



Project Notebook

Localization for Area Coverage

Caleb Lu

clu@mail.bradley.edu

Beginning April 2019

Contents

Friday, Jun 28, 2019	5
Thursday, July 4, 2019	7
Friday, July 11, 2019	9
Friday, July 12, 2019	11
Friday, July 19, 2019	13
Friday, July 25, 2019	15

Friday, Jun 28, 2019

After a meeting with Dr. Miah, the schedule of how I spend time on this project has been set. Every other Friday will be a presentation of accomplished work and future plans. Every Friday will be a day that Amr, Brian, and I will meet in the lab and work on our specific assignments. Today was the first day of working in the lab. I was able to clarify with Dr. Miah what specifically was the goal of my assignment and what was I suppose to do.

The task that I was assigned is to use multiple Zigbee boards to communicate to a BeagleBoneBlue (BBB) board. The end goal would be to have the BBB read the received signal strength , or RSS, of each Zigbee module. How to implement this was clarified by Dr. Miah.

Thursday, July 4, 2019

Continued work from last entry.

- Configuring the Zigbee modules
- Configuring the BBB
- Have the Zigbee communicate with each other.

Today I also need to accomplish the following.

- Presentation of work completed for the past week.

I have encounter a new issue, or atleast an old one has comeback. The issue of the Zigbee modules not connecting with the XCTU software has come back. This isn't a surprise since last time it was fixed, it was a result of it suddenly starting to work.

Will investigate in the future.

Thursday, July 11, 2019

Sent Dr.Miah and update email about a Eric's schedule. Eric is a graduated Bradley student who will be advising the projects. I discussed with Eric with potential problems and solutions.

Friday, July 12, 2019

Re-planned my approach on what I want to do and how to do it. My current understanding was to use the USB interface adapter with the XBee to communicate. The intended approach is to have the XBees run by themselves with out a USB interface board and in a mode that lets them run through commands from the main XBee. Will continue in this direction next time I get a chance to work on it.

Friday, July 19, 2019

Began reading through the XBee data sheet. Looking into how to use the XBee boards without the need for an interface board. My current goal is to get 2-3 XBee connected on a network and communicating. The setup will be, one board is the coordinator and the other 2 will be endpoints/beacons. The coordinator will be connected to my laptop through the USB interface board. If i have the setup working, then I should be able to request a status to a specific beacon and using the coordinator, I should be able to read the RSSI.

Thursday, July 25, 2019

A little behind in the work that needs to be done. Today I try the setup I discussed in the previous entry and will provide more details into how to do it/ what I did.

The XBee has two modes of operation: API mode and AT (Transparent) mode. The one that will be utilizing will be API mode because it has more features geared towards remote communication and will be helpful in sending commands with out a USB interface board.

Will continue to read into how to use API mode and what is required to send messages in this mode.

Friday, July 26, 2019

Current setup I have is one XBee is the coordinator and one other is an endpoint. Today I configured the other XBee board I had. SO now I have 2 endpoints and 1 coordinator. Looking into how to send a command to the coordinator to check the RSSI of the 2 endpoints.

Sunday, July 28, 2019

Continued reading and determining how to use API mode in the current setup. Also accessed the Beagle Bone Blue to check what options I have. Face an unexpected problem. The Beagle Bone Blue's (BBB) Root password is not the default password. So will need to learn how to reset the password to make changes to the BBB. The reason I need Root access is to set settings as a super user. Currently can't since the board seemed to be used by another student for the Mechatronics course. This might also explain why some of the encoder ports of the BBB are broken.

On method I wanted to try was to download XCTU, the program used to interface with the XBee boards through the USB port, onto the BBB and attempt to run commands from there. Couldn't try that after testing around with the BBB due to not having root access. This BBB Root Access issue will probably be a major issue soon. Will divert focus to getting the XBee boards communication and making a C code to send commands to the XBee boards.

By the end of the work today, I was able to send a command to the XBee board to read the RSSI value. Using XCTU, I broadcast the command to return RSSI of the last received packet. This AT command, the commands for the XBee API mode, was "DB". This has led to another issue to solve, how to send this message to a specific XBee board. Will continue next work session.

Monday, July 29, 2019

Went to Jobst and worked in the lounge. The lab was locked by the time I arrived.

Documented time: 5:15 PM - 7:15 PM

Today, the goal is to document the means of connecting to the XBee boards and attempting to communicate with the XBee board through my laptop : Windows. After successful communication from windows, will use knowlegde to communicate from Linux/Write C code to communicate.

I predict I will have issues from the BBB. Without root control, I'm not sure if I will be able to register the correct serial port. After communicating from windows will begin to write C code to send messages over Linux. A small problem is that I will not be able to see the messages I'm sending, so I don't know if I'm sending the write one. Will read incoming data and store that to check if A: the XBee is responding, and B: the XBee sent back the correct data.

Tuesday, July 30, 2019

Updated some previous entries. (Entries are kept in a physical notebook and then moved over to this document).

Today I'm trying to write code to communicate with the XBee over serial connection from the BBB. Yesterday I was able to send the commands to the XBee, but was unsure about what was sent and if the command sent was correct. One major mistake from yesterday was not trying to read any incoming values after sending the command to read RSSI. Testing that today to check response is correct.