

System Overview: RFID Network Project

Bryan Sandoval

May 2019

1 Introduction

The purpose of this document is to outline the system overview of the RFID Project and provide some helpful techniques to help narrow down what the problem is with the RFID project. It is important to note that I have been trying to replicate what Ben Duggan has done at Indiana University. He was able to successfully implement the nRF24L01 transceiver modules onto the GEN2 and the ETAG Readers while also creating a website that would collect the data recorded from the tag readers. In fact, he has begun creating a document that would act as a user manual to replicate his RFID system. His user manual is not complete, but I was working with him to try to replicate it. I have also attached a PDF copy of the user Manual with all the other documents needed. His user manual is a great start for you to get familiar with the hardware and understand some of the software set-up.

1.1 System Overview

Before going into detail of the troubleshooting, it is important to understand how the system is suppose to work. First, a RFID tag is read by ETAG Reader via the inductive loop antenna. Once the tag is read, the ETAG Reader should send that data to server that the Raspberry Pi creates via the nRF24L01 transceiver modules. The nRF24L01 module should be attached to every device that will be sending or receiving data. With that being said, when the data is being sent it is really the nRF24L01 modules that send or receive the data between the devices. Once the Raspberry Pi receives the data through the nRF24L01 module, it will process the data and post it on the website that it creates from the server.

Since the RFID network is not working the way it is suppose to, I have documented what has been trouble-shooted to be able to isolate the problem.

2 Troubling-shooting

The purpose of this section is to list out the possible problems that could be causing the system not to work. There are three parts to this system. There is the arduino IDE software scripts that were created to make a Network of nRF24L01 modules via the ETAG Readers (which is arduino based). The server, another main component, is used to communicate with the other nRF24L01 modules and act as a master node so that it can take the data and upload it to the website on the server. This server is created from the Raspberry Pi. To get more information, refer back to Ben's User manual and contact him if you have any questions regarding the server. Lastly, the

nRF24L01 modules attached to the ETAG Readers and one on the raspberry pi. The first priority of the troubleshooting section is to make sure that the transceiver modules can communicate with each other (when connected to ETAG Reader). My theory is that there is underlying problem with the software (or potentially hardware) that is causing it not output anything correctly on the serial monitor. If the radios are not communicating with each other, it could be that there is something wrong with the arduino itself or something is wrong with the nRF24L01 modules.

2.1 nRF24L01 Power Issue?

One thing I tried to trouble shoot is whether the nRF24L01 modules were receiving the correct amount of power to send and receive data. Ben mentions that you can use either a 5v to 3.3v power adapter with the nRF24L01 modules or just solder a capacitor across the VCC and GND pin. I have conducted a quick test using the RF24Network-Example script to see if either suggested solution would have an affect on the modules when attempting to transmit data. In Figure 1, I tried to run the script using a 10uF capacitor on the nRF24L01 module, but the serial monitor shows that it failed to send the data every time. In Figure 2, I tried to run the same script using the 5v to 3.3v power adapter on the nRF24L01 module. The serial monitor shows that the nRF24L01 modules are able to transmit data with the adapter connected. Lastly, I tried to run the script when I connected the nRF24L01 module directly to the 3.3v pin from the ETAG Reader. Figure 3 shows that even by connecting the transceiver directly to 3.3v, it still lacks the current needed to transmit properly.

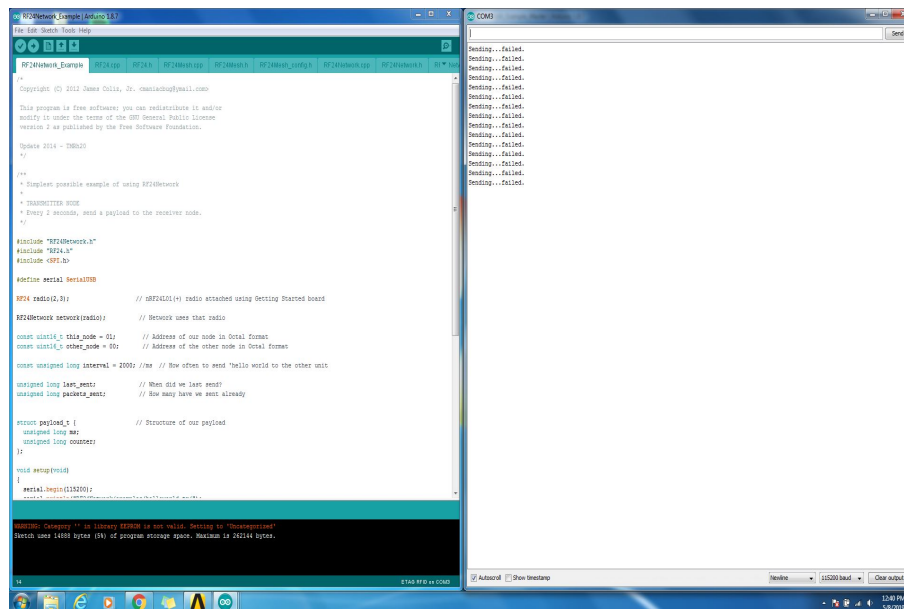
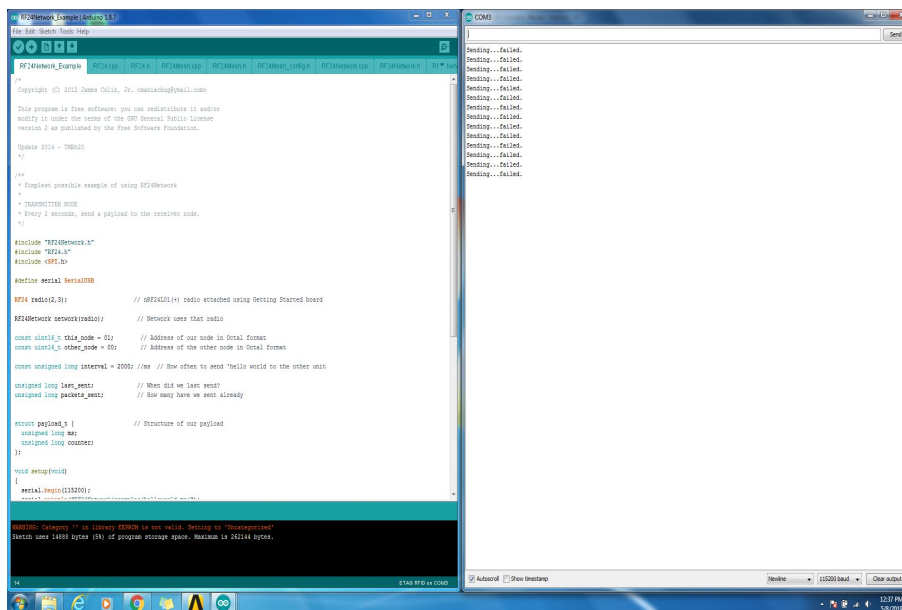
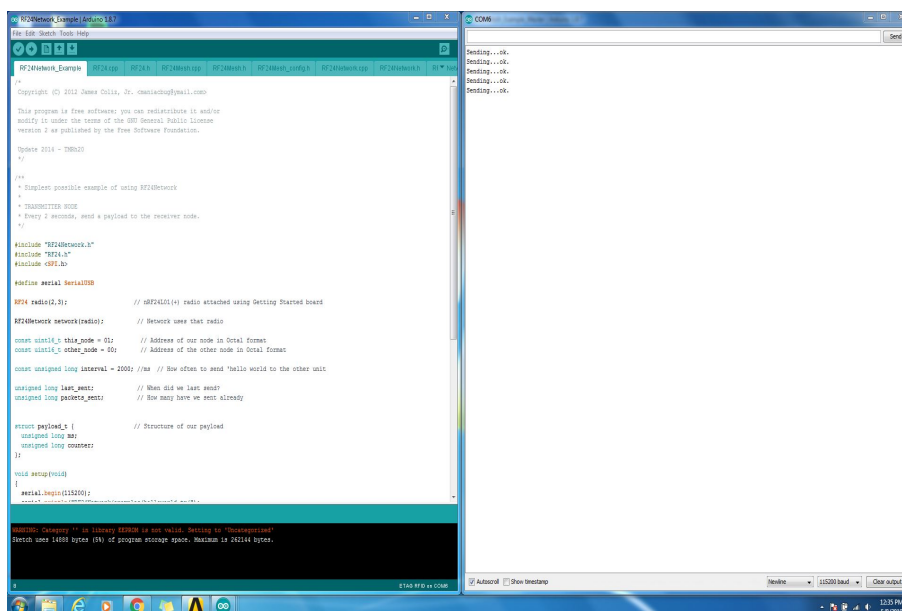


Figure 1: Transmitting with Capacitor



2.2 Using Arduino Unos to troubleshoot nRF24L01

Recently, I was able to get to Arduino Unos to be able to run a simple script that would allow us to see if the nRF24L01 modules are working properly. I found an online arduino forum that is helpful in setting this up. Here is the link to that: <https://forum.arduino.cc/index.php?topic=421081.0> I recommend that you follow the forum and see if you can get the same results that it mentioned in the forum.

2.3 Raspberry Pi Server

I realized that there wasn't much that I could to troubleshoot on the server because Ben was making modifications to it at the time. I have several screen shots of the terminal commands I used on the raspberry pi. I added them to a folder in the files given. I didn't think it was too necessary to talk about them because a lot of the commands should be done automatically when Ben finishes his auto-install script for the raspberry pi server.

3 Software Scripts

This section was made to get familiar with the RFID-Network and RF24Mesh scripts used for the ETAG Reader. I find it useful to watch videos instead of reading a report, so I have made two youtube videos to help you navigate through the arduino scripts and understand which one to use. Here is the link to the first introduction video: https://youtu.be/H1_S39CQZC4
Here is the link to the second introduction video: <https://youtu.be/x0SjcZyvWnQ>

4 Things to Know

4.1 Things to Know: Server Installation

Depending on which Raspberry Pi you use (whether its the model 3 or the Zero model), the auto-install script commands may not work correctly. To be more specific, the commands listed in the manual did not work on the Raspberry Pi Zero model for some reason. If you are try to re-install the server onto the Raspberry Pi Zero (and the original commands in the manual don't work), try inputting the follow commands into the terminal of the pi: "git clone https://github.com/BenSDuggan/RFID-Network-Build" (press enter), "mv RFID-Network-Build/install.sh ." (press enter), "chmod +x install.sh" (press enter), and lastly "./install.sh" (press enter). You may be prompted to input your github user name and password, however, if you have not contacted Ben beforehand it will not let you download the server. This is because the Pi tries to clone code from Ben's github which is private. You will need to contact him to add your github user name to give you access. His contact information is in the RFID Network User Manual PDF file.

4.2 Things to Know: Passwords

An important thing to note is what the passwords are for the raspberry pi and Ruyle Research laptop (labeled "Ruyle 2"). It is pretty simple because everything is pretty much the same password. All the passwords are set to "research9" and the user name for the website database is *AntennasRuyle*.

However, if you re-image the raspberry pi, you will have the option of setting a different passwords if needed.