

MCU TNC Design

David Cain, Kobe Keopraseuth, and Kaleb Leon

Design 2 - Final Presentation



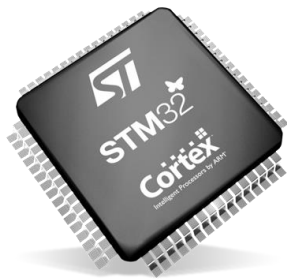
UNIVERSITY of
LOUISIANA
L A F A Y E T T E

Project Overview

- What is a TNC?
 - A modem for connecting a computer with a HAM radio
 - KISS
 - AX.25



Why every HAMer needs our TNC



Our Solution

VS

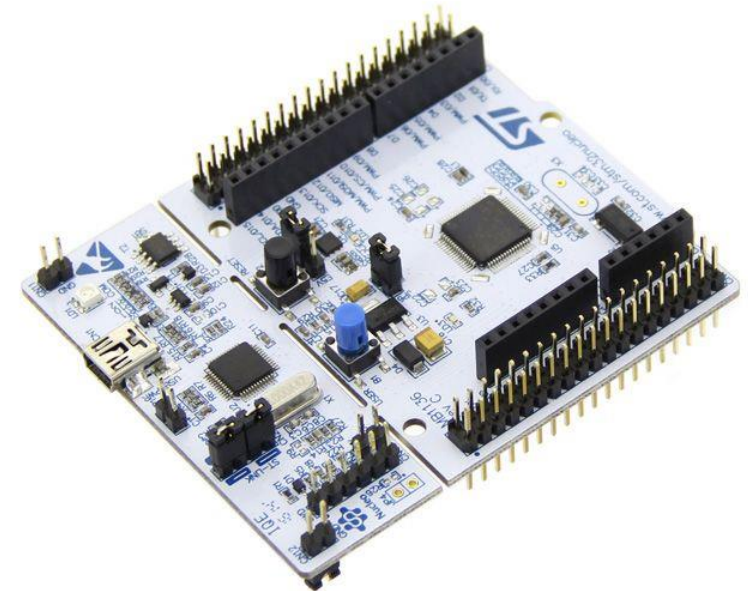
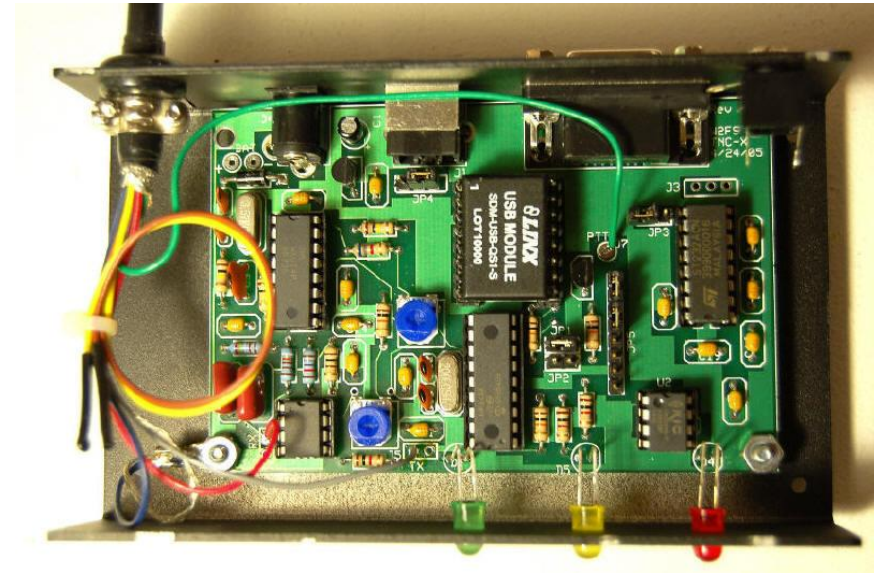


Commonly Used Hardware TNCs



MCU TNC Benefits

- Modular/Adaptable
- Open Source
- Easy to replicate
- Small form factor
- Well Documented
- Increase computational capability
- Cheap



Design Specifications

Scope of Work:

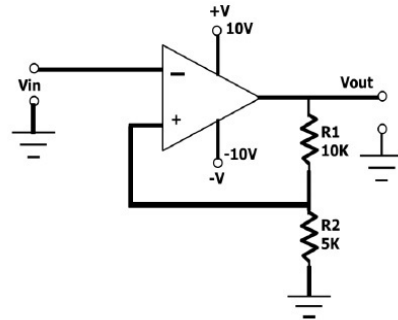
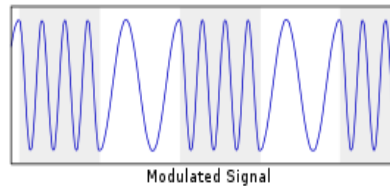
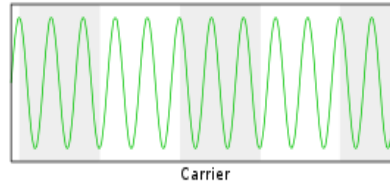
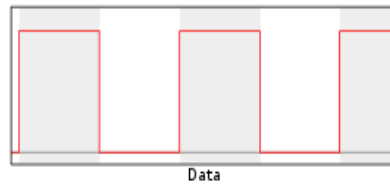
Design, implement, build, and test a TNC with an STM32 or similar microcontroller. The system must accept a string via UART or a similar interface and generate the necessary HDLC headers/packet data and then produce the correct audio waveform.

Flag	Address	Control	Information	FCS	Flag
8 bits	8 or more bits	8 or 16 bits	Variable length, $8 \times n$ bits	16 or 32 bits	8 bits

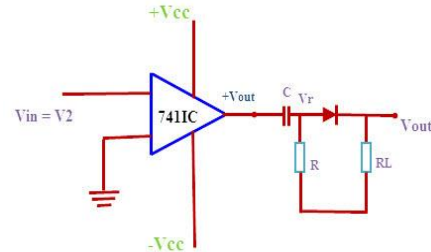


Methodology

Previous Assumptions



Schmitt Trigger



Zero Crossing



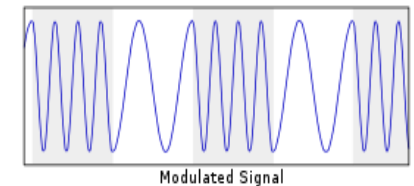
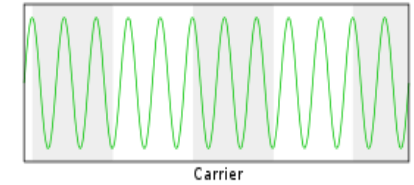
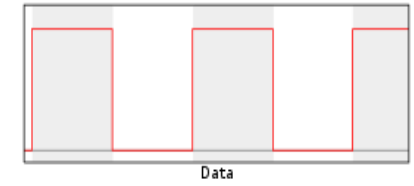
Actual Functionality

$A_{t0} \triangleq \text{Amplitude at time } t0$

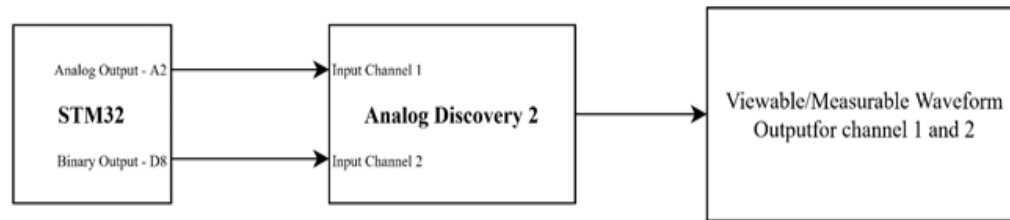
$\varphi_{t0} \triangleq \text{Phase at time } t0$

$$\varphi_{t0} = \sin^{-1}(A_{t0})$$

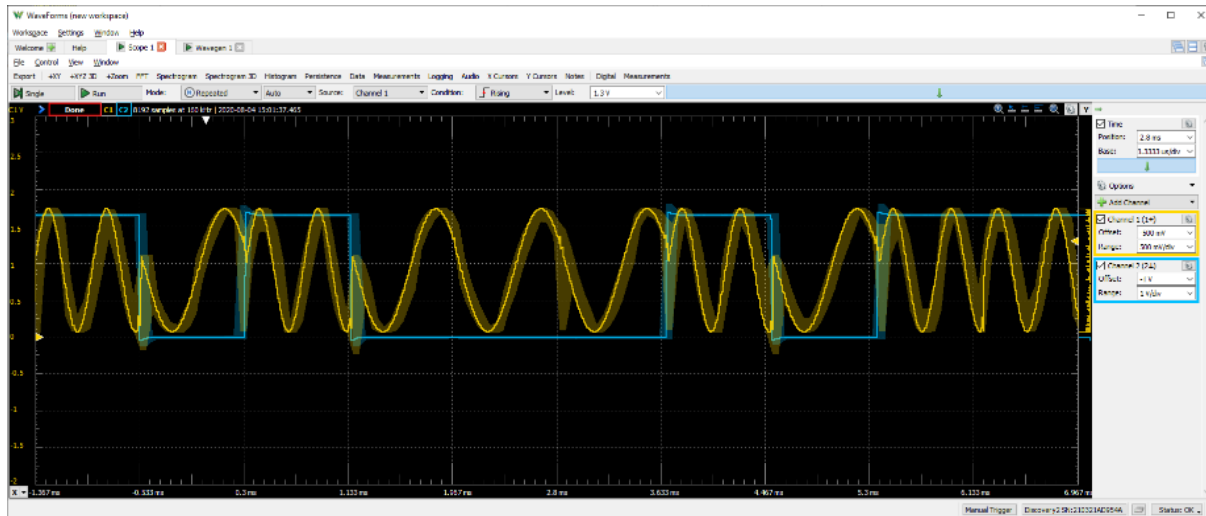
$$\omega = \frac{\varphi_1 - \varphi_2}{t_1 - t_2}$$



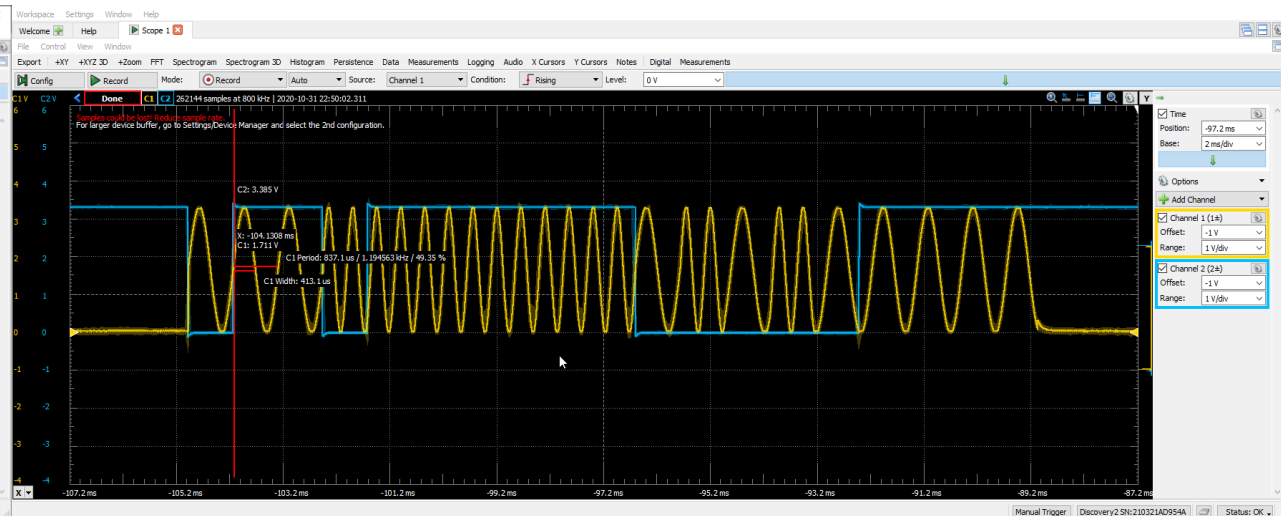
Testing and Validation



Bit Transitions FAIL



Bit Transitions PASS



Packet Generation

The screenshot displays a software interface for packet generation and analysis, likely for APRS (Amateur Radio Packet System) services. The main window shows a waveform plot with a yellow trace, indicating signal activity over time. The plot is titled "WaveForms (default)" and includes various toolbars for file operations, settings, and analysis.

Overlaid windows provide detailed information and configuration options:

- Measurements:** This window displays statistical data for two channels, C1 and C2. The table below summarizes the key measurements:

Channel	Measurement	Value
C2	Peak2Peak	14.961 mV
C2	Maximum	-6.14473 mV
C2	Minimum	-21.10570 mV
C1	Maximum	75.197 mV
C1	Minimum	-43.77890 mV
C1	Peak2Peak	118.98 mV

- AX.25 Interface:** This window is used for configuring APRS services. It includes fields for Destination Callsign (GHGH), Source Callsign (JKJK), and Repeater Callsign (COMM). It also displays calculated HEX strings and SSID binary values. The interface includes buttons for "Configure AX.25", "Override SSID", and "Convert to ASCII".

At the bottom, a code editor shows C++ code snippets related to packet handling, including a function signature `(uint8_t hex_byte_in, bool *bin_byte_out);` and a variable declaration `bool *curr_mem = local_packet->KISS_PACKET+16; //+8 is to skip the`.



Bit Formatting Validation

Destination Callsign:
KALEB

Calculated HEX String
0x96,0x82,0x98,0x8A,0x84,0x40,0xE8,

Calculated SSID Binary
11101000

License Expires 12/15/2020

Source Callsign:
DAVID

2

0x88,0x82,0xAC,0x92,0x88,0x40,0x65,

01100101

Override SSID

Repeater Callsign:

Override SSID

Configure AX25

☐ Use Repeater Address

0x96,0x82,0x98,0x8A,0x84,0x40,0xE8,0x88,0x82,0xAC,0x92,0x88,0x40,0x65,0x03,0xF0,

Convert to ASCII

Comm Port Opened

Transmitted :DAVID-2->KALEB-4
DATA

Transmitted Hex: 0xC0,0x00,0x96,0x82,0x98,0x8A,0x84,0x40,0xE8,0x88,0x82,0xAC,0x92,0x88,0x40,0x65,0x03,0xF0,0x44,0x41,0x54,0x41,0xC0,

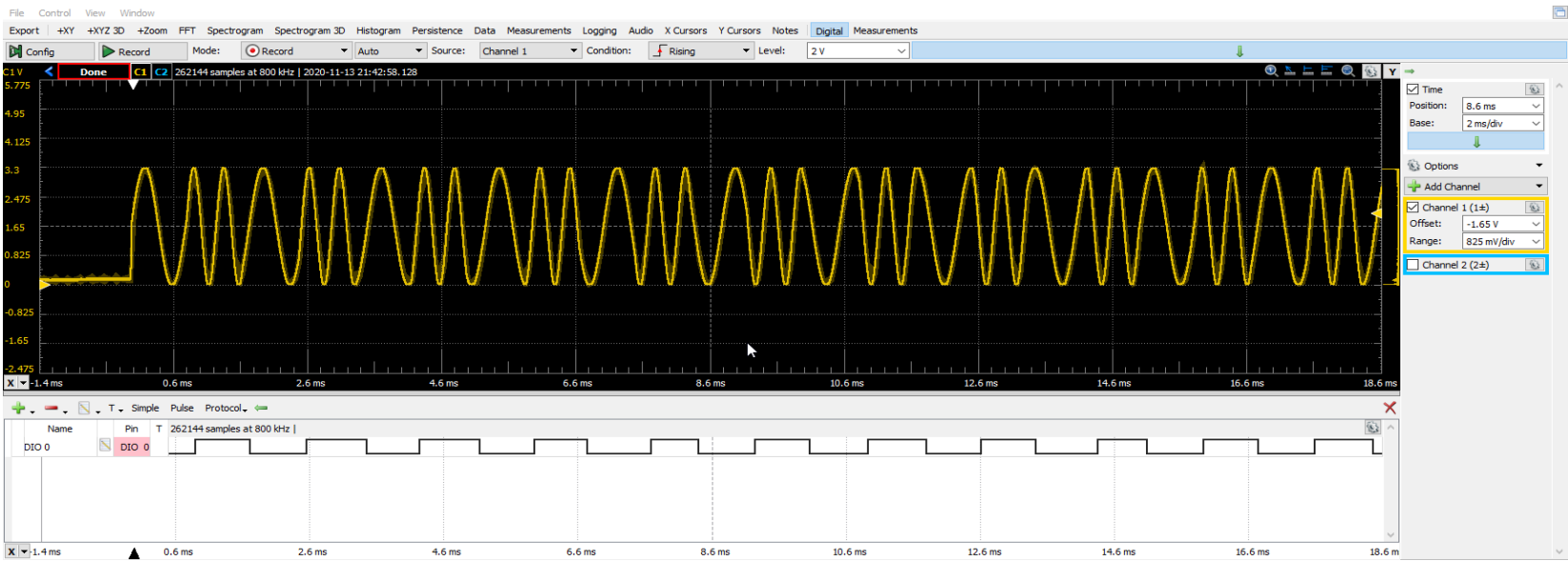
Send

Clear

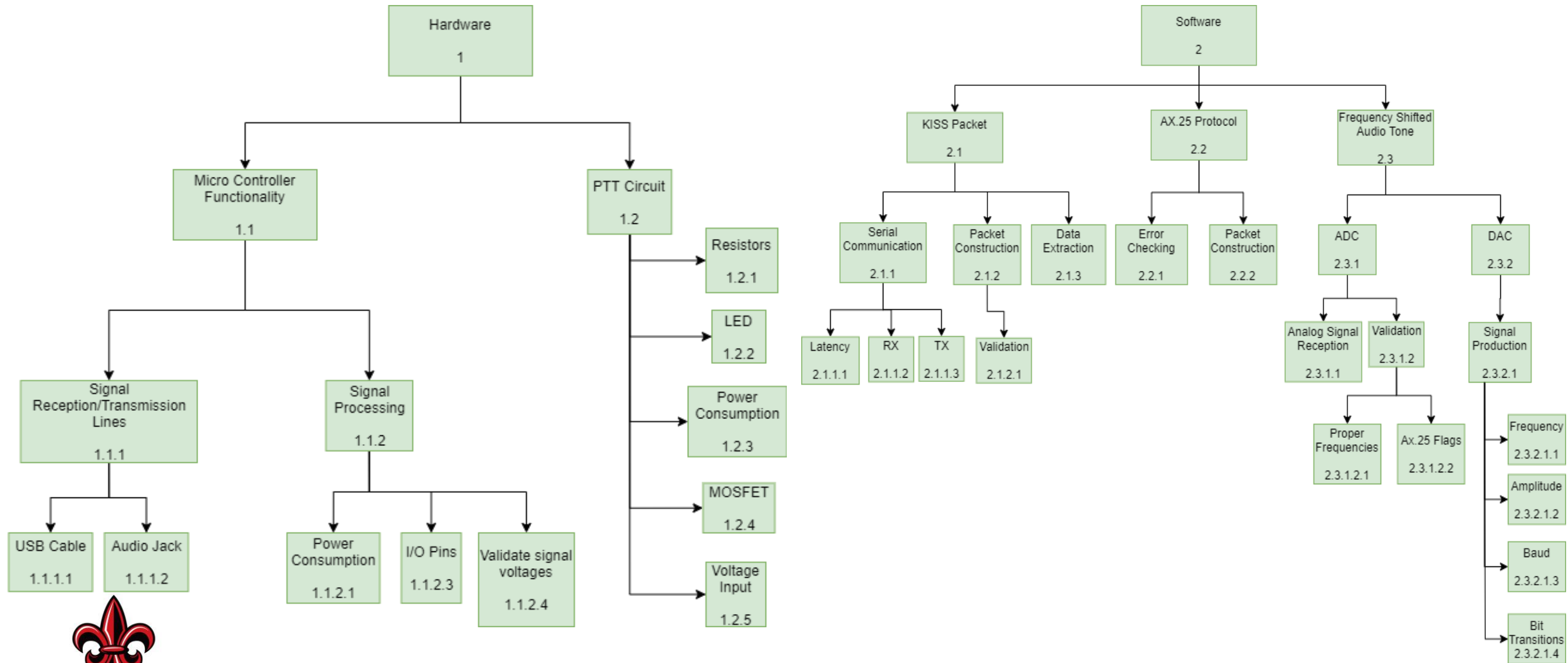
☒ Use AX25

```
Printing AX25_PACKET being sent to radio
Address Field 1 = 1 0 1 0 0 1 1 0
Address Field 2 = 0 0 0 0 0 0 1 0
Address Field 3 = 0 0 0 0 0 0 1 0
Address Field 4 = 0 1 0 1 0 0 0 1
Address Field 5 = 0 0 1 0 0 0 0 1
Address Field 6 = 0 1 1 1 1 0 0 1
Address Field 7 = 0 1 1 0 1 0 0 1
Address Field 8 = 0 1 0 0 0 1 1 1
Address Field 9 = 0 0 0 0 0 0 1 0
Address Field 10 = 0 0 0 1 0 0 0 1
Address Field 11 = 0 1 0 0 1 0 0 1
Address Field 12 = 0 0 1 1 0 1 0 1
Address Field 13 = 0 1 0 0 0 0 0 1
Address Field 14 = 0 0 0 1 0 0 0 1
Address Field extra =
Control Field  = 1 1 0 0 0 0 0 0
PID Field      = 0 0 0 0 1 1 1 1
Info Field     = 0 1 1 1 1 1 0 1 0 0 1 1 1 1 1 0 1 0 0 1 1 1 1 1 0 1 0
FCS Field      = 1 1 1 1 1 0 0 1 1 0 1 0 0 0 0 0 0
```

Proper Frequency Detection

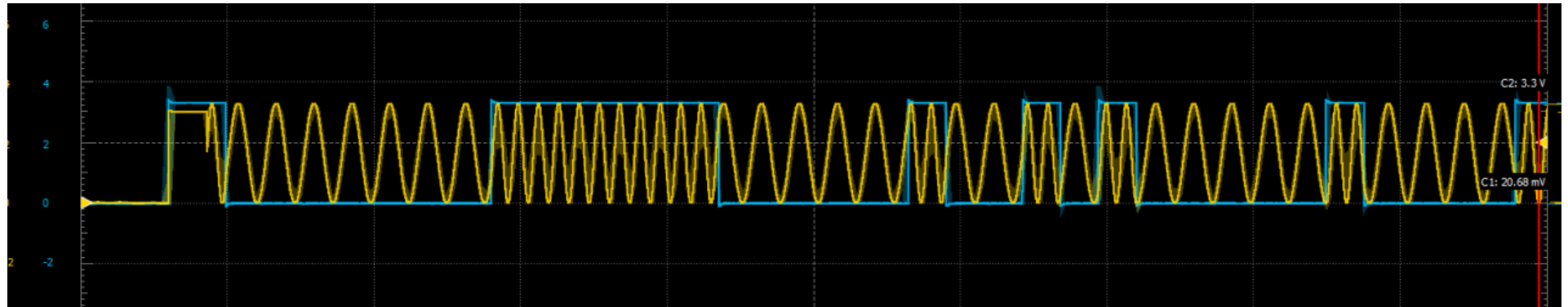


Modular Build, Testing, and Implementation Plan

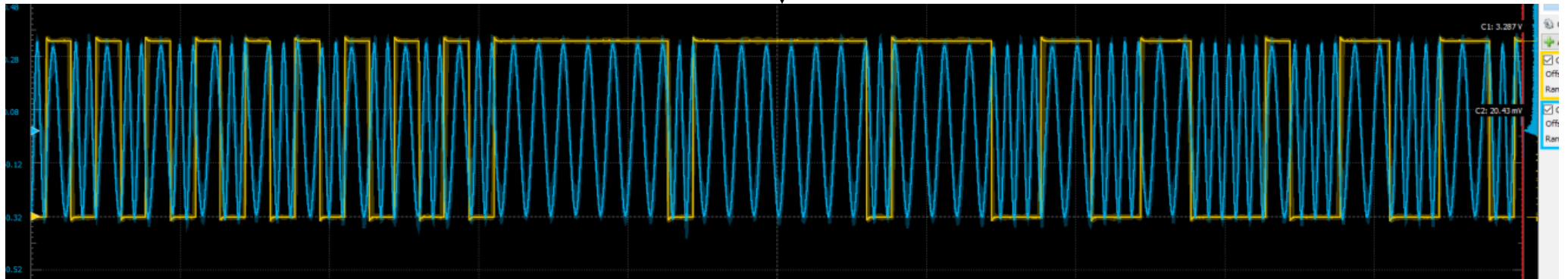


Design Modifications

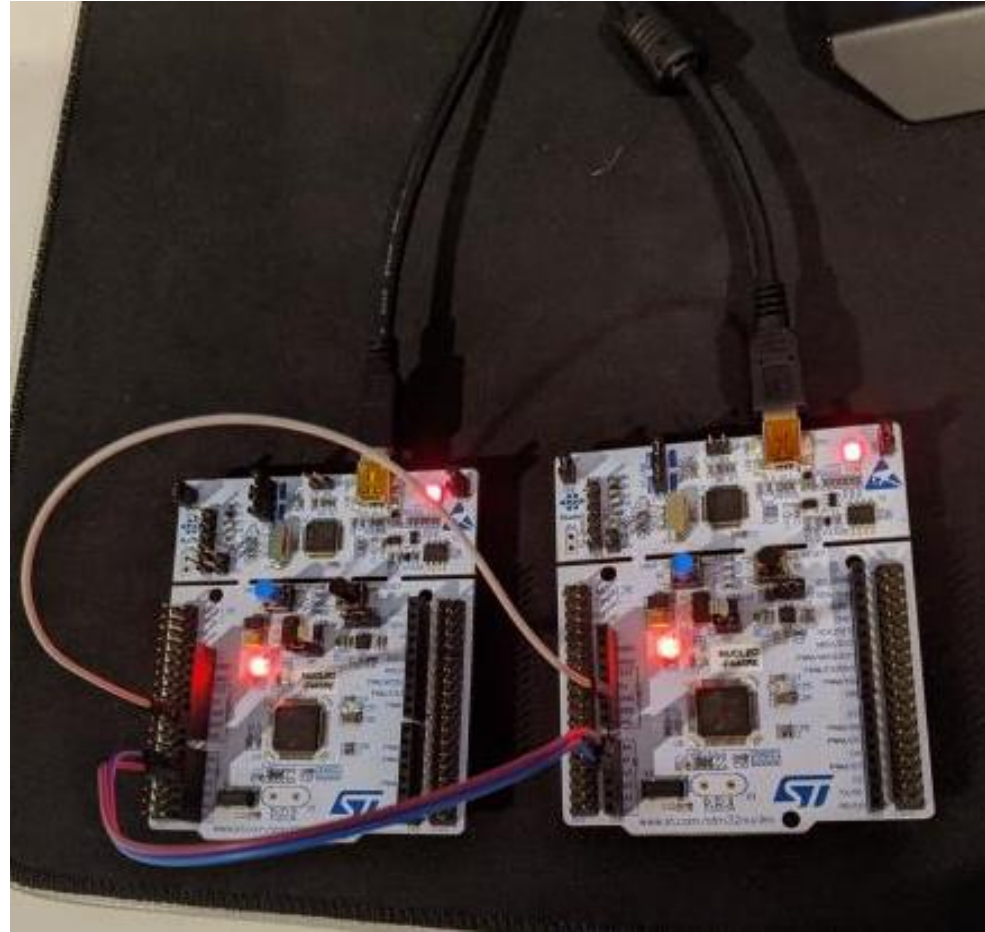
NRZ



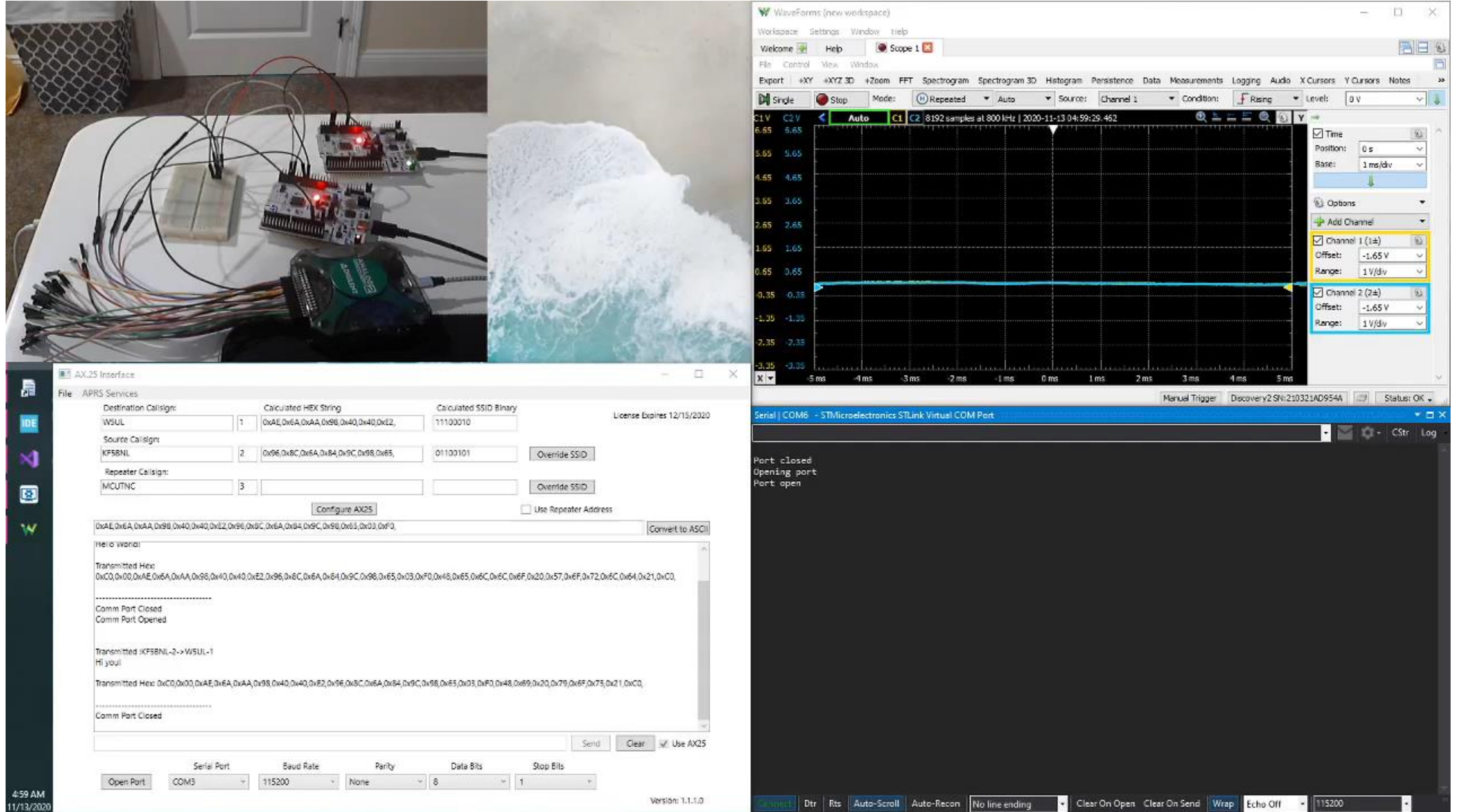
NRZI



As Built Design



Final Demo



Significance

Future Work

- APRS - an amateur radio-based system for real time digital communications of information of immediate value in the local area. Data can include object Global Positioning System (GPS) coordinates, weather station telemetry, text messages, etc..
- Distinguish between different types of packets
- Sending and receiving signals over radio

Lessons Learned

- Document Thoroughly
- Cannot trust just one resource
- There is no easy way
- One change can cause many changes in your design
- Plan ahead for life inconveniences



Acknowledgements

Project Mentors

Nolan Edwards

Rizwan Merchant

James Palmer

Nick Pugh

Special Thanks

Pelican Engineering



Questions?



UNIVERSITY *of*
LOUISIANA
L A F A Y E T T E ®

General Information

[1] What is the Baud Rate?

1200

[2] How are we detecting the frequencies?

Basing it off of change in phases over time where we can obtain the phases by performing inverse sine on amplitudes of the signal using the ADC.

[3] How would a normal user interface with your TNC?

Through a dumb terminal that sends KISS packets over UART.

[4] What does KISS stand for?

Keep It Simple, Stupid



Why is our project relevant?

- Cost affective
 - Basic cost of a micro controller
- Modular
 - Code is easily adaptable to add more functionality
- Easy to replicate and repair
 - Grab code off the GitHub and upload to controller
- Well documented
 - Compared to many other TNCs and info about them, ours is very open source

