Senior Design Progress Report

**Student**: Brian Dye **Semester**: Spring 2022

**Week**: 2

**Team**: 20 ENIGMA

**Position**: Team Leader

**Hours**: 20-30

# Progress Description

This week was used to solidify our plans to accomplish the goals of our project. During our lab on Wednesday, my team and I discussed the pros and cons of utilizing a second microcontroller to carry some of the responsibilities of the network stack. We were concern with the battery consumption of using 2 microcontrollers to accomplish our project goals. I consulted with a past professor on his thoughts about utilizing a second microcontroller and I learned that the interfaces that would utilize the wireless land area network implement the functionality of the network stack that we were going to implement on the second microcontroller. After realizing this my team can utilize one microcontroller instead of relying on two which saves money, power, and time designing. My team plans on utilizing one of the following designs:

**Figure 1:** Potential Microcontroller Layout

Diagram, schematic

Description automatically generated

# Individual Responsibilities

During this week, my team also established individual responsibilities that will contribute to the success of achieving our project goals. They include the following:

|  |  |
| --- | --- |
| **Team Member** | **Individual Responsibilities** |
| Brian Dye | Implement the embedded networking protocols of the wireless land area network. Write the embedded real-time operating software for a packet switched network. Assist Henry in the serial communication of the RFM-69 radio module and microcontroller device. Assist Nathan with developing the applications that will utilize our LAN. |
| Hanyu Zhu | Implement the embedded software necessary to facilitate a wireless packet switched network. Assist Henry & Nathan with embedded software development and user interface |
| Henry Wang | Responsible for developing the ecosystem serial communication software that communicates with the RFM-69 & Bluetooth module |
| Nathan Shenk | Responsible for implementing the application(s) that will utilize the mobile ad-hoc network. **Potential** applications include: (1) Text messaging services (2) Satellite view of the geographical locations of each node on the network (3) Audio services (4) File Transfer |

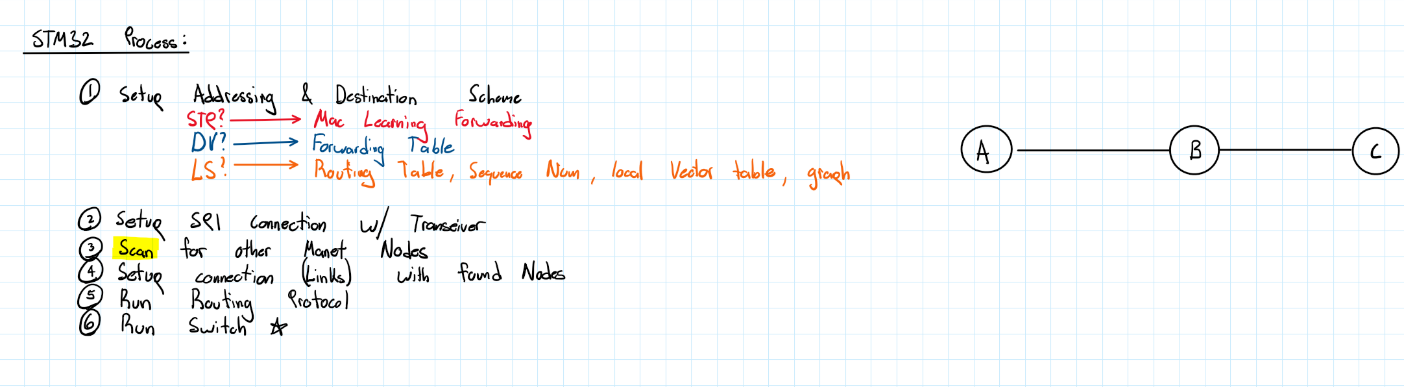
# Winter Break Progress (Pre-Semester)

Over the 2021 – 2022 winter break, I brainstormed about potential microcontrollers that could be used in our implementation. In ECE-36200, we utilized the STM32F0 microcontroller. It’s a microcontroller everyone in the group is familiar with. It’s programmed in C and comes with an ecosystem of timers, interrupts, general purpose input and output mechanisms, and serial communication protocols that include: UART, I2C, SPI, and DMA. It’s main clock operates at 48 MHz. In ECE-40862, we utilized the ESP-32 that comes with much more computational power and interfaces such as WiFi and Bluetooth. It can also be programmed in MicroPython. It has a Single/Dual Core microprocessor and advanced ultra-low-power management. I also wanted to find a radio module for the physical layer of the network stack that will allow us to communicate at distances of 500-600 meters at the **highest bandwidth possible**. I found the RFM-69HCW radio module. It’s a small device that communicates with a microcontroller via an SPI connection. It has a range of 500 meters and a maximum bitrate of 300 kbps = 37,500 Bytes/second. It operates in the ISM frequency spectrum (915 MHz) that is reserved for industrial, scientific, and medical purposes. I formulated how the concepts that we learned about in Introduction to Computer Communication and Networking (ECE-46300) can be utilized to create a Mobile Ad-Hoc network. What industry standards can we use to create the Data Link Layer? What industry standards can we use to create reliable communication (Transport Layer)?

Diagram, schematic

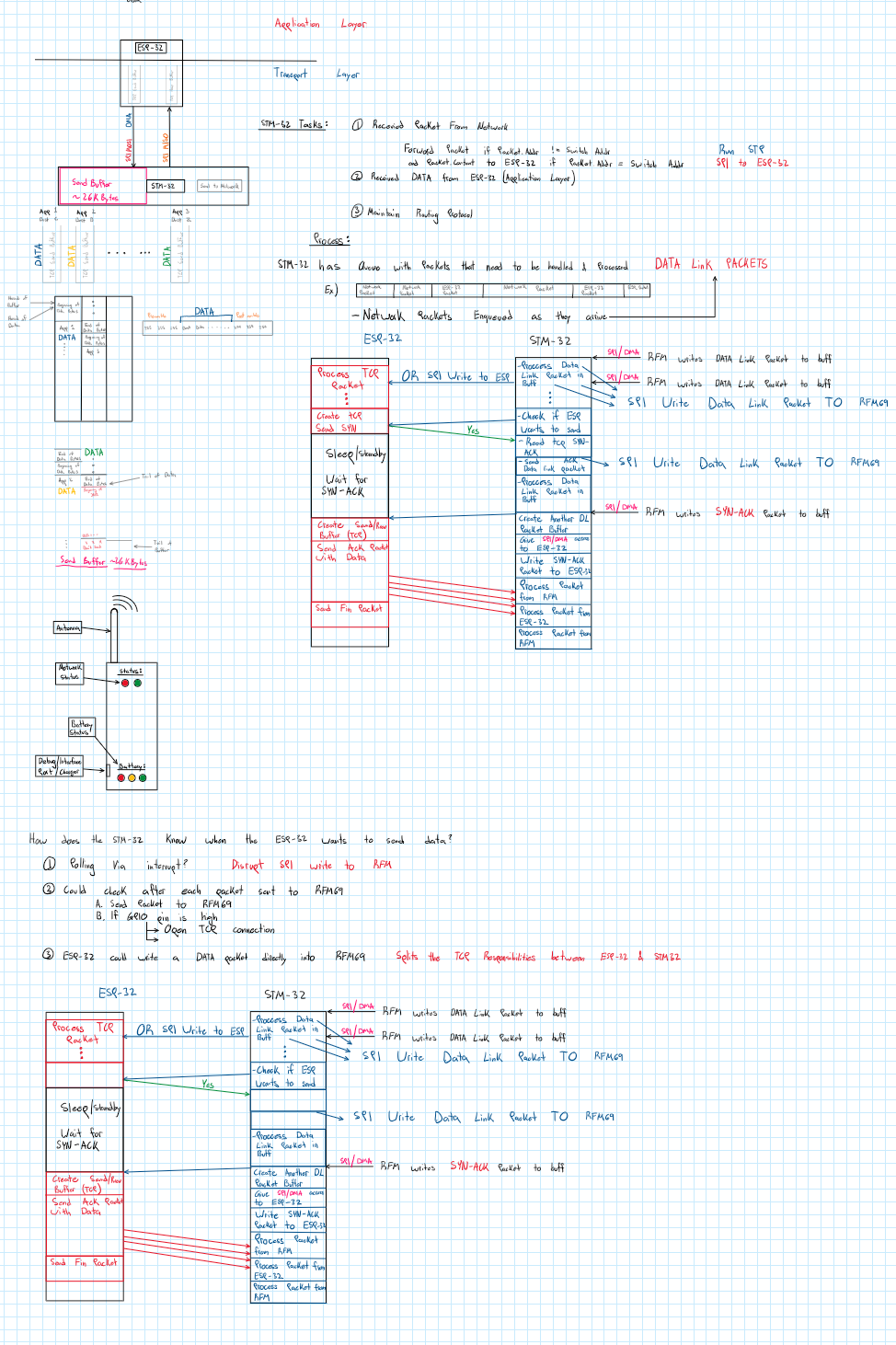
Description automatically generated

**Figure 2:** Microcontroller Network Stack Responsibilities

I also wanted to gain an understanding of the real-time scheduling that would be needed to maximize throughput in our wireless land area network. What would the STM-32 be responsible for? What would the ESP-32 be responsible for? How would the STM-32 communicate with the ESP-32? What would the device look like?

**Figure 3:** Process of STM-32 setup

My concern was concern was the rate at which data can be processed. CPUs on embedded microcontrollers sacrifice speed for general computation. To overcome this, I wanted to utilize Direct Memory Access (DMA) in conjunction with Serial Peripheral Interface (SPI) to reduce the workload of each Central Processing Unit at each node. I wrote C code that would enable the ESP-32 to directly write via SPI into a send buffer located in the STM-32’s memory. The STM-32 would then process the data into packets and wrap them with Data Link Headers and send the packets to the RFM-69 radio module. Below is a depiction of the abstraction protocol between the ESP-32, STM-32, and RFM-69 using SPI and DMA.



**Figure 4:** Communication between modules