

## Setup Raspberry Pi in headless mode

Headless Mode means system runs without any primary input and output such as keyboard or monitor. The system won't use any desktop environment and therefore GUI applications cannot run. The headless mode is set from the installation step, and all settings are configured via files or terminals. Raspberry Pi OS has a lite version which is actually a headless mode OS.

[#pi](#) [#headless](#)

---

Last update: 2021-06-25 13:40:59

## Table of Content

1. Download Lite OS Image
2. Burn Image to SDCard
3. Enable SSH
4. Add WiFi Network
5. Direct USB connection (Pi Zero / Zero W Only)
6. Login to Pi
7. Update system (optional)
8. Expand Filesystem (optional)
9. Serial Console
10. Others (optional)

## 1. Download Lite OS Image

Official images for recommended operating systems are available to download from the Raspberry Pi website [download page](#).

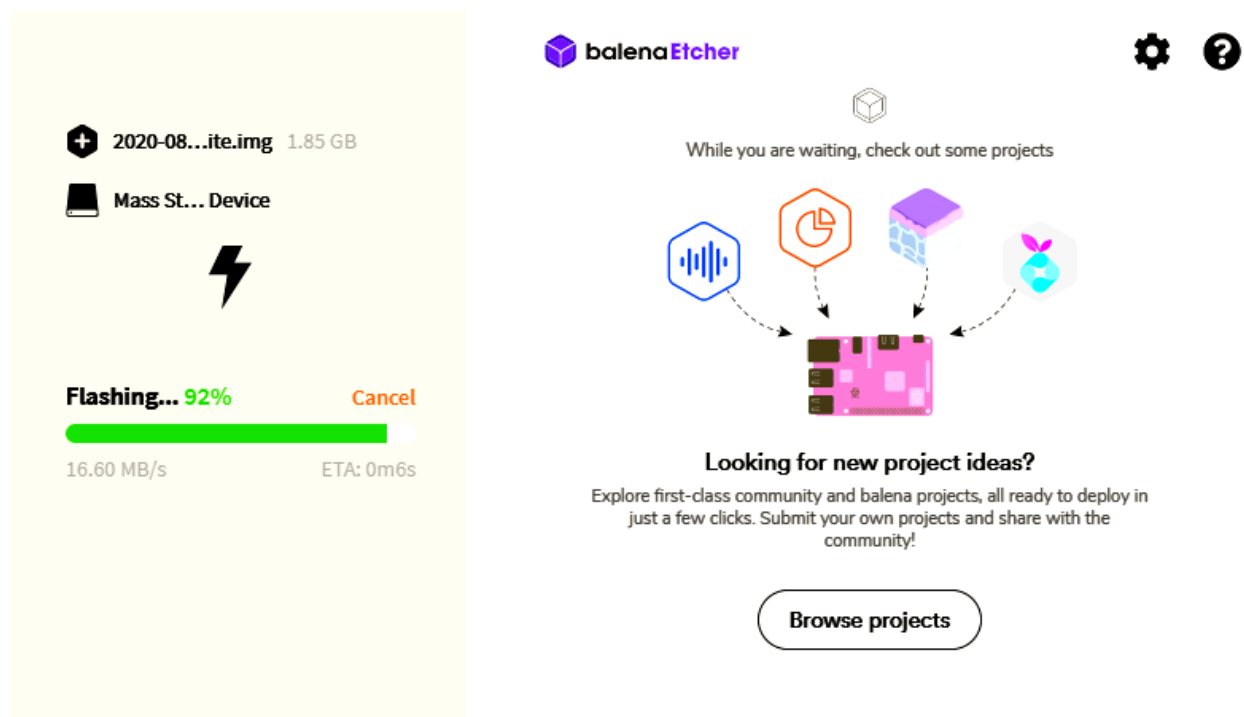
Headless Mode is available in the *Raspberry Pi OS Lite* version, please download the OS zip file in preparation. The release notes of Lite OS is listed [here](#).

## 2. Burn Image to SDCard

 **Raspberry Pi Imager** is the official Image Writer from Raspberry Pi.

The application **balenaEtcher** is a very good image writer that will write the OS image to an SDCard. Download and install it, then run it. Just follow the guided steps:

- **Select image** - browse to the zip file downloaded above
- **Select drive** - it may find the SDCard drive automatically
- Click **Flash**



*Etcher is writing the OS image*

After copying the image to the target SDCard, File Explorer may have trouble seeing the content of that SDCard. A simple fix is to pull the SDCard out then plug it back. It should appear with a partition named *boot*.

### 3. Enable SSH

For security reasons, Secure Socket Shell ( `ssh` ) is no longer enabled by default. To enable it, place a blank text file called `ssh` /\* no file extension \*/ in the root of the `boot` partition on the SDCard.

### 4. Add WiFi Network

To add a wireless network which Pi will automatically connect to, create a text file called `wpa_supplicant.conf` and place that file in the root of the `boot` partition on SDCard too.

In below config file, the `ssid` field is the WiFi Access Point name, and the `psk` field is the password of that WiFi.

wpa\_supplicant.conf

```
country=US
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="NETWORK-NAME"
    psk="NETWORK-PASSWORD"
}
```

See [ISO 3166-1 country codes](#) to fill into `country` field.

If the target network is a hidden one, an extra option in the `wpa_supplicant.conf` file, `scan_ssid=1` , may help connection.

Adding multiple wireless network configurations is allowed, with some extra fields to set the name and the priority (higher number gets connected first):

```
network={
    ssid="HomeOneSSID"
    psk="passwordOne"
    priority=1
    id_str="homeOne"
}

network={
    ssid="HomeTwoSSID"
    psk="passwordTwo"
    priority=2
    id_str="homeTwo"
}
```

boot (F:) >			Search boot (F:)
Name	Date modified	Type	
LICENCE.broadcom	5/27/2020 11:22 AM	BROADCOM	
ssh	11/27/2020 10:06 ...	File	
start.elf	8/20/2020 6:56 AM	ELF File	
start_cd.elf	8/20/2020 6:56 AM	ELF File	
start4db.elf	8/20/2020 6:56 AM	ELF File	
start4x.elf	8/20/2020 6:56 AM	ELF File	
wpa_supplicant.conf	11/27/2020 10:07 ...	CONF File	

*Added `ssh` and `wpa_supplicant.conf` in the **boot** partition*

### Direct USB connection (Pi Zero / Zero W Only)

See the below section to skip setting a WiFi network, as it will setup a virtual network over the USB connection.

## 5. Direct USB connection (Pi Zero / Zero W Only)

On Pi Zero / Zero W only, it's able to turn on USB OTG mode and the Pi will act as an USB slave with different modes: Serial, Ethernet, Mass storage device, etc. The research was published [here](#) and [here](#).


This step will setup Pi in USB OTG Ethernet mode, so that when plugging Pi into computer by the Peripheral USB port /\* labelled USB, not PWR \*/, there is virtual network will be created and Pi can be accessed over that network, and no need of an external WiFi network is required. This method also helps to power Pi over the USB port.

1. Add `dtoverlay=dwc2` on a new line in the `config.txt` file.
2. Open up the `cmdline.txt` file. Insert `modules-load=dwc2,g_ether` after `rootwait` with **only one space** between the text `rootwait` and the new text!!!

## 6. Login to Pi

Power Pi up and wait for the power led gets stable. Use any Network Scanner to detect the IP of the Pi. A plugin on [MobaXterm](#) can be used too.

For the official Raspberry Pi OS, the default user name is `pi`, with password `raspberrypi`, on the host `raspberrypi`.

 **Bonjour** is a service from Apple to discovery devices in a network using hostname. Install it and then Pi can be connected using its default hostname `raspberrypi.local`.

IP address range:  .  .  .  -->  Start scan

IP Address	Name	SSH	RDP	VNC	FTP	Telnet	Rlogin	HTTP	Other ports
192.168.1.1		X	X	X	X	X	X	✓	Deep scan 🔍
192.168.1.217		X	X	X	X	X	X	X	Deep scan 🔍
192.168.1.198	raspberrypi	✓	X	X	X	X	X	X	Deep scan 🔍

*Scan for Pi's IP*

## Review Network Settings

This command should list the network connection in the first line for *wlan0*:

```
iwconfig
```

This command should show info for *wlan0*:

```
ifconfig
```

This command should list the *wlan0* network with details:

```
iwlist wlan0 scan
```

## Connect to another Wifi network

Open the `wpa-supplciant` configuration file in nano:

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Go to the bottom of the file and add the following:

```
network={
    ssid="NETWORK-NAME"
    psk="NETWORK-PASSWORD"
}
```

Reconfigure the interface with:

```
wpa_cli -i wlan0 reconfigure
```

Verify whether it has successfully connected using `ifconfig wlan0`

## 7. Update system (optional)

To get the latest version of Pi OS and its packages, please update the system by entering below commands:

```
sudo apt-get update
sudo apt-get upgrade -y
```

## 8. Expand Filesystem (optional)

To use all of available space on the SDCard, expand the filesystem by running:

```
sudo raspi-config
```

Select **Advanced Options** → **Expand Filesystem**

Then reboot the system.

## 9. Serial Console

By default, the primary UART is assigned to the Linux console, and the secondary UART is connected to Bluetooth Module (on modle having Bluetooth feature - Pi Zero W, Pi 3, and Pi 4). Use `raspi-config` to disable the Linux serial console, then enable UART as an pheripheral.

## 10. Others (optional)

Please read the [Notes - Save power](#) post.