

Intu on Raspberry Pi

Contents

Prerequisites	1
Install Raspbian Operating System onto your Raspberry Pi.....	1
Set up your Raspberry Pi for Builds	3
Retrieve the latest self-sdk and build Self on your Raspberry Pi.....	4
Run Self on your Raspberry Pi	6

Prerequisites

For this workshop, you will need the following:

- An account on rg-gateway.mybluemix.net with an Organization and Group set up in the Intu Gateway (See materials for Workshop 1)
- Raspberry Pi with power cable (Make and model?)
- Speaker (Make and model?)
- Microphone (Make and model?)
- Camera (Make and model?)
- 32 GB SD card or larger
- Monitor (with a HDMI connection)
- Keyboard and Mouse (USB connection)

Install Raspbian Operating System onto your Raspberry Pi

1. Go to: <https://www.raspberrypi.org/downloads/noobs/>
2. Click on the **Download ZIP** button under 'NOOBS (offline and network install)', and select a folder to save it to. (Save took 10:24 AM – 11:05 AM, and it failed. Had to start over. Restarted download at 11:08 AM completed 11:42 AM)
3. Extract the files from the zip to a local directory. (11:42 AM to 11:45 AM)
4. Open the “**Read Me**” text file in the extracted folder and follow the instructions to format your SD card.
5. Once your SD card has been formatted, drag all the files within the extracted NOOBS folder and drop them onto the SD card.
6. The necessary files will then be transferred to your SD card.

7. When this process has finished, safely eject the SD card from your computer, and remove the mini SD from it, then insert it into your Raspberry Pi.
8. Connect your Raspberry Pi to a power source, and connect external keyboard, mouse and monitor your Pi.
9. You should see a window on your monitor. Click on the **Wifi networks (w)** icon at the top of the window, select your network (at devcon, it will be ROBOT_PED1), and enter your password (are we going to change the password for ROBOT_PED1 for Devcon?)
10. As this is the first time your Raspberry Pi and SD card have been used, you'll have to select an operating system and let it install. Select **Raspbian [RECOMMENDED]**.
11. At the bottom of your screen, you will see a **Language** panel for your keyboard – select **English (US)**.
12. Click on the **Install** icon on the top left of the window. A **Confirm** window will appear; select **Yes**. The installation may take up to 30-40 mins.
13. You may need to retype in your network password after the operating system has successfully installed if it has disconnected from the network. This will reboot the system.
14. Check again that you are connected to the network. On the right of the Bluetooth icon in the top right corner of the screen, you will see a successful connection to your wifi network - as shown by a blue wifi icon - or an icon with two red crosses if you have been disconnected. To reconnect, click on the two red crosses, select your network, and type in the password.
15. Get the IP address of the Pi.

1. Click on the **Terminal** icon on the top left toolbar.
2. Type in **ifconfig** and hit Enter.
Look for the wlan0 section. The **inet addr** gives you the address (e.g. 10.0.1.2).
3. Now you can ssh into the Raspberry Pi from your laptop using:
ssh pi@{ip_address}
(e.g. ssh pi@10.0.1.2)
You will be prompted for a password. The default password is: **raspberry**.

PC Users use Putty to connect to ip_address, then when prompted, username is pi and password is raspberr

If you cannot connect from any OS, reboot the Pi by **connecting to your Pi via ssh (or putty ssh) and running the command: sudo reboot**

At this point, you can disconnect the external monitor, mouse and keyboard from the Pi. Go back to your laptop (make sure it is on the same Wifi network). **Now make sure you reboot**

the Pi: **sudo reboot** by connecting to your Pi via ssh (or putty ssh) and running the command: **sudo reboot**

Set up your Raspberry Pi for Builds

Note that if any step below fails or errors, SSH to your pi and run: **sudo apt-get update**, then repeat the step.

1. Go to your local computer and download **Anaconda 4.2.0 For Linux Python 2.7 version** onto your local laptop from <https://www.continuum.io/downloads>

Make sure you download the correct version. You need the LINUX version no matter what operating system your local computer uses. PC Users may have to right click and select Save as to save the download locally.

2. On your laptop, navigate to the directory where you downloaded Anaconda locally to your laptop and copy it over to the Raspberry Pi using the following command (on your laptop terminal for Mac or command for PC, PC users may need something like Filezilla to move the files from the local machine to the Pi):

```
scp {path_to_downloaded_file}/Anaconda2-4.2.0-Linux-x86.sh  
pi@{ip}:/home/pi
```

3. ssh into your Pi: **ssh pi@{ip_address}**
PC Users use Putty to connect to ip_address, then when prompted, username is pi and password is raspberry
4. In your ssh or putty window, run the following command:
 - a. **bash Anaconda2-4.2.0-Linux-x86.sh**
 - b. Follow the steps on the screen to install Anaconda. When you get to the license, keep hitting **Enter** to jump to the bottom. Type **“yes”** to approve the license.
 - c. Hit **Enter** to install in the default location.
 - d. Note: It may take a while for the progress to update.
 - e. If you get the following error, please ignore it.

```
Anaconda2-4.2.0-Linux-x86.sh: line 484: /home/pi/anaconda2/pkgs/python-3.5.2-  
0/bin/python: cannot execute binary file: Exec format error  
ERROR:  
cannot execute native linux-32 binary, output from 'uname -a' is:  
Linux raspberrypi 4.4.21-v7+ #911 SMP Thu Sep 15 14:22:38 BST 2016 armv7l GNU/Linux
```

5. In your ssh or putty window, run: **sudo apt-get install python-pip cmake**
 - a. If this fails, run **sudo apt-get update**, and then rerun: **sudo apt-get install python-pip cmake**

6. In your ssh or putty window, run: **sudo pip install qibuild**
7. Install the **wiringPi** library by doing the following in your ssh or putty window:
 - a. Navigate to your home directory by entering: **cd /home/pi**
 - b. Then type this command: **git clone git://git.drogon.net/wiringPi**
 - c. Navigate into the **wiringPi** directory by entering: **cd wiringPi/**
 - d. Run: **./build**You should see a list of classes compiled and “All Done” at the end.
8. Create a self directory via the ssh connection by navigating back to your home directory with **cd /home/pi** and then entering the following in your ssh or putty window:
mkdir self
9. In your ssh or putty window, navigate into the self directory: **cd self**

Retrieve the latest self-sdk and build Self on your Raspberry Pi

NOTE: Need to put the self-sdk file somewhere external for conference attendees.

1. The following instructions are specific for the github location...
On your local laptop, open a browser and go the the following URL:
<https://github.ibm.com/watson-labs-austin/self-sdk>
Click on **Clone or download** and select **Download Zip**.
2. Next you need to move the self-sdk-develop.zip file that you just downloaded to your Pi using the following steps.
 - a. For Mac
 - i. On your local machine, navigate to the directory where the zip file was downloaded and copy it to the **self** directory on the Raspberry Pi using the following command:
scp self-sdk-develop.zip pi@{ip}:/home/pi/self/
 - ii. You will be prompted for the Pi’s password: raspberry
 - b. For PC
 - i. Open Filezilla
 - ii. Connect to your Pi
 - iii. In the Local Site side of the screen, navigate to the self-sdk-develop.zip file
 - iv. In the Remote site side of the screen, navigate to the directory: **/home /pi/self**
 - v. Click on the file self-sdk-develop.zip on the Local site side of the screen then drag it to the Remote side side of the screen.
 - vi. You can monitor the progress in the panel located at the bottom of the Filezilla screen.
3. Unzip the file on your Pi with the following steps:

- a. In your ssh or putty window, verify that you are still in the self directory by verifying that your prompt reads: `pi@raspberrypi:~/self $`
 - b. Then type the following command: `unzip self-sdk-develop.zip`
4. Prepare to edit your body.json with your Intu Gateway credentials with the following steps:
 - a. Retrieve the body.json from your Raspberry pi and put it on your local laptop.
 - i. Mac
 1. On your Mac, SCP the body.json file from your pi to a local directory on your Mac with the following SCP command:
`$ scp pi@[pi ip address]:/home/pi/self/self-sdk-develop/bin/raspi/etc/profile/body.json ~/`
 2. On your Mac, open the body.json with your favorite local editor
 - ii. PC
 1. On your PC, open Filezilla and connect to your Raspberry Pi.
 2. On the Remote site side of the Filezilla screen, navigate to `/home/pi/self/self-sdk-develop/bin/raspi/etc/profile`
 3. Locate the body.json file in the profile folder, right click on it and select View/Edit
5. Retrieve your Gateway credentials (You should already have completed Workshop 1 for your Mac or Windows and created an Organization and Group on the Intu Gateway)
 - a. On your local machine, open a browser and sign into `rg-gateway04.mybluemix.net`
 - b. Select View Credentials from the link on the left hand side of the page
 - c. On the View Credentials page, use the filters to select your organization and group. Click the **Get Credentials** button.
 - d. Click the **Copy** link to save the entire contents to your clipboard.
6. Edit the body.json that you opened for editing by adding the credentials you just retrieved from the gateway to it. The following steps describe how to do this.
 - a. In the body.json window, search on "m_EmbodimentCreds" to jump to the correct section.
 - b. Then highlight/select everything from "m_EmbodimentCreds" to "m_OrgId": "",
 - c. Click your delete key on your keyboard
 - d. Then paste the credentials that you copied to your clipboard into that space. Save your changes to the file. Then read on for next steps before doing anything else.
7. Save the revised body.json file to the pi with the following instructions.
 - a. On a Mac
 - i. Close the file.
 - ii. Open terminal and use the following command to copy your saved body.json from your local directory back to your pi
`$ scp ~/body.json pi@[pi ip address]:/home/pi/self/self-sdk-develop/bin/raspi/etc/profile/`
 - b. On a PC

- i. Exit the window. If you are using Filezilla, you will be prompted in a window to upload the file back on the server.
 - ii. Click the **Yes** button. This action saves your changes to your pi.
8. Build Self on your Pi with the following steps:
 - a. Navigate into the self-sdk-develop directory: `cd self-sdk-develop`
 - b. Mark the build script as executable by running:
`chmod +x scripts/build_raspi.sh`
 - c. Now run: (*NOTE: If you have done this before run scripts/clean.sh)
`scripts/build_raspi.sh`

Run Self on your Raspberry Pi

Run Self on your Pi by completing the following steps in your ssh or putty window.

Note: The following four steps will need to be repeated each time you power your Pi back up (i.e. unplug power and plug it back in).

1. First, verify that you do NOT have HDMI plugged into your pi
2. Verify that you have a microphone and speaker plugged into your pi. Your speaker may need to be charged first. It will need to be plugged in and on before proceeding to the next step.
3. Then go to your local machine and ssh/putty connect to your pi
4. Next, navigate to the raspi directory using: **`cd bin/raspi`**
5. Run: **`export LD_LIBRARY_PATH=~self/self-sdk-develop/bin/raspi`**
6. Run: **`export WIRINGPI_GPIOMEM=1`**
7. Run: **`./self_instance -c 0 -f 0`**