

MODUL 3

Pengenalan Sensor dan Aktuator dengan Arduino Uno

1. TUJUAN

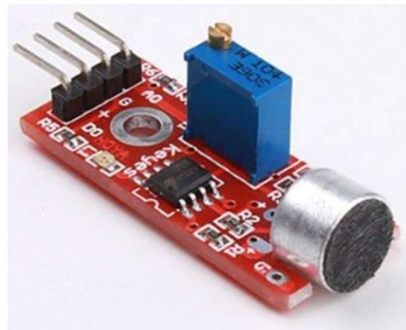
- 1.1 Memahami tentang definisi dan fungsi dari sensor dan aktuator
- 1.2 Mengetahui dan memahami jenis dari sensor dan aktuator
- 1.3 Mampu mempraktikkan penggunaan sensor dan aktuator

2. ALAT DAN BAHAN PRAKTIKUM

- Software Arduino IDE
- Laptop
- Arduino Uno
- Sensor Aktuator
- Sensor
- Jumper
- Aktuator
- Breadboard

3. DASAR TEORI

3.1 Sensor



Sensor adalah perangkat yang digunakan untuk mendeteksi perubahan besaran fisik seperti tekanan, gaya, besaran listrik, cahaya, gerakan, kelembaban, suhu, kecepatan dan fenomena-fenomena lingkungan lainnya. Setelah mengamati terjadinya perubahan, Input yang terdeteksi tersebut akan dikonversi menjadi Output yang dapat ditransmisikan secara elektronik melalui jaringan untuk ditampilkan atau diolah menjadi informasi yang bermanfaat bagi penggunanya. Sensor-sensor yang digunakan pada perangkat elektronik pada dasarnya dapat diklasifikasikan menjadi 4 kategori utama yaitu:

- a. Sensor pasif - jenis sensor yang dapat menghasilkan sinyal output tanpa memerlukan pasokan listrik dari eksternal. Contohnya Termokopel

- b. Sensor aktif - jenis sensor yang membutuhkan sumber daya eksternal untuk dapat beroperasi.
- c. Sensor digital - sensor yang menghasilkan sinyal keluaran diskrit. Sinyal diskrit akan non- kontinu dengan waktu dan dapat direpresentasikan dalam “bit”
- d. Sensor analog - sensor yang menghasilkan sinyal output yang kontinu atau berkelanjutan. Contoh Sensor Analog ini diantaranya adalah akselerometer (accelerometer), sensor kecepatan & sensor tekanan.

3.2 Aktuator



Aktuator adalah sebuah peralatan mekanis untuk menggerakkan atau mengontrol sebuah mekanisme atau sistem. Aktuator diaktifkan menggunakan lengan mekanis yang biasanya digerakkan oleh motor listrik yang dikendalikan oleh media pengontrol otomatis yang listrik analog menjadi besaran seperti kecepatan putaran atau menghasilkan daya gerakan. Aktuator juga terbagi atas beberapa jenis sebagai berikut

- a. Aktuator Pneumatik (Linier) : bekerja dengan menggunakan gas atau Udara bertekanan dalam silinder yang dibuat oleh pompa bertekanan tinggi untuk menggerakkan piston menciptakan gerakan linier.
- b. Aktuator Hidrolik : Bekerja dengan menggunakan cairan non kompresibel atau cairan hidrolik. Karena fluida tidak dapat dikompresi, ia memiliki keuntungan besar yang sangat besar, sistem ini mampu menghasilkan gaya yang sangat besar.
- c. Aktuator Putar (Rotary) : aktuator yang dikendalikan secara elektrik yang memiliki gerakan rotasi konstan.
- d. Aktuator Elektromekanis : bekerja dengan mengubah sinyal elektrik menjadi gerakan mekanik.

3.3 Arduino Uno



Arduino UNO adalah papan mikrokontroler berbasis sistem open hardware yang menggunakan mikrokontroler Atmel AVR. Arduino UNO telah dilengkapi dengan prosesor, memori, dan input/output (I/O) yang memungkinkan pengguna untuk mengendalikan berbagai perangkat elektronik.

4. PERCOBAAN

4.1 Percobaan Sensor KY-037

```
joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
Arduino Uno
joystick.ino
1  const int MIC_PIN = A0;
2
3  // Smoothing: exponential moving average (EMA), biar putaran rotor gk kasar
4  const float alpha = 0.12;
5
6  // maksimal waktu nya 300ms
7  const unsigned long PEAK_HOLD_MS = 300;
8  unsigned long peakTime = 0;
9  int peakValue = 0;
10
11 float smooth = 0.0;
12
13 void setup() {
14   Serial.begin(115200);
15   delay(200);
16   int v = analogRead(MIC_PIN);
17   smooth = v;
18   peakValue = v;
19   peakTime = millis();
20 }
21
22 void loop() {
23   int raw = analogRead(MIC_PIN);
24   smooth = (alpha * raw) + ((1.0 - alpha) * smooth);
25   int instant = raw;
```

```

joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
joystick.ino
25 int instant = raw;
26
27 unsigned long now = millis();
28 if (instant > peakValue) {
29     peakValue = instant;
30     peakTime = now;
31 } else {
32     if (now - peakTime > PEAK_HOLD_MS) {
33         peakValue = max((int)smooth, peakValue - 8);
34     }
35 }
36 int valueToMap = peakValue;
37
38 // level analog 0-1023 ke level dibaca 1-100
39 const int analogMin = 0;
40 const int analogMax = 900;
41 int level = map(constrain(valueToMap, analogMin, analogMax), analogMin, analogMax, 1, 100);
42 level = constrain(level, 1, 100);
43
44 Serial.print("raw=");
45 Serial.print(raw);
46 Serial.print(" smooth=");
47 Serial.print((int)smooth);
48 Serial.print(" peak=");
49 Serial.print(peakValue);

joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
joystick.ino
30     peakTime = now;
31 } else {
32     if (now - peakTime > PEAK_HOLD_MS) {
33         peakValue = max((int)smooth, peakValue - 8);
34     }
35 }
36 int valueToMap = peakValue;
37
38 // level analog 0-1023 ke level dibaca 1-100
39 const int analogMin = 0;
40 const int analogMax = 900;
41 int level = map(constrain(valueToMap, analogMin, analogMax), analogMin, analogMax, 1, 100);
42 level = constrain(level, 1, 100);
43
44 Serial.print("raw=");
45 Serial.print(raw);
46 Serial.print(" smooth=");
47 Serial.print((int)smooth);
48 Serial.print(" peak=");
49 Serial.print(peakValue);
50 Serial.print(" -> level=");
51 Serial.println(level);
52
53 delay(40);
54 }
    
```

Arduino	KY-037
A0	A0
GND	GND
5V	5V
D0	3

4.2 Percobaan Sensor KY-037 + Servo SG90

Install library ESP32Servo by Kevin Harrington (search ESP32Servo)

```

joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
Arduino Uno

joystick.ino
1 #include <Servo.h>
2
3 const int MIC_PIN = A0;
4 const int SERVO_PIN = 9;
5
6 Servo myServo;
7
8 // Smoothing: exponential moving average (EMA), biar putaran rotor gk kasar
9 const float alpha = 0.12;
10 float smooth = 0.0;
11
12 // maksimal waktu nya 300ms
13 const unsigned long PEAK_HOLD_MS = 300;
14 unsigned long peakTime = 0;
15 int peakValue = 0;
16
17 void setup() {
18   Serial.begin(115200);
19   delay(200);
20   int v = analogRead(MIC_PIN);
21   smooth = v;
22   peakValue = v;
23   peakTime = millis();
24   myServo.attach(SERVO_PIN);
25   myServo.write(90); // awalan disetting 90 derajat
}

joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
Arduino Uno

joystick.ino
25 myServo.write(90); // awalan disetting 90 derajat
26 }
27
28 void loop() {
29   int raw = analogRead(MIC_PIN);
30   smooth = (alpha * raw) + ((1.0 - alpha) * smooth);
31   int instant = raw;
32   unsigned long now = millis();
33
34   if (instant > peakValue) {
35     peakValue = instant;
36     peakTime = now;
37   } else {
38     if (now - peakTime > PEAK_HOLD_MS) {
39       peakValue = max((int)smooth, peakValue - 8);
40     }
41   }
42   int valueToMap = peakValue;
43
44   // level analog 0-1023 ke level dibaca 1-100
45   const int analogMin = 0;
46   const int analogMax = 900;
47   int level = map(constrain(valueToMap, analogMin, analogMax),
48                 analogMin, analogMax,
49                 1, 100);
49
joystick | Arduino IDE 2.3.6
File Edit Sketch Tools Help
Arduino Uno

joystick.ino
40 }
41 }
42 int valueToMap = peakValue;
43
44 // level analog 0-1023 ke level dibaca 1-100
45 const int analogMin = 0;
46 const int analogMax = 900;
47 int level = map(constrain(valueToMap, analogMin, analogMax),
48               analogMin, analogMax,
49               1, 100);
50 level = constrain(level, 1, 100);
51
52 // level analog 0-1023 ke level dibaca 1-100 diubah ke gerakan motor
53 int angle = map(level, 1, 100, 0, 180);
54 myServo.write(angle);
55
56 Serial.print("raw=");
57 Serial.print(raw);
58 Serial.print(" level=");
59 Serial.print(level);
60 Serial.print(" -> servo=");
61 Serial.println(angle);
62
63 delay(40);
64 }

```

Arduino	KY-037		Arduino	Motor SG90
A0	A0		GND	Kabel Coklat
GND	GND		5V	Kabel Merah
5V	5V		9	Kabel Kuning

4.3 Percobaan Menggunakan Joystick

```

1 void setup() {
2   Serial.begin(9600);
3 }
4
5 void loop() {
6   int joystickValue = analogRead(A0);
7   Serial.print("Joystick Value: ");
8   Serial.println(joystickValue);
9   delay(100);
10 }
    
```

Arduino	Joystick
GND	GND
5V	+5V
A0	VRx

4.4 Percobaan Menggunakan Joystick + Servo SG90

Install library ESP32Servo by Kevin Harrington (search ESP32Servo)

```

1 #include <Servo.h>
2
3 Servo servo1;
4
5 void setup() {
6   Serial.begin(9600);
7   servo1.attach(7);
8 }
9
10 void loop() {
11   int joystickValue = analogRead(A0);
12   Serial.println("Joystick Value:");
13   Serial.print(joystickValue);
14   delay(100);
15
16   int servoAngle = map(joystickValue, 0, 1023, 0, 180);
17   servo1.write(servoAngle);
18   delay(100);
19 }
    
```

Arduino	Joystick		Arduino	Motor SG90
GND	GND		GND	Kabel Coklat
5V	+5V		3.3V	Kabel Merah
A0	VRx		7	Kabel Kuning

5. KESIMPULAN

Sensor merupakan perangkat yang digunakan untuk mendeteksi perubahan besaran fisik seperti suhu, cahaya, tekanan, atau gerakan dan mengubahnya menjadi sinyal yang dapat diolah secara elektronik. Sensor terbagi menjadi beberapa kategori yaitu pasif, aktif, digital, dan analog. Sedangkan aktuator berfungsi untuk mengubah sinyal listrik menjadi gerakan mekanik dengan jenis jenis seperti aktuator pneumatik, hidrolik, putar, dan elektromekanis. Pada praktikum ini telah dipelajari penggunaan berbagai sensor dan aktuator yang dikendalikan dengan Arduino Uno. Melalui percobaan dengan sensor KY-037, joystick, serta aktuator servo SG90, dapat dipahami cara kerja, instalasi, serta penerapan sensor dan aktuator dalam rangkaian elektronik.

6. LATIHAN

1. bagaimana cara membedakan sensor pasif dengan aktif ?
2. Apakah jenis aktuator yang dipakai pada saat praktikum ?
3. Jelaskan hasil dari percobaan 4.4 yang sudah kamu lakukan secara lengkap dan jelas !
4. Sebutkan dan jelaskan jenis jenis aktuator, dan aktuator apakah yang di gunakan pada praktikum kali ini ?