

Progress Report

1. Goal from last week

- Applied functionality to the “MOVE” button. The user can move the robot arm using MATLAB
- Work in Seismic Surveying project

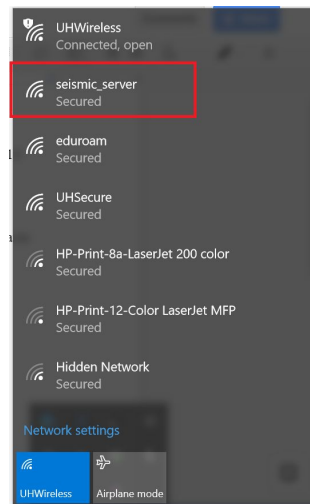
2. Accomplishments in this 2 weeks

- Make the Raspberry Pi an access point

IP address = 172.24.1.1

user = pi

pass = raspberry



- Program a Server to run in the Raspberry Pi and save received data (**Port 8000**)
- Program the M0 to collect data from the sensor and transmit to the Raspberry Pi

i. What is Working

The client constantly gathers data and every 20 seconds it saves the data in a file. At the same time, it searches for the Raspberry Pi network and measures its signal strength. When the server is close, the client tries to connect. When the client connects to the server, it starts to send the generated files. If there are no files to send, the client sends the data until the server is out of range. When the server is disconnected, the client searches again for the Raspberry Pi network and measures its signal until again it is found in the predetermined signal strength range.

ii. Limitations

The server cannot detect when the client is powered off abruptly, unlike the client which can. Also, when the connection between the server and the client is broken before reading data, the server stops, expecting to

receive data. During testing, when periodically changing among reading data, sending files and sending data no problems were found. On the other hand, when collecting data continuously for a long period of time, the server seems to freeze. Finally, all the tasks are shared within the main loop cycle and this affects the gathering of data. Currently we collect one value per second.

- Meet with Computer Network TA and discussed how to optimize the system

3. Goal for next week:

- Fix bugs found when moving robot arm using MATLAB
 - i. After one run the GUI stops.
- Change from sending string to sending bytes to the server
- Research on implemented low-level timers to interrupt time consuming high-level functions and collect more data per second.