uno

```
double x,y;
void setup() {
  Serial.begin(9600);
  x = 0;
}

void loop() {
  Serial.flush();
  y = 511+511*sin(x);
  Serial.println(y);
  x += 0.05;

  if(x >= 2*3.14)
  x = 0;
  delay(100);
}
```

Rangkaian :



PROGRAM: *(Gelombang Sinus)*

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: (Gelombang Sinus)

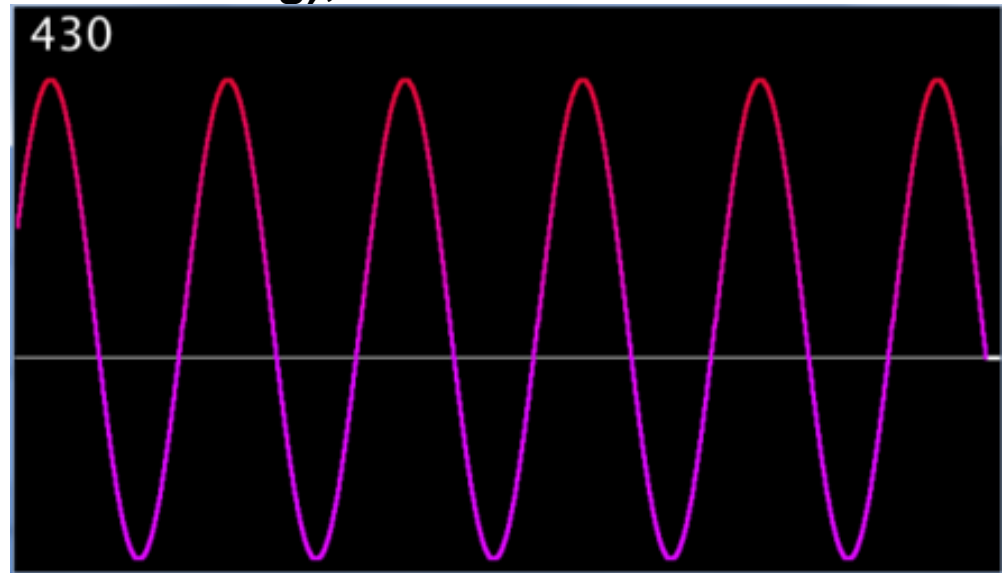
```
nilaiY[w-1] = dataSekarang;
```

Processing

```
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]);  
}  
}  
}
```



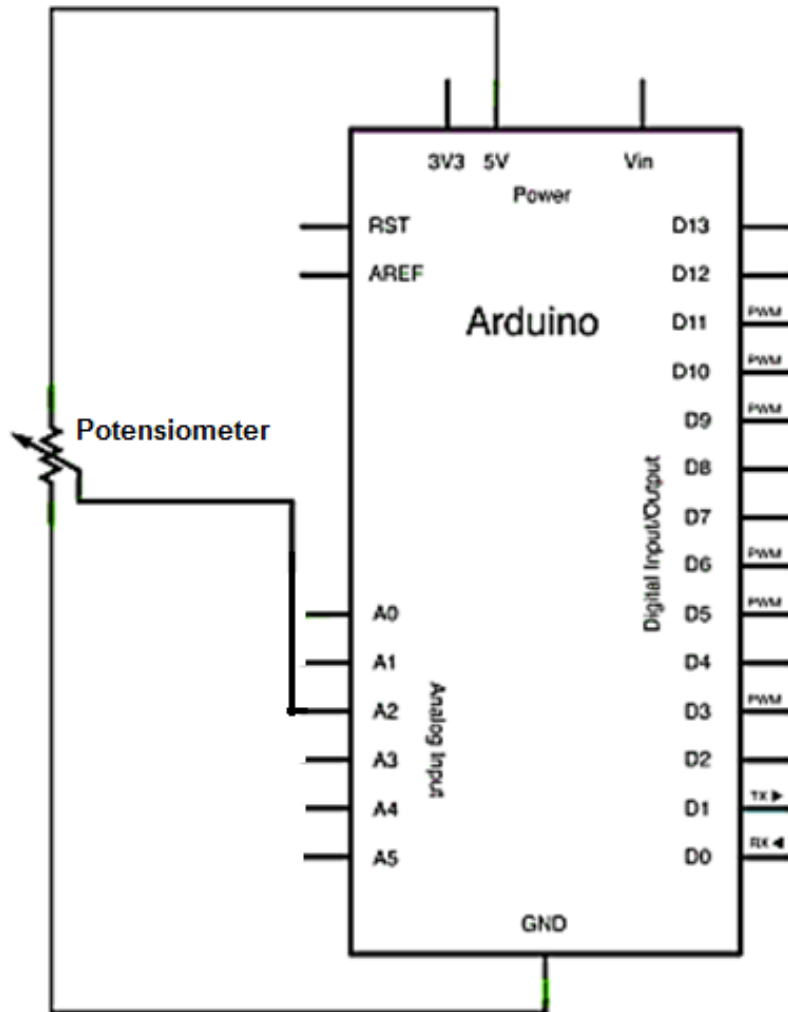
DASAR TEORI

➤ PLOTTING DATA ARDUINO MENGGUNAKAN PROCESSING

Plotting Nilai Sensor dari Arduino

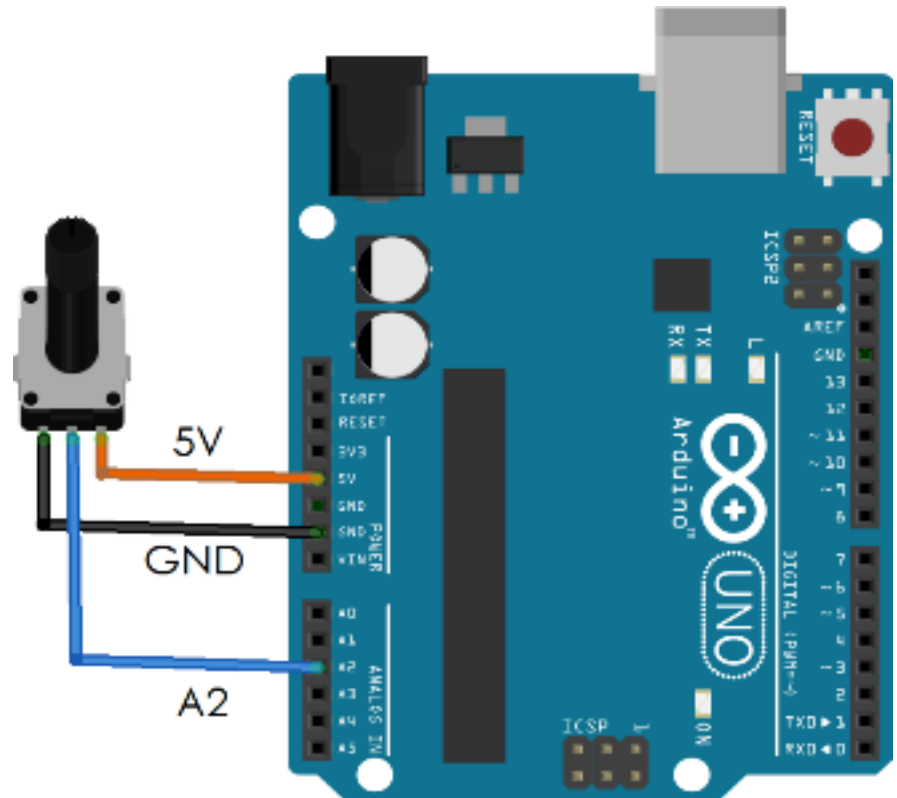
- Baca sensor atau perangkat input, pembacaan pada pin A2 (atau di pin analog lainnya).
- Memvisualisasikan data dari pin Analog Input dalam bentuk grafik.
- Mengatur grafik pada Processing menggunakan Potensiometer

Rangkaian :



Peralatan :

1. Arduino uno
2. Potensiometer
3. Software Processing



PROGRAM: **Arduino Uno**

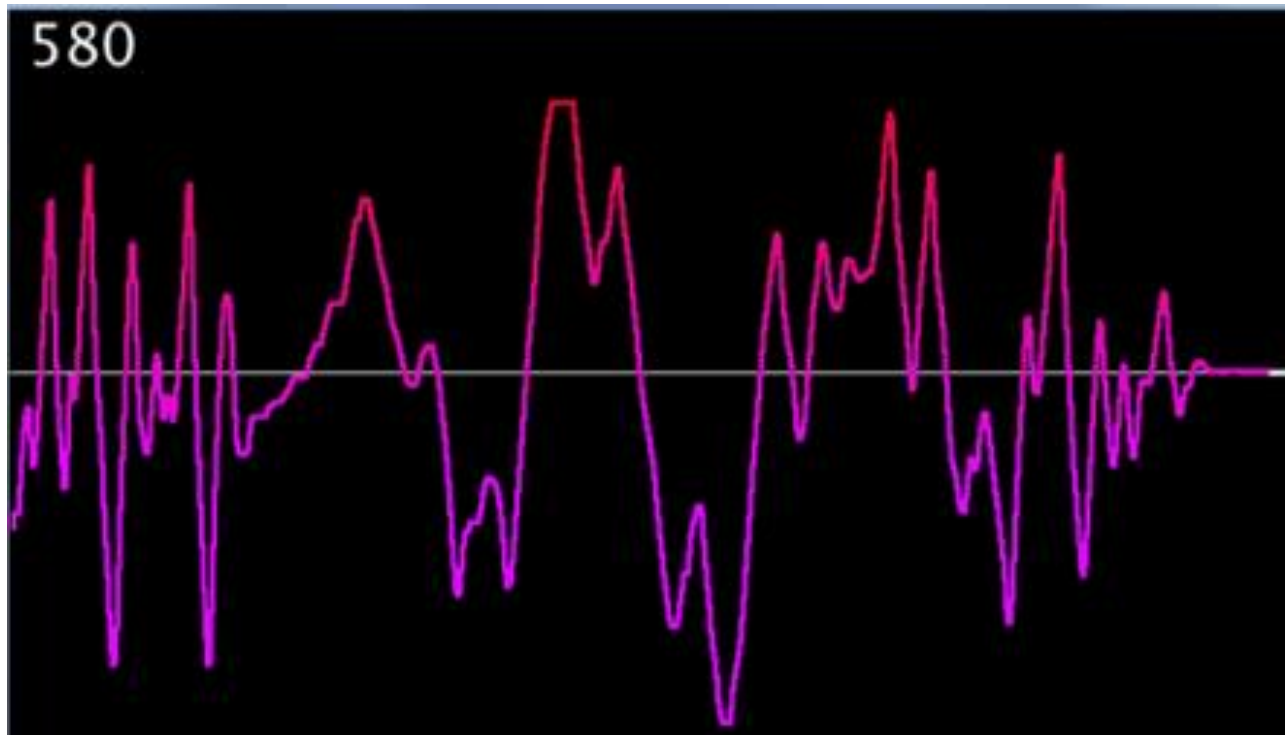
```
void setup() {  
  Serial.begin(9600);  
}
```

```
void loop() {  
  int nilaiSensor = analogRead(A2);  
  Serial.println(nilaiSensor);  
  delay(100);  
}
```

PROGRAM:

Processing

Sama dengan program Gelombang Sinus



DASAR TEORI

➤ AUDIO PROCESSING

Ploting Data Audio Dari File

- Baca data audio dari file .mp3.
- Memvisualisasikan data dari file .mp3 dalam bentuk grafik.
- Grafik data dinamis sesuai perubahan data yang diterima
- Perintah load file lagu dan play:

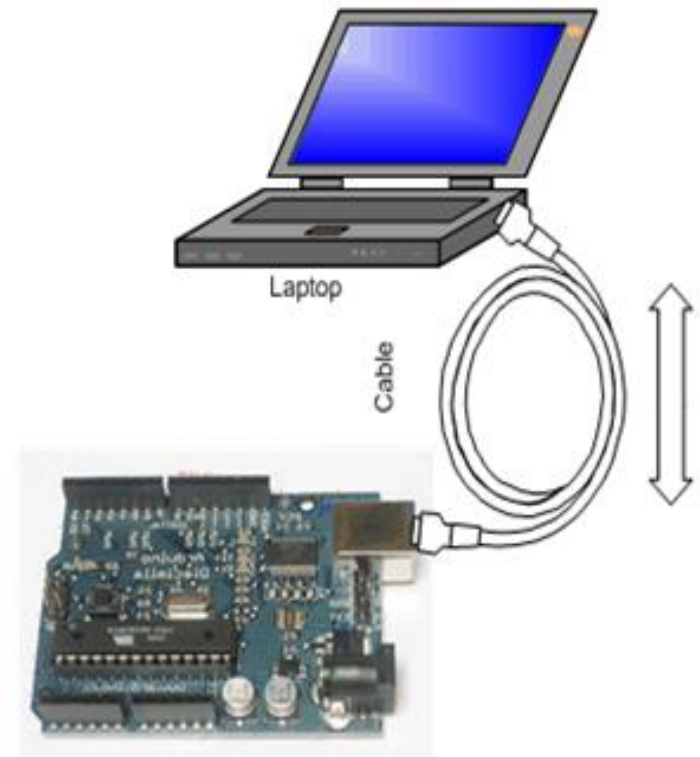
```
"lagu = minim.loadFile("nama_lagu.mp3"); "  
"lagu.play();"
```

PROGRAM:

Processing

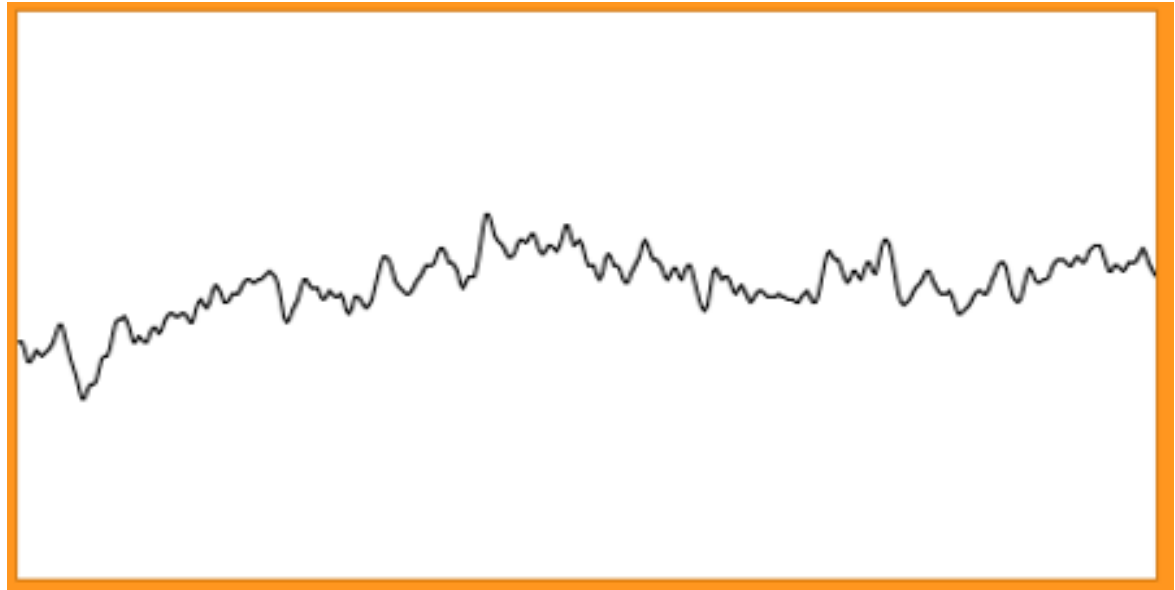
```
import ddf.minim.*;
AudioPlayer lagu;
Minim minim;
void setup()
{
  size(400,200);
  minim = new Minim(this);
  lagu = minim.loadFile("Asal.mp3");
  lagu.play();
}
void draw()
{
  background(0);
  stroke(255);
  for(int i = 0; i < lagu.bufferSize() - 1; i++)
  {
    line(i, 100 + lagu.right.get(i)*50, i+1, 100 + lagu.right.get(i+1)*50);
  }
}
```

Rangkaian :



PROGRAM: (lanjutan)

```
void stop()
{
  lagu.close();
  minim.stop();
  super.stop();
}
```



Latihan :

1. Buatlah eksperimen dengan menggunakan Analog Input Arduino dan Potensiometer. Data hasil pembacaan divisualisasikan dengan software processing seperti gambar di samping.
2. Buatlah eksperimen seperti rangkaian dan hasil di bawah ini.

