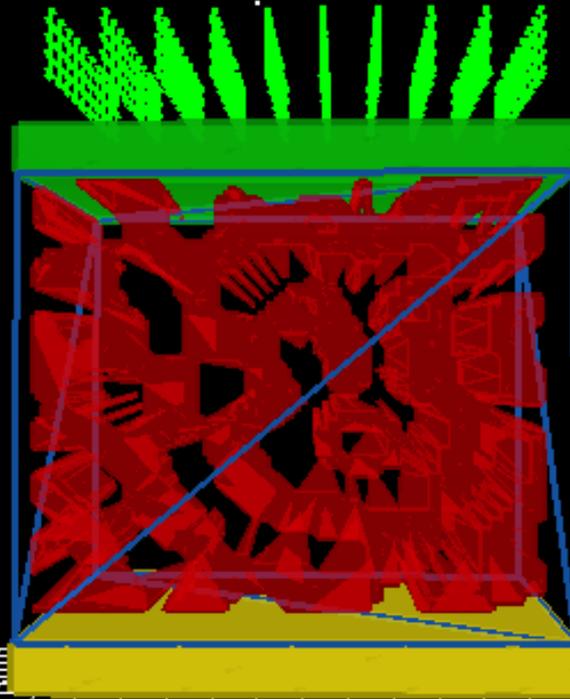
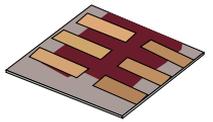


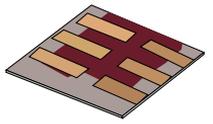
Driving OghmaNano with python



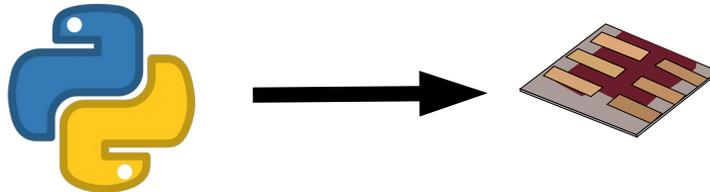


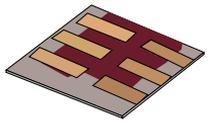
- **Why drive OghmaNano with python?**
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

Why drive OghmaNano using python?



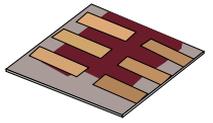
- OghmaNano has a GUI which you can be used to make and run simulations.
- However, as with all GUIs it can never be as flexible as a scripting language such as python.
- This tutorial will show you how to drive OghmaNano simulations using python.



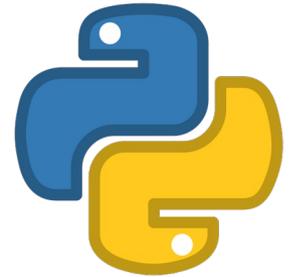


- Why drive OghmaNano with python?
- Installing**
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

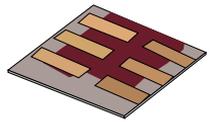
Installing Python



- I am not going to cover installing python as there are plenty of tutorials on the web
 - But I recommend you use a recent version



The two parts of OghmaNano



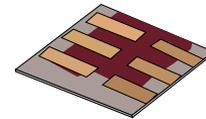
- OghmaNano usually installs to ***c:\Program Files (x86)\OghmaNano***

- If you open that directory using windows explorer you will see that along with the other files there is OghmaNano.exe and Oghma_core.exe

- OghmaNano.exe is the Graphical User Interface (GUI)

- oghma_core.exe is the numerical solver

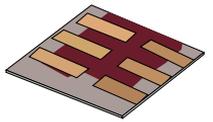
- We will be using oghma_core.exe to run our simulations.



OghmaNano.exe

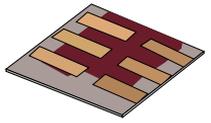


oghma_core.exe



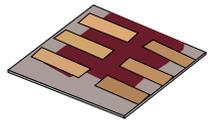
- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path**
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

Adding *c:\Program Files (x86)\OghmaNano* to the windows path

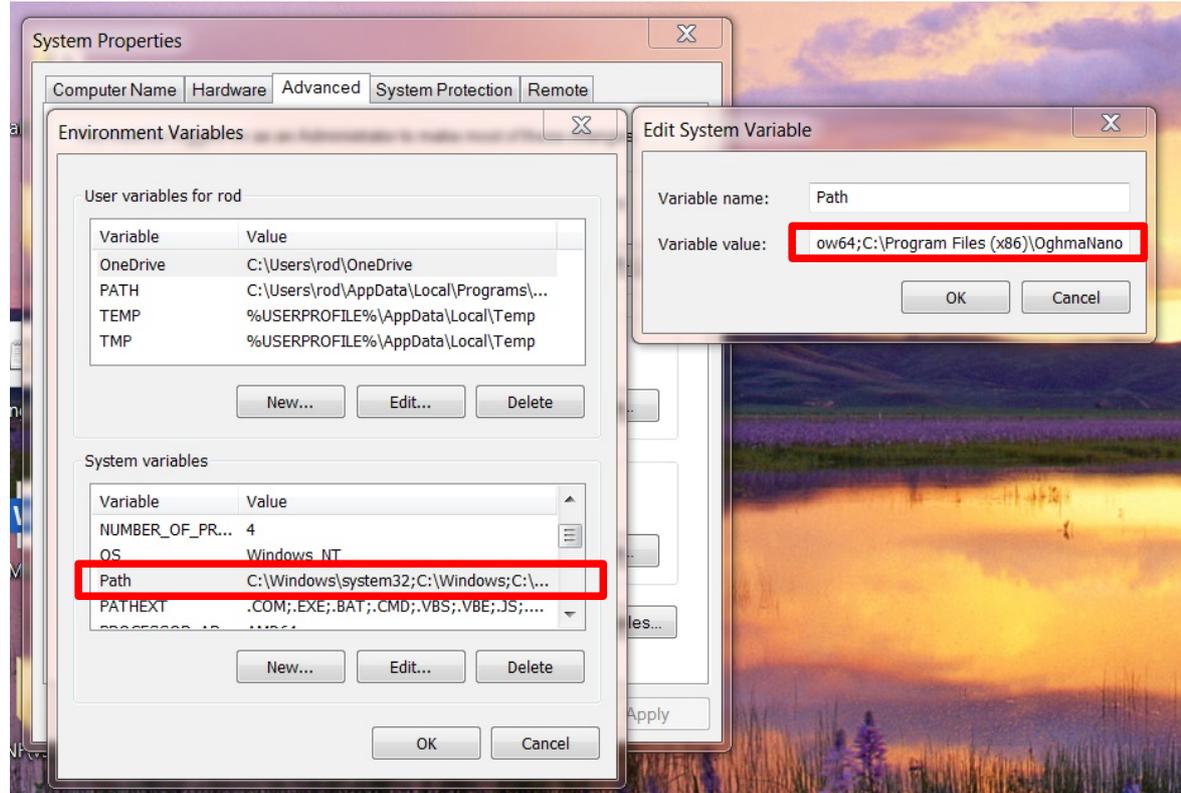


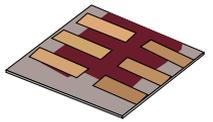
- Before we can run any python simulations we must tell windows where it can find OghmaNano.
- We need to add OghmaNano to the windows path.
- Instructions on how to do this can be found here:
 - [https://docs.microsoft.com/en-us/previous-versions/office/developer/sharepoint-2010/ee537574\(v=office.14\)](https://docs.microsoft.com/en-us/previous-versions/office/developer/sharepoint-2010/ee537574(v=office.14))
- I will show you how to do this for Windows 7
 - All versions of Windows require more or less the same steps to do this, but Microsoft has a habit of renaming things or moving things around between versions.

Adding OghmaNano to the path on windows 7



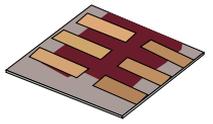
- Open the start menu and type “Environmental variables”
- Then add “;c:\Program Files (x86)\OghmaNano” to the “System Variables”
- Note the ; to separate the paths.



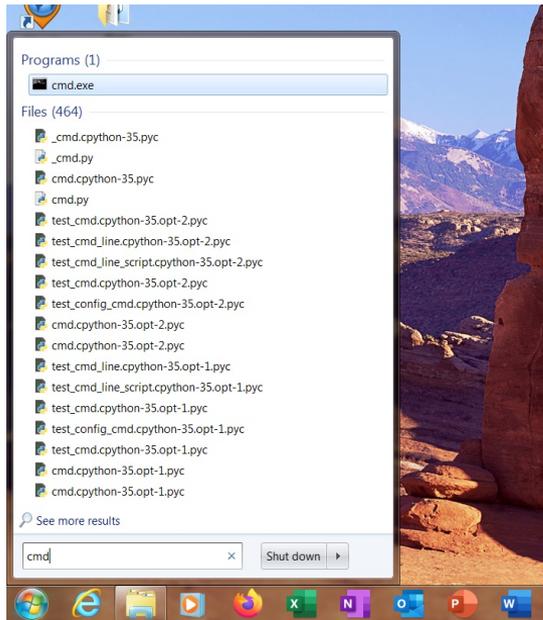


- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - **Testing OghmaNano from the command line**
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

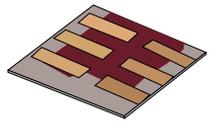
Let's now test that you did that correctly



- We are going to try to run `oghma_nano.exe` from the command line to see if it has been added to the path correctly
- Open the terminal window, by clicking on the start menu and clicking run, then typing 'cmd'. This should bring up the terminal window.



Let's now test that you did that correctly



- Type `oghma_core.exe --help` (note the dash before help is a double dash so x2 '-')
- If you get this (below) or something like it, you have installed OghmaNano correctly if you get an error or something saying `oghma_core` is not found then you have done something wrong. :)

```
C:\Windows\system32\cmd.exe
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\rod>oghma_core.exe --help
user home dir: C:\Users\rod
Share path: C:\Program Files (x86)\OghmaNano
oghma_core - Organic and hybrid Material Nano Simulation tool
Copyright (C) 2009-2022 Roderick C. I. MacKenzie, Releced the MIT License

Usage: oghma_core [options]

Options:

    --sim-root-path Simulation root path
    --version      displays the current version
    --zip_results  zip the results
    --simmode      Forces a simulation mode.
    --cpus         sets the number of CPUs

Additional information about OghmaNano is available at https://www.Oghma-Nano.com.

Report bugs to: roderick.mackenzie@oghma-nano.com

C:\Users\rod>
```

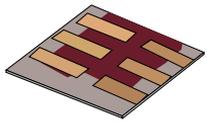
Good/ working :)

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\rod>cd Desktop
C:\Users\rod\Desktop>cd test_sim
C:\Users\rod\Desktop\test_sim>oghma_nano.exe
'oghma_nano.exe' is not recognized as an internal or external command,
operable program or batch file.

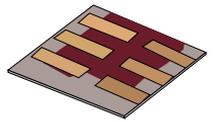
C:\Users\rod\Desktop\test_sim>
```

Bad/ not working :(

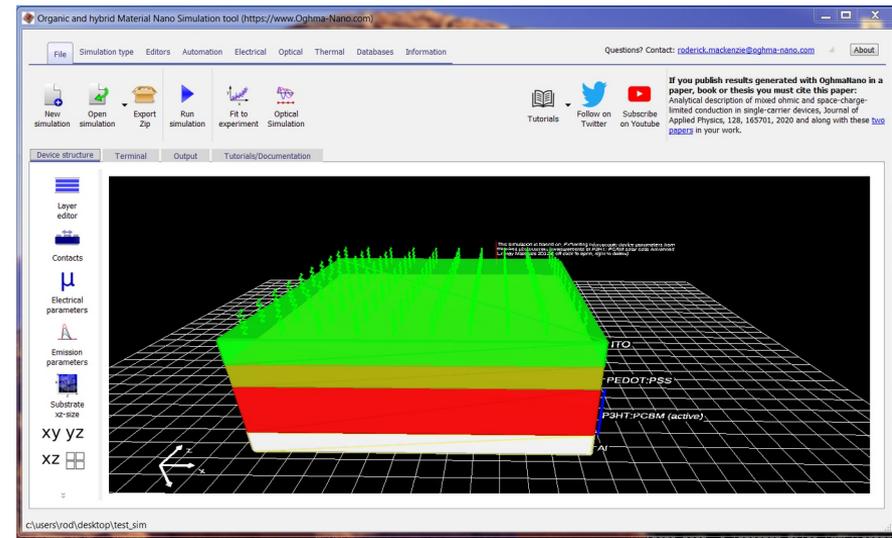
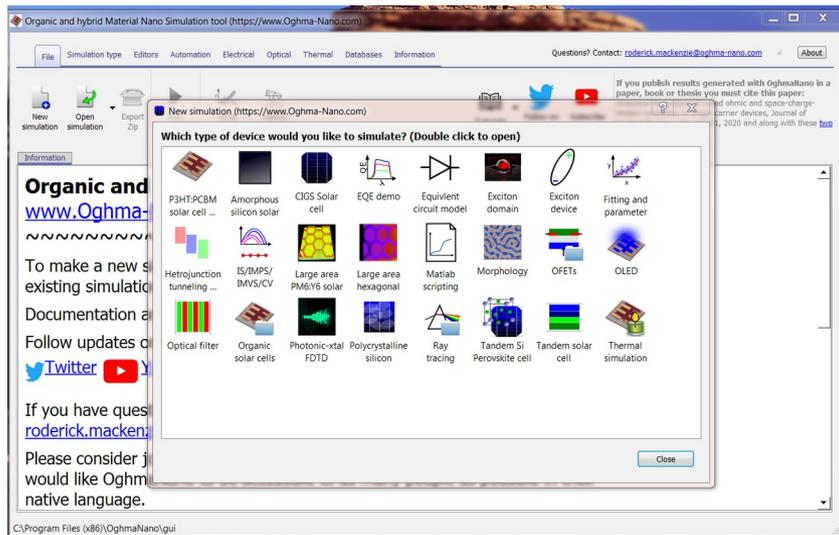


- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line**
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

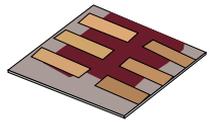
Running a real simulation from the command line



- We are now going to test that oghma_core.exe can be run with out the GUI.
- Firstly using the GUI make a new simulation and save it to your desktop. I recommend you use the P3HT:PCBM simulation for this demo.
- Check it runs by pressing the play button.



Running a real simulation from the command line



- Then in the command window navigate to this new simulation window and type oghma_core.exe
- The simulation should run from the terminal, if it does not check your path.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

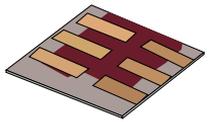
C:\Users\rod>cd Desktop

C:\Users\rod\Desktop>cd test_sim

C:\Users\rod\Desktop\test_sim>oghma_core.exe_
```

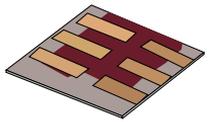
```
C:\Windows\system32\cmd.exe
top= 0.62 U, 3.79e+001 A/m^2, btm= 0.00 U, 3.79e+001 A/m^2 f(=)3.015771e-011 1
5.600098 ms
top= 0.64 U, 1.09e+002 A/m^2, btm= 0.00 U, 1.09e+002 A/m^2 f(=)2.265116e-011 0
.000000 ms
top= 0.66 U, 2.13e+002 A/m^2, btm= 0.00 U, 2.13e+002 A/m^2 f(=)2.104100e-011 0
.000000 ms
top= 0.68 U, 3.60e+002 A/m^2, btm= 0.00 U, 3.60e+002 A/m^2 f(=)2.369204e-011 0
.000000 ms
top= 0.70 U, 5.65e+002 A/m^2, btm= 0.00 U, 5.65e+002 A/m^2 f(=)3.046702e-011 1
5.599854 ms
top= 0.72 U, 8.44e+002 A/m^2, btm= 0.00 U, 8.45e+002 A/m^2 f(=)4.172242e-011 1
5.600098 ms
top= 0.74 U, 1.22e+003 A/m^2, btm= 0.00 U, 1.22e+003 A/m^2 f(=)6.144737e-011 0
.000000 ms
top= 0.76 U, 1.70e+003 A/m^2, btm= 0.00 U, 1.70e+003 A/m^2 f(=)1.444173e-010 0
.000000 ms
top= 0.78 U, 2.33e+003 A/m^2, btm= 0.00 U, 2.33e+003 A/m^2 f(=)2.338506e-010 0
.000000 ms
top= 0.80 U, 3.11e+003 A/m^2, btm= 0.00 U, 3.11e+003 A/m^2 f(=)2.006885e-010 1
5.600098 ms
Stopping because of Uexternal 1.163852e+000>1.100000e+000
Solved 9245 Equations
unload DLLs

C:\Users\rod\Desktop\test_sim>
```



- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file**
- The OghmaNano file format
- Editing/reading/writing in python

The OghmaNano file format

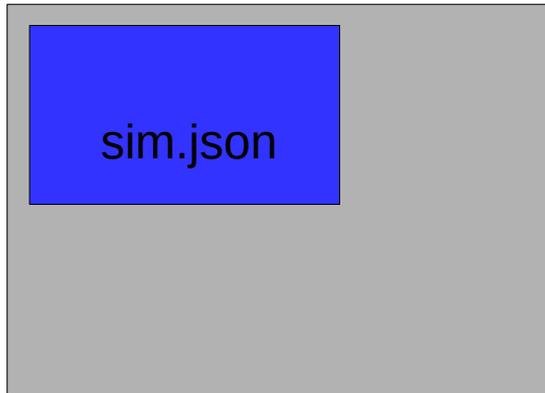


- Before we can drive OghmaNano with python we need to understand the file format.
- The .oghma file format is just a **zip file** but with a different name. If you rename it to .zip you can look inside it. Inside the zip file there is the sim.json file which is the main simulation file.

- OghmaNano does not care if the sim.json file is inside the sim.oghma file or outside it.

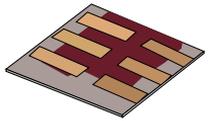
- It is easier to edit it in python if it is outside the sim.oghma file.

- We are now going to use the GUI to extract the sim.json file.



sim.oghma

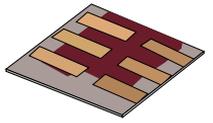
Extracting the sim.json file from the sim.oghma file



- Open your simulation you made with the GUI

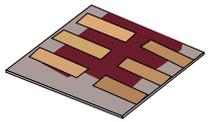
- Navigate backwards one step and right click on the Simulation and click extract

Extracting the sim.json file from the sim.oghma file



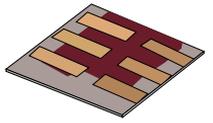
- In windows explorer you will now see the sim.json file

- Try opening it with notepad:



- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python

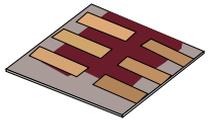
The json file format



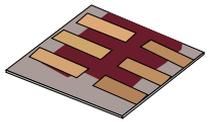
- All files in OghmaNano are stored in .json format. This consists of brackets and quote marks. This is a standard web format.

- Try opening the file in firefox, firefox has a nice json reader. Try exploring the structure.

Comparing the GUI and the json file

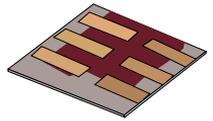


- You can see that the same information in the GUI is stored in the json file.



- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python**

Editing the file in python using python's standard json reader /writer.



```
import json
import os
import sys

#open the sim.json file
f=open('sim.json')
lines=f.readlines()
f.close()

lines="".join(lines)

#convert the text to a python json object
data = json.loads(lines)

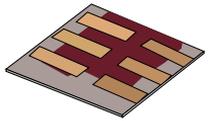
#Edit a value (use firefox as a json viewer
# to help you figure out which value to edit)
# this time we are editing the mobility of layer 1
data['epitaxy']['layer1']['shape_dos']['mue_y']=1.0

#convert the json object back to a string
jstr = json.dumps(data, sort_keys=False)

#write it back to disk
f=open('sim.json', "w")
f.write(jstr)
f.close()

#run the simulation using oghma_core
os.system("oghma_core.exe")
```

- This script will edit the mobility of layer1 to be 1.0 and then run oghma_core.exe using the os.system command.



- Why drive OghmaNano with python?
- Installing
 - The two parts of OghmaNano
 - Adding OghmaNano to the window's path
 - Testing OghmaNano from the command line
- Making a new simulation and testing OghmaNano from the command line
- Extracting the sim.json file
- The OghmaNano file format
- Editing/reading/writing in python