

Raspberry Pi Zero W + GPS + Compass + Magnetometer

🕒 Created	@Mar 15, 2021 1:22 PM
🕒 Property	@Mar 17, 2021 6:35 PM
🔗 Resources	

Установка операционной системы

Поскольку в Raspberry Pi нет внутренней памяти, установка операционной системы осуществляется на внешний носитель. Перед установкой карту памяти необходимо отформатировать. Для форматирования карты может использоваться программа SD Cart Formatter.

SD Memory Card Formatter 5.0 for SD/SDHC/SDXC

The SD Memory Card Formatter formats SD Memory Card, SDHC Memory Card and SDXC Memory Card (respectively SD/SDHC/SDXC Cards) complying with the SD File System

 <https://www.sdcard.org/downloads/formatter/>



После подготовки карты необходимо подготовить образ ОС, который будет установлен на карту. На RP может быть установлена практически любая операционная система. Для удобства работы советую использовать штатную свободно распространяемую систему Raspberry Pi OS.

Raspberry Pi OS - Raspberry Pi

The Raspberry Pi is a tiny and affordable computer that you can use to learn programming through fun, practical projects. Join the global Raspberry Pi community.

 <https://www.raspberrypi.org/software/>

Raspberry Pi OS



Для распаковки ОС на карту памяти существует большое количество программ. Наиболее удобной программой является balenaEtcher.

balenaEtcher - Flash OS images to SD cards & USB drives

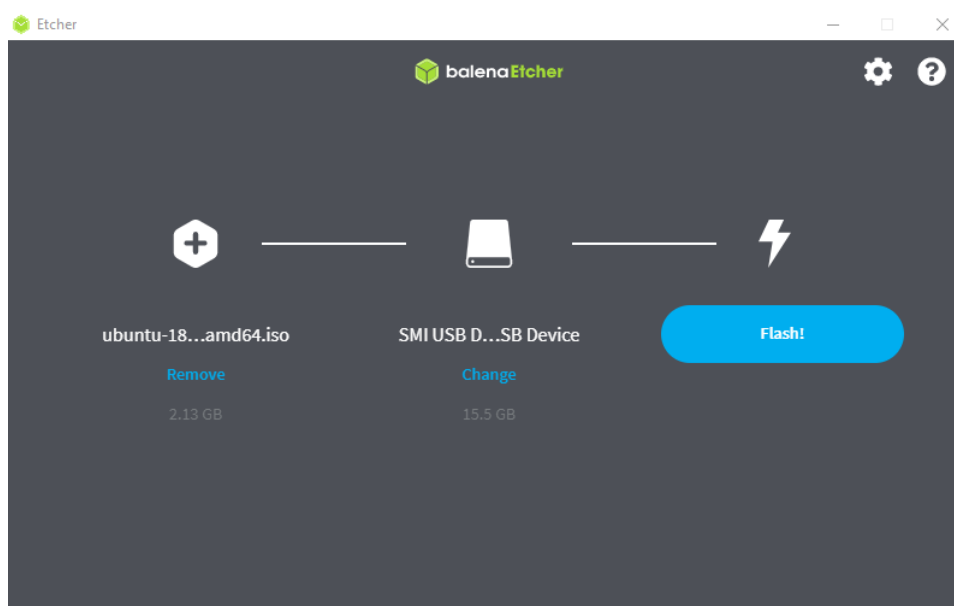
Electron is based on Gtk2, which can't run natively on Wayland. Fortunately, the XWayland Server provides backwards compatibility to run any X client on Wayland, including Etcher.

 <https://www.balena.io/etcher/>



Программа имеет интуитивно понятный интерфейс. Установка системы проходит в три этапа:

1. Выбор образа системы
2. Выбор носителя
3. Начало записи



Настройка беспроводного соединения

Настройка и управление RP могут быть осуществлены как с использованием периферийного оборудования (монитор, клавиатура, мышь), так и по беспроводному соединению.

Для удалённого доступа к RP необходимо подготовить и записать на флеш-карту с операционной системой два файла - ssh и wpa_supplicant.conf (примеры файлов есть в папке instructions).

Файл `ssh` (без расширения) должен быть пустым. Его запись на флеш-карту позволит подключиться к RP по SSH (Secure SHell).

Файл `wpa_supplicant.conf` нужен для того, чтобы при включении RP мог автоматически подключиться к сети WiFi. В файл необходимо пометстить следующий текст.

```
ctrl_interface_group=0
ap_scan=1
ctrl_interface=/var/run/wpa_supplicant
network={
    proto=WPA2
    pairwise=CCMP
    key_mgmt=WPA-PSK
    priority=2
    ssid="virg"
    psk="virg12345"
}
network={
    proto=WPA2
    pairwise=CCMP
    key_mgmt=WPA-PSK
    priority=1
    ssid="VIRG2"
    psk="39vuvvg31"
}
```

Параметр `priority` устанавливает порядок подключения микрокомпьютера к сетям WiFi. При наличии одновременного доступа к нескольким сетям, компьютер подключится к сети с параметром `priority=1`.

Блоков `network` может быть несколько, в зависимости от необходимости и доступности подключения к разным сетям WiFi.

Файлы `ssh` и `wpa_supplicant.conf` необходимо поместить в папку `boot` на флеш-карте с системой.

Теперь можно переходить к внутренней настройке RP.

Подключение к Raspberry Pi по SSH

Подключение и последующая настройка Raspberry Pi осуществляется по сетевому протоколу прикладного уровня SSH (Secure SHell). Для удалённого управления RP по SSH необходимо, чтобы микрокомпьютер и управляющее устройство были подключены к одной сети WiFi. После

подключения устройств к одной точке доступа нужно установить присвоенный микрокомпьютеру IP адрес. Для этого существует огромное число программ. Одной из наиболее удобных является программа для смартфонов Fing.

Порядок работы с программой прост.

1) Подключаемся смартфоном с установленной программой к сети WiFi.

2) Запускаем программу.

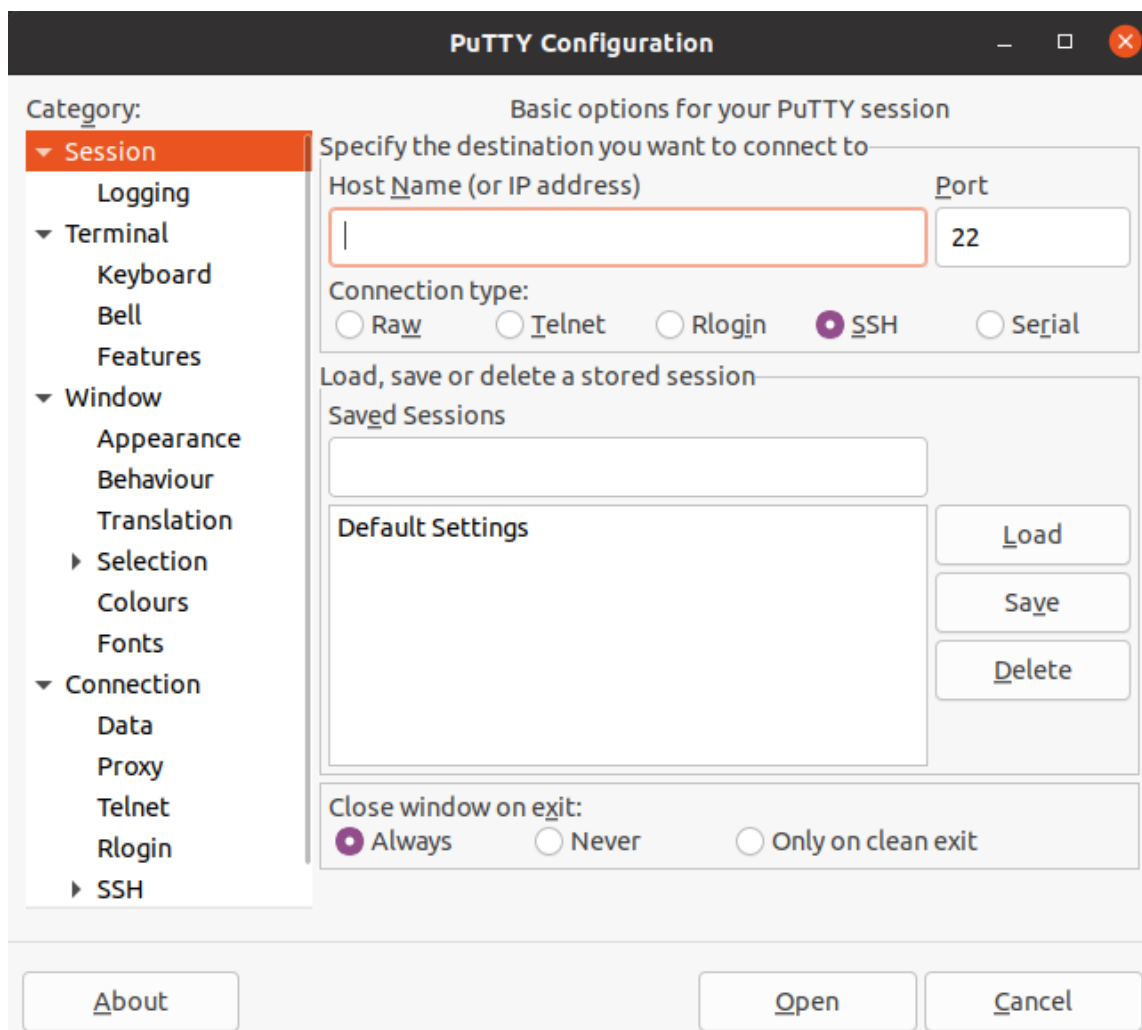
3) Нажимаем Scan Devices.

После этого появится список всех устройств в сети WiFi с их IP адресами. При правильно выполненных предыдущих шагах (создании файла `wpa_supplicant.conf`) среди прочих будет адрес RP.

Теперь можно подключаться к RP.

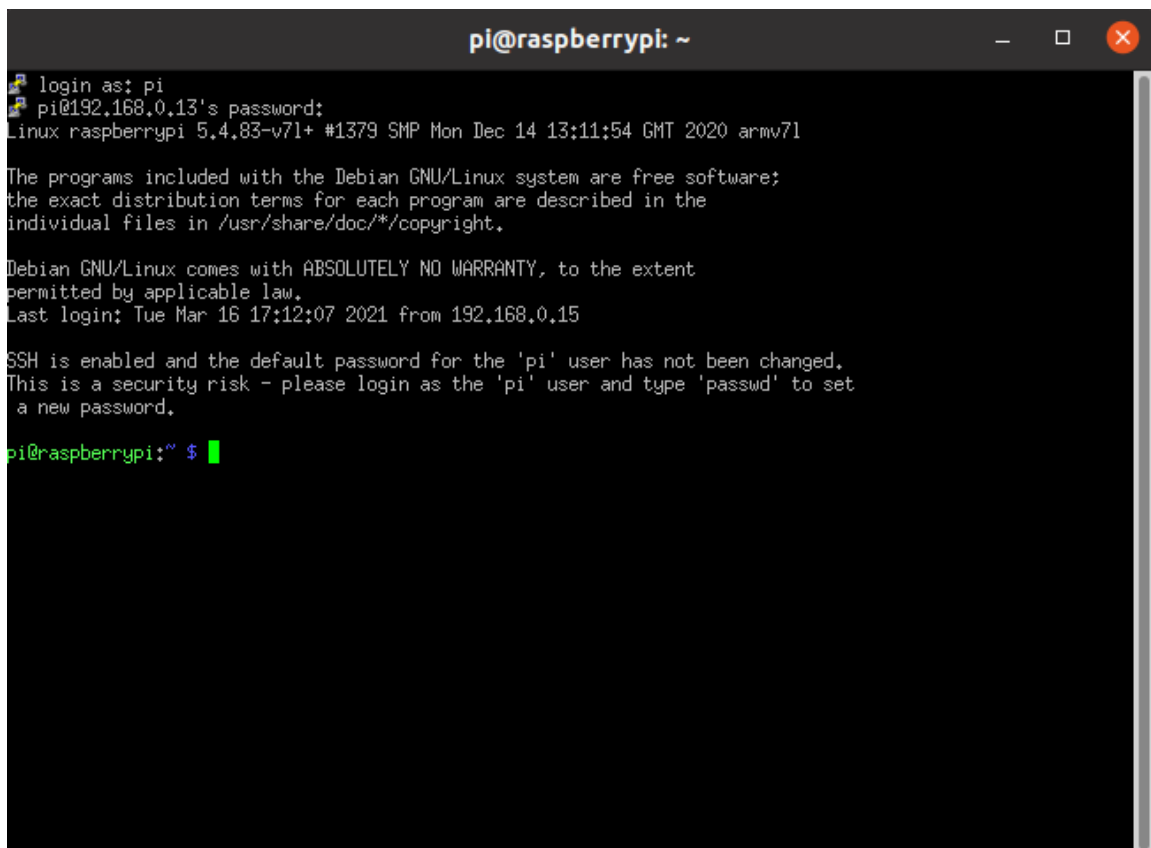
Подключение к RP с помощью программы PuTTY

Для дальнейшей настройки RP необходимо подключиться к нему по SSH. Для этого скачиваем и устанавливаем на управляющий компьютер программу PuTTY. Открываем её.



В поле Host Name (or IP address) вводим IP адресс RP.

В появившемся окне вводим логин - **pi** и пароль - **raspberry**.



```
pi@raspberrypi: ~
login as: pi
pi@192.168.0.13's password:
Linux raspberrypi 5.4.83-v7l+ #1379 SMP Mon Dec 14 13:11:54 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar 16 17:12:07 2021 from 192.168.0.15

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~$
```

Подключение к RP выполнено. Можно продолжать настройку.

Настройка параметров системы, переназначение UART

Для начала в командной строке вводим `sudo raspi-config`

В появившемся окне выбираем

`Interface Options → I2C → Yes`

`Interface Options → Serial → YES`

Выполнив данные команды, мы включили последовательную I2C шину, по которой будет осуществляться связь компаса и RP, а также последовательные порты для связи с периферийными устройствами.

Теперь необходимо настроить UART для подключения магнитометра и компаса.

Сперва в командной строке вводим `sudo nano /boot/config.txt` В открывшемся файле необходимо добавить строки

`enable_uart=1`

`dtoverlay=pi3-disable-bt`

```
dtoverlay=disable-bt
```

После этого нажимаем **Ctrl + X → Yes → Enter**.

Теперь в командной строке вводим

```
sudo systemctl stop serial-getty@ttyS0.service
```

```
sudo systemctl disable serial-getty@ttyS0.service
```

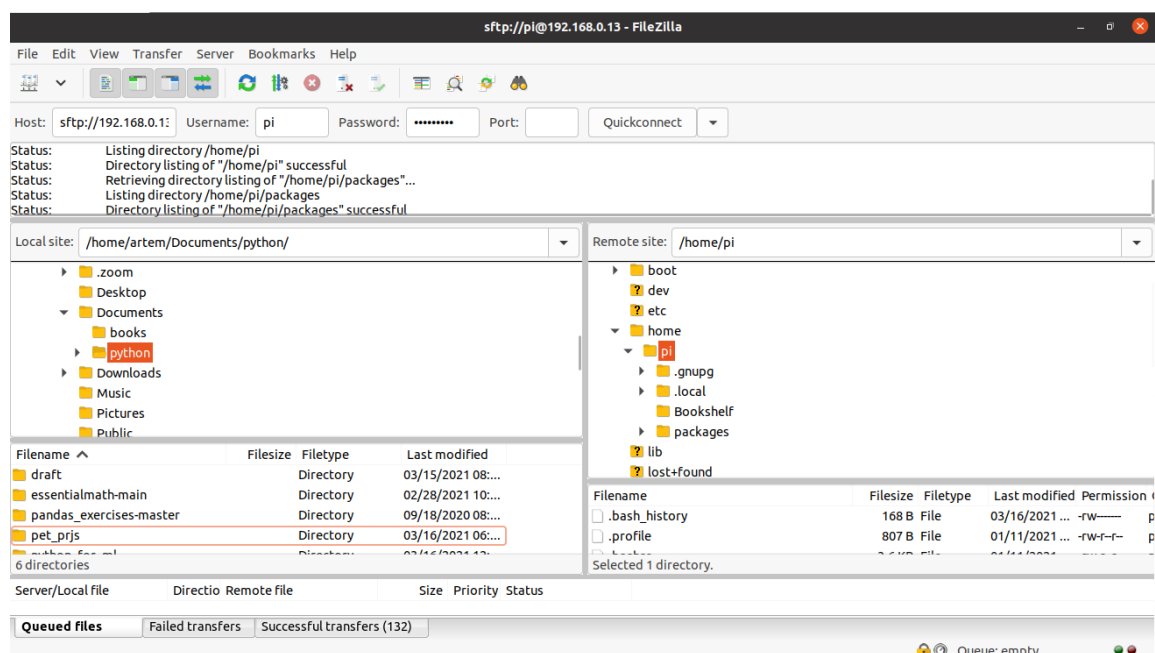
Перезагружаем RP командой `sudo reboot`.

Подключение к RP с помощью FileZilla, перенос файлов

Для переноса файлов между сторонним компьютером и RP необходимо скачать программу FileZilla.

В верхней части главного окна программы вводим IP - **192.168.0.xxx**, username - **pi**, password - **raspberrypi** и port - **22**.

После этого появится дерево каталогов RP. Переносим со стороннего компьютера в папку Documents папки data, scripts, packages.



Установка RP и необходимых библиотек

После переноса папок в RP возвращаемся в PuTTY и командную строку. Начинаем установку необходимых для работы библиотек.

Последовательно вводим в командной строке

```
sudo pip3 install -e ./Documents/packages/ublox-master
```

```
sudo pip3 install -e ./Documents/packages/hmc5883l-master
```

```
sudo pip3 install -e ./Documents/packages/py-qmc5883l-master
```

```
sudo pip3 install pyserial
```

Настройка автоматического запуска скриптов

Последним этапом перед запуском RP, как управляющего устройства, является настройка автоматического запуска скриптов. Для в командной строке вводим

```
sudo nano /etc/rc.local
```

В открывшемся файле после fi вводим***

```
sudo python3 /home/pi/Documents/scripts/heading_XXX.py &
```

```
sudo python3 /home/pi/Documetns/scripts/mag_com.py &
```

Нажимаем **Ctrl + X → Yes → Enter**.

*** В зависимости от чипа, на котором сделан компас, нужно использовать разные библиотеки. Поэтому для разных компасов написаны разные скрипты:

heading_hmc.py - для компасов на чипе HMC5883L

heading_qmc.py - для компасов на чипе QMC5883L.

XXX - нужно заменить на соответствующий постфикс.

Редактирование управляющих скриптов

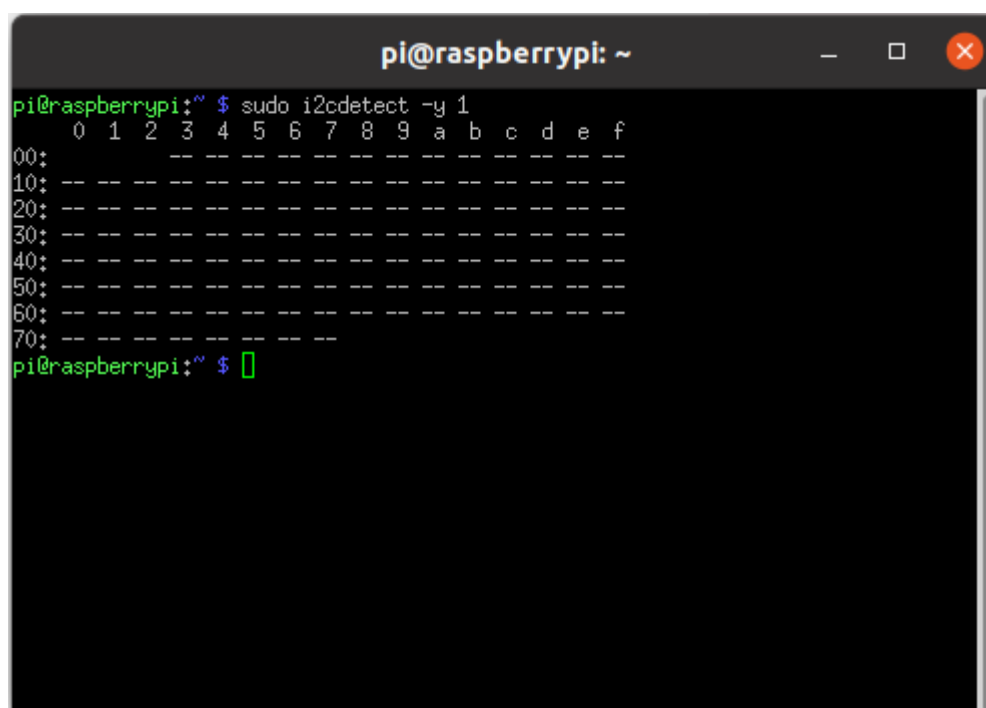
Перед перезагрузкой RP и тестировании работы скриптов советую проверить пути к файлам и директориям в скриптах. Дело в том, что иногда, например, при установке на флеш-карту системы версии Lite,

дерево каталогов может отличаться (например, отсутствовать папка Documents). В таком случае, пути к папкам и файлам в скриптах, а также пути при установке библиотек необходимо корректировать в соответствии с актуальной структурой проводника.

Кроме того, в скриптах heading_hmc.py / heading_qmc.py необходимо указать корректный адрес I2C соединения. Для этого в командной строке вводим

```
sudo i2cdetect -y 1
```

После этого в появившейся таблице должен появиться адрес компаса. (В таблице ниже подключенных устройств нет. При наличии подключенных устройств, в таблице отобразится адрес устройства (например, 1e)).



```
pi@raspberrypi:~$ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi:~$
```

Корректный адрес устройства необходимо вписать в скрипты в блок инициализации магнитометра:

в heading_hmc.py

```
mag = hmc5883l.hmc5883l(port=1, address= 0x1e , gauss=1.3,
declination=decl_tuple(decl))
```

в heading_qmc.py

```
mag = py_qmc5883l.QMC5883L(i2c_bus=1, address= 0x0d )
```

Если всё сделано правильно, после перезапуска Raspberry Pi скрипты управления периферийным оборудованием (GPS, магнитометр, компас) запустятся автоматически.