Project 2: Bluetooth and IR

Thomas Nguyen
016238935
Reed Ellison
015290260

CECS 447
Microcontroller III
05/08/2019

Table of Contents

Table of Figures

Introduction

In this project, our team created a wirelessly controlled wheeled robot which utilized UART/Bluetooth communication to receive its commands from a computer or phone. In addition, we added an IR Signal transmitter station that was used to send out a modulated IR signal with certain requirements for the transmission. Our team then created an IR signal receiver station that was used to receive the modulated IR signal from our transmitter. After receiving the signal, demodulating it, and decoding the signal, our team output a unique design to our LCD, that was connected to our receiver station. At completion, our project allows the wheeled robot to move wirelessly to a receiver station and then output a command that is an IR modulated signal which is then demodulated by the receiver station and decoded to produce a display on the LCD.

Operation

Steps to Build

1. Initialize/Setup UART communication from TM4C Tiva C Launchpad to Bluetooth Module. Initialize a UART module to transfer data from a laptop/terminal to the TM4C to allow for configuration of the Launchpad. Initialize a second UART module to handle the communication between the Bluetooth module and the TM4C. Before normal transmission can be achieved between TM4C and the Bluetooth module, the module must be sent the correct commands to initialize it including name, password, and baud rate(start/stop bit setup as well).
2. Setup Motors for Wheeled Robot. Connect Port Pins of TM4C to H-Bridge and the output of the H-Bridge goes to the motor leads. The motors can now be turned on/off and change direction according to the values supplied to the H-Bridge inputs. Using the previously setup Bluetooth, the motors can be set to go straight, backwards, or turn left or right.
3. Initialize IR Transmitter. Connect IR Transmitter Module to desired output port. Utilizing PWM for the active high states of the IR signal at the correct frequency(40kHz). Additionally use timers to ensure for each kind of signal(start bit, high bit, low bit) that the signal is oscillating and off for the right times to meet the communication requirements.
4. Initialize IR Receiver and LCD. Connect Receiver to Receiver TM4C and LCD. Utilize timers to count the number of positive edges and when they occur to decode the IR demodulated Signal.

How to Test

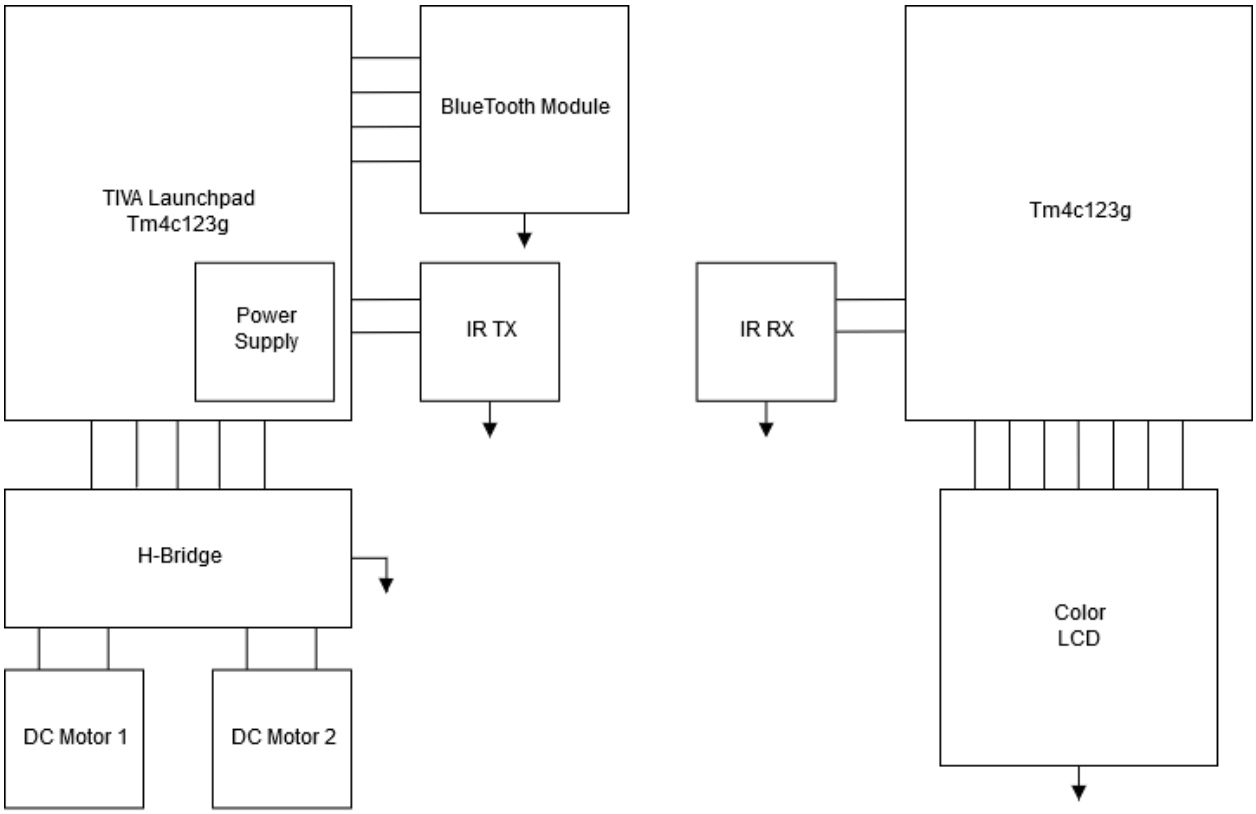
1. With hardware connected, connect a laptop/phone to the Bluetooth module of the wheeled robot. With the module in transmission mode, a bluetooth enable laptop/phone can send the following signals to send the associated commands:

- 'W': Forward
- 'S': Reverse
- 'A': Left Turn
- 'D': Right Turn
- 'Q': Stop Motion
- 'C': Change Device address
- '0': Command 0
- '1': Command 1
- '2': Command 2
- '3': Command 3
- '4': Command 4

2. By sending the corresponding commands by Bluetooth, the W,A,S,D,Q will change the motors allowing the wheeled robot to move. The C command will change the address that the IR command will be destined for. For this project, there are four addresses for different Receiver Stations, this address is what is changed by this command. 0-4 Send different command signals that will then have different LCD outputs on the receiver station.

Hardware

Figure 1:Block Diagram of Bluetooth Transmit Car and IR Receive Station

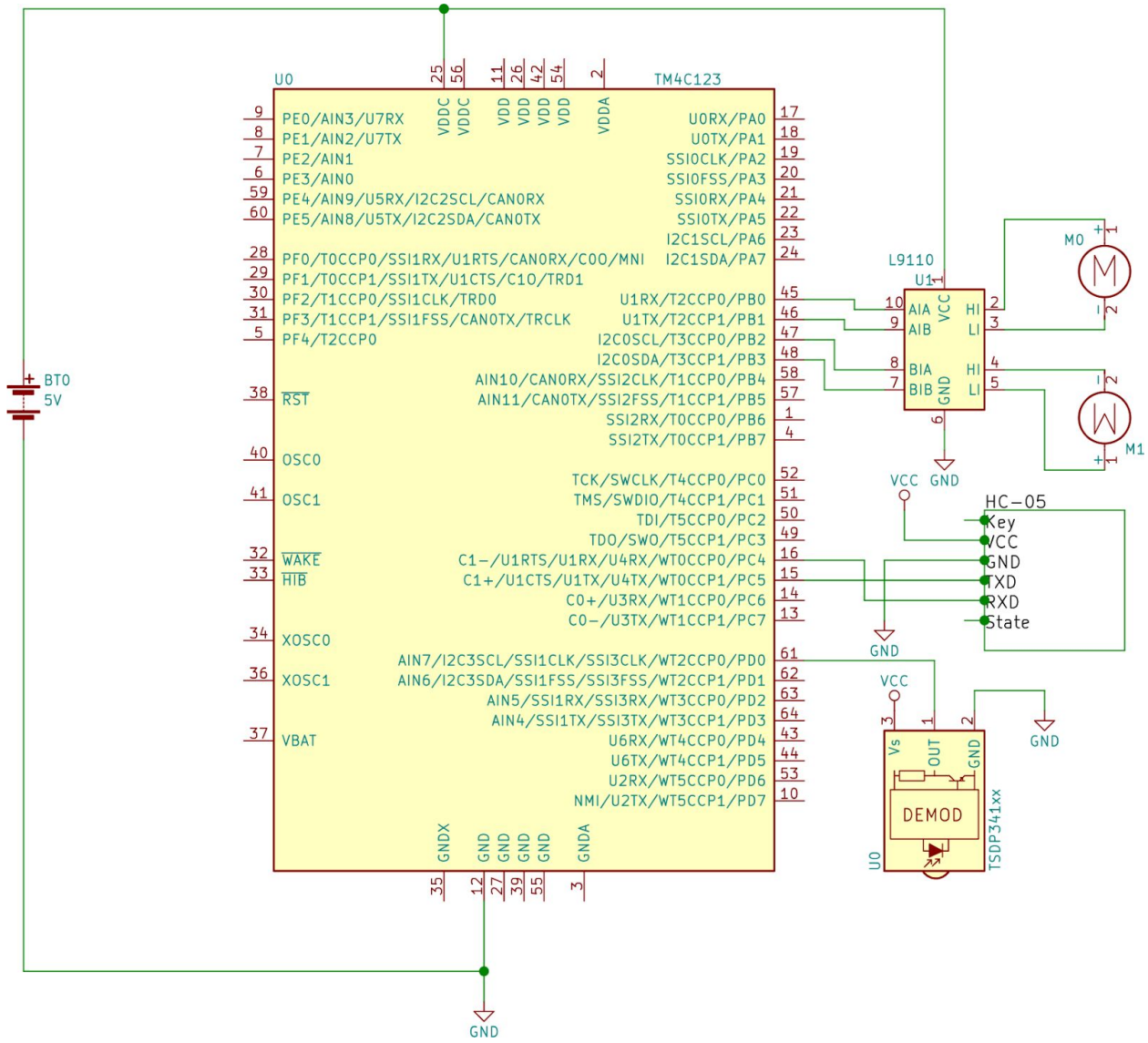


Components

- Wheeled Robot/Transmitter Station
  - Motors/Chassis
  - L9110S H-Bridge
  - Jumper Wires
  - TM4C Tiva C Launchpad
  - IR Transmitter Module
  - HC-05 Bluetooth Module

- Receiver Station
    - TM4C
    - Color LCD
    - IR Receiver Module

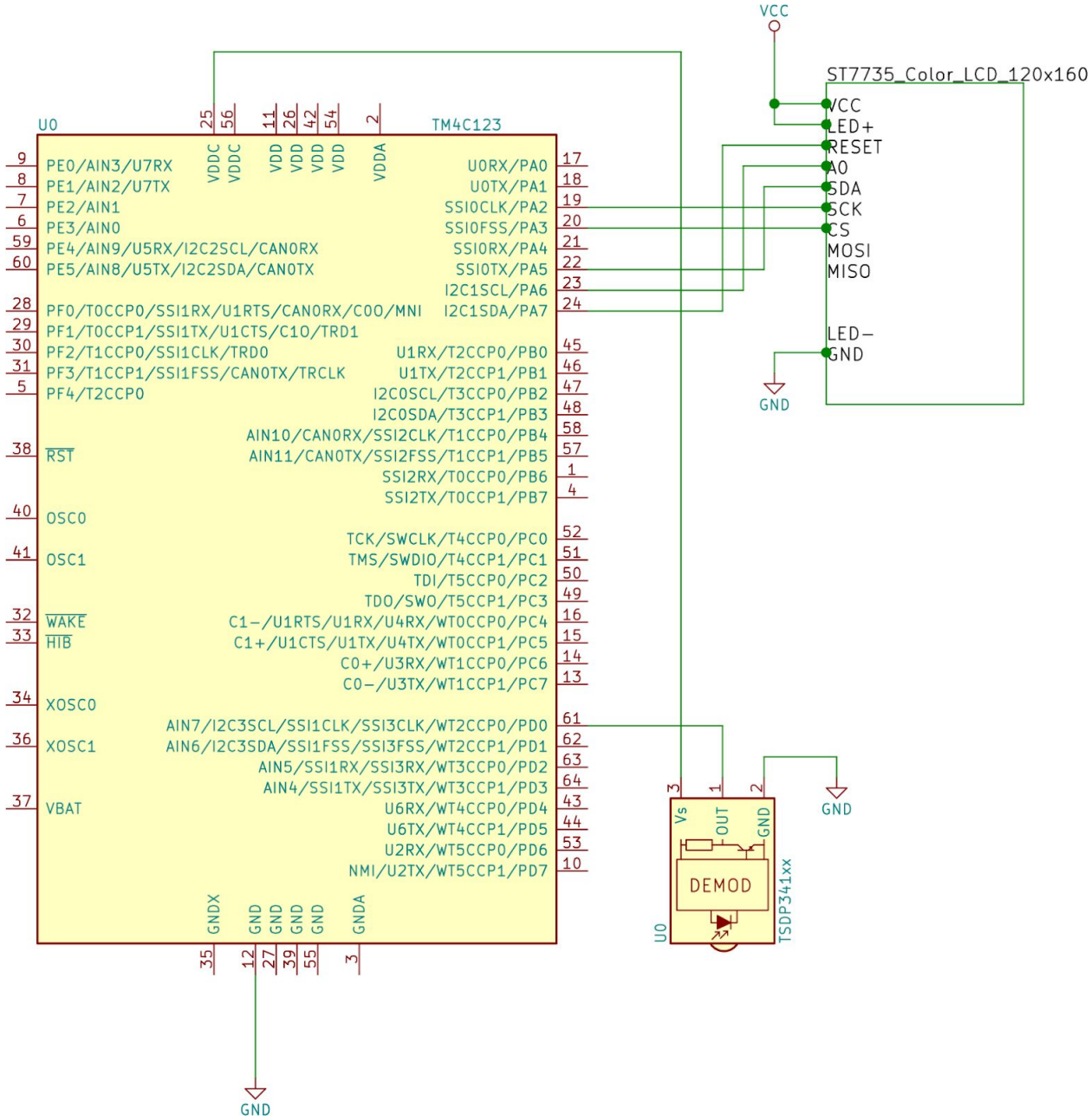
Figure 2: Bluetooth IR Transmitter Car Schematic



The bluetooth IR transmitter car uses an L9110 H-bridge to control the direction and speed of each individual DC motor, however because no PWM is used to control the speed of the motors, the motors are either on or off. The IR transmitter outputs signals based on the PWM pulse cycle at 40KHz. The pulse is either on or off depending on the specific bits needed to be sent.

The bluetooth HC-05 module is used to communicate to the microcontroller through UART1. It is configured beforehand. The bluetooth gets its signals from any bluetooth terminal device. The signals it responds to in order to control the robot car is listed in the Operations section.

Figure 3: IR Receive Station Schematic



The IR receive station receives the transmitted signals from the bluetooth car. It checks for a start bit before it starts catching the rest of the data. Two bits are the device address number. The specifications for this station is device address 0. Then

the last 4 bits are the command bits from 0 to 4. The LCD displays an image and animation depending on the command. An invalid command will result in the LCD displaying a text saying it's invalid. When no signal are detecting, the LCD displays a text saying it's waiting for a signal. The IR receiver is also a demodulator.

Software

The software design was split into two categories as there were two separate embedded systems.

Bluetooth IR Transmitter Robot Car

For the wheeled robot/IR Signal transmitter module, motor control via GPIO PORT B was initialized. UART 0 and UART 1 were initialized for communication between terminal(laptop) and TM4C and Bluetooth module to laptop. Following the initializations, a if statement allowed the program to enter a state to initialize a bluetooth connection which is only entered for setting up a new device to communicate with the bluetooth module. Following the optional setup, the main loop was entered where in a switch statement was consistently entered that allowed for the reading of the bluetooth input to select the commands given previously. The motor driving commands only required changes to output of Port B to change the H-Bridge values. The commands requiring IR modulation would enter, according to the command and current address value, send out the bits by calling each function(ie sending a startBit, then a combination of low(logic 0) and high(logic 1) bits that were determined by given requirements for IR communication. In each bit function, a PWM signal would generate the 40Khz frequency and timers would be used to ensure that the PWM signal was active for the amount of time it needed to be for the bit and then low for the other time ensuring the output signal was accurate to the requirements of the project.

IR Receive Station

The IR Receive Station was handled differently as it needed to receive and decode the demodulated IR Signal. The receiver first looks for the start bit through the use of a GPIO interrupt. After the interrupt is triggered by the input from the IR receiver on negative edge, the GPIO interrupt is then disabled and a flag indicating the start high bit has been receive starts. Next the timer interrupt goes through and counts the time it is accessed of the start low bit and then counts through all of the address and command bits. After the counter hits about 420, the checking is done and all the bits have been gathered into an array. The bits stored in the array are checked to see what command was received. If device bits were not 0, then the program doesn't care and just goes straight to an else statement. If it was a 0 then it checks the command bits to display 5 animations or an invalid statement.

Figure 4: Bluetooth Transmit Car Software Flowchart

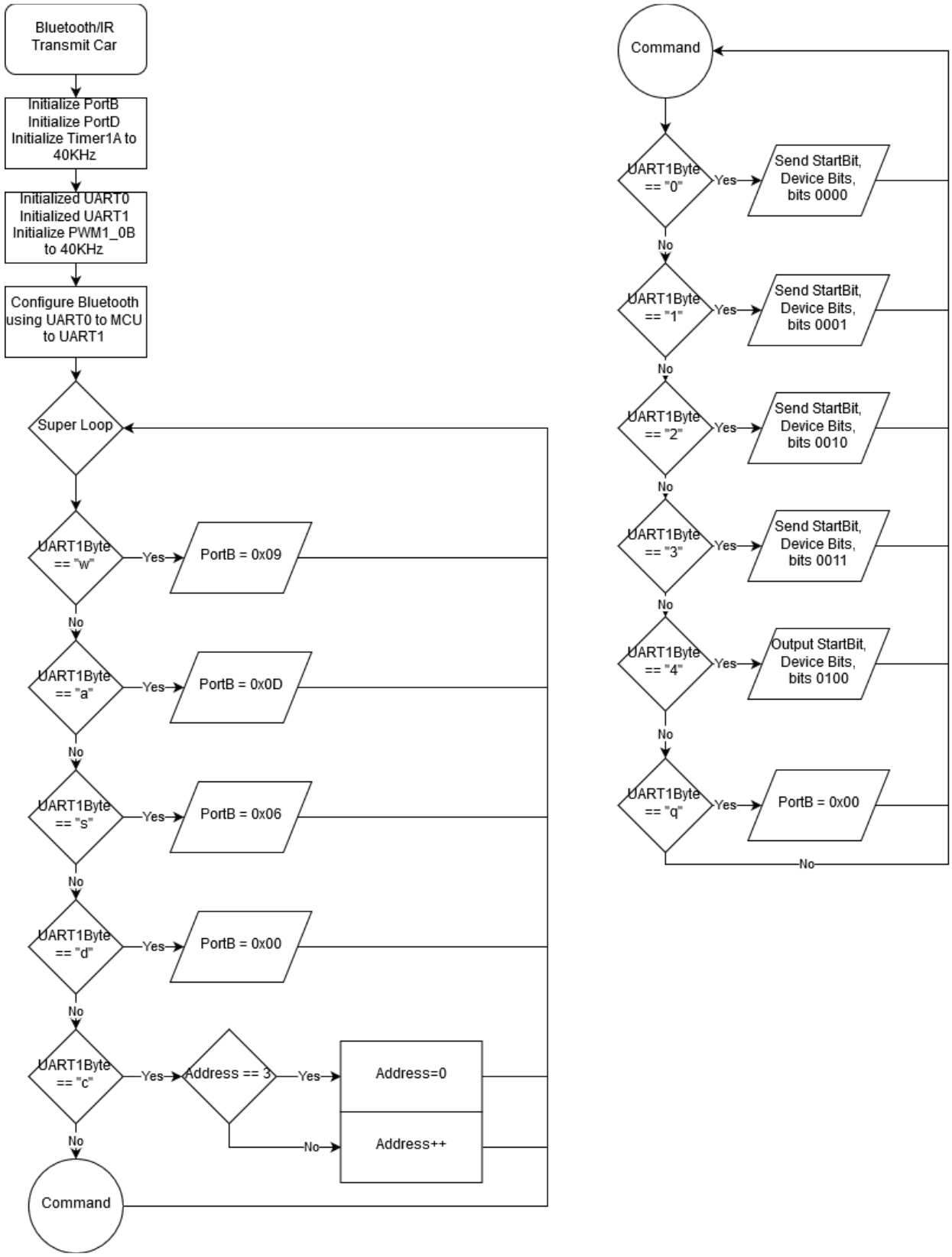
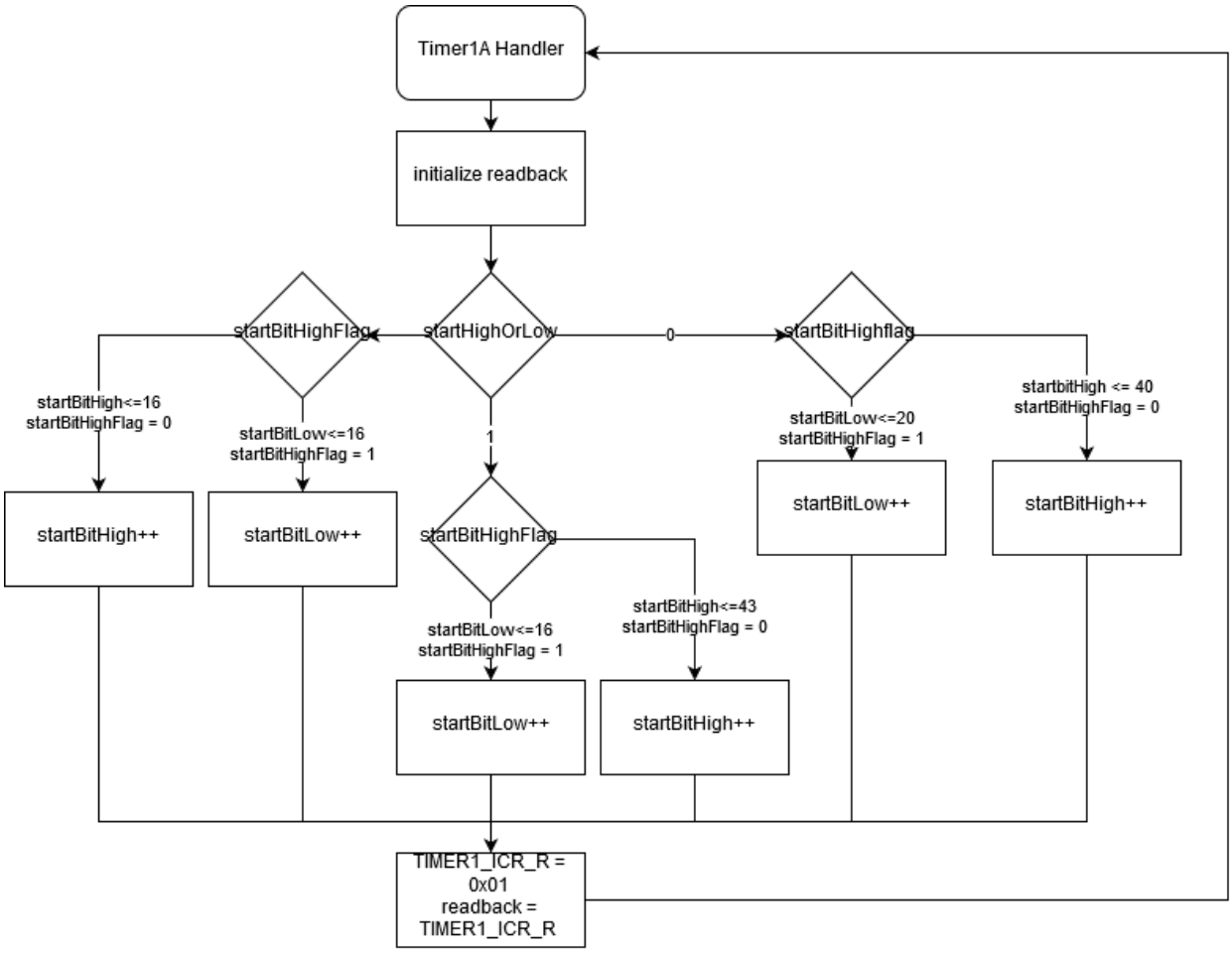


Figure 5: Bluetooth Transmit Car Part 2



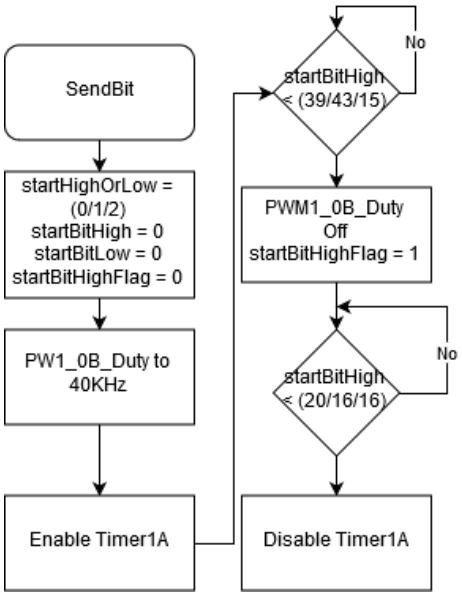


Figure 6: IR Receive Station

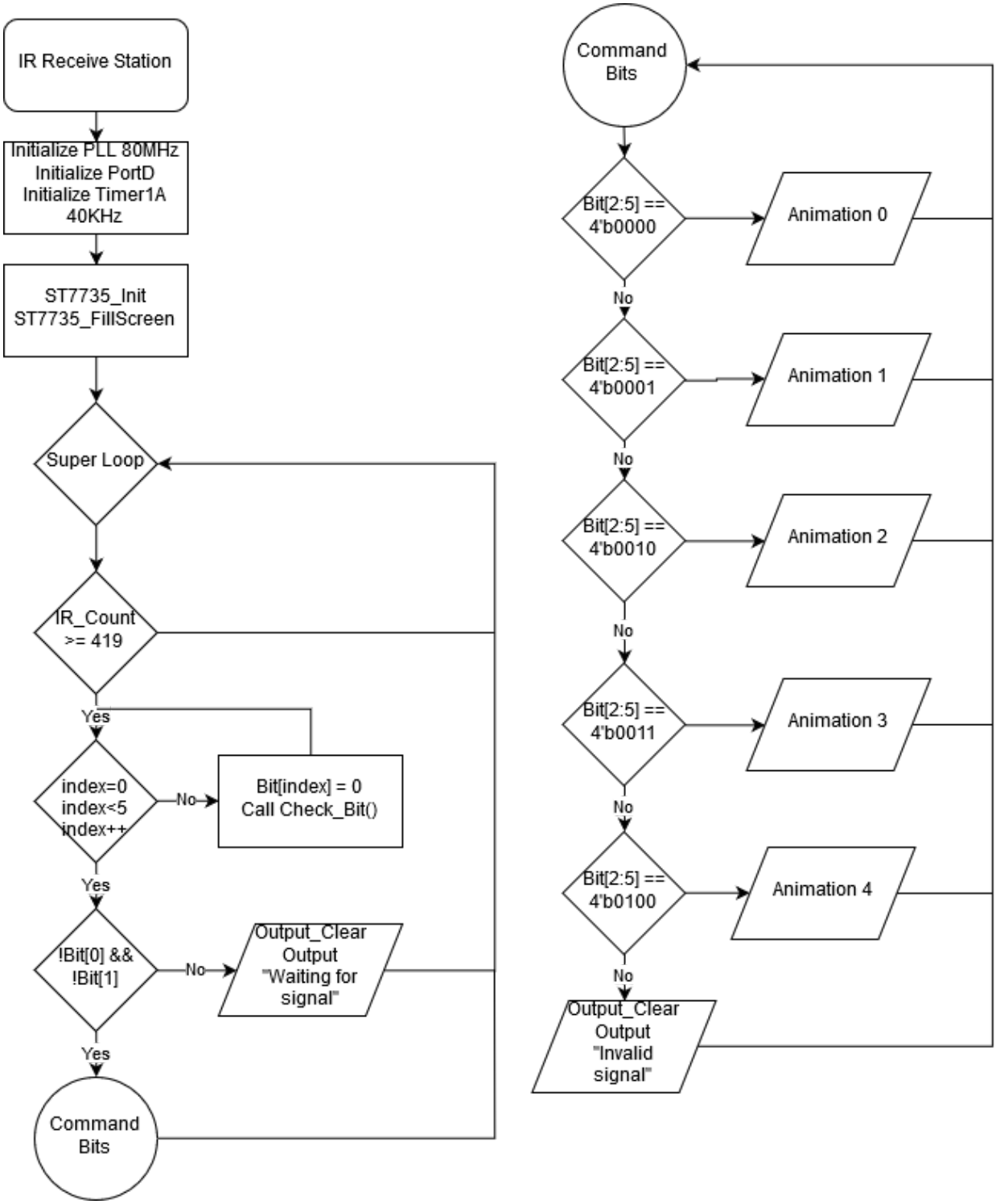
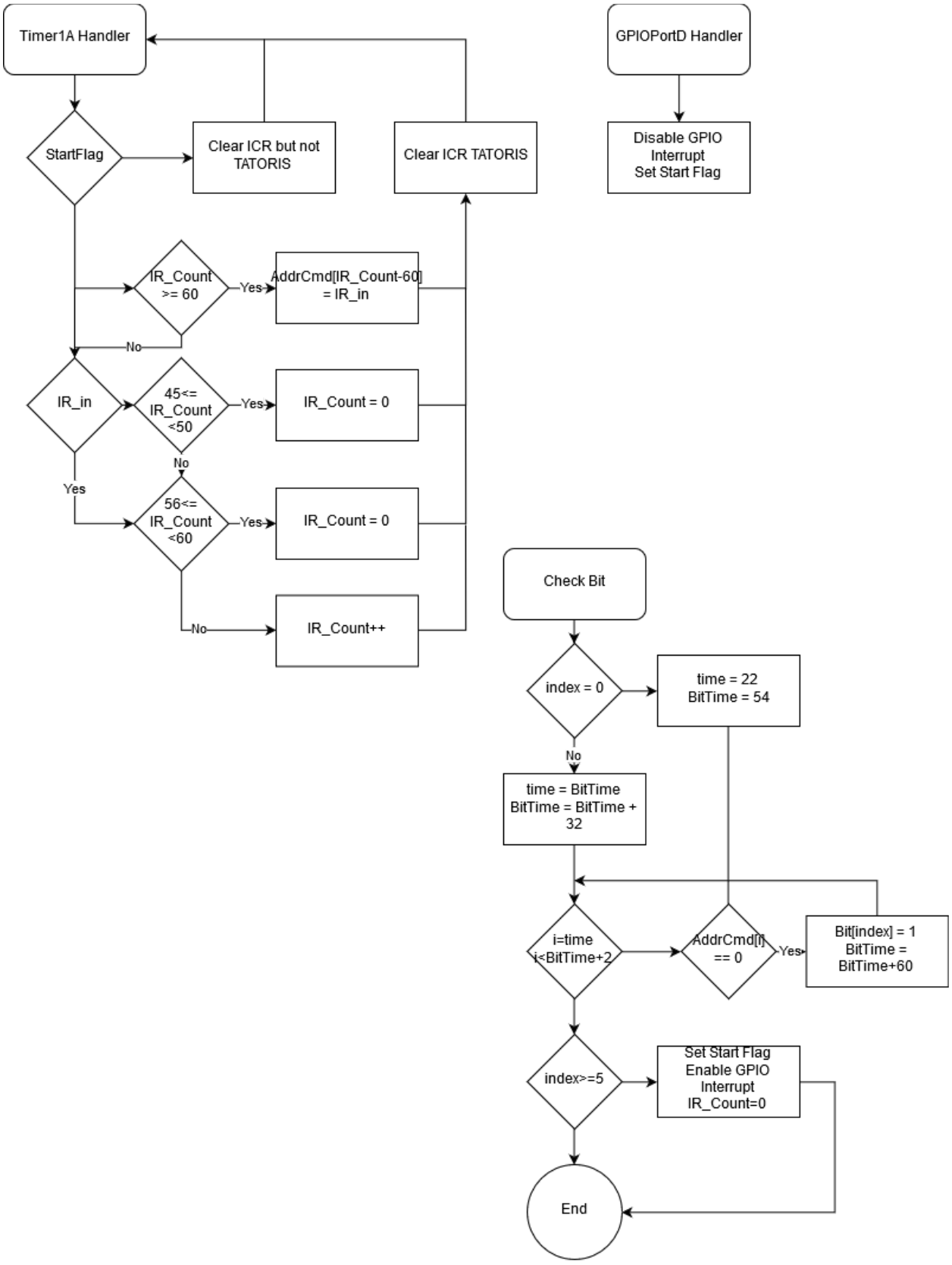


Figure 7: IR Receive Station Part 2



Waveforms

Figure 8: Send Command 0

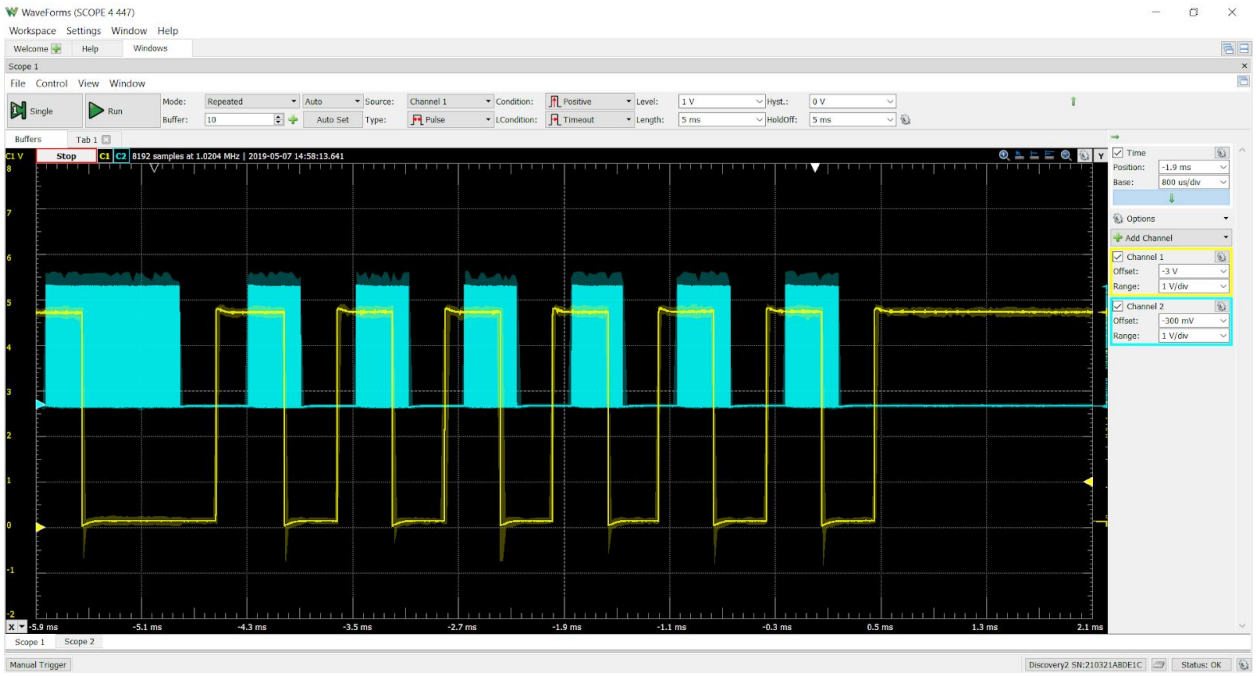


Figure 9: Send Command 1

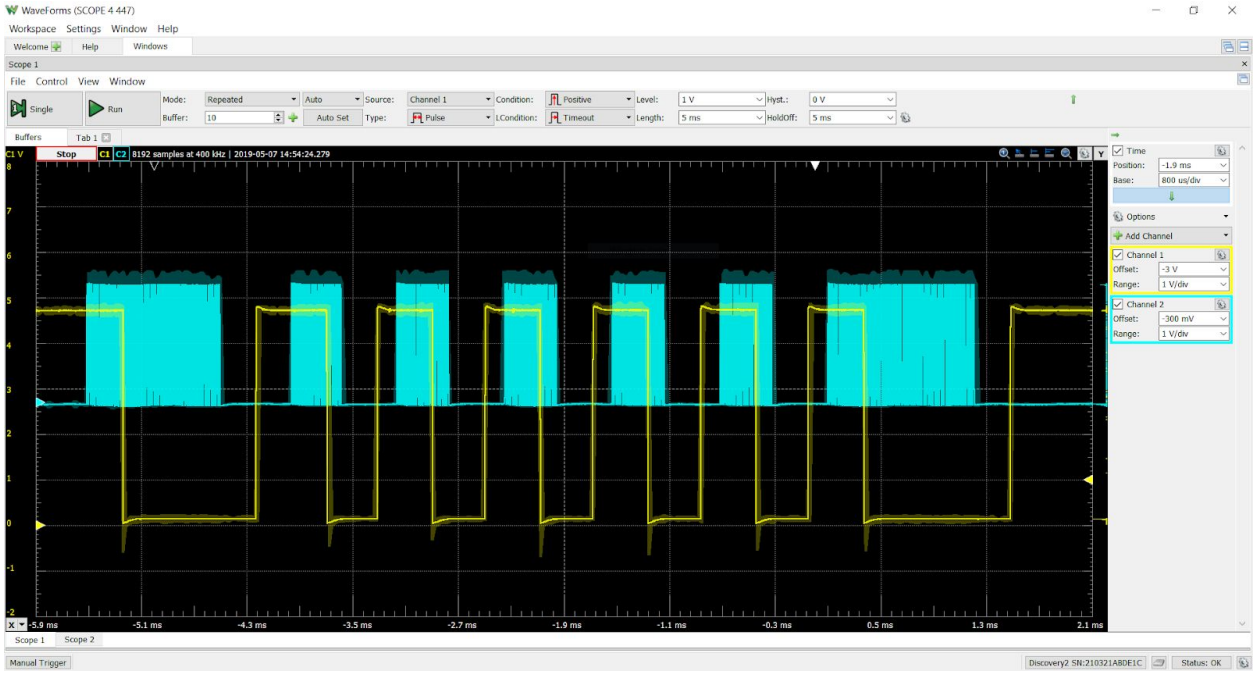


Figure 10: Send Command 2

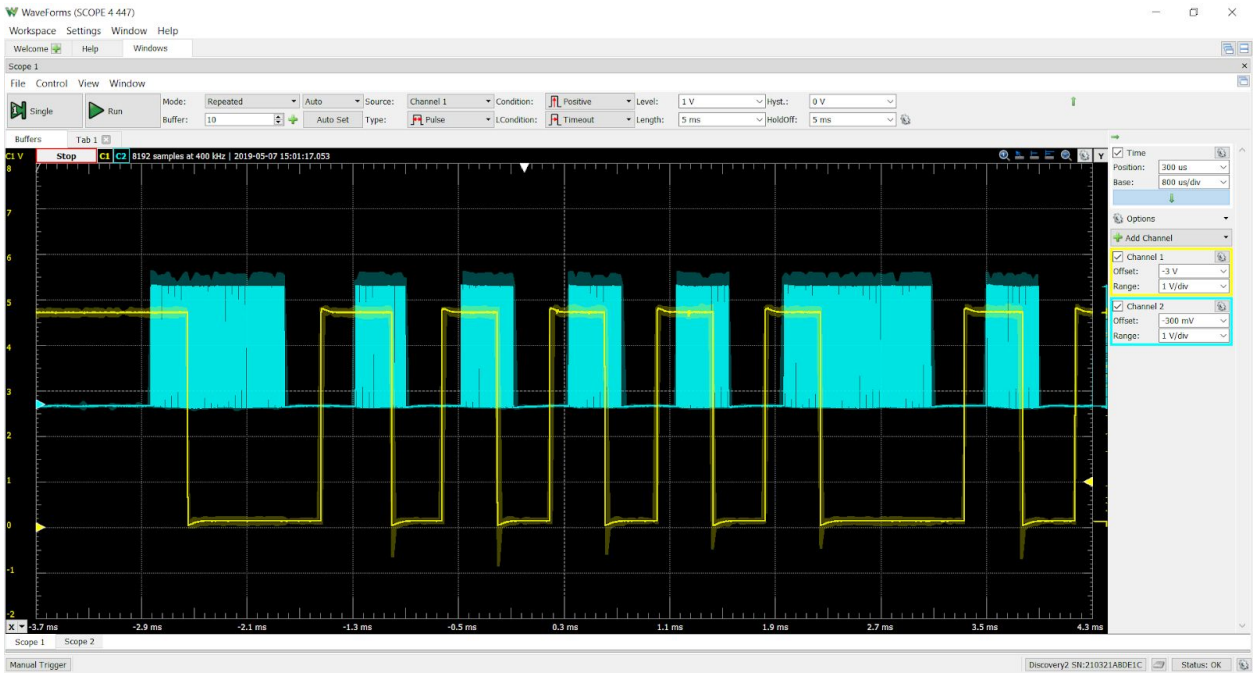


Figure 11: Send Command 3

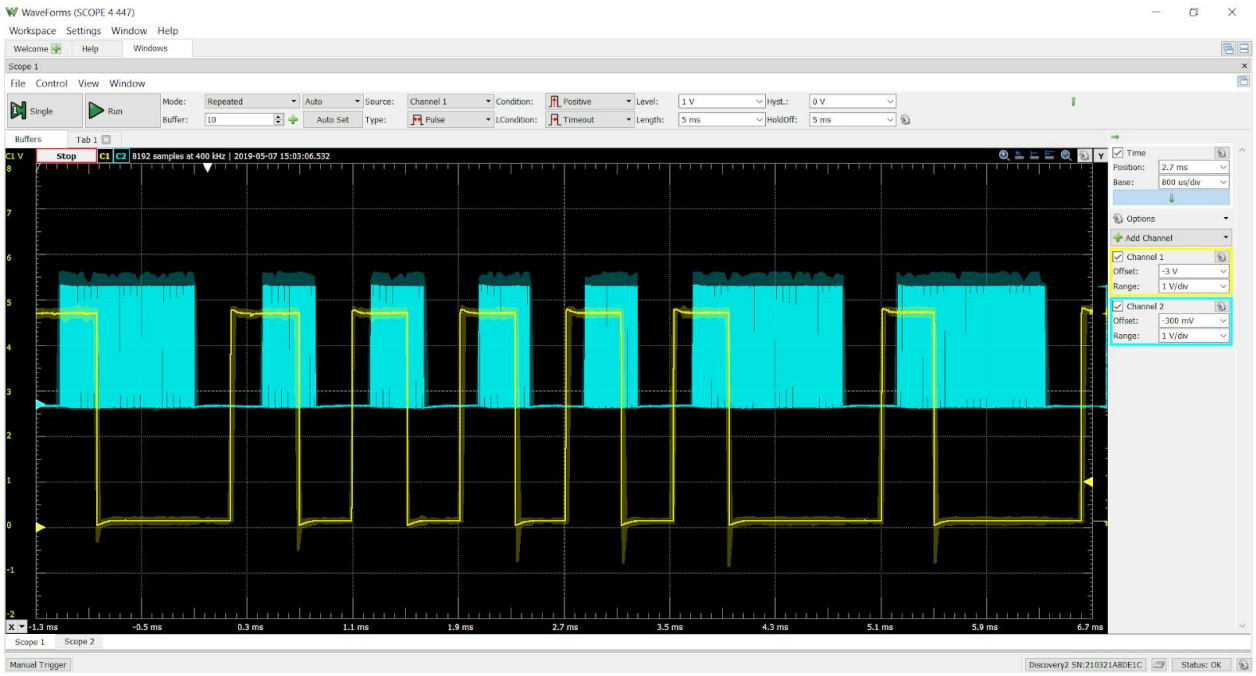
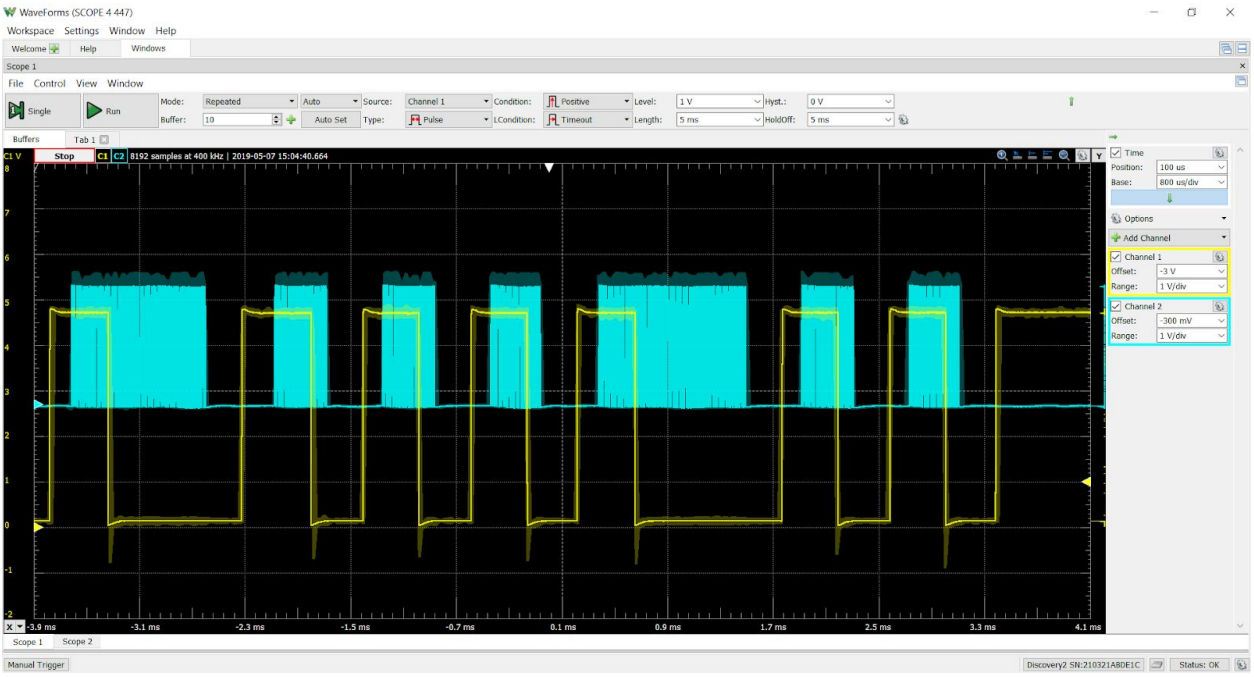


Figure 12: Send Command 4



Conclusion

Our project was able to meet the requirements of the specifications. Our team had a few difficulties when working on this project. Our only major difficulties in this project were the modulation and demodulation of the IR signal. After reviewing our logic, we were able to find a simpler and less needlessly complex Modulation method which worked quite well. Unfortunately, our demodulation took some time as well which proved to be the largest problem. After a couple attempts and long hours of debugging, we were able to successfully decode the demodulated signal we received. Overall, we learned a lot about Bluetooth and IR signal methods and practices.