

```

#include <Adafruit_Sensor.h>
#include <Adafruit_ADXL345_U.h>

Adafruit_ADXL345_Unified akcelerometer = Adafruit_ADXL345_Unified(12345);

extern float akcelerometerX;
extern float akcelerometerY;
extern float akcelerometerZ;

void displayDataRate(void) {
  Serial.print("Data Rate:   ");
  switch(akcelerometer.getDataRate()) {
    case ADXL345_DATARATE_3200_HZ:
      Serial.print("3200 ");
      break;
    case ADXL345_DATARATE_1600_HZ:
      Serial.print("1600 ");
      break;
    case ADXL345_DATARATE_800_HZ:
      Serial.print("800 ");
      break;
    case ADXL345_DATARATE_400_HZ:
      Serial.print("400 ");
      break;
    case ADXL345_DATARATE_200_HZ:
      Serial.print("200 ");
      break;
    case ADXL345_DATARATE_100_HZ:
      Serial.print("100 ");
      break;
    case ADXL345_DATARATE_50_HZ:
      Serial.print("50 ");
      break;
    case ADXL345_DATARATE_25_HZ:
      Serial.print("25 ");
      break;
    case ADXL345_DATARATE_12_5_HZ:
      Serial.print("12.5 ");
      break;
    case ADXL345_DATARATE_6_25HZ:
      Serial.print("6.25 ");
      break;
    case ADXL345_DATARATE_3_13_HZ:
      Serial.print("3.13 ");
      break;
    case ADXL345_DATARATE_1_56_HZ:
      Serial.print("1.56 ");
      break;
    case ADXL345_DATARATE_0_78_HZ:
      Serial.print("0.78 ");
      break;
    case ADXL345_DATARATE_0_39_HZ:
      Serial.print("0.39 ");
      break;
    case ADXL345_DATARATE_0_20_HZ:
      Serial.print("0.20 ");
      break;
    case ADXL345_DATARATE_0_10_HZ:
      Serial.print("0.10 ");
      break;
    default:
      Serial.print("???? ");
      break;
  }
  Serial.println(" Hz");
}

```

```

}

void displayRange(void) {
    Serial.print ("Range:      +/- ");
    switch (akcelerometer.getRange()) {
        case ADXL345_RANGE_16_G:
            Serial.print ("16 ");
            break;
        case ADXL345_RANGE_8_G:
            Serial.print ("8 ");
            break;
        case ADXL345_RANGE_4_G:
            Serial.print ("4 ");
            break;
        case ADXL345_RANGE_2_G:
            Serial.print ("2 ");
            break;
        default:
            Serial.print ("?? ");
            break;
    }
    Serial.println(" g");
}

void displaySensorDetails(void) {
    sensor_t sensor;
    akcelerometer.getSensor(&sensor);
    Serial.println("-----");
    Serial.print ("Sensor:      "); Serial.println(sensor.name);
    Serial.print ("Driver Ver:   "); Serial.println(sensor.version);
    Serial.print ("Unique ID:    "); Serial.println(sensor.sensor_id);
    Serial.print ("Max Value:    "); Serial.print(sensor.max_value); Serial.println(" m/s^2");
    Serial.print ("Min Value:    "); Serial.print(sensor.min_value); Serial.println(" m/s^2");
    Serial.print ("Resolution:   "); Serial.print(sensor.resolution); Serial.println(" m/s^2");
    Serial.println("-----");
    Serial.println("");
}

void zobrazNastavenieAkcelerometra() {
    displaySensorDetails();
    displayDataRate();
    displayRange();
}

void zapniAkcelerometer() {
    Serial.println("Nastavujem akcelerometer.");
    akcelerometer.begin();
    akcelerometer.setRange(ADXL345_RANGE_16_G);
    //akcelerometer.setDataRate(ADXL345_DATARATE_12_5_HZ);
    Serial.println("Akcelerometer zapnuty.");

    zobrazNastavenieAkcelerometra();
}

void merajZrychlenie() {
    sensors_event_t event;
    akcelerometer.getEvent(&event);
    akcelerometerX = event.acceleration.x;
    akcelerometerY = event.acceleration.y;
    akcelerometerZ = event.acceleration.z;
}

```