Post-Lab 3: IEEE 802.15.4 & Thread

## What to submit?

Please use this document as a template, add your responses directly, and export it as a PDF to Gradescope. Each group should submit one post-lab.

**Group name:**

**Team member names: Rajan Verma (Group Activities with Connors and Anand)**

**Link to GitHub repository:**

# E. Setting up a Receiver and Sender

1. **What did you set the 1st byte of the payload to?**
2. **Prove that you can receive the packets with the updated payload (A screenshot of serial output works here),**

# F. Filtering out packets not meant for your receiver

1. **How long does it take to send an 802.15.4 preamble?**
2. **What is the updated extended address of your receiver?**
3. **What nrf\_802154\_ function did you use to change the channel?**
4. **Which channel did you use?**
5. **Prove you can receive packets with the updated address, channel, and PAN id changes (A screenshot of serial output works here).**

# G. Getting acknowledgements from the receiver

1. **Include a screenshot of the updated sender packet showing acknowledgements being requested.**
2. **Demonstrate that the sender transmissions are successful (A screenshot of serial output works here).**

# H. Reducing the packet size

1. **By how many bytes did you reduce the packet size by switching to short addresses?**
2. **What did you change the sender’s and receiver’s addresses to?**
3. **Set short address for the receiver (Include your updated sender code and receiver code. A GitHub link to a specific commit or lines of code works here.)**

# I. Low Power Listening (LPL)

1. **Code for the transmitter and receiver. A GitHub link to the two main.c files is fine.**
2. **The terminal outputs from both the transmitter and receiver for the 10 packets using both check periods. This should be four separate text files. A GitHub link to these text files is fine.**
3. **A short (paragraph or two) description of your results. How do you know your receiver was correctly duty cycling to save energy? Did this version of LPL work? Did you run into any challenges?**
4. **CHECKOFF: We will test your two boards with each other and against our reference implementation. Make sure to #define the check period and parameters of the network so you can easily change it.**
   1. We will change your check window to a large value to verify the board does not receive packets while sleeping.
   2. We will make your check window reasonable to make sure it works correctly on both the receiver and sender. The sender should print how long the packet train was.

Start of 3B Material

# K. Join the class Thread network

1. **IP addresses of both boards:**
2. **Router table:**
3. **Child table:**
4. **Neighbor table:**

**TASK 1:** List the IP addresses of both connected boards and include them in your report.

A computer screen with white text

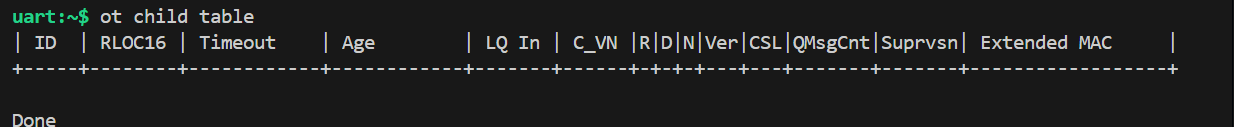
AI-generated content may be incorrect.

**TASK 2 :** Run the ot router table command in the CLI app and copy or screenshot the output.

A screenshot of a computer

AI-generated content may be incorrect.

**TASK 3:** Run the ot child table command in the CLI app and copy or screenshot the output.



**TASK 4:** Run the ot neighbor table command in the CLI app and copy or screenshot the output.

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Task 5 : TASK 5:** Run a ping and show the output.

**Ping output:  
A computer screen with white text

AI-generated content may be incorrect.**

# L. Send UDP messages between your devices

**Show evidence (with sufficient explanation) to show your devices sending and receiving UDP packets.**Sender :Com4

A screen shot of a computer program

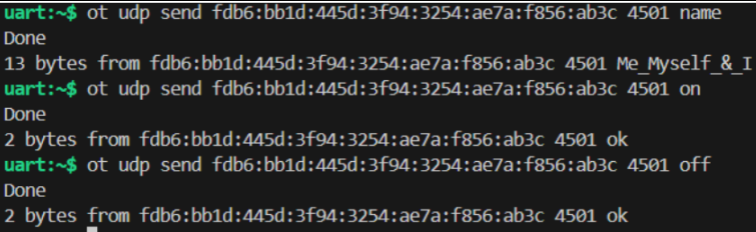
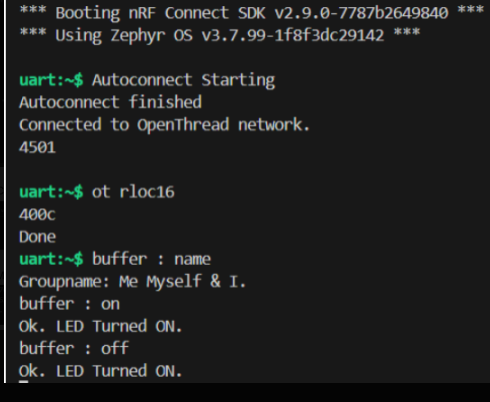
AI-generated content may be incorrect.

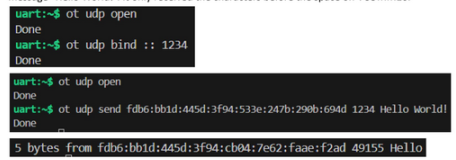
Receiver : Pita Bread

A screen shot of a computer code

AI-generated content may be incorrect.

# M. UDP Server

1. **Your code for your UDP device. A GitHub link is fine here.**
2. **Terminal output showing your UDP device working (with all three commands).  
     
     
   **

****

## EC2. OPTIONAL: Visualize the network topology

***This section is worth bonus points if you complete it.***

1. **Show the network topology.**
2. **Show the network topology after changing your device’s role (highlighting the change).**

## EC3. OPTIONAL: LED Service

***This section is worth bonus points if you complete it.***

1. **Your code for your LED service. A GitHub link is fine here.**
2. **A video showing the button presses controlling the other board with the three commands.**