

PRAKTIKUM 5 – BERKOMUNIKASI DENGAN ARDUINO

Dengan adanya komunikasi serial dapat memungkinkan terjadinya pertukaran informasi antara papan Arduino dengan komputer, hal ini bermanfaat untuk memantau status proyek-proyek yang sedang diuji langsung melalui komputer

a. Komunikasi dengan Papan Arduino

Buat koding seperti *sketch* di bawah ini, simpan dengan nama "KomunikasiArduino"

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

void loop()
{
  while (Serial.available() !=0)
  {
    char data = Serial.read();
    Serial.print(data);

    delay(20); //Tunda sebentar untuk menyiapkan
               //pembacaan data berikutnya
  }
}
```

Sketch di atas digunakan untuk membaca semua karakter yang dikirimkan ke *port* serial (oleh komputer). Untuk memulai komunikasi, `Serial.begin` perlu dipanggil terlebih dahulu.

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

Argumen 9600 menentukan kecepatan data baik untuk keperluan pengiriman maupun penerimaan.

Arduino memantau data yang berada di *port* serial secara terus-menerus berkat perintah :

```
while (Serial.available() != 0)
{
  .....
}
```

Dalam hal ini, `Serial.available()` akan menghasilkan nol sekiranya tidak ada data di *port* serial. Jika *port* tersebut mengandung karakter, satu karakter akan dibaca melalui :

```
char data = Serial.read();
```

Pernyataan di atas mendeklarasikan variable `data` sebagai variabel `char` bertipe (untuk menampung satu karakter). Variabel ini akan menampung karakter yang dibaca dari *port* serial.

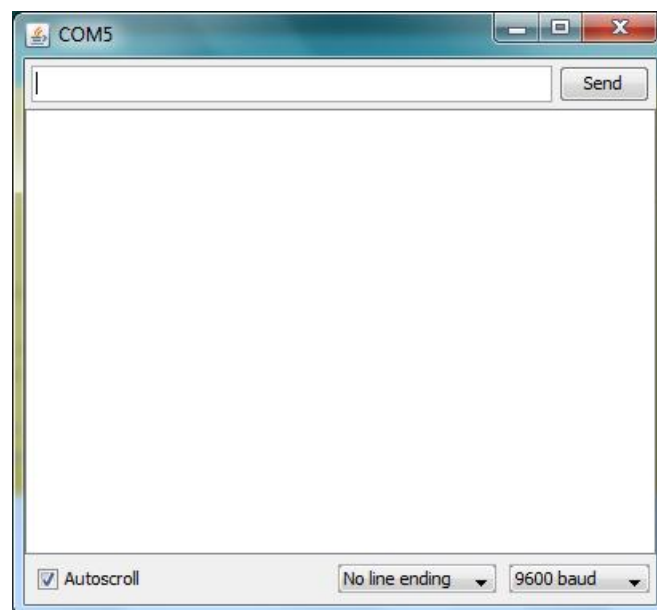
Pernyataan :

```
Serial.print(data);
```

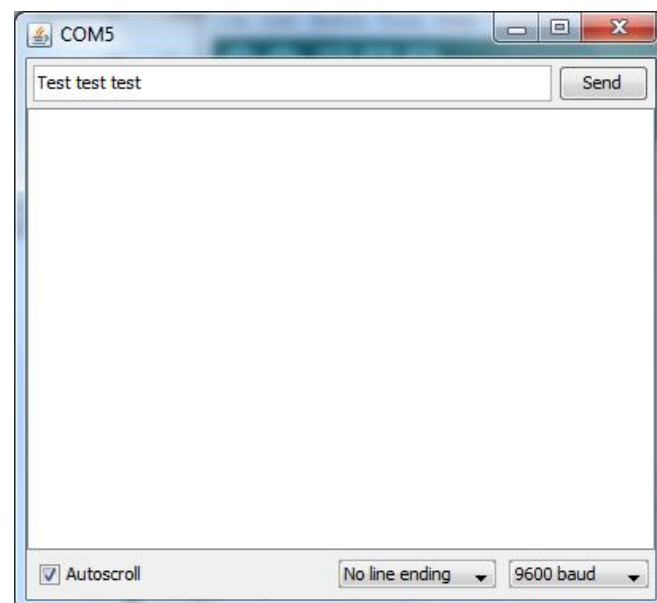
Digunakan untuk mengirim isi variabel `data` ke *port* serial dan implikasinya akan membuat isi data tersebut di penampil informasi *port* serial di komputer.

Untuk menguji hasil *sketch* tersebut dilakukan dengan cara :

1. Klik TOOLS pada MenuBar > Klik Serial Monitor dan muncul seperti di bawah ini :

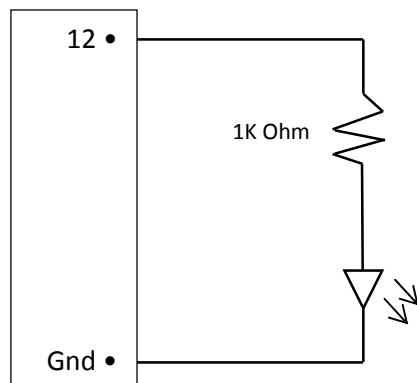


2. Cobalah mengetikkan string seperti terlihat pada gambar di bawah ini, lalu klik SEND :



b. Mematikan dan Menghidupkan LED melalui Keyboard

Dengan memanfaatkan komputer serial, dimungkinkan untuk mengendalikan LED melalui keyboard. Sebagai contoh, angka 1 digunakan untuk menyalakan LED dan angka 0 digunakan untuk mematikan LED. Buatlah rangkaian seperti di bawah ini



```
const int PIN_12 = 12;

void setup()
{
  //Atur kecepatan penerimaan dan pengiriman data
  Serial.begin(9600);

  //Atur PIN_12 sebagai keluaran
  pinMode(PIN_12, OUTPUT);
}

void loop()
{
  while (Serial.available() !=0)
  {
    //Baca tipe data bilangan integer
    int bilangan = Serial.parseInt();

    //Buang sisanya jika ada
    Serial.flush();

    //atur LED
    if (bilangan == 1)
    {
      digitalWrite(PIN_12, HIGH);
      Serial.println("LED NYALA");
    }
    else
    {
      if (bilangan == 0)
      {
        digitalWrite(PIN_12, LOW);
        Serial.println("LED MATI");
      }
      delay(20); //Tunda sebentar untuk menyiapkan
                //pembacaan berikutnya
    }
  }
}
```

Pada sketch di atas, pernyataan

```
int bilangan = Serial.parseInt();
```

Digunakan untuk mendeklarasikan variabel `bilangan` sebagai variabel bertipe `int` dan memberikan sebuah bilangan bulat yang berada di port serial ke variabel tersebut

Pernyataan :

```
Serial.flush();
```

Digunakan untuk membuang semua data tersisa di port serial. Adapun pernyataan berikut dipakai untuk mengendalikan LED :

```
if (bilangan == 1)
{
    digitalWrite(PIN_12, HIGH);
    Serial.println("LED NYALA");
}
else
    if (bilangan == 0)
    {
        digitalWrite(PIN_12, LOW);
        Serial.println("LED MATI");
    }
```

Kode di atas akan membuat LED dihidupkan sekiranya `bilangan` bernilai 1 dan mematikan LED jika `bilangan` bernilai 0. Pernyataan di atas juga mengirim informasi ke port serial berupa "LED NYALA" atau "LED MATI". Untuk bilangan selain 0 atau 1 akan diabaikan.

Setelah koding diverifikasi dan di *upload* ke dalam Arduino, selanjutnya lakukan pengujian dengan Klik TOOLS pada MenuBar > Klik Serial Monitor > ketik 1 > SEND. Jika LED menyala maka pengujian Anda berhasil. Dan ketik 0 > SEND, maka LED akan padam.

Referensi :

Kodir, Abdul. *From Zero to a Pro Arduino*. 2015. CV. Andi OFFSET: Yogyakarta

<https://www.arduino.cc/>