MCU TNC Design

Kaleb Leon, Kobe Keopraseuth, David Cain

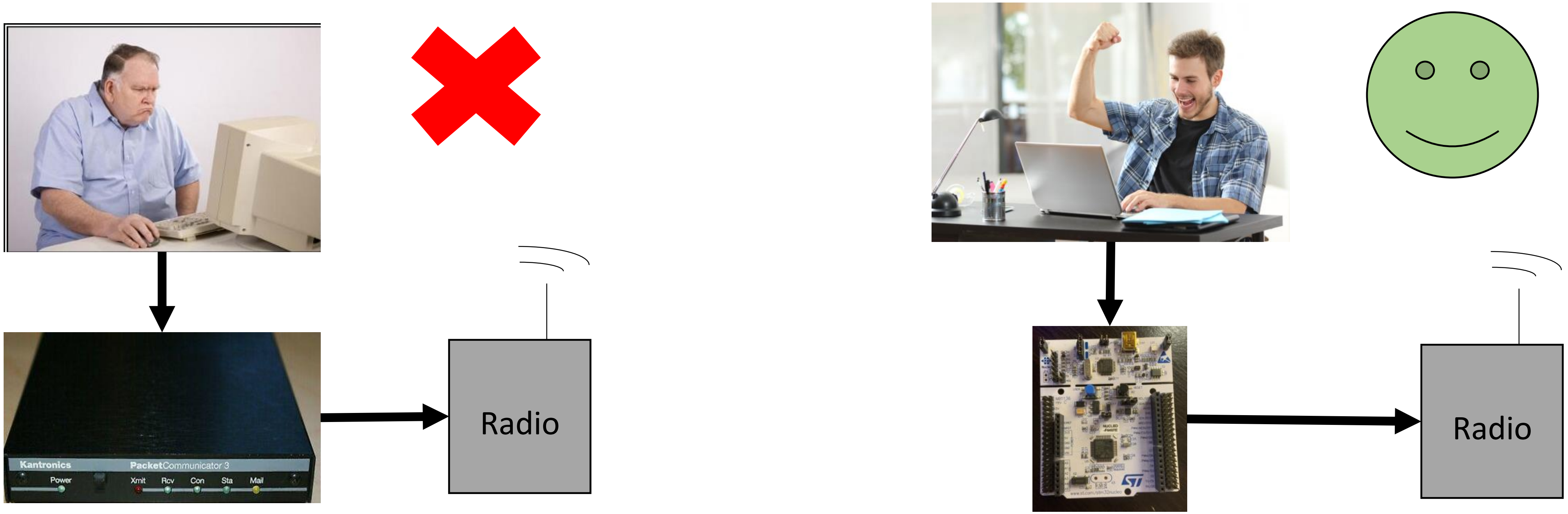
Design 1

FINAL PRESENTATION

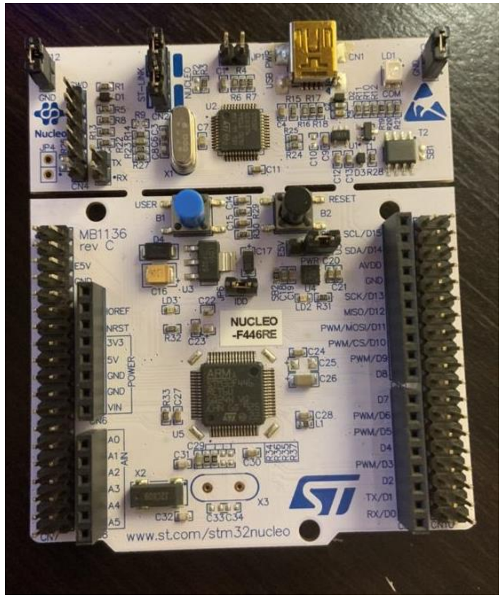
UNIVERSITY of LOUISIANA LAFAYETTE

Project Overview

Design a compact efficient Terminal Node Controller using the STM32 platform



STM32 TNC



Full Setup:
$53

Size: 2''x2''

Logic:

Coded in C without much external hardware of chips

Hardware:

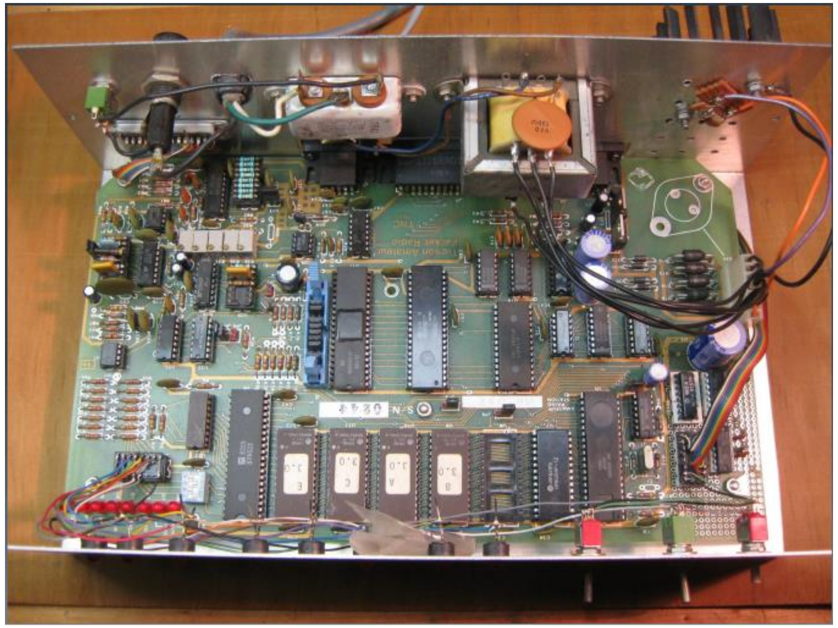
Nucleo STM32 Board

Documentation:

- Easier to document code (also easier to find and reuse)
- Nucleo Board is proprietary hardware and well documented and tested

VS.

TAPR TNC-1



Full Setup:
~ $140

Size: 1'x6''

Logic:

Data formatted mostly by analog logic chips

Hardware:

Many circuits in addition to many integrated chips

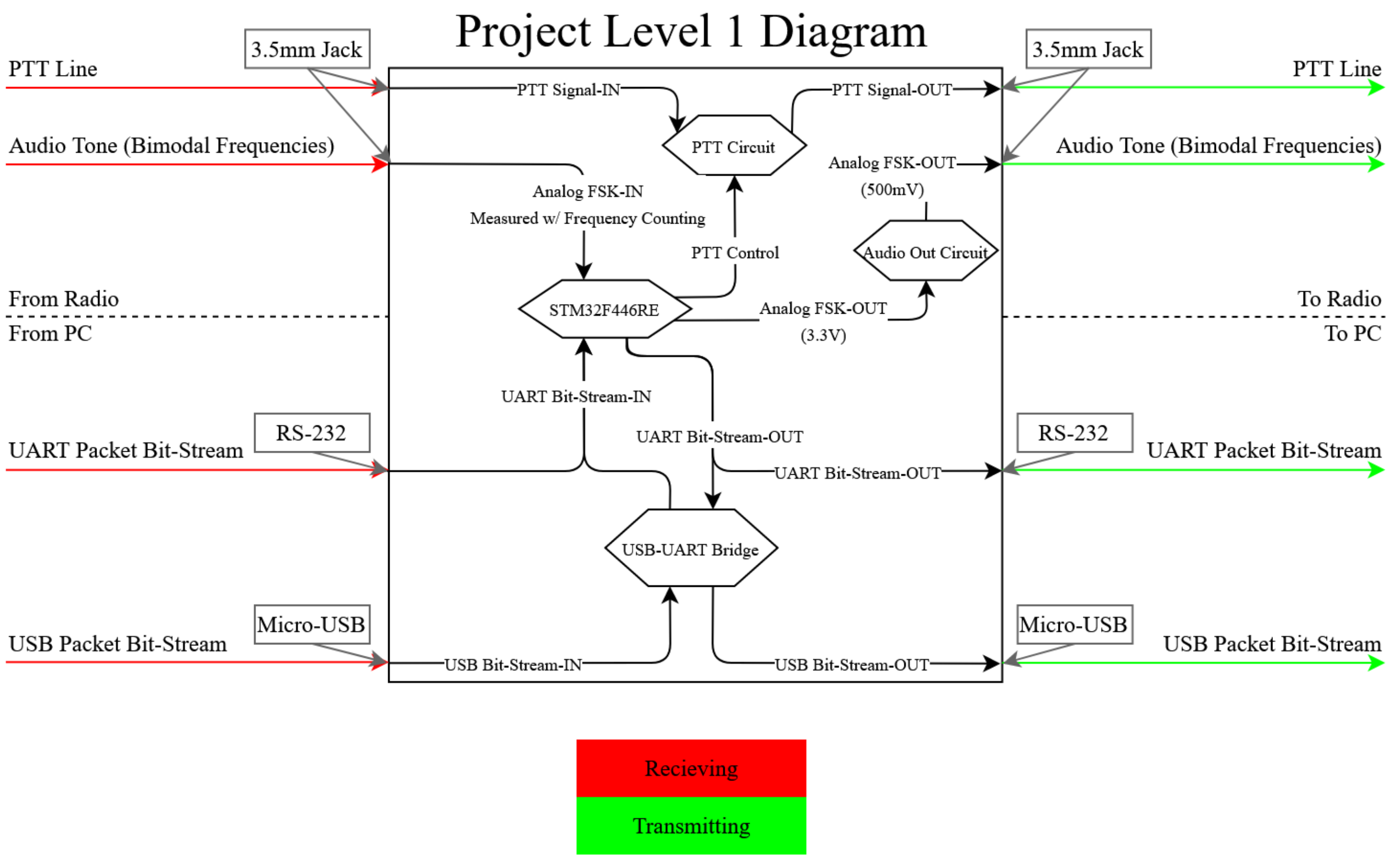
Documentation:

- Not well documented
- hard to find info to repair if something is wrong
- many different designs so only people who make these know how to fix them

Design Process

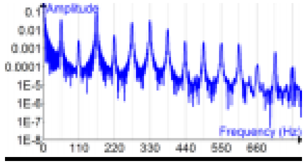
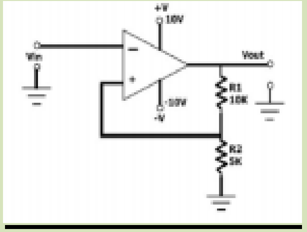
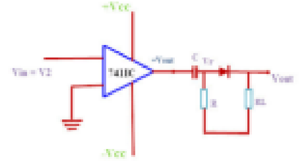
Scope of Work & Functional Block Diagram

To Implement and design a Terminal node Controller using the STM32 platform that is capable receiving and transmitting data packets, serving as a modem between a PC and a radio.

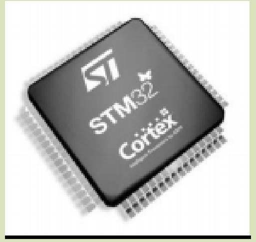
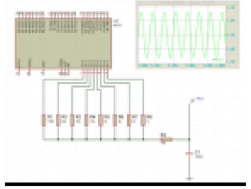
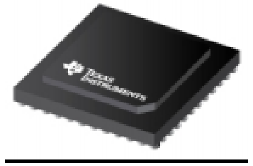


Alternatives and Tradeoffs

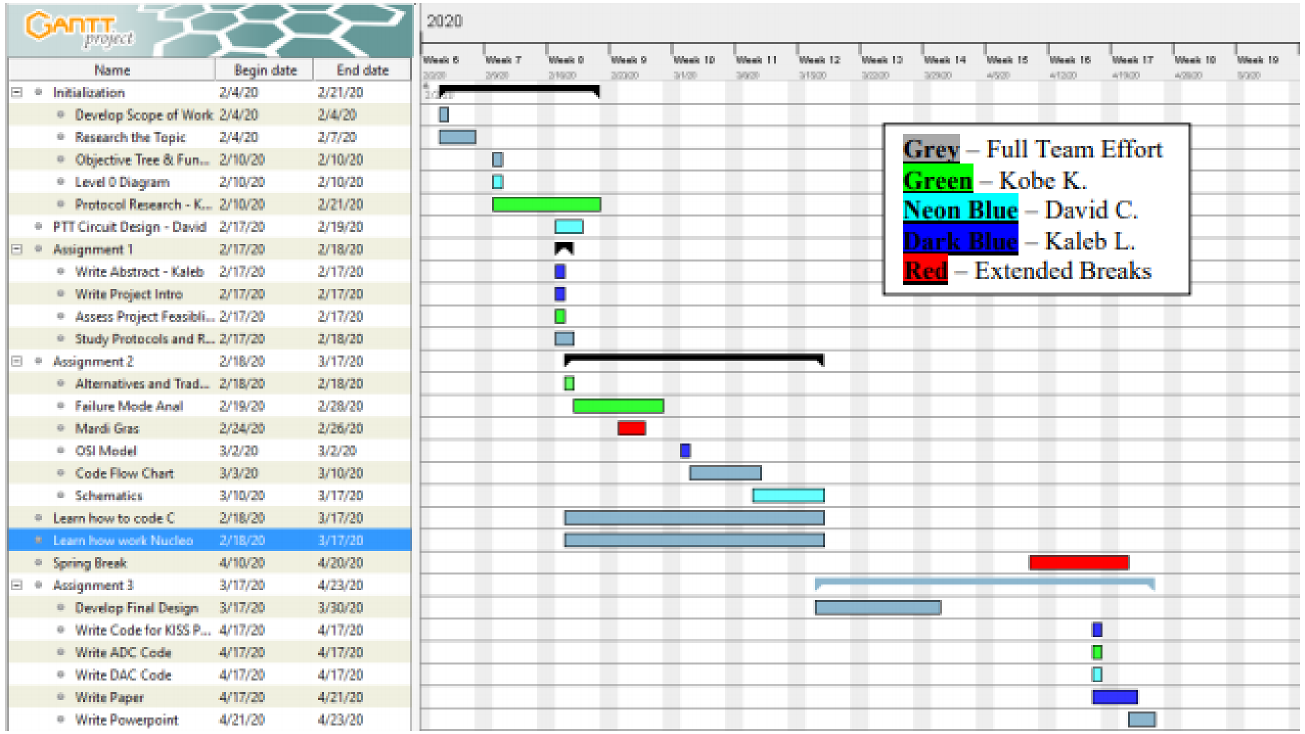
MICROCONTROLLER	**STM32L4433**	**Teensy 4.0**	**Arduino Mega**
			
Description	This microcontroller contains 16 external ADC channels, 1 12-bit ADC, 2 12-bit DAC output channels, an on board RTC, 2 CAN buses, 2 ultra-low-power comparators, CRC calculation unit, and a Schmitt trigger I/O.	This microcontroller contains 40 digital pins (all interrupt capable), 14 analog pins, 2 ADCs on chip, a RTC for date/time, an ARM Cortex-M7 at 600 MHz, 1024K RAM (512K is tightly coupled), and a 2048K Flash (64K reserved for recovery & EEPROM emulation).	This microcontroller contains 16 Analog read pins, 53 Digital pins,and 6 interrupt pins.
Cost	14.90	19.99	18.99
Pros	• Contains CRC calculation unit • Low Cost • Many GPIOs	• Fast clock speed • Has RTC	• Easy to use • Many GPIOs
Cons	• Embedded C programming	• Highest Cost • No CRC calculation unit	• Does not contain RTC • Does not contain DACs or ADCs

Signal Analysis Method	**Fourier Analysis**	**Schmitt trigger**	**Zero Crossing**
			
Description	In our case, this project would use this method to add multiple analog signals of different frequencies to generate digital signals	Simple transistor gate to create the desired active low needed for radio circuits.	Uses comparator to output a toggling logic signal when analog voltage goes to zero.
Pros	It doesn't involve any hardware.	Built in on STM 32 boards, great for filtering out oscillation in digital signal, when noise is in audio/analog signa	Easy implementation with a comparator
Cons	Not efficient because we would need multiple waves of different frequencies to generate a digital signal when we are only working with two different frequencies	If not built on microcontroller we would have to buy a comparator IC	Device may not be capable of supplying needed current for the system. Device should be capable of passing ~20mA

Alternatives and Tradeoffs

Circuit Design	**Built into Controller**	**Resistor Switching Network**	**DAC IC**
			
Description	If the design were to include any of the STM32 line, the MCUs have built in DACs.	Would only consist of using ~4-6 GPIO, connected to different resistor values to represent variable step voltage output. This output would be passed through an LPF to generate a smooth sinusoid.	This would be using a dedicated High-Speed DAC ICs (such as DAC38RF82) that only requires digital input translated to an analog wave for us.
Pros	Similarly, to the dedicated IC, the benefit is there will only be a need to generate digital values.	Simplicity and lack of components needed to generate waveform at low power cost.	Ease of use, only needing to generate digital values that will quickly be converted to sinusoidal waveform.
Cons	Often built in DACs are slow and this may not work within the strict timing constraints of AX.25	Requirement to create code to drive a resistor network meaning more time would be spent on the DAC	With the dedicated silicon, this will raise the price and power consumption of the board.

Feasibility Analysis



Feasibility Analysis

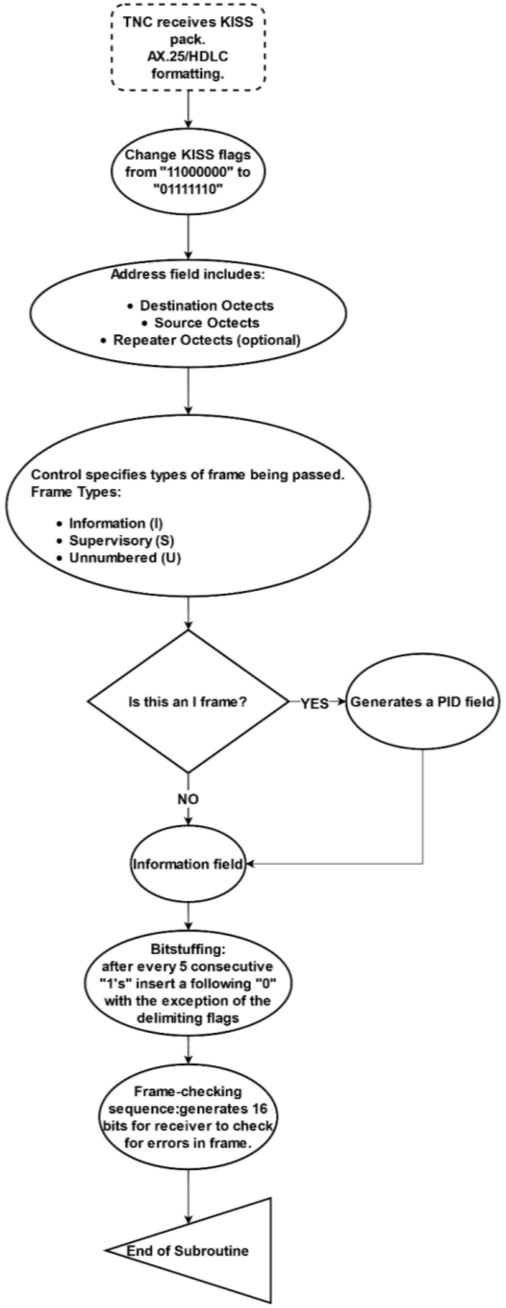
Bill of Materials					
	Quantity	Package	Mfr. Part #/Vendo Link	Mouser Part #	Price
PNP BJT	1	Through-Hole	ZTX951	522-ZTX951	$7.80
STM32F446RE Nucleo board	1	uController	NUCLEO-F446RE	511-NUCLEO-F446RE	$42.00
3.5mm Audio Jack	1	Connector	1699	485-1699	$0.95
10k Resistor	1	Through-Hole	MFR5-10KFI	756-MFR5-10KFI	$0.79
.1uF Capacitor	2	Through-Hole	RDE5C1H104J2K1H03B	81-RDE5C1H104J2K1H3B	$0.25
1.5k Resistor	1	Through-Hole	MFR4-1K5FI	756-MFR4-1K5FI	$0.79
220 Resistor	1	Through-Hole	MFP1-220RJI	756-MFP1-220RJI	$0.79
				Total Cost	$53.37

Software Flow Breakdown

KISS Packet to AX.25

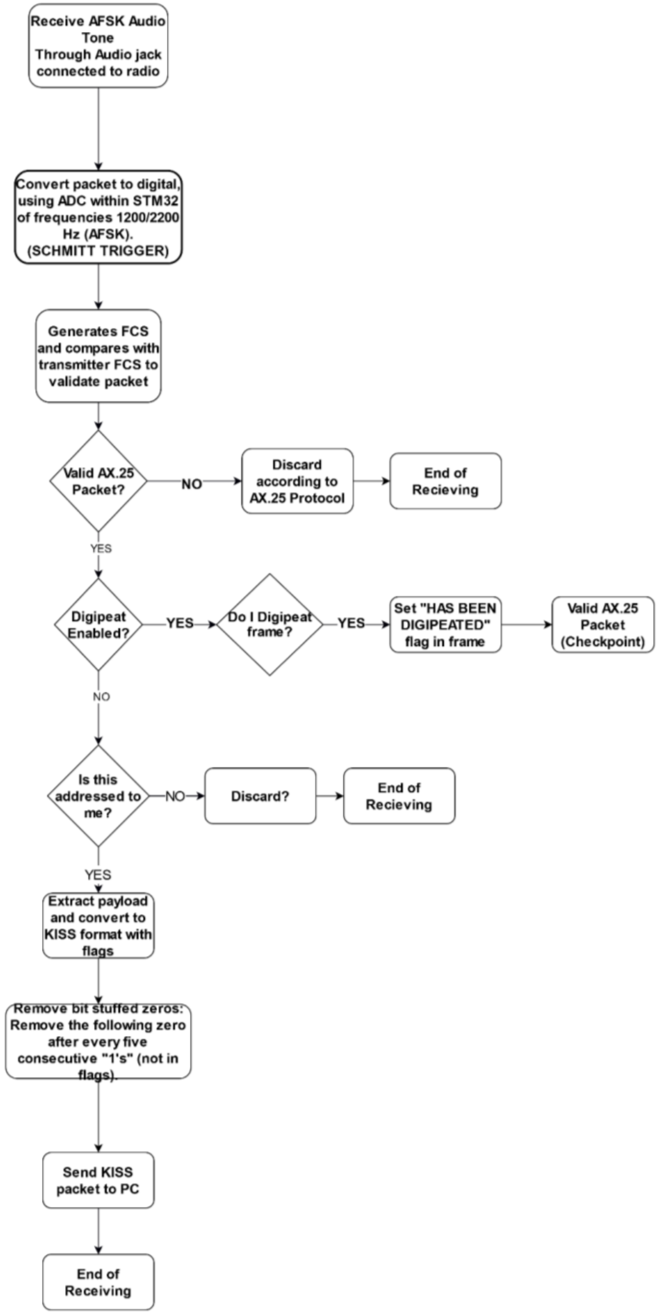
VALIDATION/TESTING:

Create our own packets of which we know what the data section says (example in HEX “Hello world”). This HEX packet is sent to the TNC over the USB. It is then picked apart for its data section. This data section is then translated back into HEX then Ascii and outputted to serial. We monitor this serial to see if it is outputting the correct data.



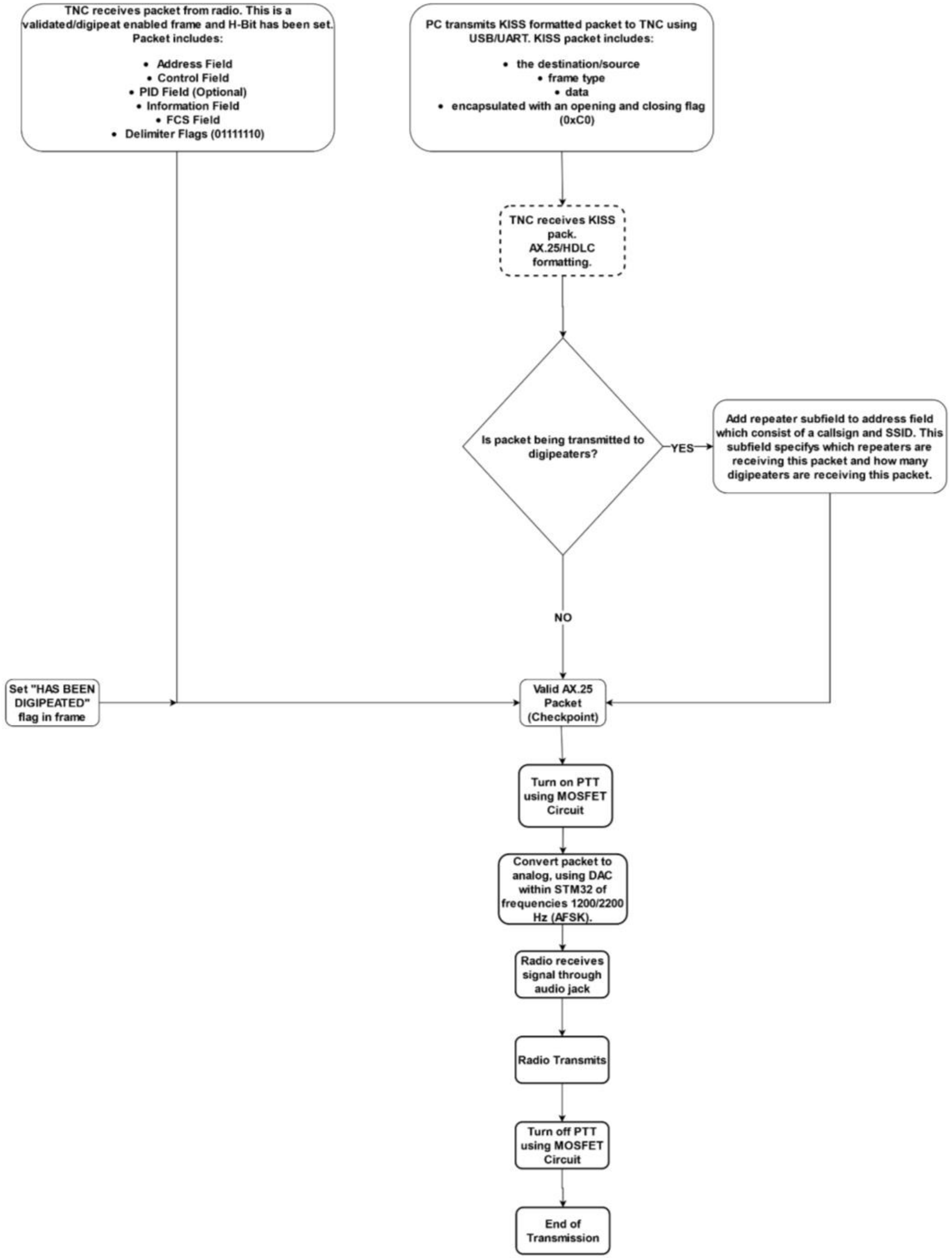
Receiving Mode
Analog to Digital

- In this mode the TNC will be receiving and measuring the frequency of the audio tones from the radio.
- If a tone matches 2200/1200 Hz (5% tolerance) then a 1/0 will be stored in an array, respectively.
- Once the conversion is finished, the STM32 will generate a CRC value to compare with FCS section of packet. If it does not match, then discard packet.
- If it does match, check if digipeat is enabled. If so, turn on PTT and got into transmitting mode.
- If digipeat is not enabled, then check if packet's address matches the TNC's address, if not discard.
- Convert packet stored in array to KISS format.
- Transmit packet to PC through USB or UART.



Transmitting Mode
Digital to Analog

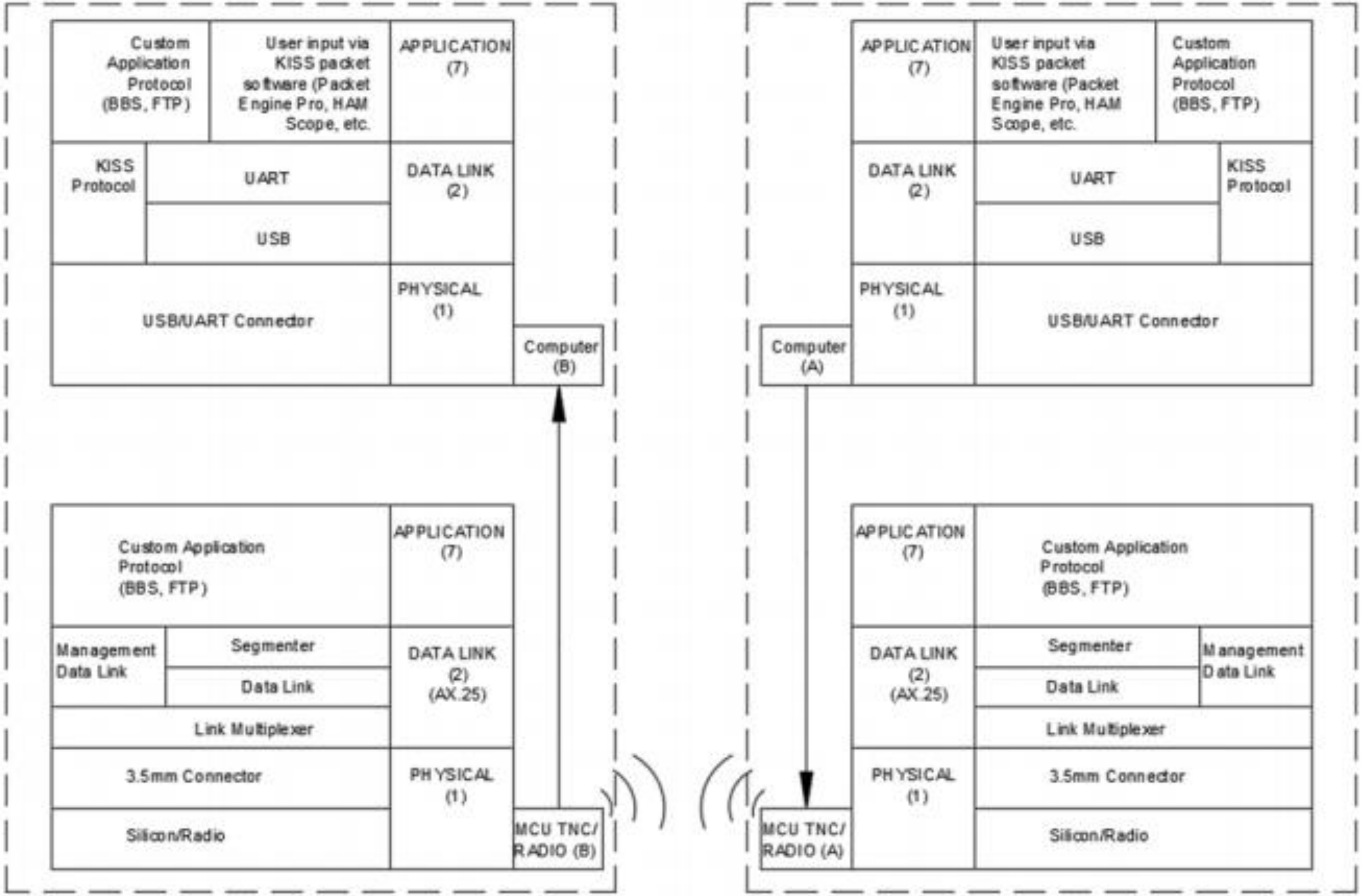
- In this mode the TNC will be producing audio tones to be received by the radio.
- The audio output will be a binary AFSK representation of the packet being sent
- The waveform values for 1200Hz/2200Hz are stored as an array to be converted using the onboard DAC.
- Before audio is being produced, the TNC will pull the PTT line low, indicating to the radio to begin receiving.
- Once the AFSK waveform has been sent to the radio, the PTT line will be released, putting the TNC back in to receiving mode.



Modulation and Demodulation Validation

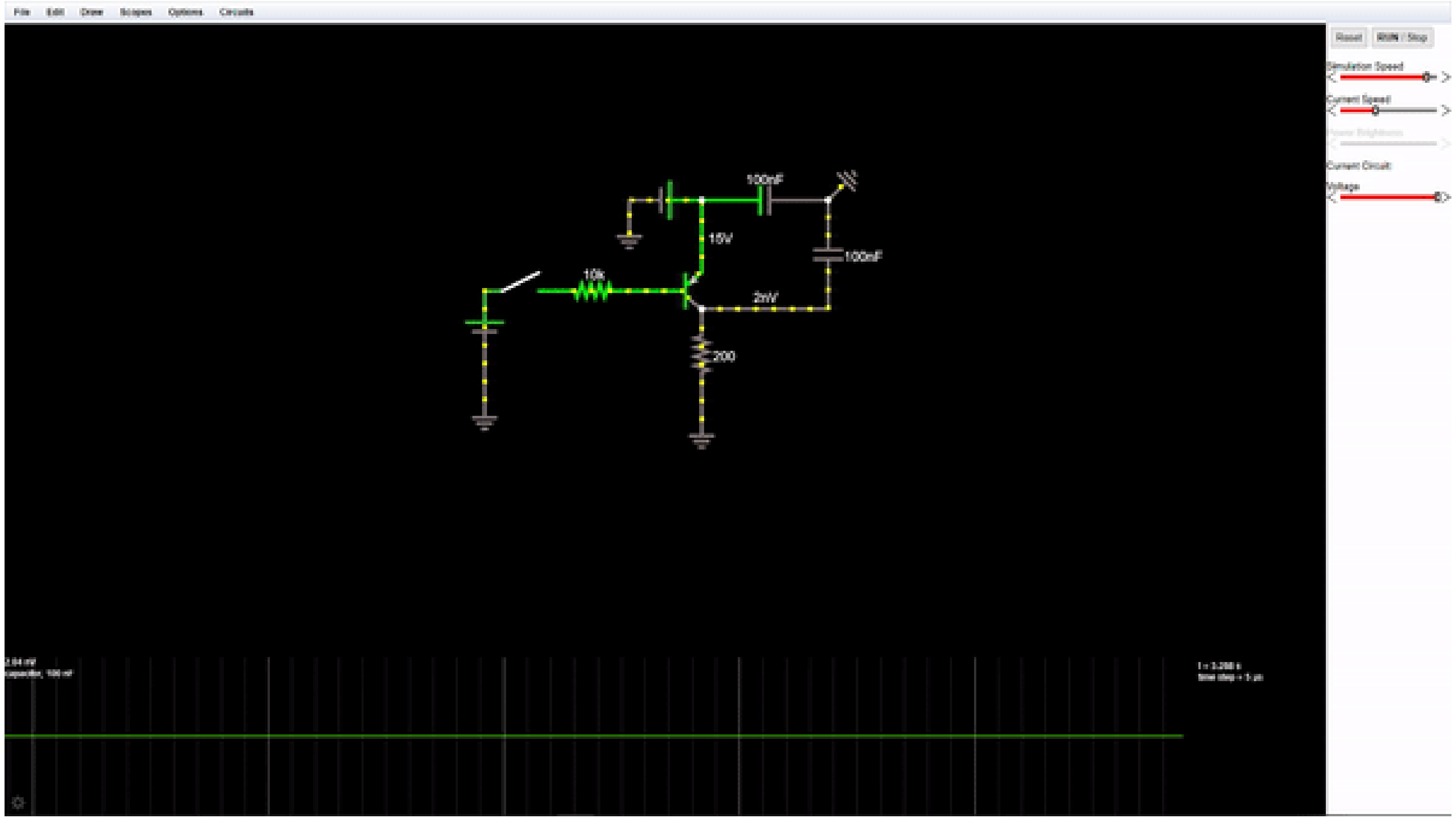
1. A short binary bitstream is hardcoded to be sent to the DAC
2. The microcontroller saves this bit stream to a memory location that is reference and then parsed through as it is sent to the DAC
3. The DAC creates an audio signal that is sent over wire back into the TNC where it is demodulated at the ADC code block
4. If the data returns the same as what was sent out the process is valid.

OSI Layered Communication Model



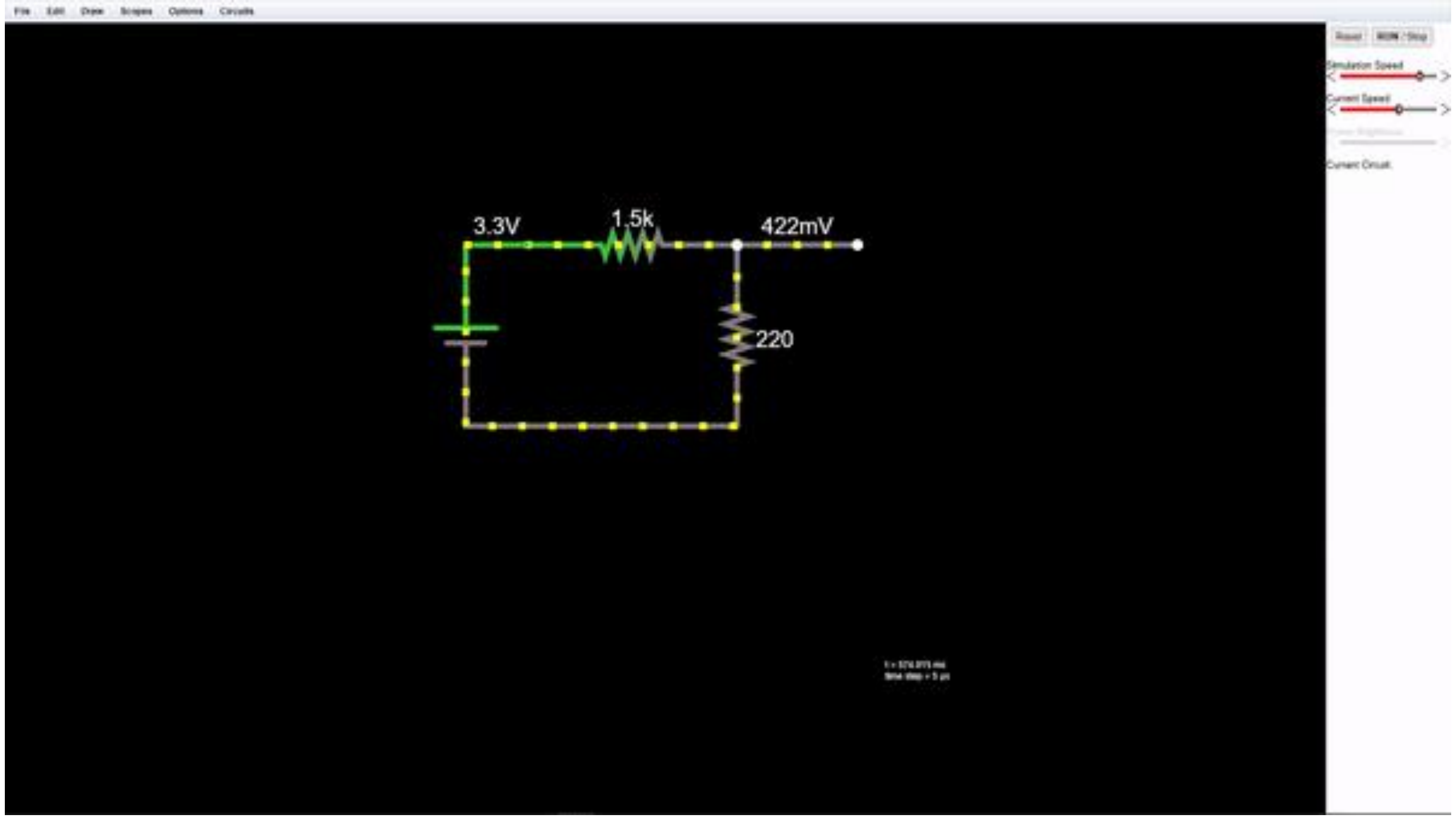
Hardware Validation and Simulation

BJT Push to Talk Circuit

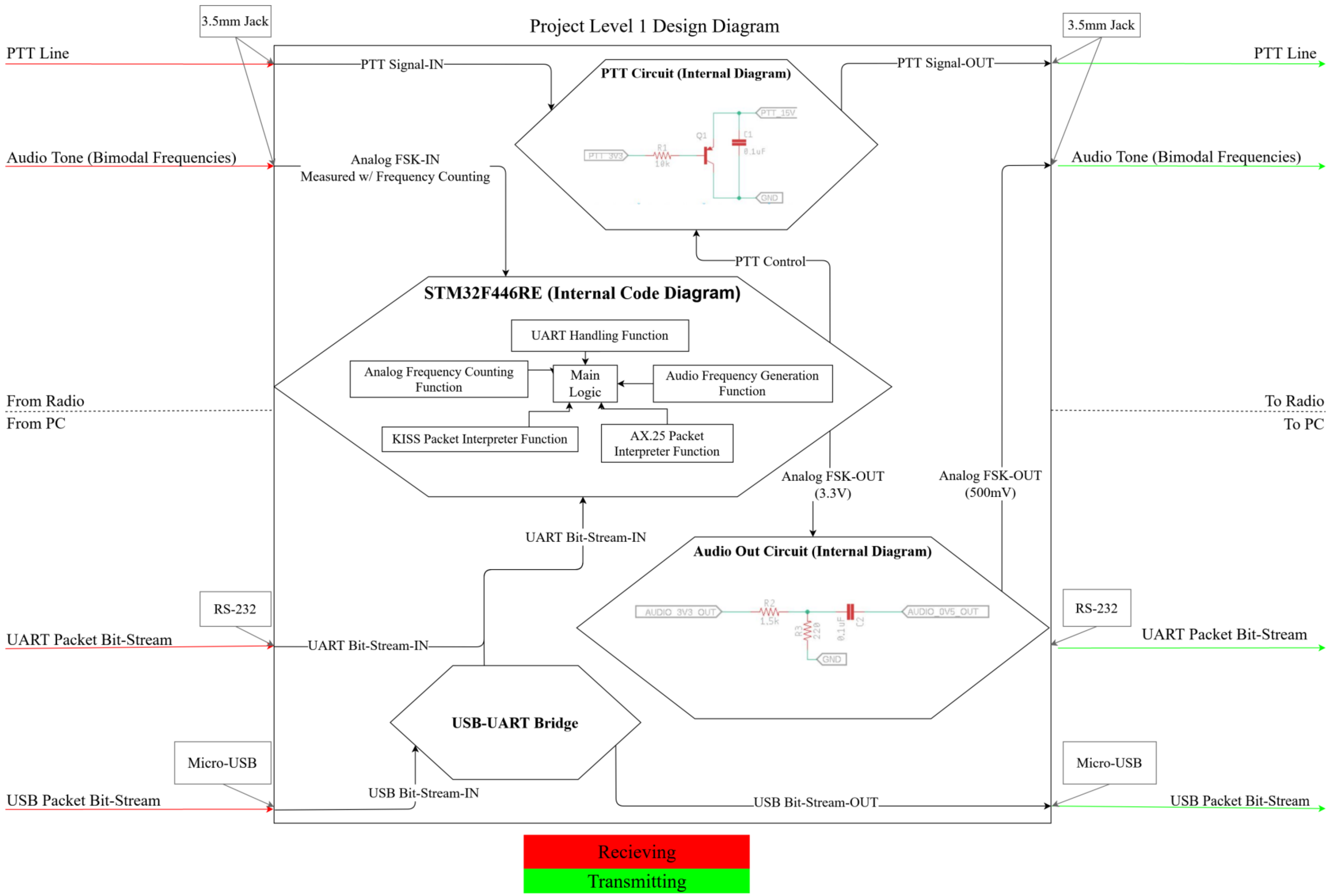


Hardware Validation and Simulation

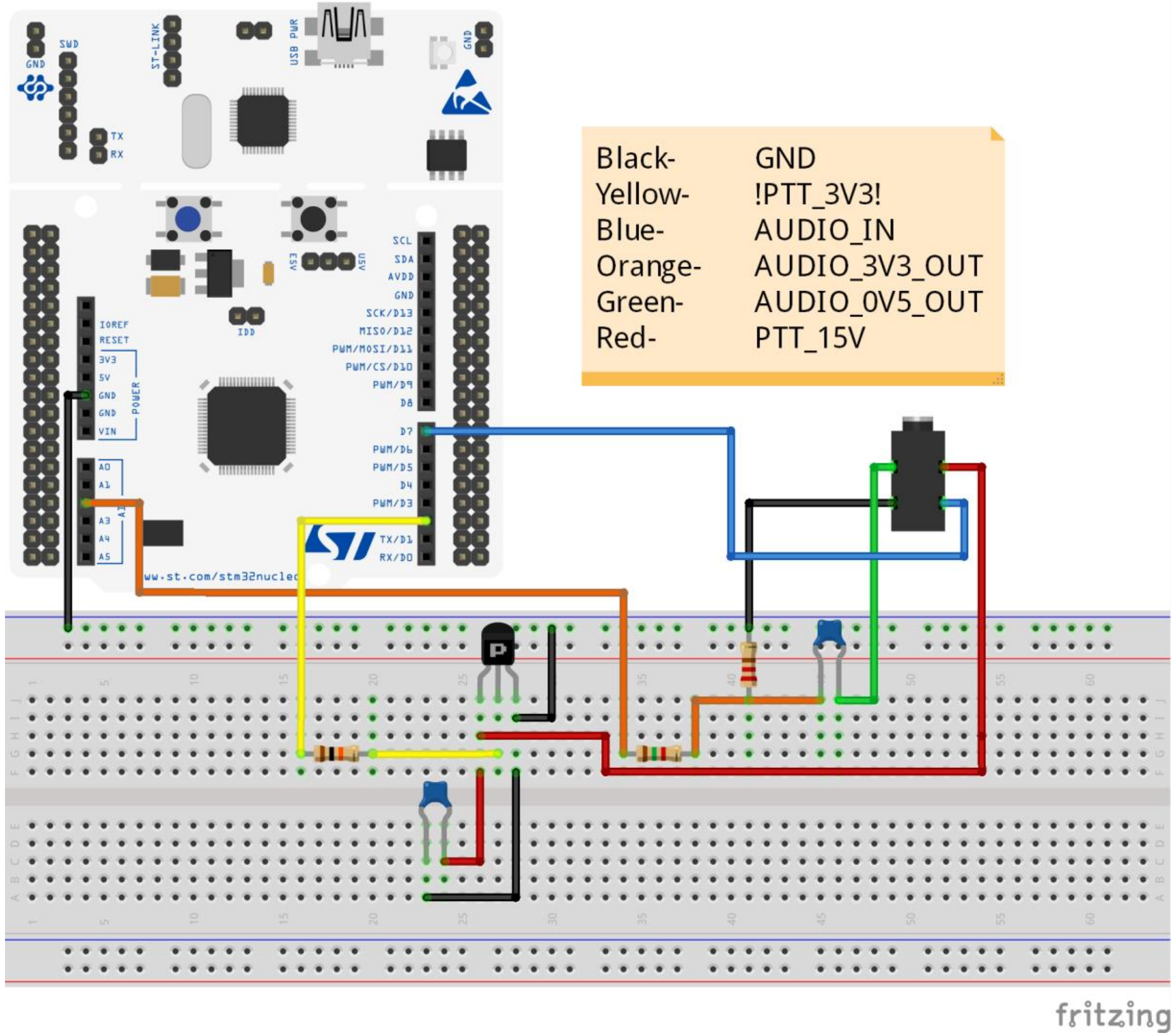
Voltage Divider to Control Voltage Output



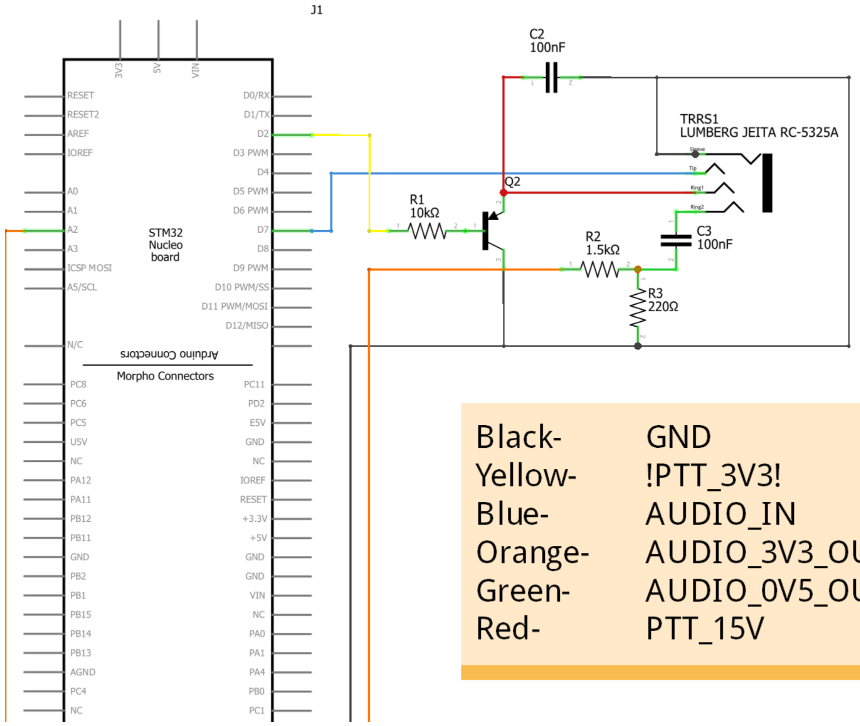
Final Design: Level 1 Design Diagram



Modular Wring Schematic



Black-	GND
Yellow-	!PTT_3V3!
Blue-	AUDIO_IN
Orange-	AUDIO_3V3_OUT
Green-	AUDIO_0V5_OUT
Red-	PTT_15V



Black-	GND
Yellow-	!PTT_3V3!
Blue-	AUDIO_IN
Orange-	AUDIO_3V3_OUT
Green-	AUDIO_0V5_OUT
Red-	PTT_15V

Acknowledgements

Project Mentors:

Mr. Nolan Edwards

Mr. Rizwan Merchant

Mr. Nick Pugh

Mr. James Palmer

Special Thanks:

CAPE Team

Pelican Engineering

Future Plans

- Design PCB board for layout, for efficiency in space and power distribution.
- Possibly wireless communication to other radios.
- Design higher level software structures, such as APRS.
- Design casing to protect components.

Lessons Learned

- How to write a scholarly paper.
- How to write in an embedded C environment.
- A successful design requires considerations of extreme operation conditions.

Questions?