

Seriell kommunikasjon Arduino

Eirik Tennøfjord

December 1, 2020

1 PuTTY og Seriell kommunikasjon



Figure 1: PuTTY

- Last ned PuTTY
- Koble til Arduino.
- Prøv å sette opp seriell printing i Arduino.
- Fungerer dette med Seriell-vinduet til Arduino-IDE?
- Prøv å få opp kommunikasjonen med PuTTY.
- Hva skjer dersom baudrate er feil?

2 Arduino til Arduino

- Sett dere sammen i grupper.
- Koble arduinoene sammen, ved hjelp av RX/TX.
- På ene arduinoen kobler dere til en knapp.

- På andre kobler dere til en lampe.
- Få lampen til å lyse når knappen er trykt ned.
- Sett en diode på begge arduinoene. Deretter få disse til å blinke (ikke samtidig).
- Dimming av lys. Potmeter på én arduino og diode på den andre.

2.1 Tutorial

For å få kommunikasjon mellom to arduinoer er en av metodene å benytte seriell kommunikasjon. Denne benytter RX/TX på arduino-brettet. Når man skal benytte disse pinnene kan det være problematisk å bruke USB. Dette er fordi USB kommunikasjonen mellom PC og arduino benytter også disse pinnene (selvom det ikke er koblet til noe på pinne 0 og 1 fra USB).

2.1.1 Oppkobling

Når vi skal kommunisere én vei mellom to arduinoer, trenger man kun to ledere. En som kobler sammen GND mellom de to arduinoene og en fra TX til RX. Den Arduinoen som har tilkoblet TX er den som sender, og den som er tilkoblet RX er den som mottar.

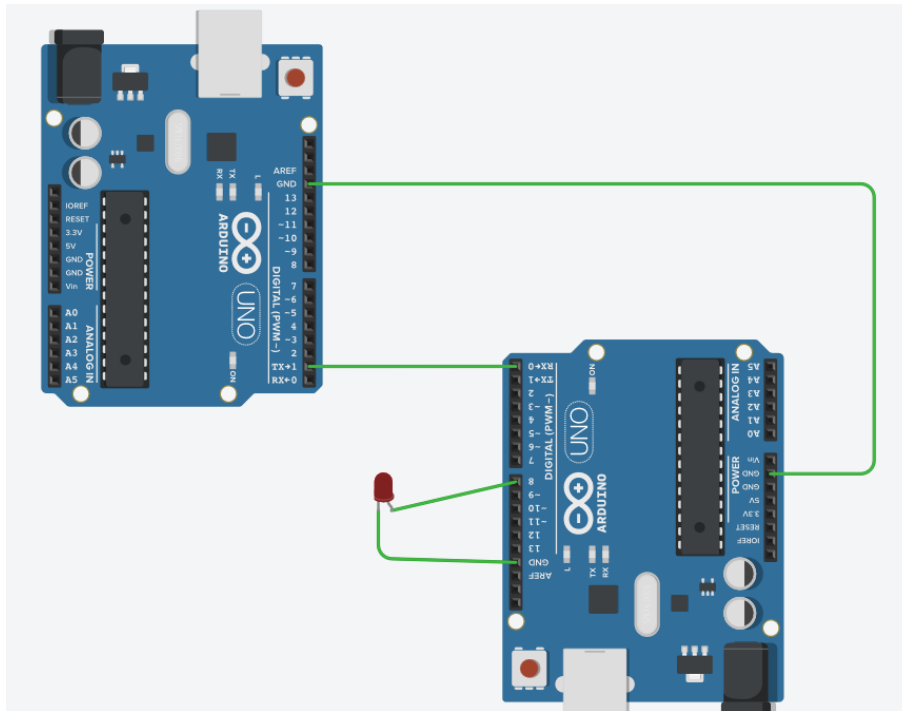


Figure 2: Kommunikasjon mellom to arduinoer (UART)

2.1.2 Koding for å sende (TX)

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

void loop()
{
  Serial.write(0);
  delay(1000);
  Serial.write(1);
  delay(1000);
}
```

2.1.3 Koding for å motta (RX)

```
void setup() {
  Serial.begin(9600);
  pinMode(8, OUTPUT);
}

int incomingByte = 0; // for incoming serial data

void loop() {
  if (Serial.available() > 0) {
    // read the incoming byte:
    incomingByte = Serial.read();

    // say what you got:
    Serial.print("I received: ");
    Serial.println(incomingByte, DEC);
  }

  if (incomingByte == 1) {
    digitalWrite(8, HIGH);
  }
  else {
    digitalWrite(8, LOW);
  }
}
```

3 Python til Arduino

- Last ned pyserial, ved hjelp av PIP.

```
pip install pyserial
```

- Finn COM porten som er tilkoblet Arduino (slik som tidligere).
- Sett opp en seriell kommunikasjon ved hjelp av python serial biblioteket, og Arduino.
- Prøv å få lese noe ut fra den serielle kommunikasjonen (tastetrykk, temperatur, potmeter ++).
- I python lager du et program som logger til en fil hver gang det blir skrevet noe i den serielle kommunikasjonen.
- Legg til tidsstempling til loggingen (hint: python timestamp).
- Lag et program i python/arduino som logger når en knapp blir trykt inn, og deretter lagrer dette på fil.

Dokumentasjon for PySerial finner dere her: [PySerial](#).

Et starteksempel kan du se på denne kodesnutten:

```
import serial
import time
from datetime import datetime

ser = serial.Serial(
    'COM3', 9600, timeout=0, parity=serial.PARITY_NONE, rtscts=1
)

while True:
    try:
        s = str(ser.readline(100).decode())
        if s != "":
            f = open("Adgangslogg.txt", "a+")
            f.write(s)
    except:
        print('ERROR')

    time.sleep(1)
```
