Contents

1 Arduino Wii Nunchuk

1.1 Nunchuk Library

1.2 Send to STDOUT

2 Netcat TCP piping

2.1 Setting up netcat

This is in a linux environment. The setup may be different for Windows.

1. Open two terminals. One terminal will be the server, the other is the client.

2. The server listens for incoming messages. So on the server terminal, enter the command:

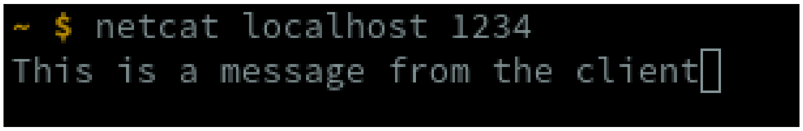
   ```
   netcat --listen --local-port=1234
   ```

3. On the client terminal, connect to the server via:

   ```
   netcat localhost 1234
   ```

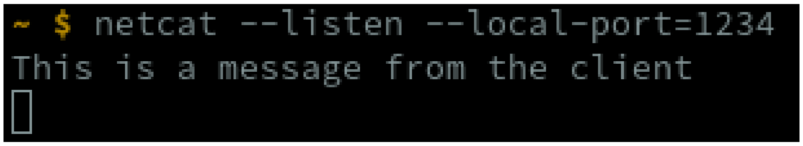
   localhost is 127.0.1.1 and indicates that I am connecting the client to my own computer. In later sections, once the client is the esp8266 module, the esp8266 will use the public ip of my computer instead.

4. You should now be able to write a message from the client to the server, and vice versa.
   Client Terminal:



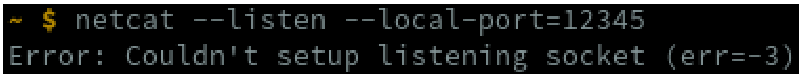
```
~ $ netcat localhost 1234
This is a message from the client
```

Server Terminal:



```
~ $ netcat --listen --local-port=1234
This is a message from the client
```

The port you choose is almost trivial. Generally, it should be above 1024, because 0-1024 contain well-known ports (e.g. HTTP, FTP, SSH, DHCP. . .). Other ports may be taken too. In my case, for example, port 12344 worked, but port 12345 did not.



```
~ $ netcat --listen --local-port=12345
Error: Couldn't setup listening socket (err=-3)
```

2.2 Piping Nunchuk Data to a Local TCP Client

Now that we have tested a local TCP server and client connection, we can send nunchuk data from the local server to the local client. We will be doing this in Bash.

The baud rate (9600) is set in the Nunchuk Arduino sketch.

```
baud=9600
port=1234
device=/dev/ttyACM0
```

We will be using picocom as a serial communications program to get data from the Arduino.

```
picocom --imap lfcrlf --baud $baud $device
```

The netcat command will be used to send picocom data from the server to the client.

```
netcat --listen --local-port=$port
```

The data from picocom is piped to the client and to the STDOUT (with the tee command). The final script is:

```
baud=9600
port=1234
device=/dev/ttyACM0

main() {
        getNunchukData | tee >(sendToClient)
}

getNunchukData() {
        picocom --imap lfcrlf --baud $baud $device
}

sendToClient() {
        netcat --listen --local-port=$port
}

main
```

2.3 Piping Nunchuk Data to the ESP8266

To set up the esp8266, I am using an Arduino Mega. Eventually, the ZyboZ7 will replace the Arduino Mega.