

NetworkZero

Easy Network Discovery & Messaging





g IDE
Debug Tools Window Help

py
Unindent

```
event in pygame.event.get():  
    if event.type == pygame.JOYBUTTONDOWN:  
        do MY Game  
    if event.type == pygame.JOYAXISMOTION:  
        print("Number of joysticks: 1")  
        print("Joystick 0")  
        print("Joystick name: usb gamepad")  
        print("Number of axes: 2")  
        print("Axis 0 values -0.007843017578125")  
        print("Axis 1 values -0.007843017578125")
```

screen
extPr
extPr

joysti
extPr
extPr

for i
jo
jo

te
te
na
te

av
sta

Whole s
next



Setting up a Raspberry Pi as a WiFi access point

Make the most adorable little WiFi router

[Overview](#)

[What you'll need](#)

[Preparation](#)

[Check Ethernet & Wifi](#)

[Install software](#)

[Connect and Test](#)

[Compiling hostapd](#)

[Multiple Pages](#)

[Download PDF](#)

[Contributors](#)

[lady ada](#)

[Feedback? Corrections?](#)

and remove the # so it says

[Download file](#)

[Copy Code](#)

```
1. # If this DHCP server is the official DHCP server for the local
2. # network, the authoritative directive should be uncommented.
3. authoritative;
```

```
GNU nano 2.2.6 File: /etc/dhcp/dhcpd.conf

# option definitions common to all supported networks...
#option domain-name "example.org";
#option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 600;
max-lease-time 7200;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.

[ line 21/118 (17%), col 1/15 (6%), char 654/3762 (17%) ]
^G Get Help ^C WriteOut ^R Read File ^V Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^N Next Page ^U UnCut Text ^T To Spell
```

Then scroll down to the bottom and add the following lines

[Download file](#)

[Copy Code](#)

```
1. subnet 192.168.42.0 netmask 255.255.255.0 {
2.     range 192.168.42.10 192.168.42.50;
3.     option broadcast-address 192.168.42.255;
4.     option routers 192.168.42.1;
5.     default-lease-time 600;
6.     max-lease-time 7200;
7.     option domain-name "local";
8.     option domain-name-servers 8.8.8.8, 8.8.4.4;
9. }
```

```
COM53 - PuTTY
GNU nano 2.2.6 File: /etc/dhcp/dhcpd.conf Modified

# range 10.17.224.10 10.17.224.250;
# }
# pool {
#     deny members of "foo";
#     range 10.0.29.10 10.0.29.230;
# }
#)

subnet 192.168.42.0 netmask 255.255.255.0 {
    range 192.168.42.10 192.168.42.50;
    option broadcast-address 192.168.42.255;
    option routers 192.168.42.1;
    default-lease-time 600;
    max-lease-time 7200;
    option domain-name "local";
    option domain-name-servers 8.8.8.8, 8.8.4.4;
}

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

Save the file by typing in **Control-X** then **Y** then **return**

```
Administrator: C:\Windows\SYSTEM32\cmd.exe - python
ed=py.exe c:\tools\ted.py $*
iv="%ProgramFiles(x86)%\IrfanView\i_view32.exe" $*
pip2=py.exe -2 -mpip $*
pip3=py.exe -3 -mpip $*
venv2=pushd %VENVS% && py -2 -mvirtualenv $* && popd
venv3=pushd %VENVS% && py -3 -mvirtualenv $* && popd
vls=dir /b/ad/on %VENVS%
voff=deactivate.bat && cls
von=pushd %VENVS% && $1\scripts\activate.bat && popd
vrm=pushd %VENVS% && rmdir /s $1 && popd

c:\work-in-progress>von nw0

(nw0) c:\work-in-progress>python
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import networkzero as nw0
>>>
```

```
config.ini * SciTE
File Edit Search View Tools Options Language Buffers Help
1 notes.txt 2 config.ini *
1 - [setup]
2 ip_address=192.168.0.2
3 port=12345
```

```
Administrator: C:\Windows\SYSTEM32\cmd.exe - python
ed=py.exe c:\tools\ted.py $*
iv="%ProgramFiles(x86)%\IrfanView\i_view32.exe" $*
2=py.exe -2 -mpip $*
3=py.exe -3 -mpip $*
v2=pushd %VENVS% && py -2 -mvirtualenv $* && popd
v3=pushd %VENVS% && py -3 -mvirtualenv $* && popd
=dir /b/ad/on %VENVS%
f=deactivate.bat && cls
=pushd %VENVS% && $1\scripts\activate.bat && popd
=pushd %VENVS% && rmdir /s $1 && popd

work-in-progress>von nw0

0) c:\work-in-progress>python
hon 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit
MD64)] on win32
e "help", "copyright", "credits" or "license" for more information.
import networkzero as nw0
```

```

class Robot(object):

    def __init__(
        self,
        output,
        stop_event=None,
        listen_on_ip=config.LISTEN_ON_IP, listen_on_port=config.LISTEN_ON_PORT
    ):
        log.info("Setting up Robot on %s:%s", listen_on_ip, listen_on_port)
        log.info("Outputting to %s", output)
        self.stop_event = stop_event or threading.Event()
        self._init_socket(listen_on_ip, listen_on_port)
        self.output = output
        self.output._init()

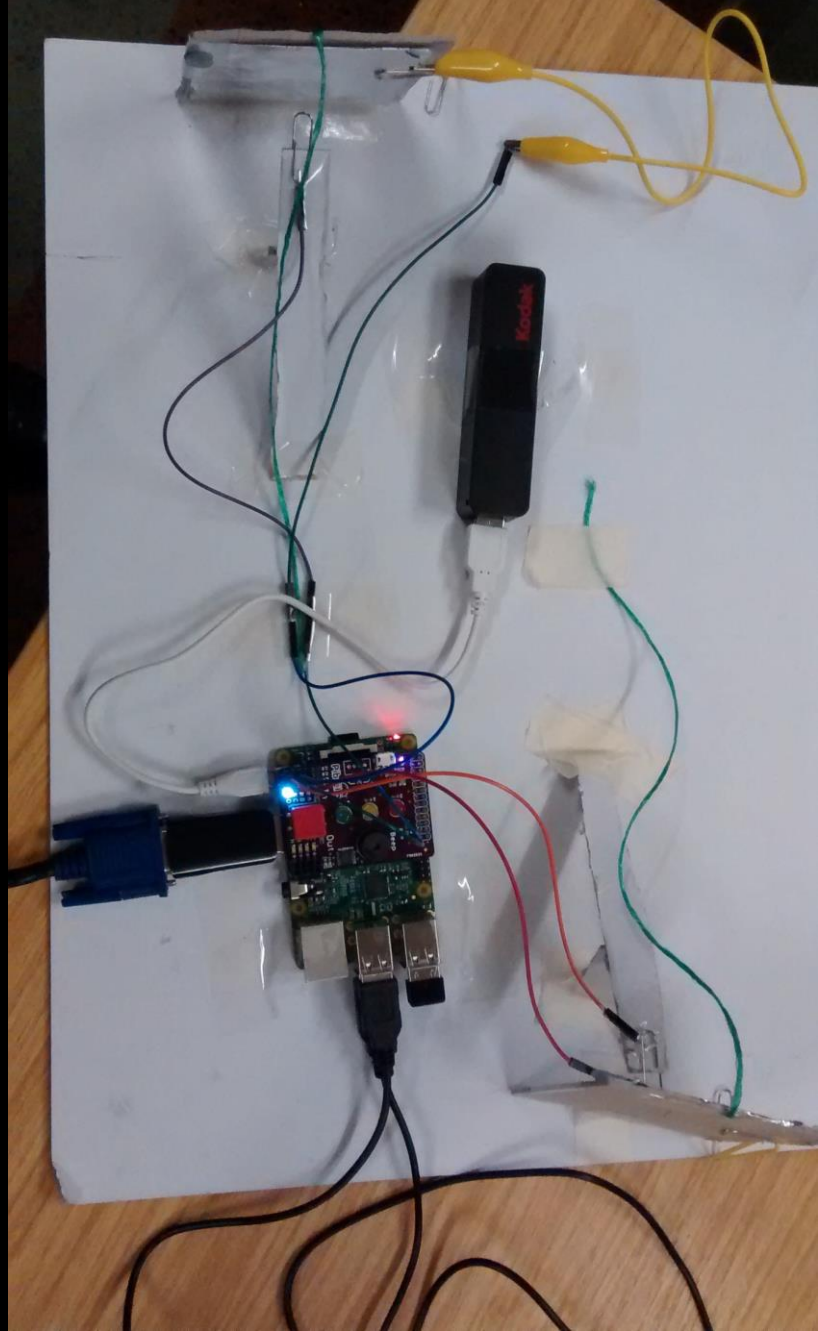
    def _init_socket(self, listen_on_ip, listen_on_port):
        context = zmq.Context()
        self.socket = context.socket(zmq.REP)
        self.socket.bind("tcp://%s:%s" % (listen_on_ip, listen_on_port))

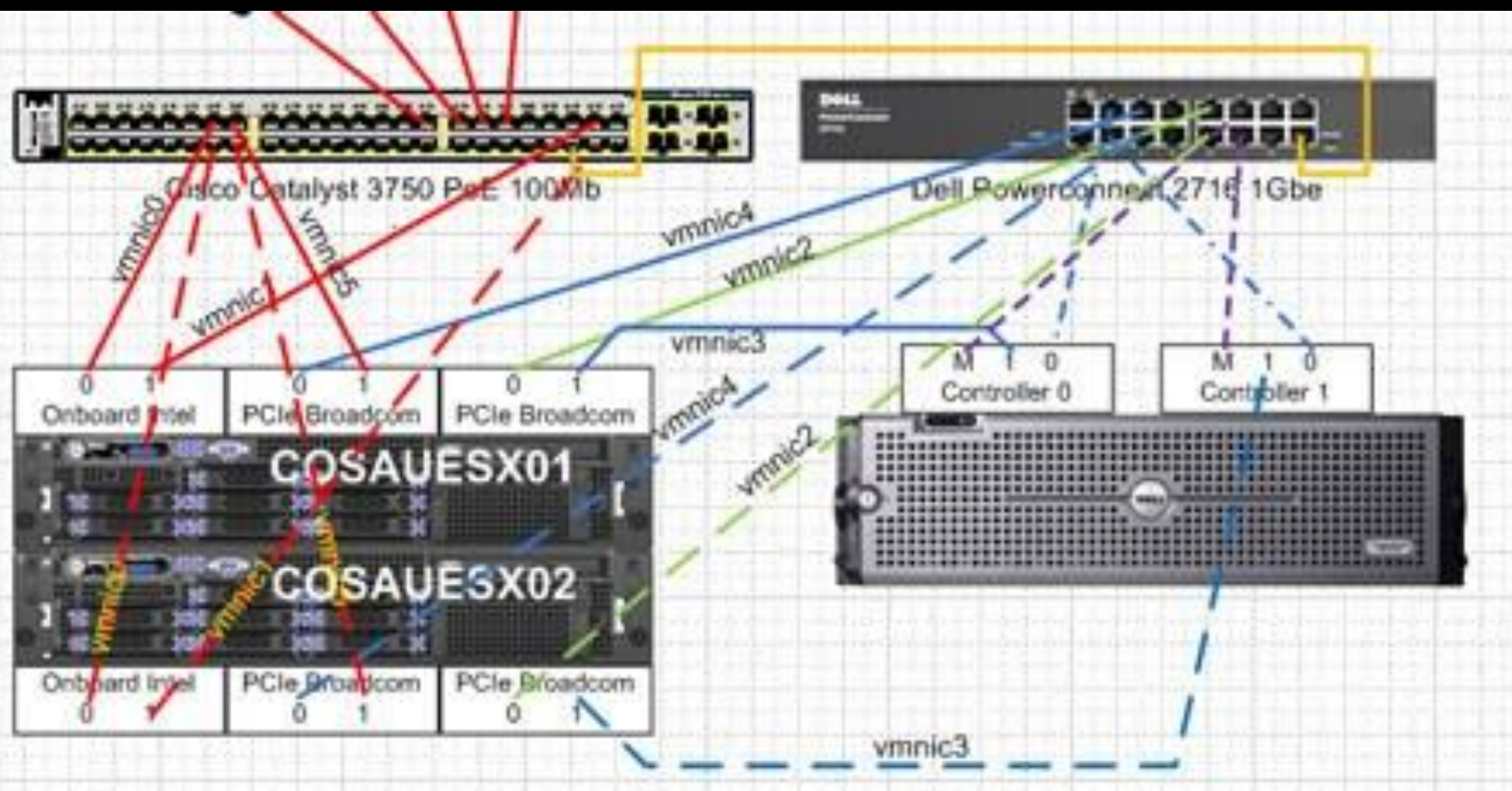
    def get_command(self):
        """Attempt to return a unicode object from the command socket

        If no message is available without blocking (as opposed to a blank
        message), return None
        """
        try:
            message_bytes = self.socket.recv(zmq.NOBLOCK)
            log.debug("Received message: %r", message_bytes)
        except zmq.ZMQError as exc:
            if exc.errno == zmq.EAGAIN:
                return None
            else:
                raise
        else:
            return message_bytes.decode(config.CODEC)

    def send_response(self, response):
        """Send a unicode object as reply to the most recently-issued command
        """
        response_bytes = response.encode(config.CODEC)

```

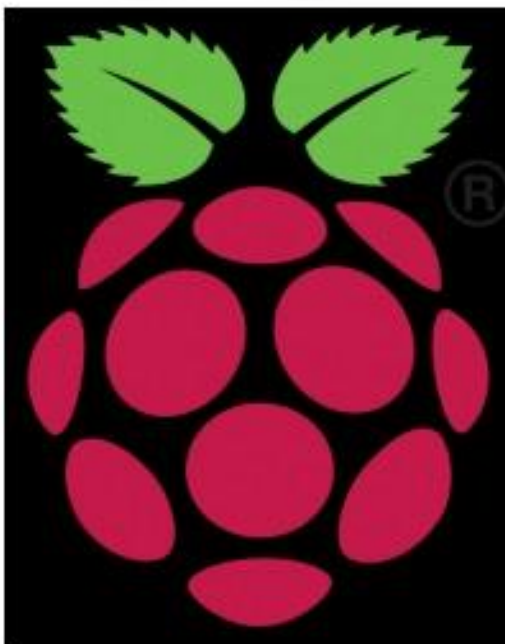













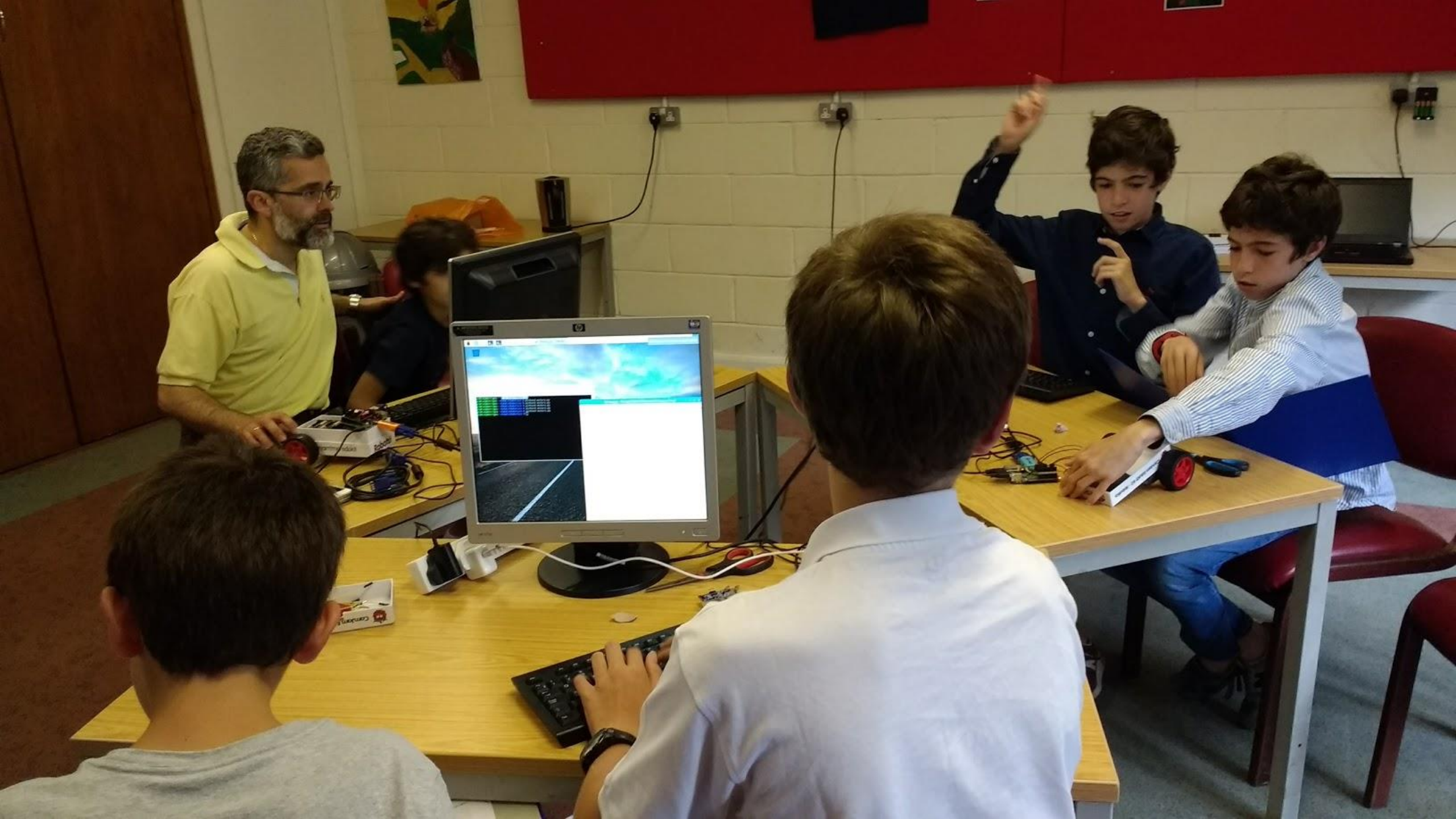


Table Of Contents

NetworkZero

- Functions
 - General
 - Discovery
 - Message-Sending

Previous topic

Questions to be answered

Next topic

Advertising & Discovering
Network Services

This Page

Show Source

Quick search

Enter search terms or a module,
class or function name.

NetworkZero

Easy network discovery & messaging

Aimed at a classroom or club situation, networkzero makes it simpler to have several machines or several processes on one machine discovering each other and talking across a network. Typical examples would include:

- Sending commands to a robot
- Sending scores to a scoreboard
- Having a remote sensor ping a central controller
- A peer-to-peer chat / instant messenger

To send a message and wait for a reply:

```
[Computer 1]
import networkzero as nw0

echo_address = nw0.advertise("echo")
while True:
    name = nw0.wait_for_message_from(echo_address)
    nw0.send_reply_to(echo_address, "Hello " + name)
```

```
[Computer 2]
import networkzero as nw0

echo_address = nw0.discover("echo")

reply = nw0.send_message_to(echo_address, "Alice")
print(reply)
reply = nw0.send_message_to(echo_address, "Bob")
print(reply)
```



PyGame Zero

GPIO Zero



GUI Zero

```
[Computer 1]
import networkzero as nw0

echo_address = nw0.advertise("echo")
while True:
    name = nw0.wait_for_message_from(echo_address)
    nw0.send_reply_to(echo_address, "Hello " + name)
```

```
[Computer 2]
import networkzero as nw0

echo_address = nw0.discover("echo")

reply = nw0.send_message_to(echo_address, "Alice")
print(reply)
reply = nw0.send_message_to(echo_address, "Bob")
print(reply)
```