

MIKROPROSESOR & Teknik Antarmuka 1 ARDUINO

PROGRAM STUDI
TEKNIK TELKOMUNIKASI
Semester 4

Akuwan Saleh, MT

PENILAIAN

⇒ UTS	= 35%
⇒ UAS	= 50%
⇒ Tugas-1	= 5%
⇒ Tugas-2	= 10%

REFERENSI

- Julien Bayle, “C Programming for Arduino”, Packt Publishing Ltd, Birmingham, May 2013.
- James Floyd K & Harold T , “Arduino Adventure Escape from Gemini Station”, Apress, 2013.
- Famosa Studio Arduino Starter Kit Manual – V1.0, Famosa Studio, 2013.
- Martin E, Joshua N, & Jordan H, “Arduino in Action“,Manning Publications.Co, USA, 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”, Springe, New York, 2011.
- _____, Sistem minimum Arduino Uno/ATmega328, Instruction Manual, 2010.

MATERI

1. PENDAHULUAN
2. ANTARMUKA MIKROKONTROLER DENGAN LIGHT EMITTING DIODE (LED)
3. ANTARMUKA MIKROKONTROLER DENGAN SAKLAR
4. ANTARMUKA MIKROKONTROLER DENGAN LED DOT MARIK
5. ANTARMUKA MIKROKONTROLER DENGAN KEYPAD
6. ANTARMUKA MIKROKONTROLER DENGAN SEVEN SEGMENT (7-S)
7. ANTARMUKA MIKROKONTROLER DENGAN LCD 2x16
8. PEMROGRAMAN MELODY
9. **ANALOG INPUT (ADC)**
10. **KOMUNIKASI SERIAL**
11. ANTARMUKA MIKROKONTROLER DENGAN LM 35
12. ANTARMUKA MIKROKONTROLER DENGAN LDR
13. ANTARMUKA MIKROKONTROLER DENGAN LAMPU AC 220V
14. ANTARMUKA MIKROKONTROLER DENGAN MOTOR DC
15. ANTARMUKA MIKROKONTROLER DENGAN SENSOR ULTRASONIC
16. Demo Tugas Proyek Semester

9. ANALOG INPUT (ADC)

TUJUAN

- Membaca tegangan analog yang masuk pada pin analog Arduino
- Membuat program untuk proses konversi dari nilai analog menjadi digital atau ADC (*Analog to Digital Conversion*)

DASAR TEORI

Analog I/O

- **Kontrol dan pembacaan tegangan**

- **Kelebihan :**

- Antarmuka sederhana
- Low Cost untuk low-resolutions
- Speed tinggi
- biaya Pemrograman rendah

Voltage type: Typical ranges

- 0 to 2.5V
- 0 to 4V
- 0 to 5V
- +/- 2.5V
- +/- 4V
- +/- 5V

- **Kekurangan :**

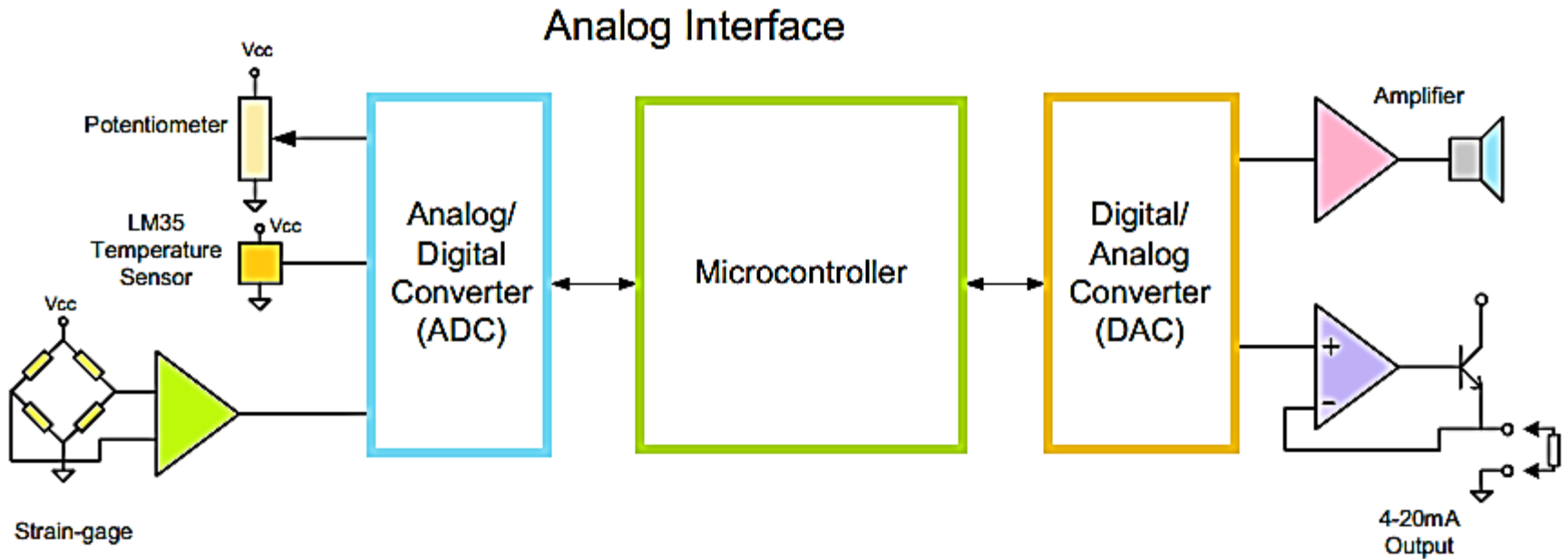
- High cost untuk high resolutions
- Tidak semua mikrokontroler memiliki analog I/O
- Desain rangk. Semakin kompleks ketika membutuhkan eksternal ADC/DAC

Current type: Typical ranges

- 0-20mA
- 4-20mA

DASAR TEORI

Contoh Antarmuka



DASAR TEORI

ADC

- Tegangan yang masuk dikonversi terlebih dahulu menjadi data digital
- Pin analog Arduino dapat menerima nilai hingga 10 bit sehingga dapat mengkonversi data analog menjadi 1024 keadaan ($2^{10} = 1024$)
- Artinya nilai 0 merepresentasikan tegangan 0 volt dan nilai 1023 merepresentasikan tegangan 5 volt.

DASAR TEORI

ADC

- Pada papan Arduino, memiliki pin A0 sampai A5 untuk membaca tegangan analog.

Kaitan Nilai ADC dengan Tegangan

- Nilai ADC menunjukkan ratio perbandingan dengan tegangan yang terbaca.

DASAR TEORI

Cara mencari nilai ADC:

$$\frac{\textit{Resolution of the ADC}}{\textit{System Voltage}} = \frac{\textit{ADC Reading}}{\textit{Analog Voltage Measured}}$$

$$\frac{1023}{5} = \frac{\textit{ADC Reading}}{\textit{Analog Voltage Measured}}$$

Jika Tegangan Analog **2.12 Volt** , maka nilai ADC:

$$\frac{1023}{5V} = \frac{X}{2.12V}$$

$$\frac{1023}{5V} * 2.12V = X$$

$$X = 434$$

DASAR TEORI

Cara mencari nilai ADC:

jika tegangan 5 volt dikonversi menjadi data digital **10 bit**

$$\frac{5}{1023} = 0,004887585$$

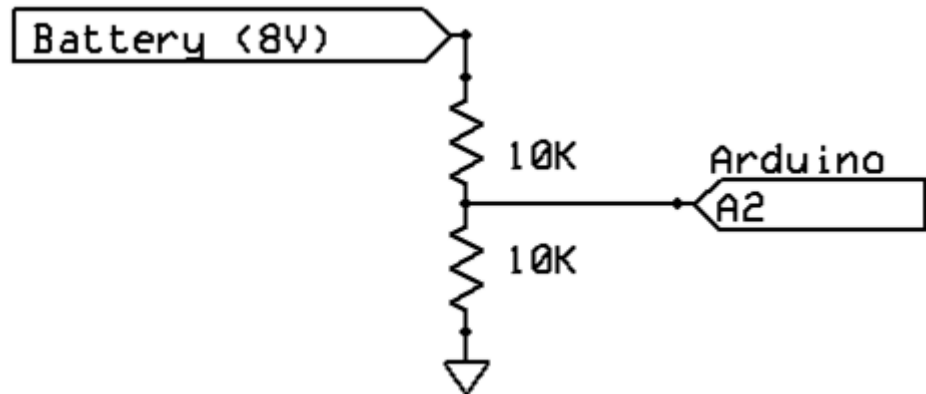
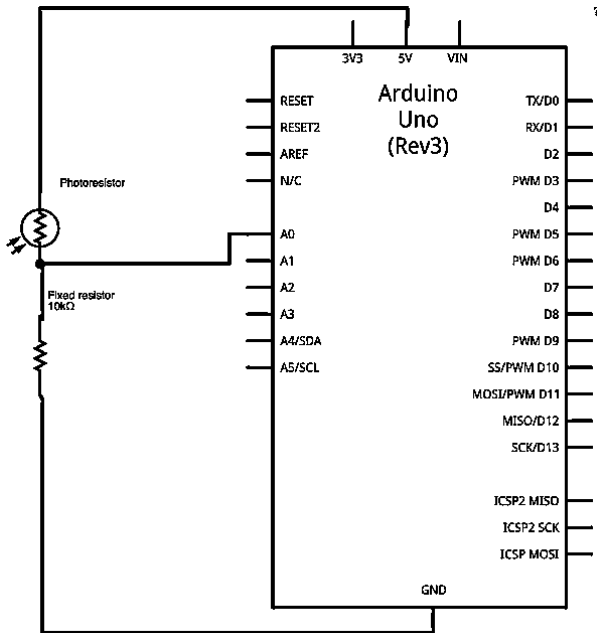
Artinya setiap 1 angka desimal mewakili tegangan sebesar 0,004887585 volt

Berapa besar tegangan yang diwakili angka **512**?

$$\frac{5V}{1023} * 512 = 2,50244 \text{ Volt}$$

DASAR TEORI

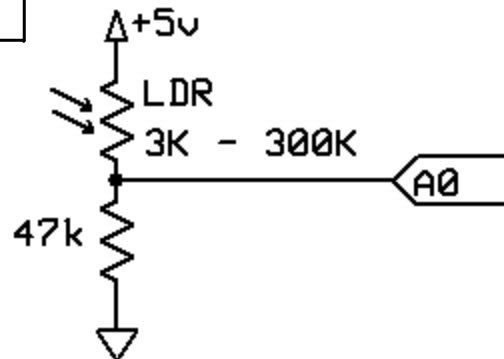
voltage-divider



$$\text{Pin A2} = 10k / (10k + 10k) * 8 = 4V$$

Jika tegangan battteray 12V dan resistor terbesar bernilai 22K, maka tegangan pada

$$\text{A2} = 10k / (10k + 22k) * 12 = 3.75V$$

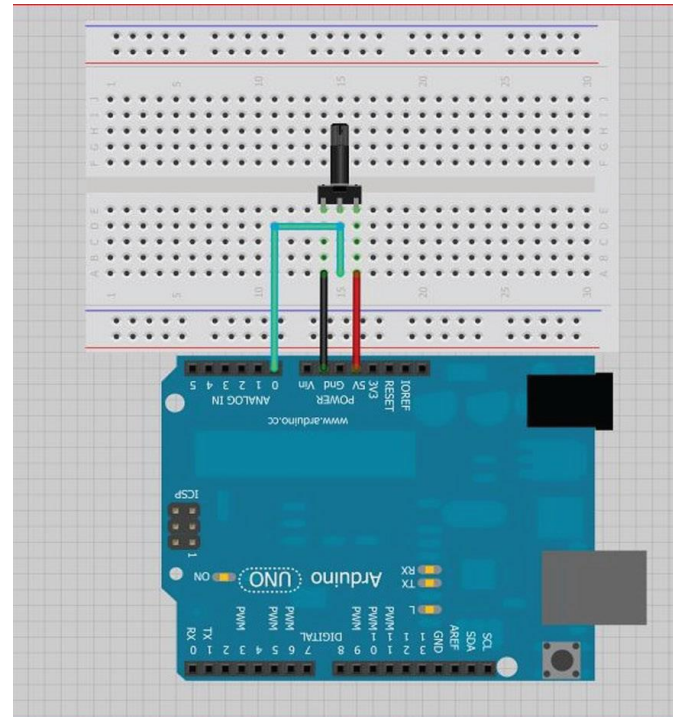
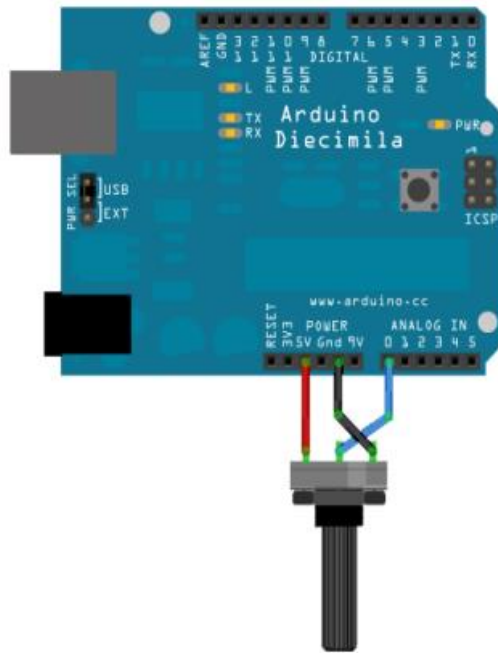


Pin A0 ?

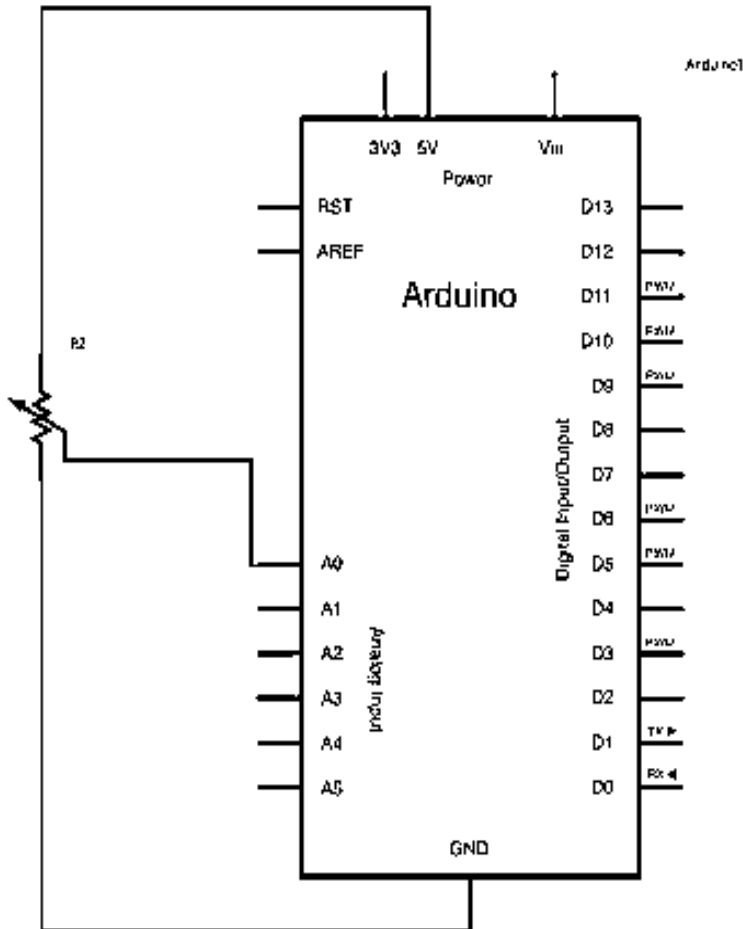
Hardware :

- Arduino Uno Board
- 1 Potentiometer
- Built-in LED = pin 13
- Kabel jumper

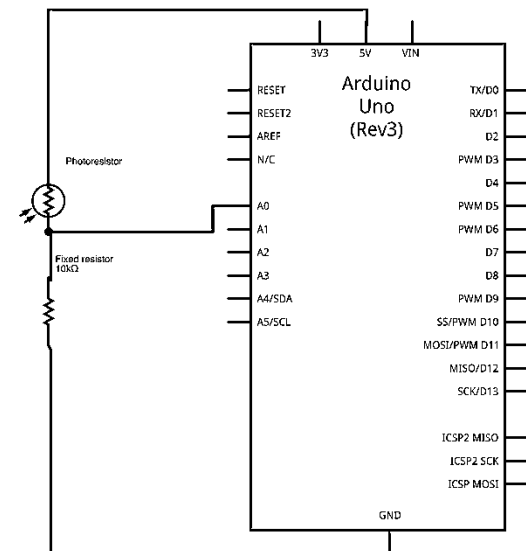
Rangkaian:



Rangkaian:



- Hubungkan Kaki 1 Potentiometer dengan pin 5v Arduino
- Hubungkan Kaki 2 (tengah) Potentiometer dengan pin A0 (analog input) Arduino.
- Hubungkan kaki 3 Potentiometer ke pin Gnd arduino



PROGRAM-1:

```
int sensorPin = A0;    // pilih pin input untuk potentiometer
int ledPin = 13;      // pilih pin untuk LED
int sensorValue = 0;  // variabel simpan nilai dari sensor
void setup() {
  pinMode(ledPin, OUTPUT); // deklarasi ledPin untuk OUTPUT
}
void loop() {
  sensorValue = analogRead(sensorPin); // baca data dari sensor:
  digitalWrite(ledPin, HIGH); // ledPin on
  delay(sensorValue); // stop program u/ <sensorValue> milidetik
  digitalWrite(ledPin, LOW); // ledPin off
  delay(sensorValue);
}
```

Ket:

- Perintah **analogRead()** untuk mengkonversi tegangan input dari range, 0 sd 5 volts, ke nilai data digital antara 0 sd 1023.
- Arduino memanggil fungsi ADC

PROGRAM-2a dan b:

```
int analogPin = A0;
int lang = 0;
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  lang = analogRead(analogPin);
  delay(100);
  Serial.println(lang);
}
```

```
void setup() {
  Serial.begin(9600);
}
void loop() {
  // baca input analog :
  int NilaiSensor = analogRead(A0);
  // Konversi (0 – 1023) ke teg.(0 – 5V):
  float volt = NilaiSensor * (5.0 / 1023.0);
  Serial.println(volt);
}
```

PROGRAM-2c:

```
void setup() {  
  Serial.begin(9600); // setup port serial  
}  
void loop() {  
  int sensorValue = analogRead(A0); // baca input analog pin 0  
  int mappedSensorValue = map(sensorValue, 0, 1023, 0,  
  9);  
  Serial.println(mappedSensorValue, DEC); // kirim ke port serial.  
}
```

PROGRAM-2d:

```
const int analogInPin = A0;  
const int analogOutPin = 9;  
int sensorValue = 0;           // variabel posisi awal pin A0  
int outputValue = 0;          // var posisi awal nilai output  
  
void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  sensorValue = analogRead(analogInPin);  
  outputValue = map(sensorValue, 0, 1023, 0, 255);
```

PROGRAM-2d - lanjutan

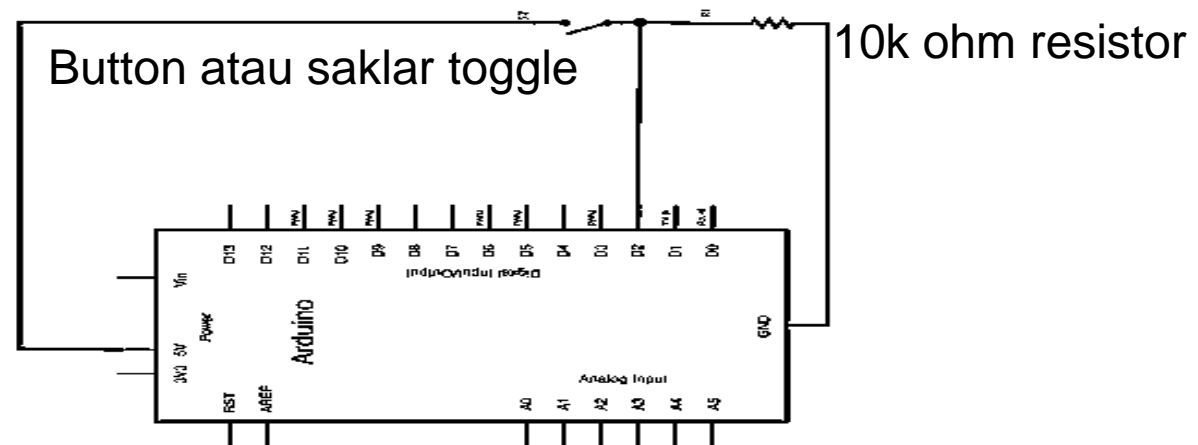
```
analogWrite(analogOutPin, outputValue);  
Serial.print("sensor = " );  
Serial.print(sensorValue);  
Serial.print("\t output = ");  
Serial.println(outputValue);  
delay(2);  
}
```

Hasil :

- Catat hasil data yang tampil di LED
- Catat hasil data yang tampil di serial monitor
- Buat laporan hasil dari percobaan

Latihan :

1. Buatlah eksperimen dengan menggunakan komunikasi serial dengan rangkaian sebagai berikut



2. Buatlah eksperimen ADC dengan menggunakan komunikasi serial untuk menampilkan data di serial monitor dan di LED 4-bit:
 - dari nilai 0 = 0000 hingga nilai 255 = 1111

10.KOMUNIKASI SERIAL

TUJUAN

- Membuat aplikasi komunikasi serial antara PC dengan modul mikrokontroler arduino
- Menampilkan dan mengamati data setelah pengiriman data dari arduino ke serial monitor PC
- Membuat aplikasi komunikasi serial antara 2 modul mikrokontroler arduino

DASAR TEORI

- **komunikasi serial** adalah salah satu protokol komunikasi yang paling banyak diadopsi dan juga terdokumentasi dengan baik.
- **Komunikasi serial** adalah komunikasi yang pengiriman datanya per-bit secara berurutan dan bergantian.
- **Komunikasi serial** ada dua macam, asynchronous serial dan synchronous serial.

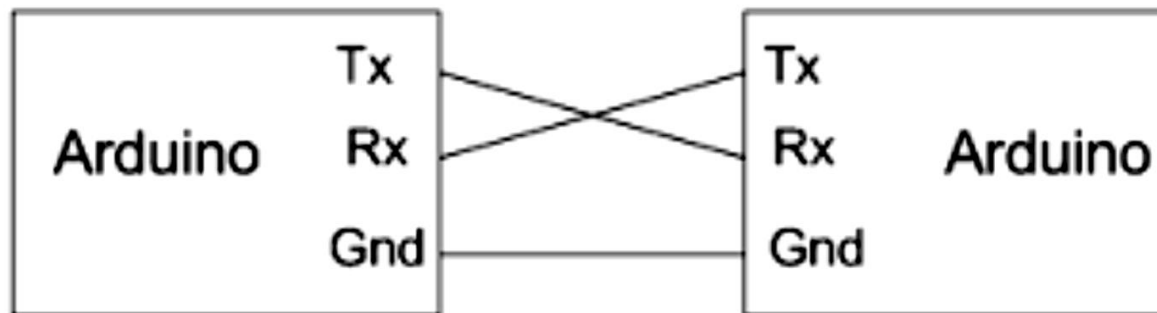
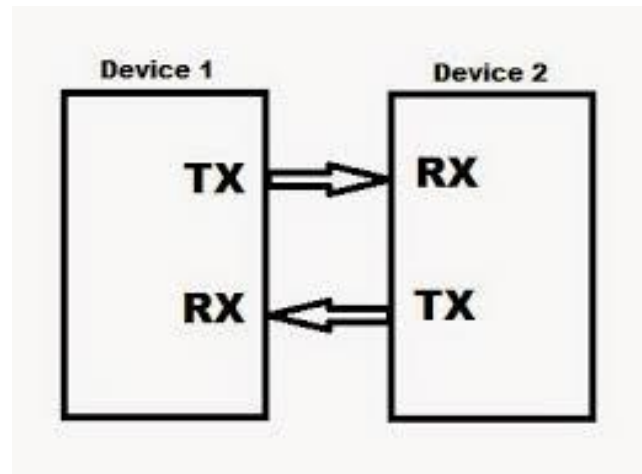
- ***Synchronous serial*** adalah komunikasi dimana hanya ada satu pihak (pengirim atau penerima) yang menghasilkan clock dan mengirimkan clock tersebut bersama-sama dengan data.

Contoh : terdapat pada transmisi data keyboard.

- ***Asynchronous serial*** adalah komunikasi dimana kedua pihak (pengirim dan penerima) masing-masing menghasilkan clock namun hanya data yang ditransmisikan, tanpa clock.

Contoh : UART pada serial port (COM) komputer.

- *Asynchronous serial* (UART)



Ilustrasi wiring komunikasi serial UART/asinkron

Konsep Baud Rate

- Menghubungkan dua perangkat melalui komunikasi serial adalah kedua perangkat berkomunikasi dengan konfigurasi yang sama.
- Parameter komunikasi secara serial, diantaranya adalah ***baud rate***, ***paket data***, ***parity bit***, dan ***synchronization bit***.
- ***Baud rate*** : kecepatan data dikirim melalui komunikasi serial. Satuan **bit-per-second (bps)**
- Standar kecepatan : 1200, 2400, 4800, 9600, 19200, 38400, 57600, dan 115200 ***bps***.

Contoh Pengiriman Data

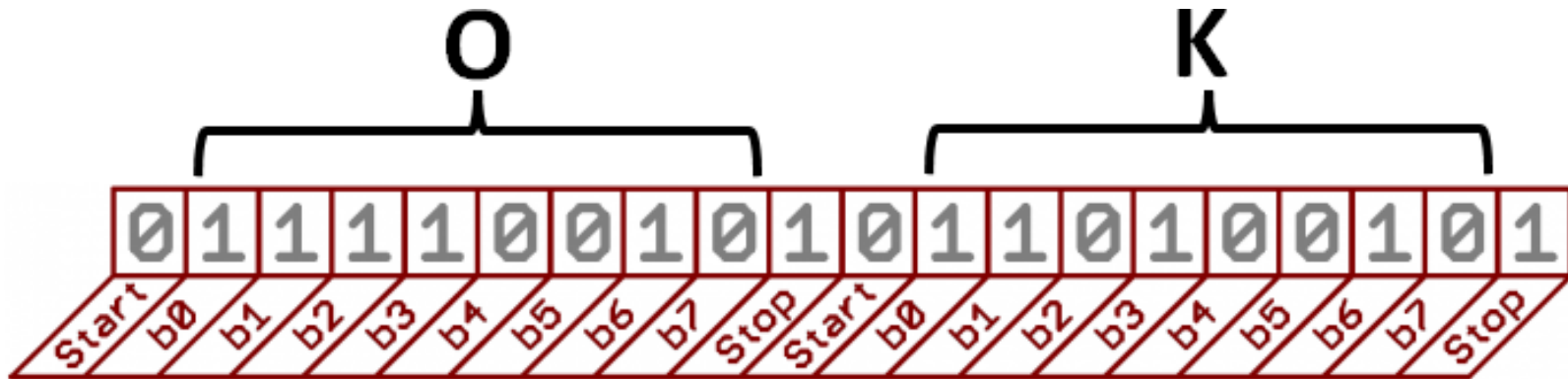
- Scenario pengiriman, yaitu **9600 8N1**. Kode 9600 8N1 bermakna bahwa kecepatan yang digunakan **9600 baud**, **8-bit data**, tidak terdapat **parity**, dan **1-bit stop**.
- Data akan dikirim dalam format ASCII (***American Standard Code for Information Interchange***). Dikirim kata **OK** (O dan K), maka komunikasi akan memiliki dua buah paket data.
 - Karakter 'O' = 79d (0100 1111 b)
 - Karakter 'K' = 75d (0100 1011b).

Contoh Pengiriman Data

- Kecepatan **9600 bit/detik**, maka setiap bitnya memerlukan waktu selama **$1/9600 = 104$ mikrodetik/bit**.
- Satu paket data untuk satu karakter terdiri dari **10 bit (8-bit data, 1-bit start dan 1-bit stop)**. Pengiriman satu karakter (yang terdiri dari 10-bit) akan membutuhkan waktu selama **10×104 mikrodetik = 1.040 mikrodetik = 1,04 milidetik**.
- Pengiriman kata '**OK**' akan membutuhkan waktu sekitar **2 milidetik** pada kecepatan **9600 bps**.

Contoh Pengiriman Data

Parameter = 9600 8N1

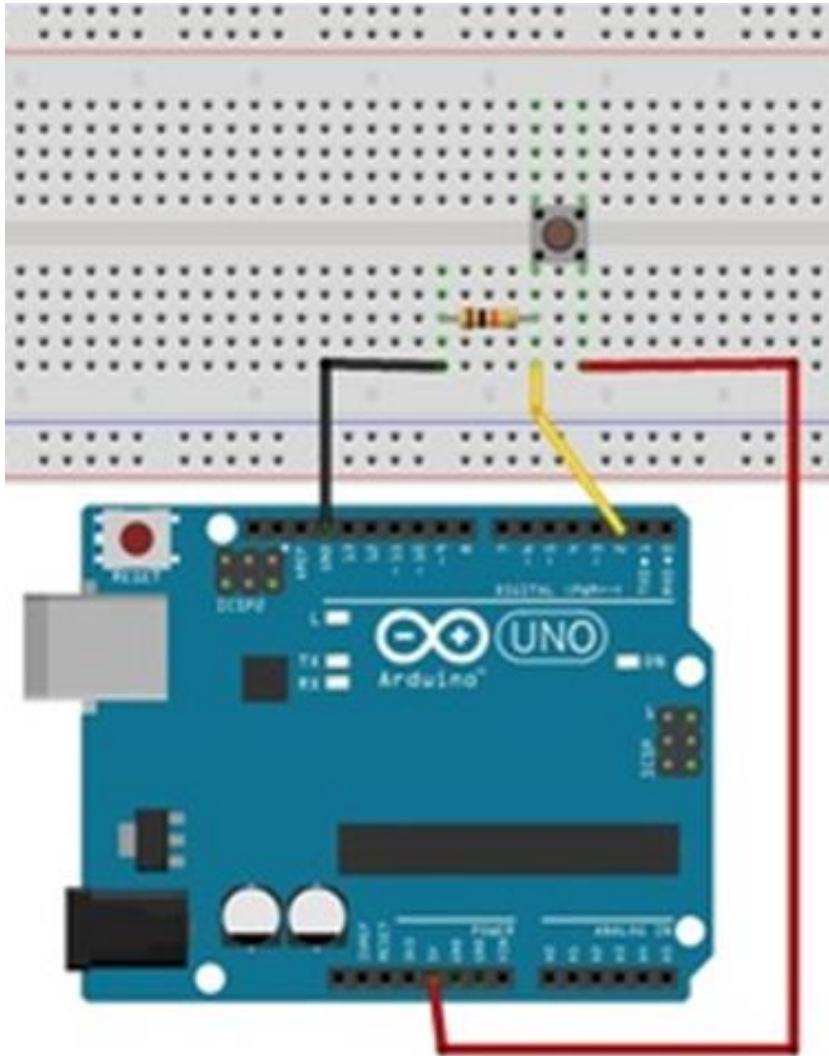


Gambar Paket data serial 'OK'

Kode ASCII untuk 'O' (kapital) = **0100 1111**,
sedangkan karakter 'K' (kapital) = **0100 1011**.
waktu **± 2 milidetik**, kecepatan **9600 bps**.

- Setiap port **komunikasi serial** akan menggunakan sepasang pin digital.
- Satu pin untuk menerima, dan satu lagi untuk mengirim. Sebagai contoh, di Arduino UNO yang memiliki satu port komunikasi serial, **pin #0** untuk **menerima (RX)**, dan **pin #1** untuk **mengirim (TX)**.
- Semua board Arduino memiliki sedikitnya satu buah controller komunikasi serial.
- Hal ini menyebabkan komunikasi serial dengan board Arduino mudah untuk dilakukan.

Rangkaian-1:



Hardware :

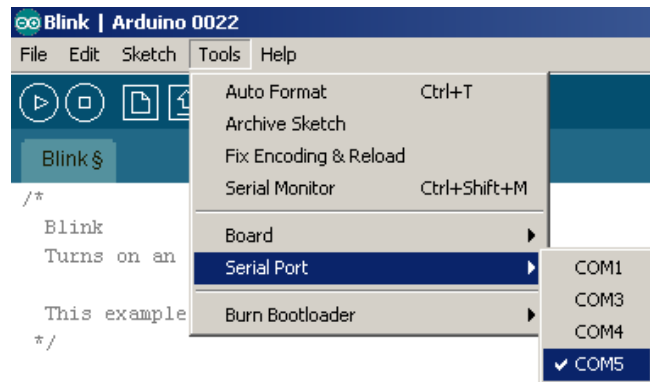
- Arduino Uno Board
- 1 buah Pushbutton
- 1 resistor 10K
- Kabel jumper

Rangkaian-1:

1. Hubungkan Kaki 1 Push Button dengan pin 5v Arduino
2. Hubungkan Kaki 2 Push Button dengan salah satu kaki Resistor 1K Ohm dan pin digital 2 Arduino.
3. Hubungkan kaki Resistor yang belum terhubung ke pin Gnd arduino

Langkah-langkah eksperimen:

- Tancapkan usb serial ke kemputer.
- Pastikan drivernya sudah terinstall dan dikenali dengan benar.
- Cek driver, lihat port COM dan samakan pada software arduino

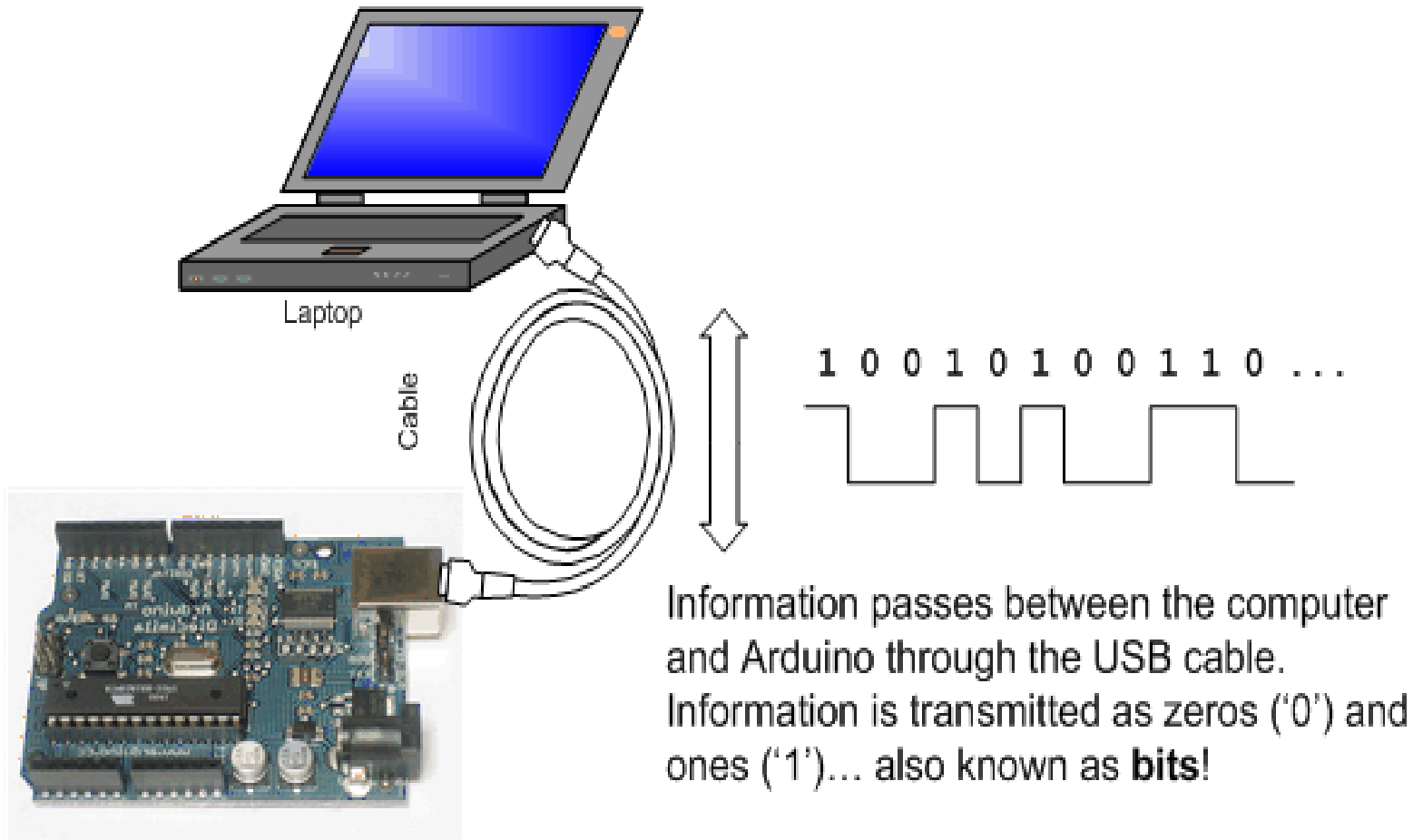


- Jalankan arduino dan buat programnya
- Buka jendela komunikasi serial pada IDE arduino caranya, pilih Tools > Serial Monitor.

PROGRAM-1

```
int pushButton = 2;  
void setup()  
{  
  Serial.begin(9600);  
  pinMode(pushButton, INPUT);  
}  
void loop()  
{  
  int kondisiButton = digitalRead(pushButton);  
  Serial.println( kondisiButton);  
  delay(1);  
}
```

Rangkaian-2:



PROGRAM-2

```
void setup()
```

```
{
```

```
  Serial.begin(9600);
```

```
}
```

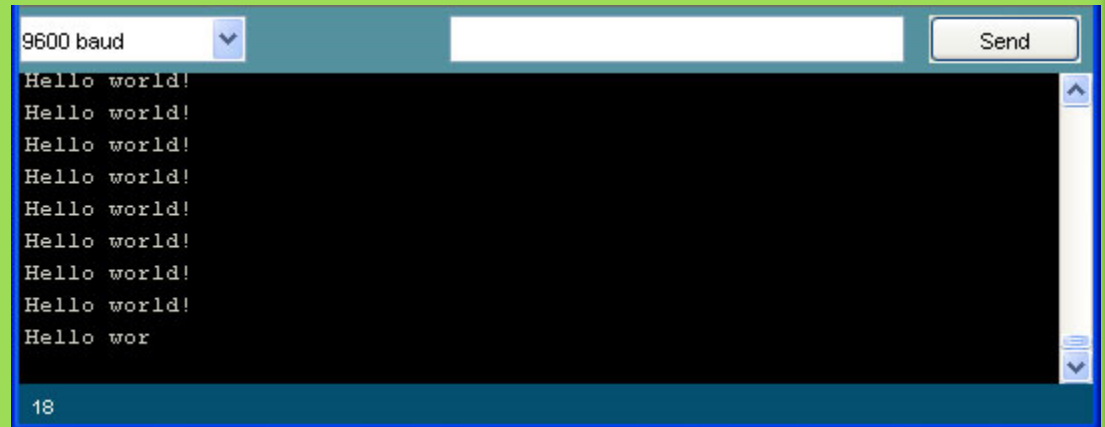
```
void loop()
```

```
{
```

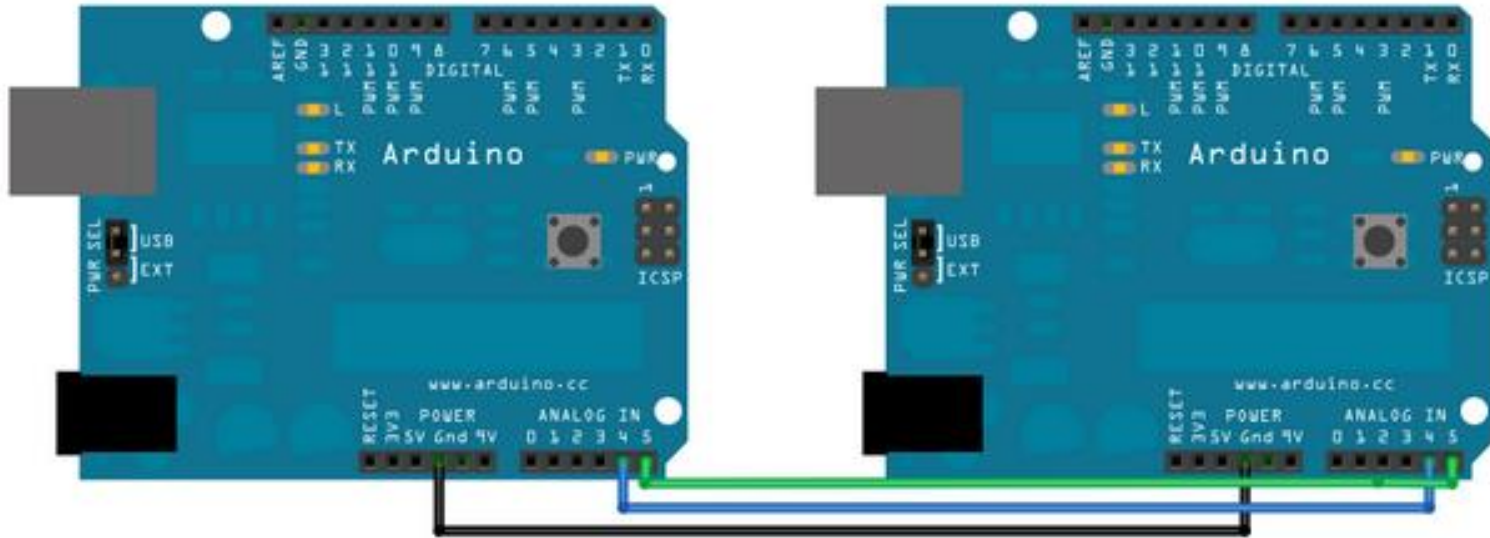
```
  Serial.println("Hello world!");
```

```
  delay(1000);
```

```
}
```



Rangkaian-3:



- Hubungkan pin 4 Arduino-1 (Tx) dengan pin 4 Arduino-2 (Rx)
- Hubungkan pin 5 Arduino-1 (Rx) dengan pin 5 Arduino-2 (Tx)
- Hubungkan pin Gnd Arduino-1 (Tx) dengan pin Gnd Arduino-2 (Rx)

PROGRAM-3a1:

```
#include <Wire.h>
void setup()
{
  Wire.begin(2);
  Wire.onRequest(requestEvent);
}

void loop()
{ delay(1000); }

Void requestEvent()
{
  Wire.write("test ");
}
```

Pengirim (Tx)

PROGRAM-3b1:

```
#include <Wire.h>
void setup()
{
  Wire.begin();
  Serial.begin(9600);
}
void loop()
{
  Wire.requestFrom(2,6);
  while(Wire.available())
  {
    char c=Wire.read();
    Serial.print(c);
  }
  delay(500);
}
```

Penerima (Rx)

PROGRAM-3a2:

```
#include <Wire.h>
```

```
void setup() {  
  Wire.begin();  
}
```

```
  byte x = 0;
```

```
  void loop() {
```

```
    Wire.beginTransmission(8);
```

```
    Wire.write("x is ");
```

```
    Wire.write(x);
```

```
    Wire.endTransmission();
```

```
    x++;
```

```
    delay(1000);
```

```
  }
```

Pengirim (Tx)

PROGRAM-3b2:

```
#include <Wire.h>

void setup() {
  Wire.begin(8);
  Wire.onReceive(receiveEvent);
  Serial.begin(250000);
}

void loop() {
}

void receiveEvent(int howMany) {
  while (0 < Wire.available()) {
    char c = Wire.read();
    Serial.print(c);
  }
  int x = Wire.read();
  Serial.println(x);
}
```

Penerima (Rx)

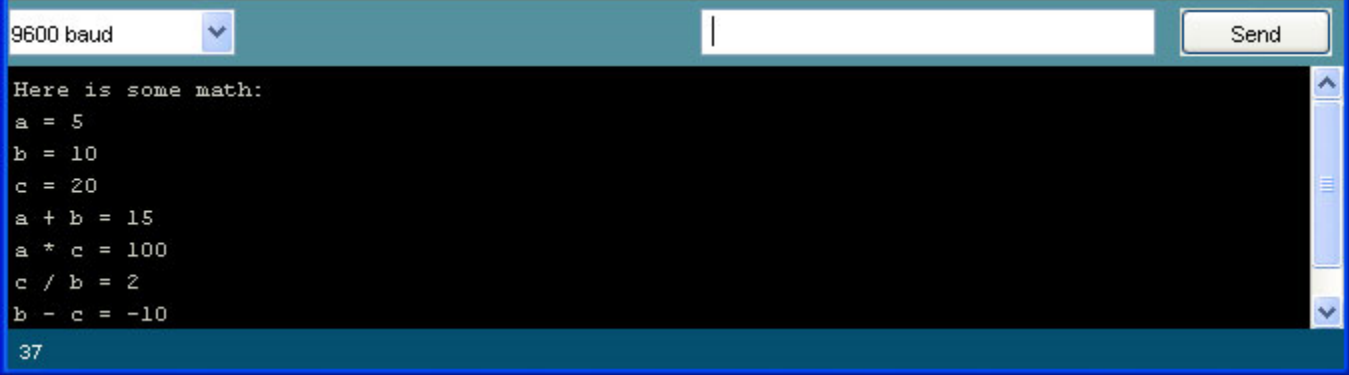
Hasil :

- Catat hasil data yang tampil di serial monitor
- Buat laporan hasil dari percobaan

Latihan :

1. Buatlah eksperimen dengan menggunakan komunikasi serial untuk menampilkan hasil dari ekspresi matematika.

Contoh:



The screenshot shows a serial monitor window with a teal header. On the left, a dropdown menu is set to '9600 baud'. On the right, there is an empty input field and a 'Send' button. The main area is a black terminal with white text displaying the following output:

```
Here is some math:  
a = 5  
b = 10  
c = 20  
a + b = 15  
a * c = 100  
c / b = 2  
b - c = -10
```

At the bottom left of the terminal, the number '37' is displayed. On the right side of the terminal, there are vertical scroll and zoom controls.

Latihan :

2. Buatlah eksperimen dengan menggunakan komunikasi serial antara 2 modul Arduino yang dapat berkomunikasi seperti Chatting