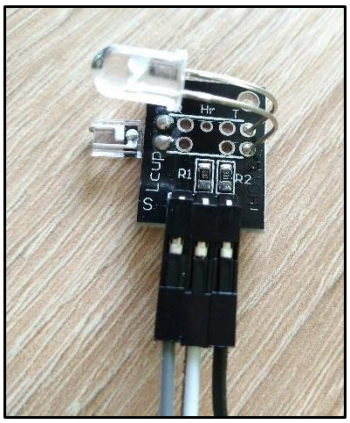
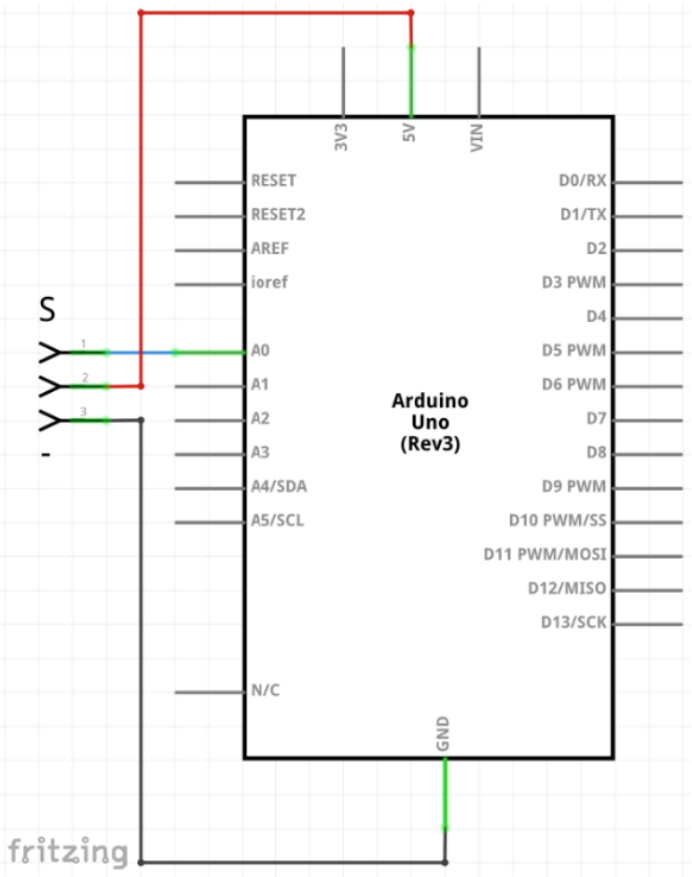
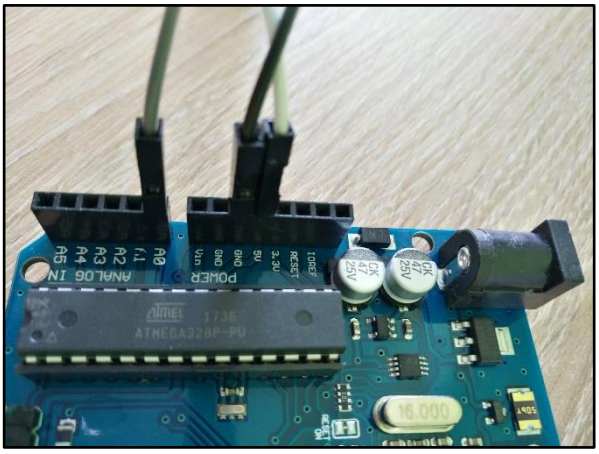
HEART PULSE SENSOR

Heart Pulse Sensor is an input module that includes a combination of an infrared LED and a phototransistor. IF we insert a finger between the IR LED and the phototransistor, the infrared light from the LED will light up our finger and we will be able to measure the variable voltage through the phototransistor. Changes in the electrical voltage will be caused by the blood flowing in the finger, so we can measure the heart rate in BPM (beats per minute) with fairly good accuracy.





For the successful connection of the Heart Pulse Sensor with the Arduino board, it is necessary to connect 3 connecting pins. We connect the extreme pin S with the A0 pin, the middle pin with a 5 V Arduin pin and the extreme pin "-" with the Arduin ground.



Sample Code

```
// Arduino heart pulse sensor KY039

// pin number setting
#define indikLED 13
#define analogPin A0
// creating a constant for detection delay
const int zpozdeniMereni = 60;

void setup() {
  // communication over a seriál line at 9600 b
aud
  Serial.begin(9600);
  // initialization of an analog pin as input
  // and a digital pin as output
  pinMode(analogPin, INPUT);
  pinMode(indikLED, OUTPUT);
}

void loop()
{
  // create temporary variables to store result
s
  static int uderyZaMinutu = 0;
  int tepovaFrekvence = 0;
  // heart rate control by subroutine detekceTe
pu
  if (detekceTepu(analogPin, zpozdeniMereni)) {
    // heart rate calculation
    tepovaFrekvence = 60000 / uderyZaMinutu;
    // LED on when detected
    // heart rate measurement
    digitalWrite(indikLED, HIGH);
    // print the measured heart rate informatio
n
    if (tepovaFrekvence > 50 & tepovaFrekvence
< 200) {
      Serial.print("Heartbeat: ");
      Serial.print(tepovaFrekvence);
      Serial.println(" beat per minute (BPM).
");
    }
    // reset the variable for next measurements
    uderyZaMinutu = 0;
  } else {
    // when there is no detection, switch off t
he LED
    digitalWrite(indikLED, LOW);
  }
  // pause the program to the next measurement
  delay(zpozdeniMereni);
  // adding a delay for next measurements
  uderyZaMinutu += zpozdeniMereni;
}
// subroutine for pulse detection and frequency
 calculation
bool detekceTepu(int senzorPin, int zpozdeni) {
  // creating auxiliary variables
  static int maxHodnota = 0;
  static bool SpickovaHodnota = false;
  int analogHodnota;
  bool vysledek = false;
  // reading the analog value from the sensor
  analogHodnota = analogRead(senzorPin);
  // conversion of analog values for next calcu
lations
  analogHodnota *= (1000 / zpozdeni);
  // adjust the maximum value
  if (analogHodnota * 4L < maxHodnota) {
    maxHodnota = analogHodnota * 0.8;
  }
  // detection of max value
  if (analogHodnota > maxHodnota - (1000 / zpoz
deni)) {
    // setting a new maximum with detected tip
    if (analogHodnota > maxHodnota) {
      maxHodnota = analogHodnota;
    }
    // setting the validity of the result, when
    // the tip was not detected
    if (SpickovaHodnota == false) {
      vysledek = true;
    }
    SpickovaHodnota = true;
  } else if (analogHodnota < maxHodnota - (3000
 / zpozdeni)) {
    SpickovaHodnota = false;
    // adjusting the maximum value when changin
g the measured values
    maxHodnota -= (1000 / zpozdeni);
  }
  // return the subprogram result
  return vysledek;
}
```