



**Matanuska-Susitna College
CST Program DOCUMENT**

**Open EI Wind for Schools Design
Procedural Document**

Summary

The following instructions are designed to aid organizations participating in Open Energy Information's Wind for Schools community program. Its intent is to instruct on the configuration and deployment a Raspberry Pi to communicate with one or more SkyStream wind turbine. **It is recommended that this document is read in its entirety if you are deploying a project of similar scope.**

Revision History

Date	Version	Author	Description
03.30.15	<u>1.0</u>	<u>Jacob Straub</u>	Created Original Document
4.04.15	<u>1.1</u>	<u>Jacob Straub</u>	Updated Task Environment
4.12.15	<u>2.0</u>	<u>Jacob Straub</u>	Created Individual Instructions
4.13.15	<u>2.1</u>	<u>Jacob Straub</u>	Omitted Sensitive Credentials

Special Instructions:

These instructions exist specifically for the documentation of the University of Alaska Anchorage – Matanuska-Susitna's application of Open EI's Wind for School program as there are minute changes in the configurations of deployed hardware. For instructions supported by Open EI please refer to http://en.openei.org/wiki/Turbine_Support and contact Mark Jacobson at mark.jacobson@nrel.gov. The Open EI website directs to software, contact information, and instructional support. Organizations that wish to follow the procedure as documented by Open EI may go to <http://goo.gl/GOXDS4>.

Contents

Introduction.....	3
Task Conditions	3
Environment.....	3
Required Hardware	3
Required Software	3
Initial Setup.....	4
Writing image file to SD Card.....	4
Connecting to your Device	5
Setting Password.....	5
Disabling VNC.....	5
Enabling SSH.....	5
Configuring check_ip.py.....	6
Connecting Remotely.....	6
Creating Startup Items	8
Automating Startup Scripts.....	9
Configuring WindInterface.....	10
Searching for turbines	10
Editing windinterface.xml.....	10
Setup Open EI Account	10
Verifying WindInterface Configuration	11
Navigate to WindInterface.....	11
Troubleshooting	11
Enabling GUI.....	11
Enabling VNC.....	11
Acknowledgements.....	12
Appendix.....	12

Introduction

The following procedure is intended for organizations that wish to participate in Open EI's Wind for Schools community program utilizing a Raspberry Pi with a SkyStream 3.5. This procedure assumes you already have deployed a SkyStream wind turbine on site. This process has been tested on Raspberry Pi B and B+ models.

Non-UAA affiliated organizations shall refer to official Open EI documentation for design and deployment. This documentation can be found at <http://goo.gl/iRzbwJ>.

Task Conditions

Environment

The XBEE (pronounced ZigBee) radio has a max range of 300 feet. Distance may be reduced due to interference caused by material and cross channel communication. It is recommended to deploy the Raspberry Pi and ZigBee within view for the turbine. Neither the Raspberry Pi nor XBEE is inherently waterproof, avoid deploying outside or create an enclosure to displace water and moisture from the device. It is not recommended to deploy the appliance utilizing a Wi-Fi dongle this may cause issues with the Wind Interface java applet.

Required Hardware

1. Raspberry Pi 1 B (Revision 2.0) or B+ (Revision 2.1)
 - a) A Raspberry Pi computer may be obtained through OpenEI. Contact Ruth Millar, rdmiller@ksu.edu, for further details.
2. XBEE Pro2 2.4GHz wireless radio
 - a) A XBEE Pro2 may be obtained through OpenEI. Contact Ruth Millar, rdmiller@ksu.edu, for further details.
3. USB/Bluetooth Keyboard
4. USB/Bluetooth Mouse (Optional)
5. HDMI Cable
6. HDMI female to DVI male adaptor (Optional)
7. Monitor or other display with HDMI or DVI you have an adapter
8. SD card reader

Required Software

- **Raspberry Pi Image**
 - Image by Kansas University for Open EI - <http://goo.gl/C7YTUz>
 - Matanuska-Susitna College - <http://goo.gl/aGIKwv>
- **Imaging Software**
 - **Imaging Tool: USBit** - <http://www.alexpage.de/usb-image-tool/>
 - **Remote Management Software: Putty** - <http://goo.gl/lFZgnw>

Initial Setup

Writing image file to SD Card

Download image from your respective source detailed in Required Software.

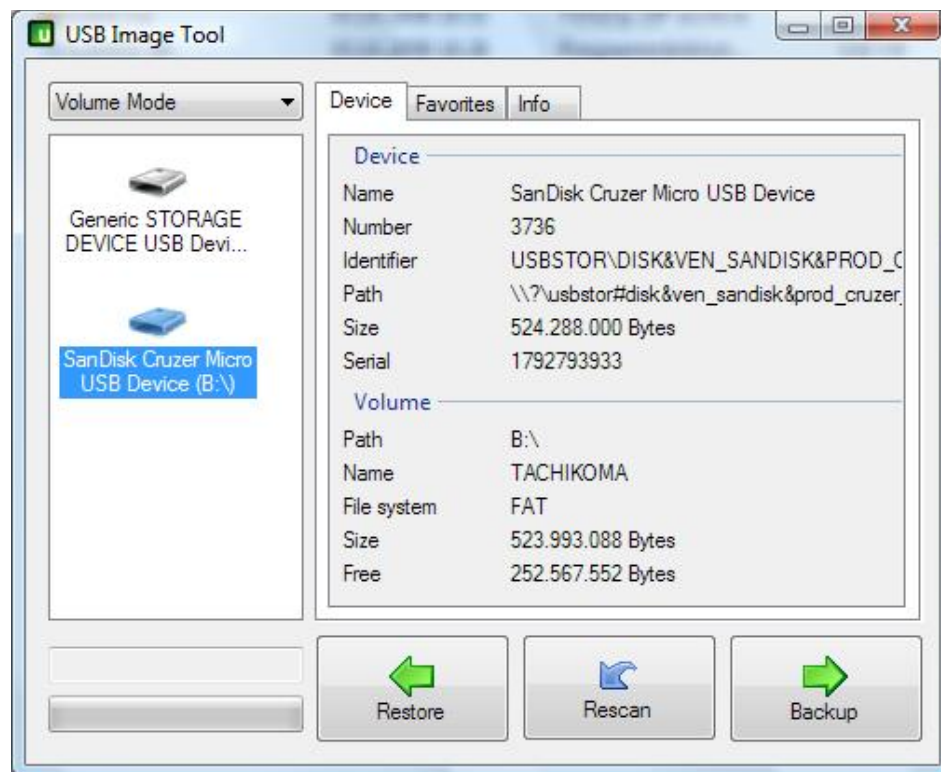
Write the image to the SD card

USB Image Tool can create images of USB flash drives that are mounted as USB drives. It allows you to switch between images or to make an exact backup image of your USB Stick. USB Image Tool works with any device that implements the USB Mass Storage protocol. This includes flash drives, card readers, cell phones, and mobile music players.

1. If applicable, unzip the image file from the compressed directory/folder.
2. Insert SD card into a card reader
3. Download and launch USBit
4. Select the correct drive letter from the left most panel in USBBit
 - a. Make sure that the device selected matches the SD card's location (e.g F:/ or E:/)

NOTICE: ENSURE SELECTION OF CORRECT DRIVE TO AVOID DATA LOSS!

5. If applicable, select 'Backup' to create an image file of data currently stored on the SD device.
6. Select a location to save the image for later retrieval.
7. Select 'Restore' to write the image file to the SD card.
8. Browse to the location of the downloaded image and select it.



Connecting to your Device

1. Insert SD card into the push-push slot located on the bottom of the Raspberry Pi.
2. Insert USB keyboard/Mouse into USB port(s).
3. Insert HDMI cable into HDMI port.
4. Plug in your Raspberry Pi to the network using an Ethernet cable. (Optional)
5. Connect to the Raspberry Pi EITHER with monitor (HDMI cable), mouse and keyboard (both USB) OR thru remote Connection over Network.

NOTICE: Although mouse and keyboard can be used for initial setup, the remote connection is pre-installed and highly recommended for access and configuration of the Raspberry Pi.

Setting Password

1. Enter 'sudo raspi-config'
2. Using arrow keys, navigate to 'Change User Password'
3. Enter your new password. Be sure to document
4. Using tab to navigate to 'Finish' and select

Disabling VNC

Depending on your organization's security requirements, VNC may be considered an insecure service.

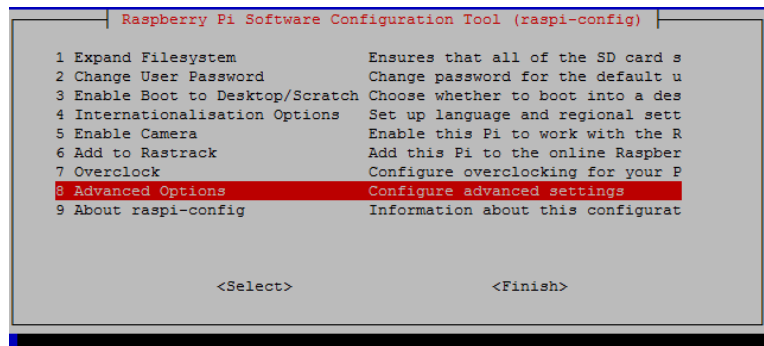
NOTICE: UAA MSC's image has disabled VNC. To enable VNC, refer to Raspberry Pi foundation's procedural document: <http://goo.gl/cAXXZy>

1. Open a new terminal (CTRL+ALT+T or CMD/WinLogo+R and enter 'lxterminal')
2. Enter 'sudo apt-get remove x11 && sudo apt-get autoremove'
3. A reboot will be required.

Enabling SSH

Secure Shell (SSH) allows a user to run commands on a machine's command prompt without them being physically present near the machine. It also allows a user to establish a secure channel over an insecure network

1. Enter "sudo raspi-config" in the terminal
2. Navigate to "Advanced Options" through the arrow keys and hit enter
3. Navigate to 'ssh' and hit enter
4. Select "Enable".



Configuring check_ip.py

Check_ip.py is a python script used to send the IP address and configuration of Wind Interface to a specified set of email recipients. The following instruction set will configure check_ip.py to add additional email recipients. It is a well-commented configuration file and navigation should be a breeze.

1. Enter 'sudo nano check_ip.py'
2. Line [#] contains the list of email recipients
3. Using the format of '{email@address.TLD}', enter desired email addresses.
4. Be sure to not remove the 'azwind4schools@gmail.com' address.
5. CTRL+X and 'Y' to exit and retain changes.

```
#      FILL OUT THE VALUES BETWEEN THE COMMENT BLOCKS
#####
# Specify the sender credentials and recipient email address here
SENDER = {}
SENDER['addr'] = 'azwind4schools@gmail.com'
SENDER['pass'] = 'windenergy!'
SENDER['serv'] = 'smtp.gmail.com'
SENDER['port'] = 465 # default GMAIL SMTP port

RECEIVER = {}
RECEIVER['addr'] = [ 'azwind4schools@gmail.com', 'email@domain.com' ]:
# Note: RECEIVER['addr'] must be a list (ie don't delete the brackets)!
# If you would like to add your own email address, just type between the second set of quotes
# This allows the specification of a list of addresses ['me@addr.com', 'you@addr.com', 'them@addr.com']

# specify the interface to query
INTERFACE = 'eth0' # default interface for a Raspberry Pi

# specify the name of the computer to display in the email
HOSTNAME = Popen(['hostname'], stdout=PIPE).communicate()[0].replace('\n', '')

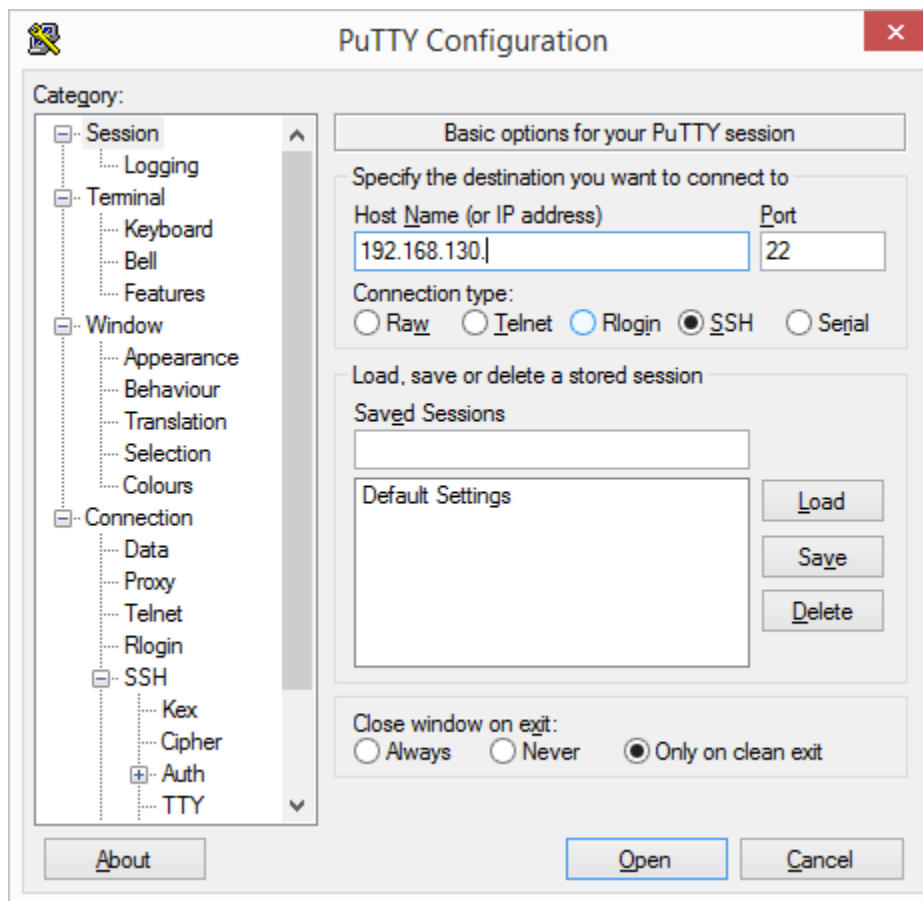
#####
# Execute the call to ifconfig
```

Connecting Remotely

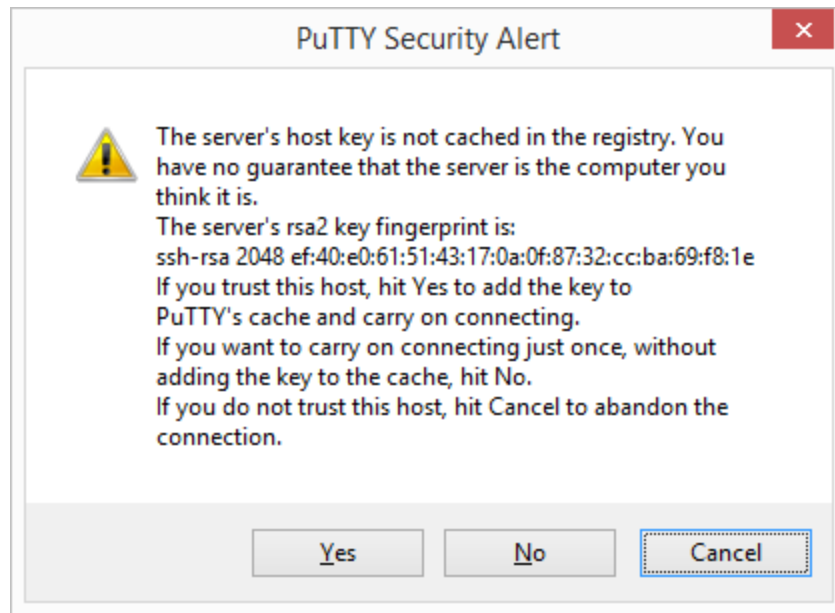
Depending on the location of your deployed Raspberry Pi, local administration through conventional USB keyboard and mouse may not be possible. If you have opted to enable SSH, the following instruction set will allow you to connect to the Raspberry Pi remotely. Connecting will require an IP address, be sure to document the IP address through the email sent by the device or through running the 'ifconfig eth0' or 'ifconfig wlan0' from the command terminal.

1. Enter your connection settings:

- a. Host Name: pi@example.com or a 'pi@[IP address]
- b. Port: 22 (leave as default)
- c. Connection Type: SSH (leave as default)



2. Click Open to start the SSH session.
 - a. If this is your first time connecting to the server from this computer, you will see the following output. Accept the connection by clicking 'Yes'.



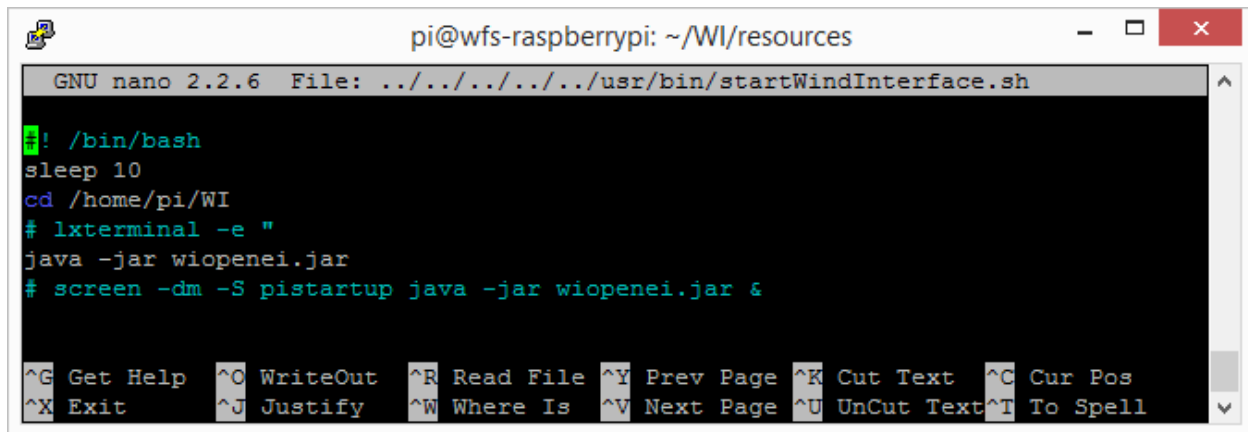
3. SSH Connection will open, you should see a terminal prompt asking for your username:
4. Enter 'pi'
5. Enter your password.
 - a. Please note that you will NOT see your cursor moving. Hit enter.
6. You are now logged in with SSH. You may begin typing commands at the prompt.

Creating Startup Items

Open EI's image has startup items configured to launch with 'x server', a Linux windows system display server. We find this to be an inadequate method and have moved startup items to launch with the system itself through the rc.local file. This removes the need for a 'graphical user interface' GUI unless preferred to apply configuration changes.

NOTICE: If you are using the image provided by UAA-MSC then the following automation steps have already been completed

1. Open a new terminal (CTRL+ALT+T or CMD/WinLogo+R and enter 'lxterminal')
2. Enter 'sudo nano /usr/bin/startWindInterface.sh'
3. Comment out line 3 and 4 by appending a '#' to the beginning
4. In a new line enter "java -jar wiopenei.jar" retain changes



```

pi@wfs-raspberrypi: ~/WI/resources
GNU nano 2.2.6 File: ../../../../../../usr/bin/startWindInterface.sh
#!/bin/bash
sleep 10
cd /home/pi/WI
# lxterminal -e "
java -jar wiopenei.jar
# screen -dm -S pistartup java -jar wiopenei.jar &

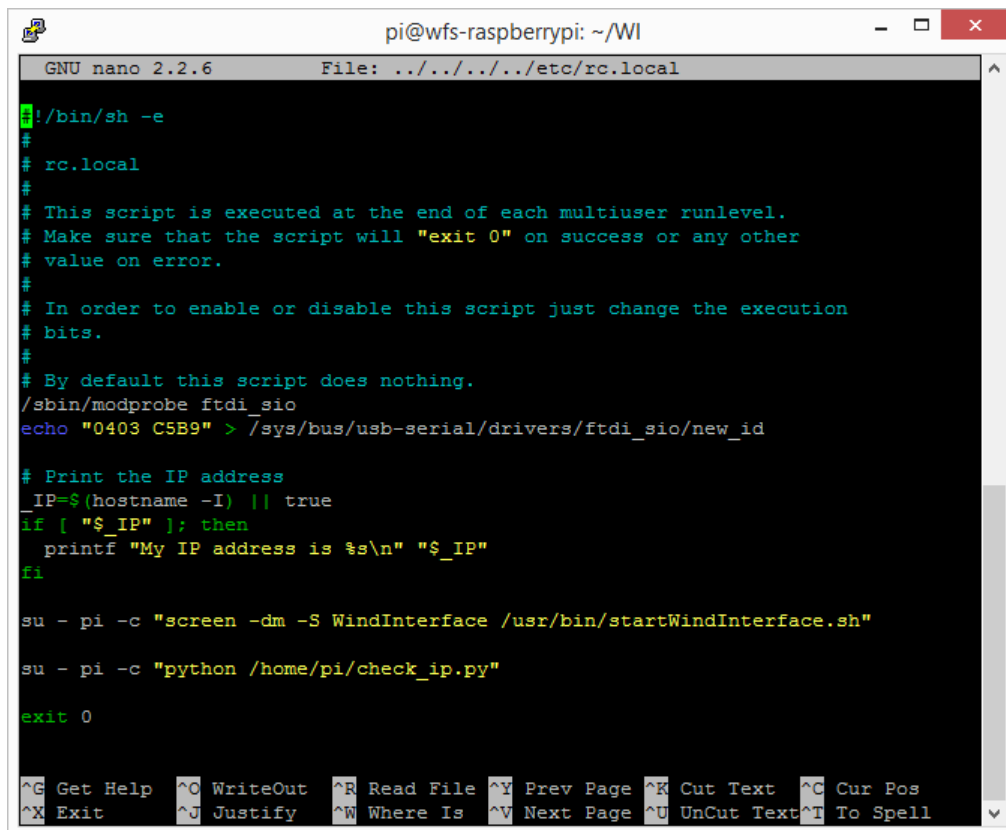
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell

```

Automating Startup Scripts

In order to have a command or program run when the Pi boots, you can add commands to the rc.local file. This is especially useful if you want to be able to plug your Pi into power headless and have it run a program without configuration or a manual start.

1. Open a new terminal
2. Enter 'sudo nano /etc/rc.local'
3. Enter 'su - pi -c "screen -dm -S WindInterface /usr/bin/startWind Interface.sh"'
4. Enter 'su - pi -c "python /home/pi/check_ip.py"'
5. CTRL+X and enter 'Y' to retain changes.



```

pi@wfs-raspberrypi: ~/WI
GNU nano 2.2.6 File: ../../../../../../etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
/sbin/modprobe ftdi_sio
echo "0403 C5B9" > /sys/bus/usb-serial/drivers/ftdi_sio/new_id

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
    printf "My IP address is %s\n" "$_IP"
fi

su - pi -c "screen -dm -S WindInterface /usr/bin/startWindInterface.sh"

su - pi -c "python /home/pi/check_ip.py"

exit 0

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell

```

Configuring WindInterface

Searching for turbines

1. Access Terminal
2. Enter “cd resources”
3. Enter “./s2zcmd zls’ the leading period is important do not omit.

```

pi@wfs-raspberrypi ~/WI/resources $ ls
s2zcmd  skzcmd.exe  windinterfacepref.xml
pi@wfs-raspberrypi ~/WI/resources $ ./s2zcmd zls
# 0013a200-4062139a (3146) 'SKY37A_2010-1339' 0xffffffff
+ found 1
;other 0

```

Editing windinterface.xml

The XML document, windinterface.xml, contains the configuration settings used by University of Kansas’ WindInterface java applet. Here, you will enter

Setup Open EI Account

To report your turbine data for new or newly configured systems to OpenEI, you will need to register your turbine ID with Jay Huggins, the OpenEI administrator at the National Renewable Energy Laboratory. Once you have your turbine ID in hand email jay and request to be registered and to receive an API key. Once the turbine is registered and configured with the API key, successful submissions to the API uploader will receive an HTTP response code of 200 and a JSON status object. Jay may be reached at jay.huggins@nrel.gov

Once you're logged into OpenEI, enter a name for the school's project. Note that the name you enter here will become the wiki page name. You will enter something like "My School Wind Project" (without quotes), which will create http://en.openei.org/wiki/My_School_Wind_Project.

After you enter the school wind facility name and submit, you will see a form for entering detailed information.

- Facility: If the page name is My School Wind Project, enter "My School"
- Sector: Select Wind Energy from the dropdown menu.
- Facility Type: Enter Community Wind.
- A number of Generating Units: Enter a number.
- Wind Turbine Manufacturer: Enter the manufacturer's name if you know it.
- Wind for Schools Portal Turbine ID: Enter the ID.
- Location: Enter the two-letter state abbreviation.
- A number of units: Enter the number of turbines at this location.

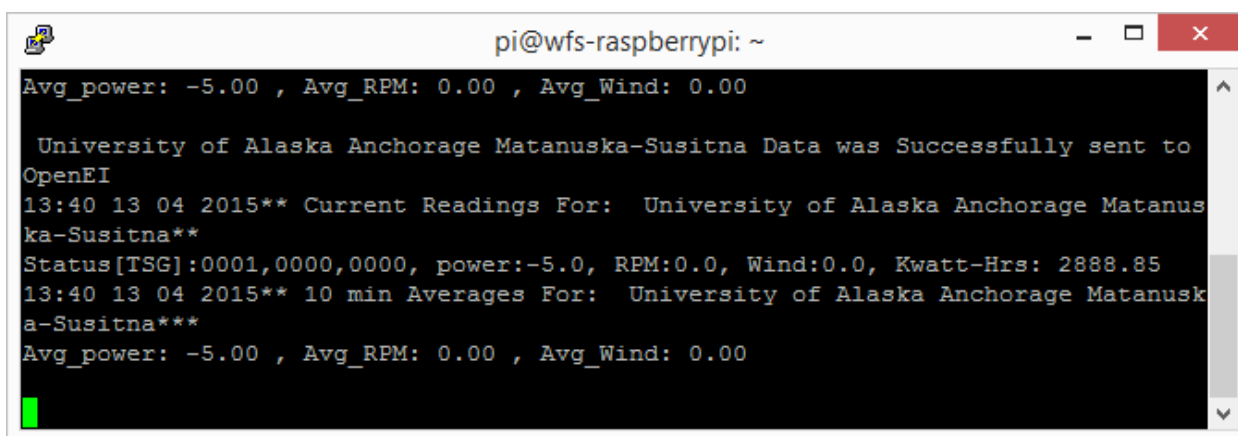
- Coordinates: Enter the coordinates; OpenEI provides a Look Up Coordinates field below. Simply enter the school address.

Within 24 hours, if the above instruction sets were followed, facility name and a link to the facility wiki page will appear on Open EI's Wind for Schools' google map overlay, located at http://en.openei.org/wiki/Wind_for_Schools_Portal.

Verifying WindInterface Configuration

Navigate to WindInterface

1. Enter 'screen -x WindInterface'
2. The following Prompt should appear with new values appearing every 30 seconds.



```

pi@wfs-raspberrypi: ~
Avg_power: -5.00 , Avg_RPM: 0.00 , Avg_Wind: 0.00

University of Alaska Anchorage Matanuska-Susitna Data was Successfully sent to
OpenEI
13:40 13 04 2015** Current Readings For: University of Alaska Anchorage Matanus
ka-Susitna**
Status[TSG]:0001,0000,0000, power:-5.0, RPM:0.0, Wind:0.0, Kwatt-Hrs: 2888.85
13:40 13 04 2015** 10 min Averages For: University of Alaska Anchorage Matansk
a-Susitna***
Avg_power: -5.00 , Avg_RPM: 0.00 , Avg_Wind: 0.00

```

Troubleshooting

Enabling GUI

The image used by UAA has VNC disabled. To enable VNC enter the following commands:

- Open a new terminal
- Install X11vnc (Sudo apt-get install x11vnc)
- Direct X11 to use the user's password
 - X11vnc -usepw
 - Add the following to /etc/rc.local to automate X11 on system start


```
su - pi -c "/usr/bin/x11vnc -xkb -forever -auth /var/run/lightdm/root/:0
-display :0 -rfbauth /etc/x11vnc.pass -rfbport 5900 -bg -o
/var/log/x11vnc.log"
```

Enabling VNC

- To enable VNC, refer to the Raspberry Pi foundation's documentation: <https://www.raspberrypi.org/documentation/remote-access/vnc/README.md>

Acknowledgments

This project, and subsequently this document, would not be possible without the efforts and expertise of the Raspberry Pi Foundation and the individuals involved with Open EI's Wind for a Schools community project.

The windinterface.xml instruction sets were retrieved from Kansas State University's "Setup Guide for FsS Raspberry Pi" which can be found at <http://goo.gl/BQXM23>.

Appendix

What is an Image?

In computing, a system image is a copy of the entire state of a computer system stored in some non-volatile form such as a file. A system is said to be capable of using system images if it can be shut down and later restored to exactly the same state. In such cases, system images can be used for backup.

What is Raspberry Pi?

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing and to learn how to program in languages like Scratch and Python.

What is an XBee Pro?

XBees is a family of 2.4 GHz compatible radio modules. The XBee Pro is XBee module which uses Digi Mesh, a sleeping mesh networking protocol developed by Digi International.

What is Sky Stream?

The Sky Stream turbine was obtained through grants and collaboration with OpenEI's Wind for Schools program. Sky Stream is a wind-powered electricity generator containing an integral AC power inverter. It is designed to supplement the electrical power provided by the local electrical utility company in residential applications by connecting directly to the main AC utility panel. Sky Stream may also be utilized to provide power with battery-based residential electrical systems or utility grid-connected systems with battery backup.

What is the application programming interface (API)?

An application programming interface (API) is a set of programming instructions and standards for accessing a Web-based software application or Web tool. In layman's terms, an API is an agreement between two people stating: "If you submit this instruction, I will perform this action, or return this information."