

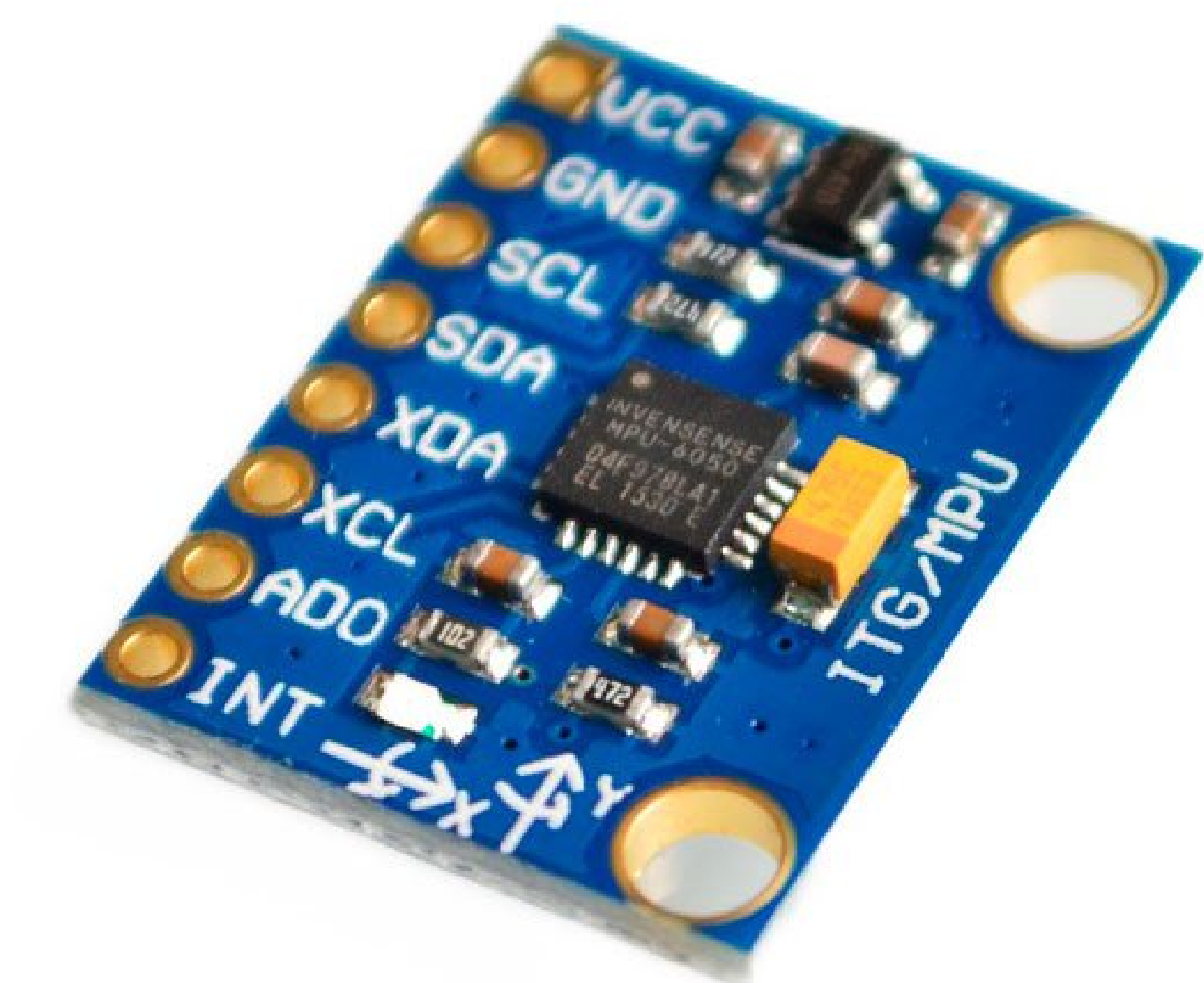
Подключение гироскопа-акселерометра MPU6050 к Wemos d1

Wemos D1MPU6050Ardulogic

30 июня 2017

91390

Для того чтобы наше устройство могло отображать и записывать в лог ускорение и перегрузки во время разгона, мы подключим к нему популярный высокоточный модуль **GY-521**, на основе чипа **MPU-6050**, являющегося трех-осевым гироскопом и трех-осевым акселерометром. Гироскоп нам пока не понадобится, поэтому в этой части я буду рассказывать только про акселерометр.

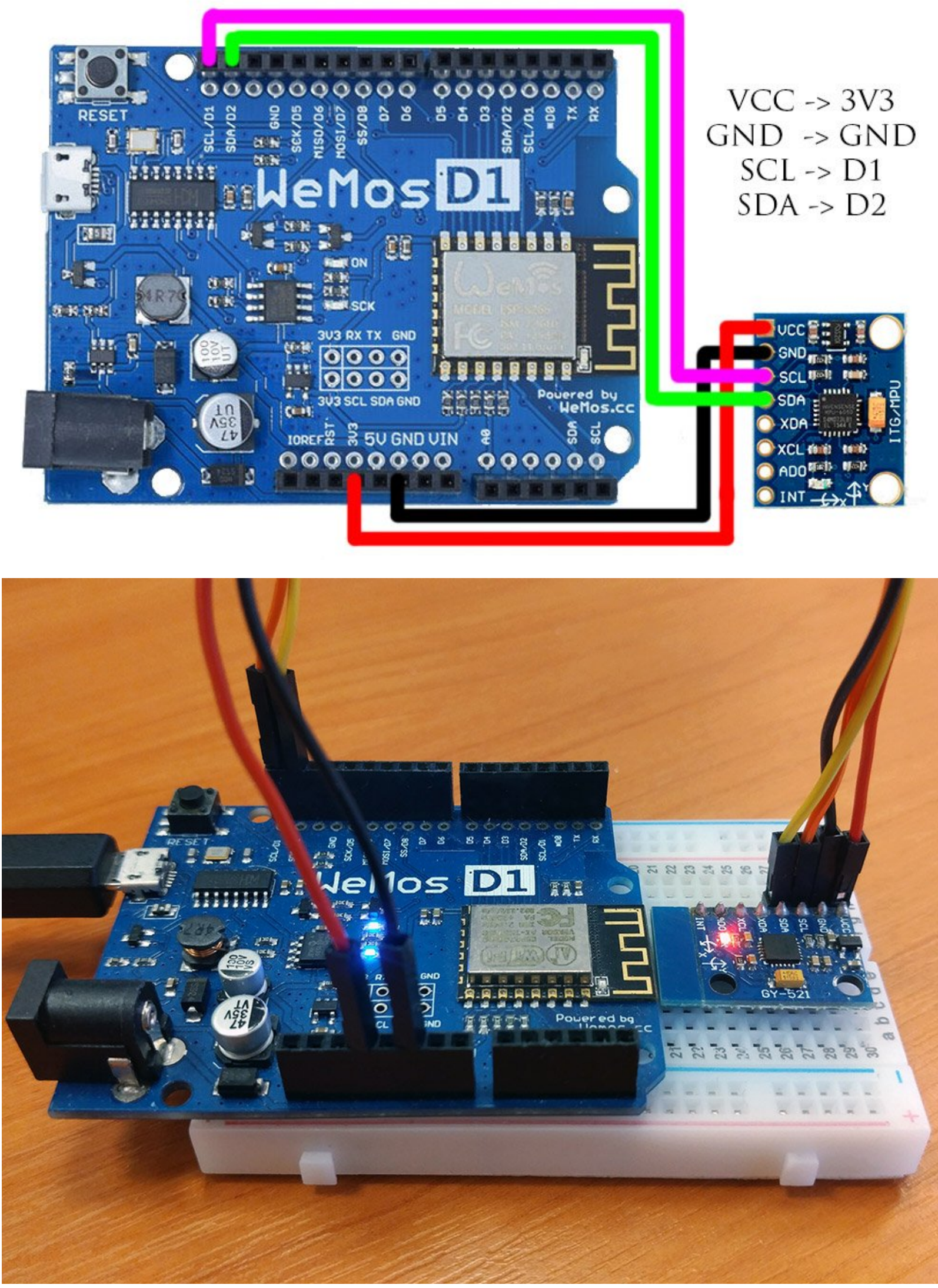


Основные характеристики MPU6050:

- 16-битный АЦП
- Напряжение питания 3-5В
- Поддержка протокола I2C
- Диапазон ускорений: ± 2 ± 4 ± 8 ± 16g
- Диапазон «гирос»: ± 250 500 1000 2000 °/s

Подключение MPU6050 к Wemos d1

Модуль работает по I2C, поэтому подключить его весьма просто, соединим акселерометр с платой **Wemos D1**: VCC -> 3V3, GND -> GND, SCL -> D1, SDA -> D2.

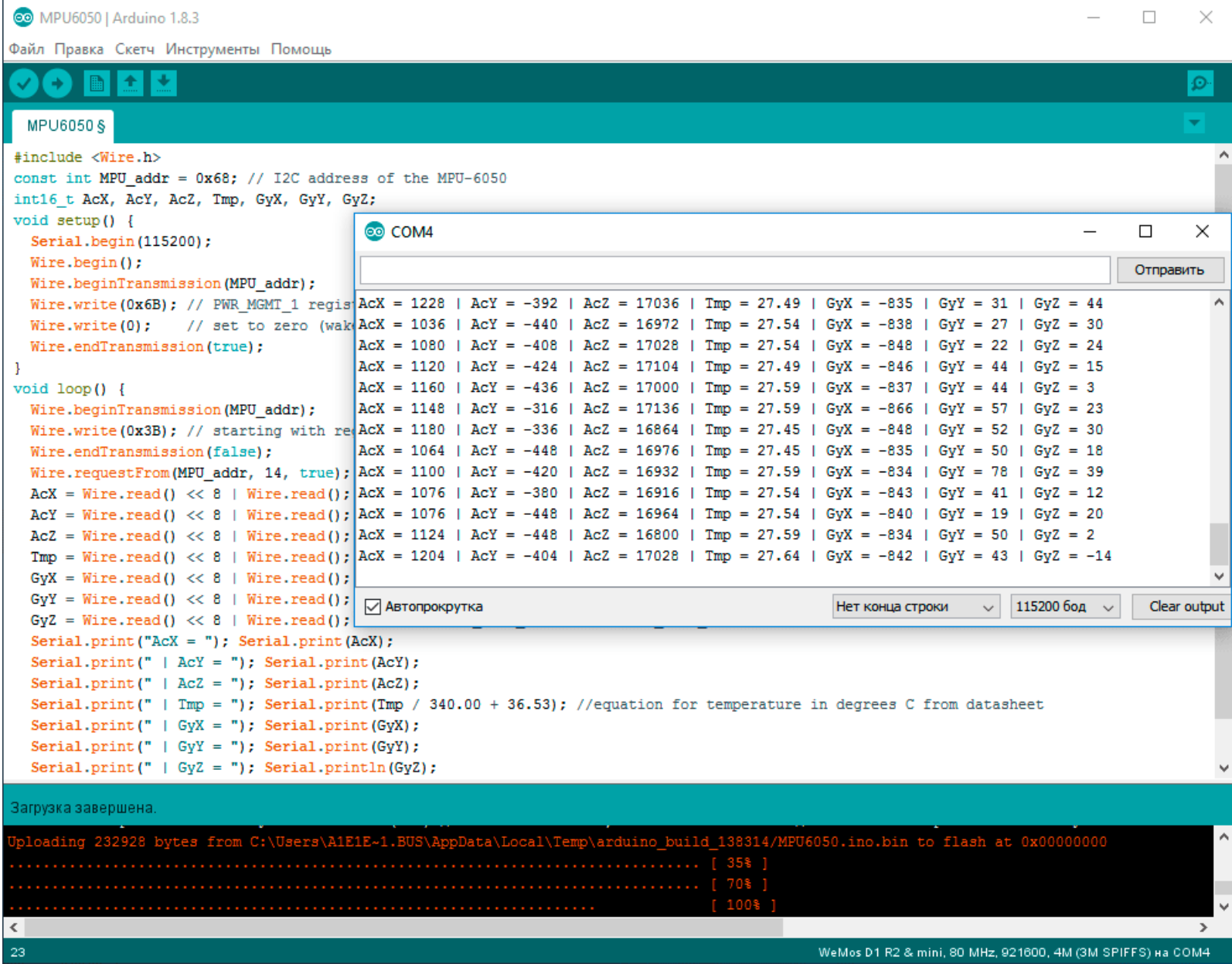


Для проверки давайте зальём в плату следующий скетч:

```
1 #include <Wire.h>
2
3 const int MPU_addr = 0x68; // I2C address of the MPU-6050
4 int16_t AcX, AcY, AcZ, Tmp, GyX, GyY, GyZ;
5
6 void setup() {
7   Serial.begin(115200);
8   Wire.begin();
9   Wire.beginTransmission(MPU_addr);
10  Wire.write(0x6B); // PWR_MGMT_1 register
11  Wire.write(0); // set to zero (wakes up the MPU-6050)
12  Wire.endTransmission(true);
13 }
14
15 void loop() {
16   Wire.beginTransmission(MPU_addr);
17   Wire.write(0x3B); // starting with register 0x3B (ACCEL_XOUT_H)
18   Wire.endTransmission(false);
19   Wire.requestFrom(MPU_addr, 14, true); // request a total of 14 registers
20
21   AcX = Wire.read() << 8 | Wire.read(); // 0x3B (ACCEL_XOUT_H) & 0x3C (ACCEL_XOUT_L)
22   AcY = Wire.read() << 8 | Wire.read(); // 0x3D (ACCEL_YOUT_H) & 0x3E (ACCEL_YOUT_L)
23   AcZ = Wire.read() << 8 | Wire.read(); // 0x3F (ACCEL_ZOUT_H) & 0x40 (ACCEL_ZOUT_L)
24   Tmp = Wire.read() << 8 | Wire.read(); // 0x41 (TEMP_OUT_H) & 0x42 (TEMP_OUT_L)
25   GyX = Wire.read() << 8 | Wire.read(); // 0x43 (GYRO_XOUT_H) & 0x44 (GYRO_XOUT_L)
26   GyY = Wire.read() << 8 | Wire.read(); // 0x45 (GYRO_YOUT_H) & 0x46 (GYRO_YOUT_L)
27   GyZ = Wire.read() << 8 | Wire.read(); // 0x47 (GYRO_ZOUT_H) & 0x48 (GYRO_ZOUT_L)
28 }
```

Скачать архив со скетчем: [MPU6050.zip](#)

Теперь, если вы откроете терминал com порта, то увидите примерно следующую картину: ускорение по трём осям, температура, углы положения по трем осям.



Из **даташита**, мы знаем, что при настройке чувствительности акселерометра до 2g на каждую единицу g приходится 16384 единиц измерения датчика. Самые наблюдательные наверно уже заметили по фото, что лежа на столе датчик почему-то показывает ускорения по горизонтальным осям, а по вертикальной показывает значение больше 16384 (я надеюсь все помнят со школы, что на все объекты на земле действует сила тяжести в 1g). Это проблема всех подобных акселерометров – они очень шумят. Их обязательно нужно калибровать и фильтровать получаемые данные.

Фильтр Калмана MPU6050

Давайте добавим в скетч упрощенный фильтр Калмана, для того чтобы сгладить шумы датчика.

```
1 #include <Wire.h>
2
3 const int MPU_addr = 0x68; // I2C address of the MPU-6050
4 int16_t AcX, AcY, AcZ, Tmp, GyX, GyY, GyZ;
5
6 // переменные для Калмана
7 float varVolts = 78.9; // среднее отклонение (щем в excel)
8 float varProcess = 0.5; // скорость реакции на изменение (подбирается вручную)
9 float Pc = 0.0, G = 0.0, P = 1.0, Xp = 0.0, Zp = 0.0, Xe = 0.0;
10 // переменные для Калмана
11
12 // Функция фильтрации
13 float filter(float val) {
14   Pc = P + varProcess;
15   G = Pc / (Pc + varVolt);
16   P = (1 - G) * Pc;
17   Xp = Xe;
18   Zp = Xp;
19   Xe = G * (val - Zp) + Xp; // "фильтрованное" значение
20   return (Xe);
21 }
22
23 void setup() {
24   Serial.begin(115200);
25   Wire.begin();
26   Wire.beginTransmission(MPU_addr);
27   Wire.write(0x6B); // PWR_MGMT_1 register
28   Wire.write(0); // set to zero (wakes up the MPU-6050)
29 }
```

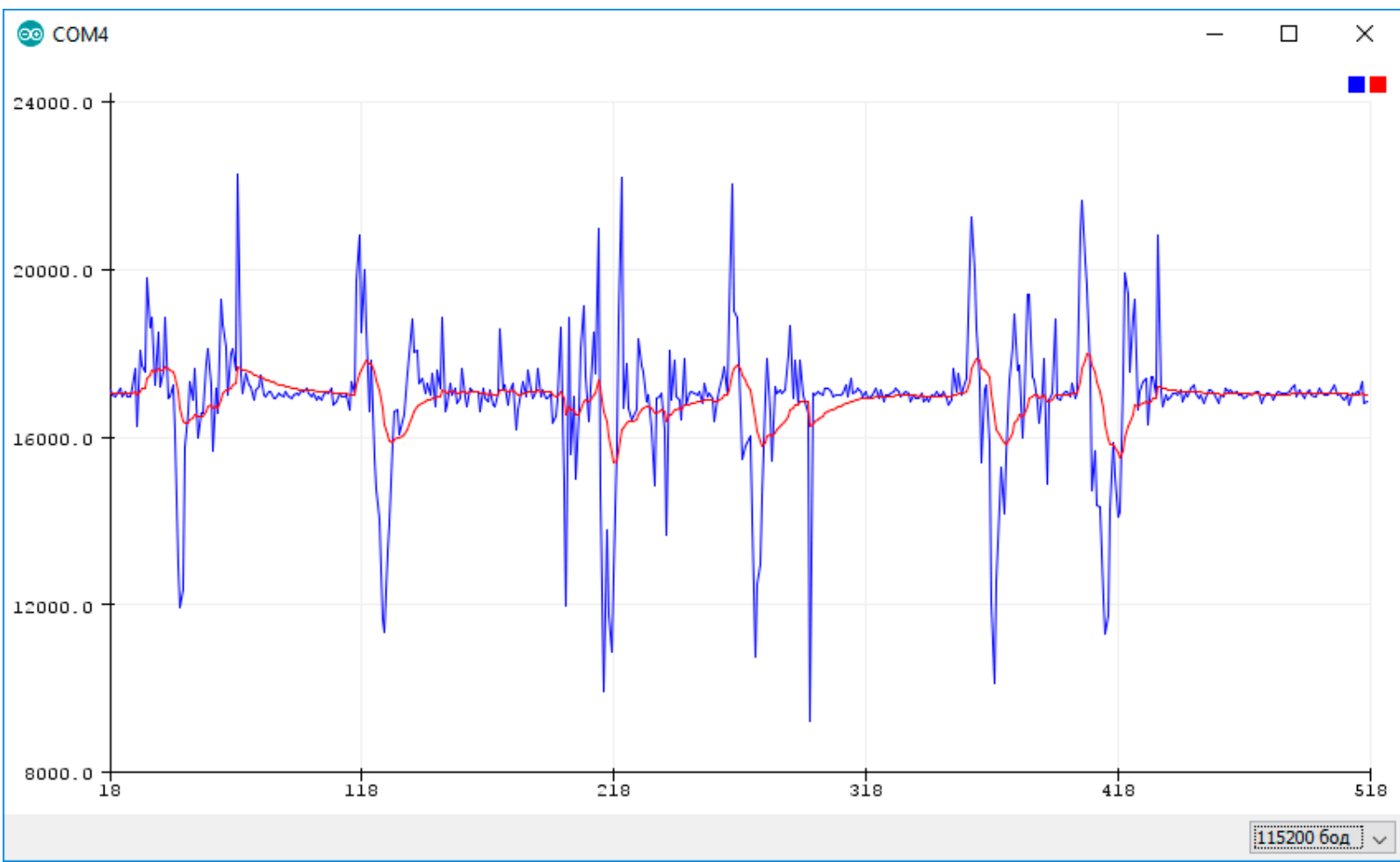
Скачать архив со скетчем: [MPU6050Kalman.zip](#)

Для фильтра нам необходимо подобрать два значения:

- varVolt - это среднее отклонение, его можно либо высчитать самостоятельно либо забить значения в файл excel который я положил в архив со скетчем
- varProcess - это время реакции, чем меньше это число, тем плавнее будут меняться значения.

Попробуйте поиграться с этими параметрами.

Теперь зайдем в меню **«Инструменты – Плоттер по последовательному соединению»**, откроется вот такое окно:



Синяя линия - это не фильтрованные данные, а красная это после обработки фильтром Калмана. Пики это - я поднимал и опускал акселерометр примерно на пол метра над столом. С такими данными уже можно будет комфортно работать.

На этом пока всё, в одной из следующих частей мы с вами соберем полноценный G-meter.