

## Setting up a BeagleBone Blue for robotics using Cloud9 and Python

- A [BeagleBone Blue](#) is an amazing tiny robotics-oriented computer
- [Cloud9](#) is a browser-based integrated development environment
- [Python](#) is a programming language that's good for robotics

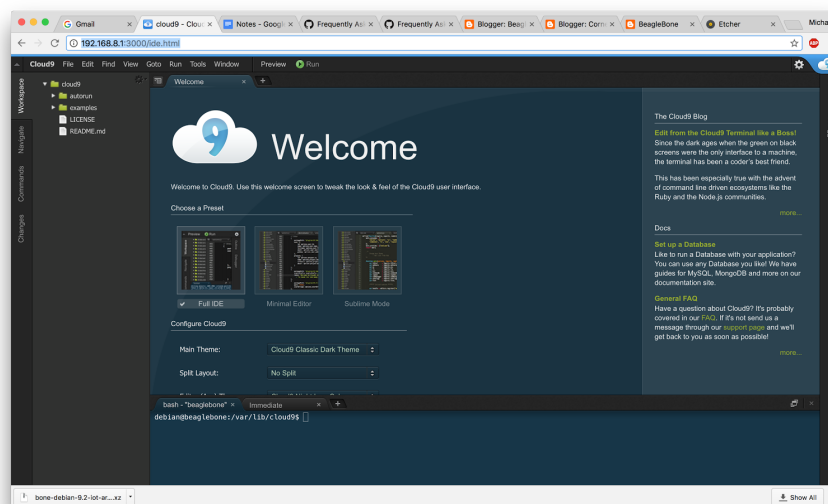
## Make a bootable microSD card using the [debian](#) (linux) operating system

1. Go to <http://beagleboard.org/latest-images>
2. Choose: *Stretch IoT (non-GUI) for BeagleBone and PocketBeagle via microSD card* ("Stretch" refers to the major version of debian and IoT (non-GUI) omits user interface components that are not needed)
  - 2.1. Install [Etcher](#)
  - 2.2. Use Etcher to move the downloaded image to a microSD card
  - 2.3. Etcher is ridiculously easy to use – but the process takes a while
  - 2.4. Install the microSD card into the BeagleBone

## Power up the BeagleBone and connect to Cloud9, by using either (1) or (2):

1. using a USB cable: browse to either <http://192.168.6.2:3000> or <http://192.168.7.2:3000>, depending on the USB networking drivers provided by your operating system.
2. using an external power supply: connect to the BeagleBoard wifi access point (SSID: *BeagleBone-XXXX*, where XXXX is a unique ID; Password: *BeagleBone*) and browse to <http://192.168.8.1:3000>.

Either way, Cloud9 should appear in the browser. Notice that the lower right pane is actually a terminal (shell) session giving command-line access:



### Alternative way to gain command-line access

1. As an alternative to gain command-line access, connect using a USB cable and use:  
`ssh debian@192.168.7.2` (password = temppwd)

### Connect the BeagleBone to the internet by configuring its wifi adapter

Note that the token **highlighted in yellow** is an example; use a choice listed by “services”:

```
debian@beaglebone:~$ sudo -s (become superuser/root)
[sudo] password for debian: temppwd (use the default debian password)
root@beaglebone:/home/debian# connmanctl (starts the connmanctl program)
connmanctl> tether wifi off (not really necessary on latest images)
connmanctl> enable wifi (not really necessary)
connmanctl> scan wifi (scans for available network access points)
connmanctl> services (lists services in a strange technical format)
connmanctl> agent on
connmanctl> connect wifi_f45eab2f1ee1_6372797774616c_managed_psk
connmanctl> quit
```

### Install any pending software updates for debian & roboticscape

```
debian@beaglebone:~$ sudo apt-get update
[sudo] password for debian: temppwd

debian@beaglebone:~$ sudo apt-get upgrade roboticscape
[sudo] password for debian: temppwd
```

If prompted to choose a program to run when roboticscape starts, choose “existing”.

### Reboot the BeagleBone

```
debian@beaglebone:~$ sudo reboot
(reconnect using Cloud9 or ssh)
```

### Install Python3.x, pip, and rcpy

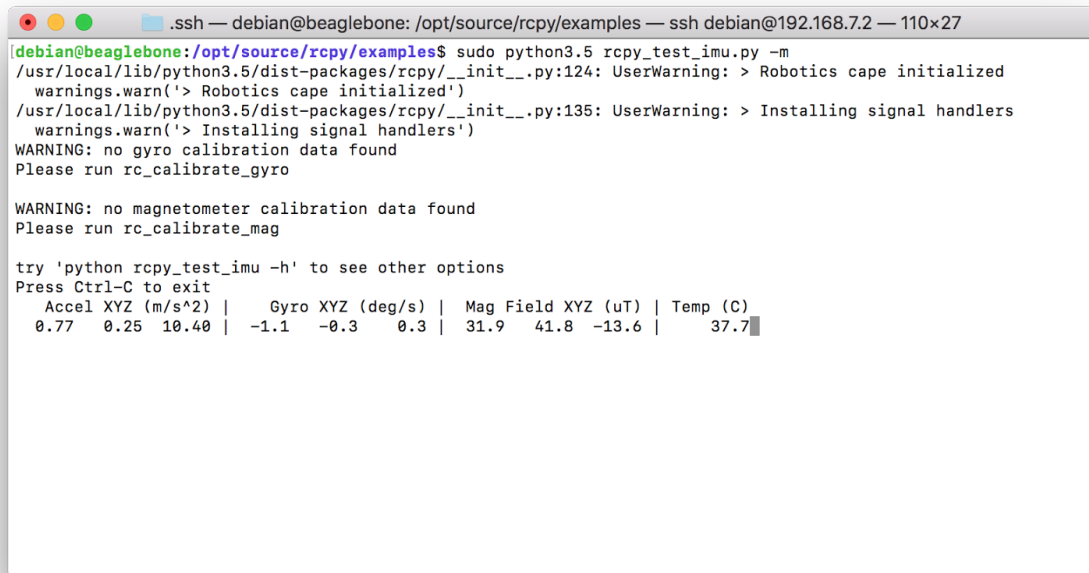
```
debian@beaglebone:~$ sudo apt install python3 python3-pip
debian@beaglebone:~$ sudo pip3 install rcpy
```

Questions? Contact [jeff@jvon.org](mailto:jeff@jvon.org) or [michael@jvon.org](mailto:michael@jvon.org)

**Try running some of the rcpy examples... note: roboticscape only runs as root user**

```
debian@beaglebone:~$ cd /opt/source/rcpy/examples
```

```
debian@beaglebone:/opt/source/rcpy/examples$ sudo python3.5 rcpy_test_imu.py -m
```



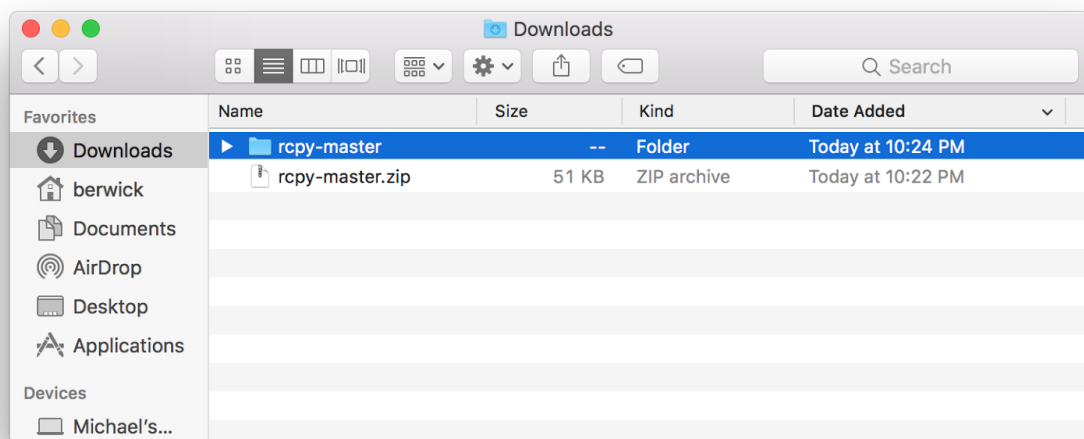
```
ssh — debian@beaglebone: /opt/source/rcpy/examples — ssh debian@192.168.7.2 — 110x27
[debian@beaglebone:/opt/source/rcpy/examples$ sudo python3.5 rcpy_test_imu.py -m
/usr/local/lib/python3.5/dist-packages/rcpy/__init__.py:124: UserWarning: > Robotics cape initialized
warnings.warn('> Robotics cape initialized')
/usr/local/lib/python3.5/dist-packages/rcpy/__init__.py:135: UserWarning: > Installing signal handlers
warnings.warn('> Installing signal handlers')
WARNING: no gyro calibration data found
Please run rc_calibrate_gyro

WARNING: no magnetometer calibration data found
Please run rc_calibrate_mag

try 'python rcpy_test_imu -h' to see other options
Press Ctrl-C to exit
  Accel XYZ (m/s^2) |   Gyro XYZ (deg/s) |  Mag Field XYZ (uT) | Temp (C)
    0.77   0.25  10.40 |   -1.1   -0.3    0.3 |  31.9  41.8 -13.6 |   37.7
```

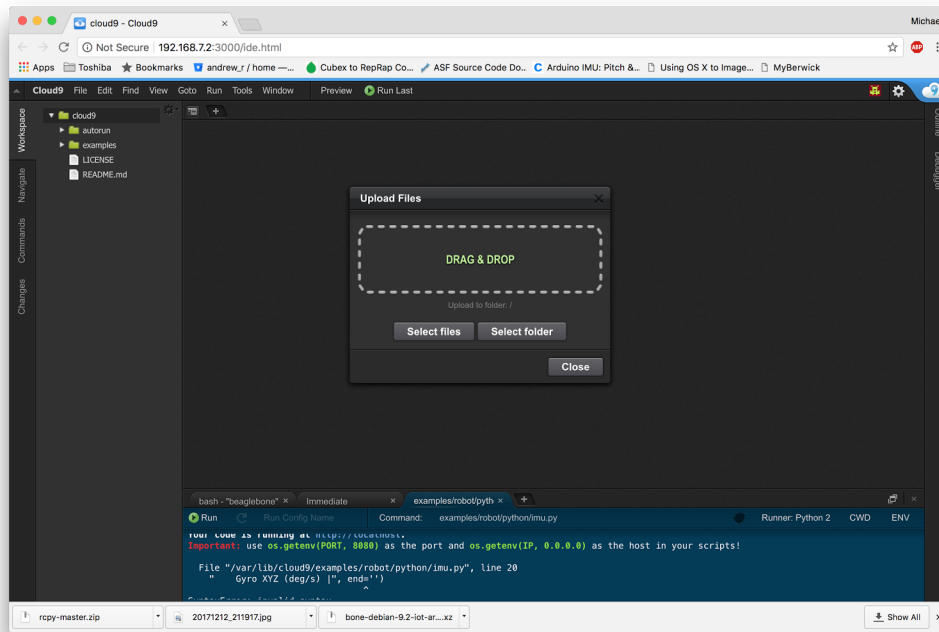
## Introduce a copy of rcpy as a Cloud9 project

1. Download a copy from <https://github.com/mcdeoliveira/rcpy> (use “Clone or download” button to download a zip copy of the entire project)
2. Unzip locally:

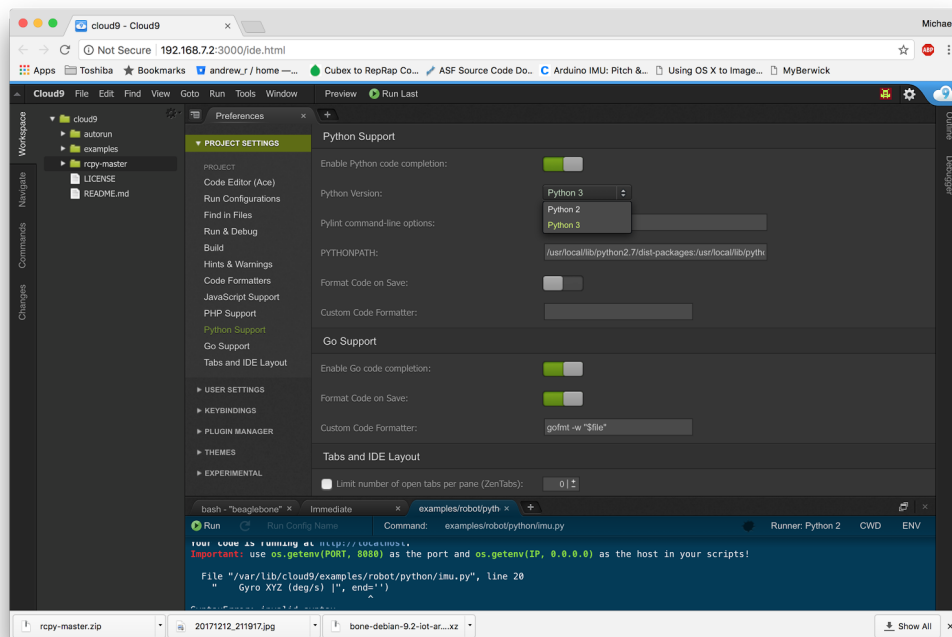


Questions? Contact [jeff@jvon.org](mailto:jeff@jvon.org) or [michael@jvon.org](mailto:michael@jvon.org)

3. Use the Cloud9 *File/Upload Local Files...* menu option to upload the unzipped folder:

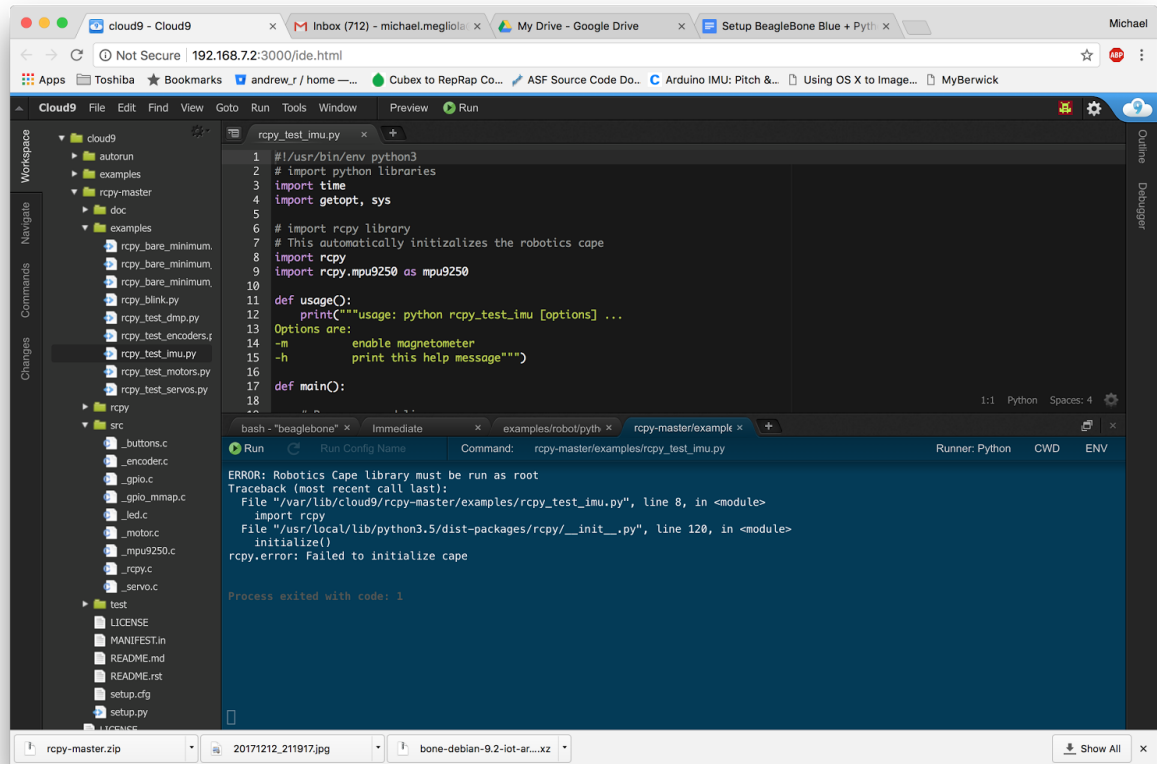


4. Use the Cloud9/Preferences menu option to switch from Python2 to Python3:



Questions? Contact [jeff@jvon.org](mailto:jeff@jvon.org) or [michael@jvon.org](mailto:michael@jvon.org)

5. Selecting and running **rcpy\_test\_imu.py** will fail, because Cloud9 does not start as the root user, and the roboticscape software requires root access:



The screenshot shows a Cloud9 IDE interface. The left sidebar displays a file tree with a project named 'rcpy-master'. The main editor window shows the file 'rcpy\_test\_imu.py' with the following code:

```
1 #!/usr/bin/env python3
2 # import python libraries
3 import time
4 import getopt, sys
5
6 # import rcpy library
7 # This automatically initializes the robotics cape
8 import rcpy
9 import rcpy.mpu9250 as mpu9250
10
11 def usage():
12     print("""usage: python rcpy_test_imu [options] ...
13 Options are:
14 -m enable magnetometer
15 -h print this help message""")
16
17 def main():
18     ...
```

Below the code editor, a terminal window shows the command 'rcpy-master/examples/rcpy\_test\_imu.py' being run. The output is an error message:

```
ERROR: Robotics Cape library must be run as root
Traceback (most recent call last):
  File "/var/lib/cloud9/rcpy-master/examples/rcpy_test_imu.py", line 8, in <module>
    import rcpy
  File "/usr/local/lib/python3.5/dist-packages/rcpy/__init__.py", line 120, in <module>
    initialize()
rcpy.error: Failed to initialize cape
Process exited with code: 1
```

6. To correct this problem: remove the line **User=1000** from `/lib/systemd/system/cloud9.service` (allowing Cloud9 to run as root) and reboot:

`debian@beaglebone:/lib/systemd/system$ sudo vi cloud9.service`



The screenshot shows a terminal window with the following content:

```
Unit
Description=Cloud9 IDE
ConditionPathExists=/var/lib/cloud9

[Service]
WorkingDirectory=/opt/cloud9/build/standalonebuild
EnvironmentFile=/etc/default/cloud9
ExecStartPre=/opt/cloud9/cloud9-symlink
ExecStart=/usr/bin/nodejs server.js --packed -w /var/lib/cloud9
SyslogIdentifier=cloud9ide
User=1000

"cloud9.service" [readonly] 11L, 306C
```

Questions? Contact [jeff@jvon.org](mailto:jeff@jvon.org) or [michael@jvon.org](mailto:michael@jvon.org)

**The BeagleBone Blue should now run rcpy under Python3.x in Cloud9:**

The screenshot shows a Cloud9 IDE window with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a project structure with folders like 'cloud9', 'examples', and 'rcpy-master'. The code editor displays the contents of 'rcpy\_test\_imu.py', which is a Python script for testing the rcpy library. The terminal shows the output of running the script, including warnings about signal handlers and a table of sensor data.

```
1 #!/usr/bin/env python3
2 # import python libraries
3 import time
4 import getopt, sys
5
6 # import rcpy library
7 # This automatically initializes the robotics cape
8 import rcpy
9 import rcpy.mpu9250 as mpu9250
10
11 def usage():
12     print("usage: python rcpy_test_imu [options] ...")
13     print("Options are:")
14     print("-m enable magnetometer")
15     print("-h print this help message")
16
17 def main():
18     # ...
```

bash - "beaglebone" x Immediate x examples/robot/pyth x rcpy-master/example  
Stop Run Config Name Command: rcpy-master/examples/rcpy\_test\_imu.py Runner: Python CWD: ENV  
/usr/local/lib/python3.5/dist-packages/rcpy/\_\_init\_\_.py:124: UserWarning: > Robotics cape initialized  
warnings.warn('> Robotics cape initialized')  
/usr/local/lib/python3.5/dist-packages/rcpy/\_\_init\_\_.py:135: UserWarning: > Installing signal handlers  
warnings.warn('> Installing signal handlers')  
WARNING: no gyro calibration data found  
Please run rc\_calibrate\_gyro  
  
try 'python rcpy\_test\_imu -h' to see other options  
Press Ctrl-C to exit  
Accel XYZ (m/s^2) | Gyro XYZ (deg/s) | Temp. (C)  
0.61 0.23 10.34 | -1.1 -0.4 -0.1 | 37.0

*Special thanks to Mauricio de Oliveira at UCSD for creating **rcpy** and to Robert Nelson at Digi-Key for helping out with item #6.*