Progress Review 3

Individual lab report – 04 || November 12, 2015

Team Daedalus
Richa Varma

Team Members:

Mohak Bhardwaj
Dorothy Kirlew
Pranav Maheshwari
Shivam Gautam

INDIVIDUAL PROGRESS

- Learning Python and ROS
- Setting up XBee DigiMesh adapters and testing them as ROS nodes
- PCB Design

1. Learning Python and ROS:

   The first decision I had to make before starting off with writing the code for the Xbee node in ROS, was the programming language to move forward with. I was more comfortable with C++ and went ahead with writing a simple publisher and subscriber node for serial messages to be sent through XBees.
   It was decided soon after though, that it would be more convenient to work in Python as Pranav had a lot of experience in it and we were working together on this task.
   I spent some time learning Python and using it to write a ROS publisher. In this process, I earned some more experience with ROS as well which is critical for our project.

2. XBee

   a. Setting up equipment and preliminary testing:
      The first step was to go through with the datasheet of the Xbees to understand configuration settings and functioning of the XBee 900 DigiMesh adapters.
      The XCTU software provides a graphical network view for configuring settings and working in API mode.
      The XBees can be operated in one of the two modes:
      1. Transparent mode
      2. API mode

      Transparent mode does not require configuration settings and the modules operate as serial line replacements. The frame-based API mode extends the level to which a host application can interact with the networking capabilities of the module. The data is

contained in frames that define events and operations as per requirements of the user.

We explored the API mode and established that for our application, the transparent mode is sufficient as we aim for a mesh topology in which all the nodes talk to each other and no addresses need to be specified. The data link layer of the adapters takes care of reliability of the data transfer.

Packets were sent and received in Transparent Mode for preliminary testing.

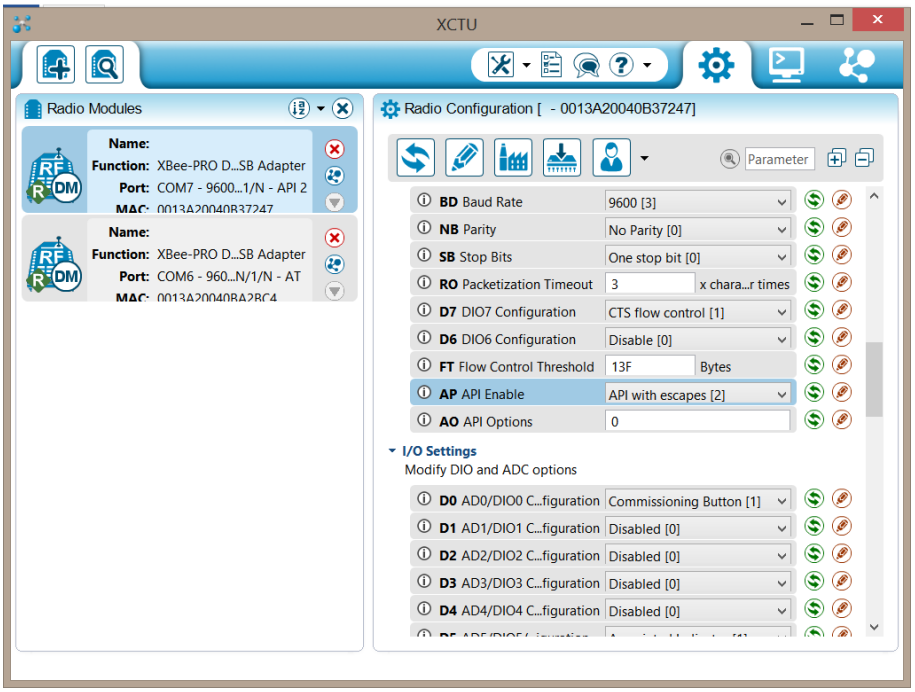

Figure 1. Config settings on XCTU

XCTU shows the two adapters connected to the COM ports having unique MAC addresses. The UART parameters like Baud Rate, parity, start and stop bits, flow control threshold can be set through this interface.

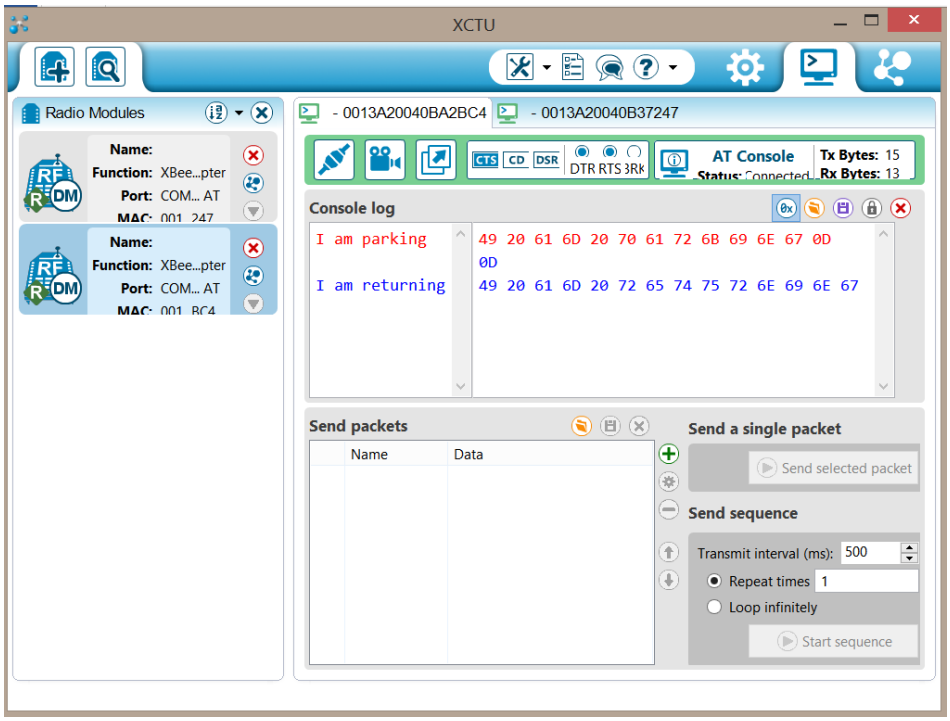

Figure 2

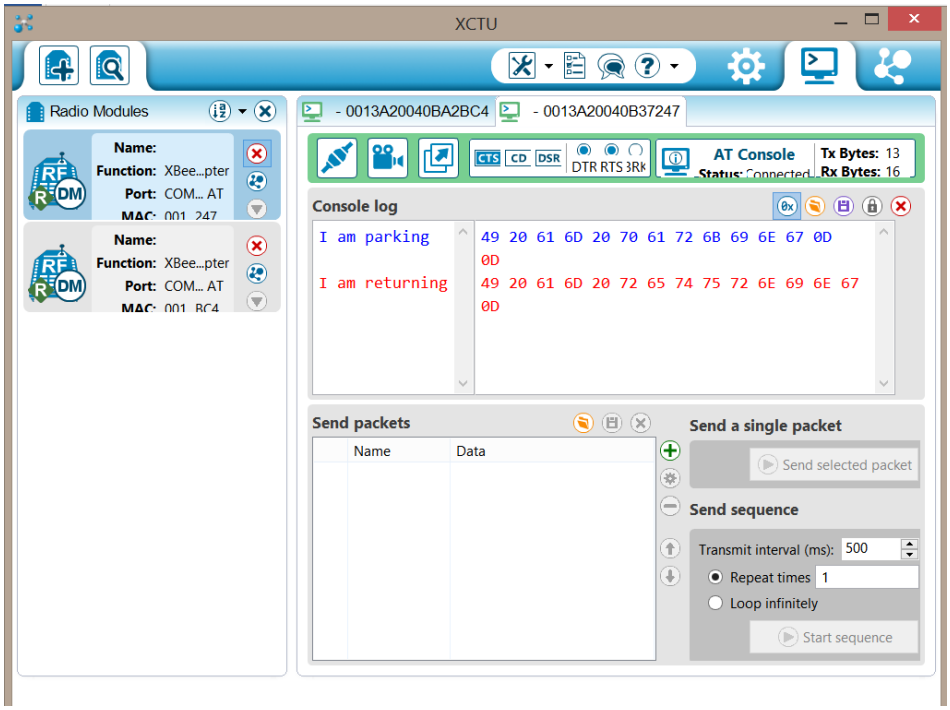

Figure 3

The figures show two modules connected and sharing information in HEX format over serial.

The next step was moving to ROS and creating a serial comm node to send and receive data. This was done using the ROS serial library. For testing purposes, I worked with Pranav to write the node in Python which sent and received data at regular intervals in different formats.

Testing the network:

- Testing carried out using 2 Xbee Pro 900 adapters
- Data sent using timestamps between two stations
- Range tested : 15 feet
- Latency test:
  Timestamped 100 element integer list (502 bytes) sent over serial
  Latency observed: 1.8 seconds

Performance:

- 95% of messages are received
  - At tested range, set buffer size : 100% of messages are received
- Network able to accommodate at least 3 vehicles
  - Network is scalable within the 450 ft indoor range
- Rejoin network within 30 seconds of connection loss
  - Connection established instantaneously; no authentication required

I also created a block diagram for our communication infrastructure, shown below.

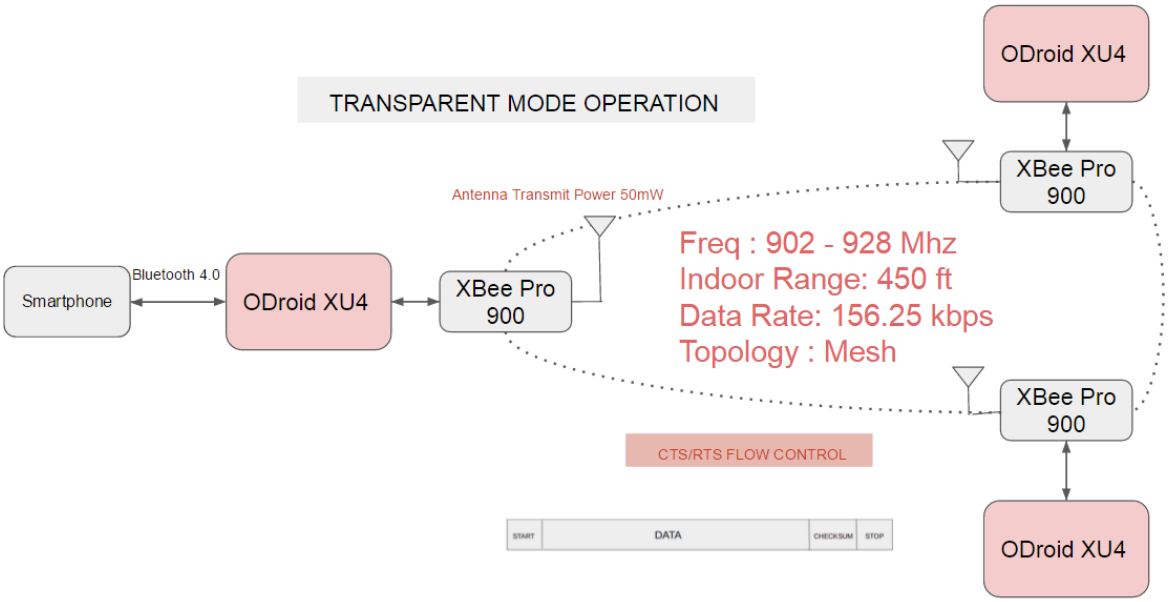

Figure 4. Communication Infrastructure

PCB Power Distribution Board

I worked with Shivam to design a power distribution board suitable for our application.

The subsystems that required regulated power were:

- ODroid XU4
- Kinect sensor
- Actuator Control Board

I selected the components and designed the schematic for the circuit. Shivam worked on routing the board and I compiled the Bill of Materials.

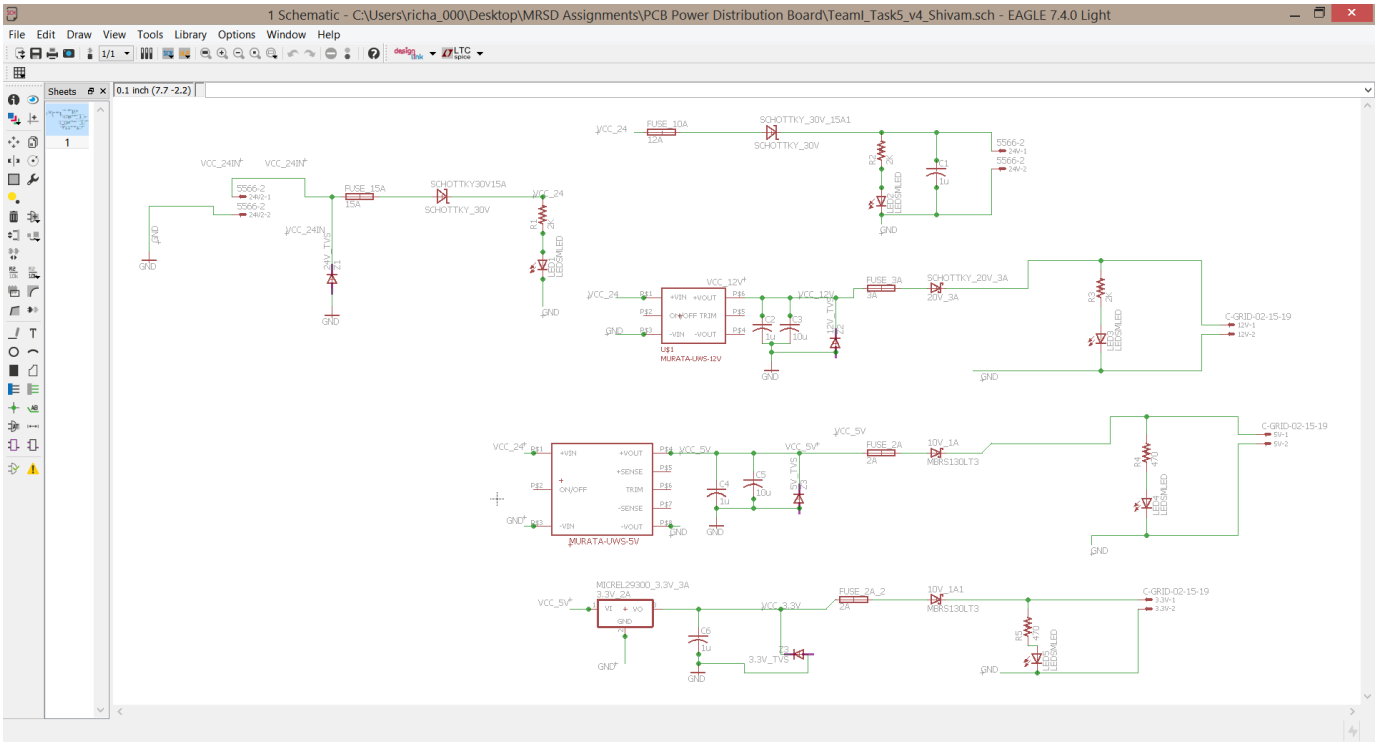

Figure 5. PCB Schematic

After further discussion, we realized that since our new platform, the Oculus Prime, comes with its own power distribution board, it would be of little use to fabricate another one. Based on the inputs from Prof. Dolan and Luis, we decided to go for a sensor interfacing board that would house our three SHARP IR sensors for proximity sensing connected to an on-board Arduino Nano.

Shivam designed the schematic design and board and I prepared the Bill of Materials. I am currently finalizing the additional connectors that we would require to connect this board to our system and will place the components order through Keyla.

TEAMWORK

- I have been working with Pranav over the past two weeks on the communication subsystem. I have received significant help from him in getting familiar with Python and getting better with ROS. We have achieved a good amount of clarity on the functionality of this

subsystem in terms of the information that needs to be sent over the network and ways to handle the issue of scalability.

- I worked with Shivam and Mohak throughout the PCB task. We collaboratively defined the needs of our system and efficiently divided tasks for all the deliverables in this task.
- I am currently working with Shivam on electrical setup of the mobile platform. We have also started work on serial protocol definition and over the next week will clearly define the same.

CHALLENGES

The main challenges I faced during the past two weeks were:

1. Choosing the programming language for working in ROS was tough decision and was taken keeping in mind that Pranav had a lot of experience in ROS and Python and it would be wise going with it keeping in mind the tight schedule that we're on, in case some issues come up. The challenge was that I had to spend some time learning the language before starting work with the XBees. It seems to be working out fine and we are able to deliver on our goals.

2. The decision on the kind of PCB we needed for our system was affected by the fact that we ordered our platform around the same time. We were aware that it comes with a distribution board. But in order to meet the requirements of the task, we went ahead with a power distribution board and only after the process was complete did we alter our plans and designed a new sensor control board. The extra time spent in this process could have been avoided.

3. The main challenge that I am currently facing is designing the standard protocol for the serial messages. Being the task owner of the communication subsystem, I have to make a decision based on the long term requirements that the network needs to fulfill. The protocol should be general enough, so that in future the format can be modified to accommodate additional data fields without making too many changes. Moreover, I need to make sure that the network is scalable, i.e. take account of the fact that our system will have

multiple vehicles in the near future. I am actively working to define it in such a way that not only ensures the success of our Fall validation, but also creates a proper framework for the Spring.

FUTURE PLANS

At this point, all the subsystems are functional and the mobile platform has arrived. Pranav and Mohak have partially assembled the platform and I am currently working with Shivam to fix up the wiring and make sure there are no errors in the connections before the first test run.

Next, integration will be the prime focus. The team will work together to integrate individual subsystems according to the defined software and hardware architecture.

We have ordered a third XBee module and I will be working on creating and testing the mesh network functionality.

Serial Protocol definition is an important task that will be taken care of this week, and I'll be working on it with Shivam.

Apart from this:

1. Mohak and Pranav shall set-up the platform to work with and be controlled by the Single Board Computer via ROS.
2. Dorothy is the task owner for the Mobile App and is working on refining and testing the app to ensure that it works robustly. Pranav shall be working with her to test the bi-directional communication between the Single Board Computer and the smartphone.
3. Mohak is the task owner for Vision system and will collaborate with Shivam to implement and test the Obstacle Detection algorithm on Kinect.
4. Shivam and Dorothy are working on the IR proximity sensors + Arduino Nano for close range obstacle detection. Once their setup is ready, Pranav will create a CAD of the mounting base needed to attach all of this to Oculus Prime and get it 3D printed.