# Installing the MiniVIE on a Raspberry Pi 3

This guide will walk you through installing the MiniVIE onto a raspberry pi as a stand-alone unit. The Pi serves to:

1. Connect to two Myo Bands simultaneously using built-in Bluetooth
2. Hosts the ‘Mobile App’, which can be accessed by a smart-phone
3. Controls a virtual MPL (vMPL) within the unity environment either on a PC or an iOS device

## Flash the SD Card image

1. Download the SD card image file as a zip file and extract it.
2. Flash the .img image on an SD card >= 8GB
   1. On Windows/Mac, the ‘Etcher’ program is recommended
3. Insert the SD Card into a Raspberry Pi 3 and power on.  Note you don’t really need a keyboard or mouse (even if you do it will just give you a command prompt) – all the software runs ‘headless’.  Note: should you need it, the username is vradmin and the pass is \*\*\*\*\*

## Set the Myo band MAC Address

This step requires modifying configuration files on the raspberry pi

1. Determine the MAC address of the myo band(s) (<https://support.getmyo.com/hc/en-us/articles/207523646-Finding-the-MAC-address-of-a-Myo-armband>)
2. Open the vmpl\_user\_config.xml file for editing. Note there are a couple ways to do this:
   1. You can plug in a keyboard and monitor directly to the Pi to establish a console session with the Pi:
   2. You can ‘ssh’ into the device using a program like putty.exe on windows, or the command prompt on Mac/Linux
   3. You can ‘scp’ the files
3. Login to the device using the username and password above
4. > cd minivie
5. > nano vmpl\_user\_config.xml
6. Edit the following lines, updating the number of devices and MAC addresses as appropriate:

<!-- Myo Data Server Streaming Ports

Use these for establishing a Myo UDP Server that reads from BTLE and forwards

Packets to UDP from the local port to the remote port -->

<add key="MyoUdpServer.num\_devices" value="2"/>

<add key="MyoUdpServer.mac\_address\_1" value="xx:xx:xx:xx:xx:xx"/>

<add key="MyoUdpServer.mac\_address\_2" value="xx:xx:xx:xx:xx:xx"/>

<!-- Myo Data Client Streaming Ports

Use these parameters for reading from a Myo Data Source in a client application -->

<add key="MyoUdpClient.num\_devices" value="2"/>

1. Save and close the file (CTRL+O, CTRL+X)
2. Restart the Pi

## Connect the devices

1. First put your myo armband in the vicinity of the RPi and shake it to wake it up (initially the myo band should be ‘breathing’ with the blue myo logo).  If the connection is successful, a blue bar will appear at the bottom of the armband – this verifies it is connected.
2. Next, open your smartphone and connect to the MPLHOST wifi network.  Password is the same as above.  Note you can click on the (i) info button and confirm that you get a valid connection and IP address (e.g. 192.168.8.[X], where [X] is 5-20). You should also see that the wifi host address is 192.168.8.1
3. You can now open a web browser and direct it to the address:   [192.168.8.1:9090](http://192.168.8.1:9090)
4. The training interface should appear.
5. Once in the training interface, check the status message at the bottom to ensure the myo armband is streaming at 200Hz and the date/time is incrementing.  Swipe the pictures to select a movement and while performing the action, toggle the ‘add data’ switch to record examples of each movement.  As you are training, periodically ‘test’ the motion by observing the ‘current class’ status message and ensure it follows what you are doing.  Typically 150-400 samples of data is sufficient for a good training set.
6. The last item is to display the 3d rendered virtual arm.  We do have a windows, linux, mac, and iOS version of the software however the iOS version needs to be compiled from source and is a little buggy.  If you have a spare windows or linux computer that would be best.
7. Once you download the virtual arm, you need to connect the computer to the MPLHOST wifi and use a static IP address of 192.168.8.10
8. Once you have this it should move when you send motion commands.

That’s it!