

Beacon-Controlled Autonomous Network for Trains (BCANT)

By

Susan Chen

Prithvi Garimalla

Jordi Pakey-Rodriguez

Final Report for ECE 445, Senior Design, Spring 2017

TA: Jacob Bryan

03 May 2017

Project No. 5

Abstract

BCANT is a system for model trains that consists of infrared beacons and a laser time-of-flight sensor in order to enable true set and go operation. Over the course of the project we developed the beacon reading system and it demonstrated it works, as well as a complete control system on the locomotive that interfaces the beacon and time-of-flight data to control to motor and other peripherals. Unfortunately, due to last minute hardware failures, we were unable to integrate all the components onto a locomotive, but most of the components were demonstrated to work on their own.

Contents

1	Introduction	1
1.1	Objective	1
1.2	High-Level Requirements.	1
2	Design.	2
2.1	Beacon.	2
2.1.1	Schematic	3
2.1.2	Bridge Rectifier	4
2.1.3	Voltage Regulator	4
2.1.4	Microcontroller	4
2.1.5	LED Driver	4
2.1.6	IR LED	4
2.2	Locomotive Power.	4
2.2.1	Schematic	5
2.2.2	Bridge Rectifier	5
2.2.3	Voltage Regulator	5
2.3	Locomotive Control.	5
2.3.1	Schematic	6
2.3.2	Microcontroller	6
2.3.3	Flash Storage.	6
2.3.4	Voltage Scaler	6
2.4	Locomotive Drivetrain.	7
2.4.1	Schematic	7
2.4.2	Motor Controller	7
2.5	Obstacle Detector.	8
2.6	Beacon Reader	8
2.7	IR Protocol	8
2.8	Beacon Firmware	8
2.9	Locomotive Firmware	8
2.9.1	Feature List	9
2.9.2	Flow Chart	9

2.10 PC Software	10
2.10.1 Beacon Configurator	10
2.10.2 Feature List	10
3 Cost	11
3.1 Labor	11
3.2 Parts	11
4 Conclusion.	12
4.1 Accomplishments	12
4.2 Uncertainties	12
4.3 Ethical considerations	12
4.4 Future work	12
Reference	13
Appendix A Requirement and Verification Tables.	14
Appendix B Calculations.	17
B.1 Beacon MOSFET Gate Resistor.	17
B.2 Beacon LED Current-Limiting Resistor	17
B.3 Beacon LED Half Viewing Angle	17
B.4 Locomotive Voltage Regulator Resistors	17
B.5 Locomotive Voltage Scaler Resistors.	17
Appendix C Schematics.	18
C.1 Beacon.	18
C.2 Locomotive.	19
C.3 Obstacle Detector	20
C.4 Beacon Reader	20
Appendix D Pinouts	21
Appendix E Software	22
Appendix F Cost	23
F.1 Beacon BOM	23
F.2 Locomotive BOM	23
F.3 Obstacle Detector BOM	23
F.4 Beacon Reader BOM.	23

1 Introduction

1.1 Objective

Model trains have become more and more advanced over the last few decades, but they are still controlled entirely by a human operator. These human operators are solely responsible for ensuring that no harm comes to the train. A lapse in attention can be disastrous, causing damage that could be very costly to repair. Model trains currently do not have the ability to detect obstacles or other trains in front of them. If not stopped in time a crash may occur, often resulting in derailment or physical damage. Additionally, derailment may occur when a train is traveling at high speeds on a curved tracks. The train has no recourse for these situations other than for the operator to manually slow or stop it to avoid an accident.

Currently, the National Model Railroad Association (NMRA) defines the Digital Command Control (DCC) standard for model trains. DCC allows for digital control of various train functions such as lights, speed, direction, sound, and much more[1]. One downside to DCC is that smooth operation is completely dependent on the human operator who must manually control the train as it is unable to react to its environment autonomously. In addition, DCC has a steep learning curve and setting up a system without prior experience can be quite challenging.

Our solution is to give each locomotive the ability to respond to the track on its own by detecting obstacles and reacting to programmable beacons placed on the track. A distance sensor is placed on the front of the train to detect obstacles directly in front of it. Small beacon modules placed on the track are set to send out unique identifiers to the train to let it know what speed it should travel at. Essentially, the end goal is to have the train sense potential issues along the track that would cause collision or derailment and act to avert such problems. Our solution is a convenient alternative to those who wish to have speed control for their locomotives. The sensors integrated with the locomotive also provide some measure of protection against human negligence by avoiding obstacles and preventing derailment.

1.2 High-Level Requirements

- Must be able to operate indefinitely on existing methods of track power, both analog (DC) and bipolar Digital Command Control (DCC).
- Must be able to detect oncoming obstacles on the track and stop to avoid derailment or damage.
- Must be able to read customizable speed limit beacons that are powered by the tracks and adjust its speed accordingly.

2 Design

Our solution consists of two independent systems: the beacon boards and the locomotive controller. The beacons are small standalone boards on the track that continuously broadcast a unique ID that is read as the train passes over. The locomotive controller consists of three PCBs on the train and is responsible for reading beacons, identifying obstacles, and controlling the motor. The block diagram is shown in Figure 1.

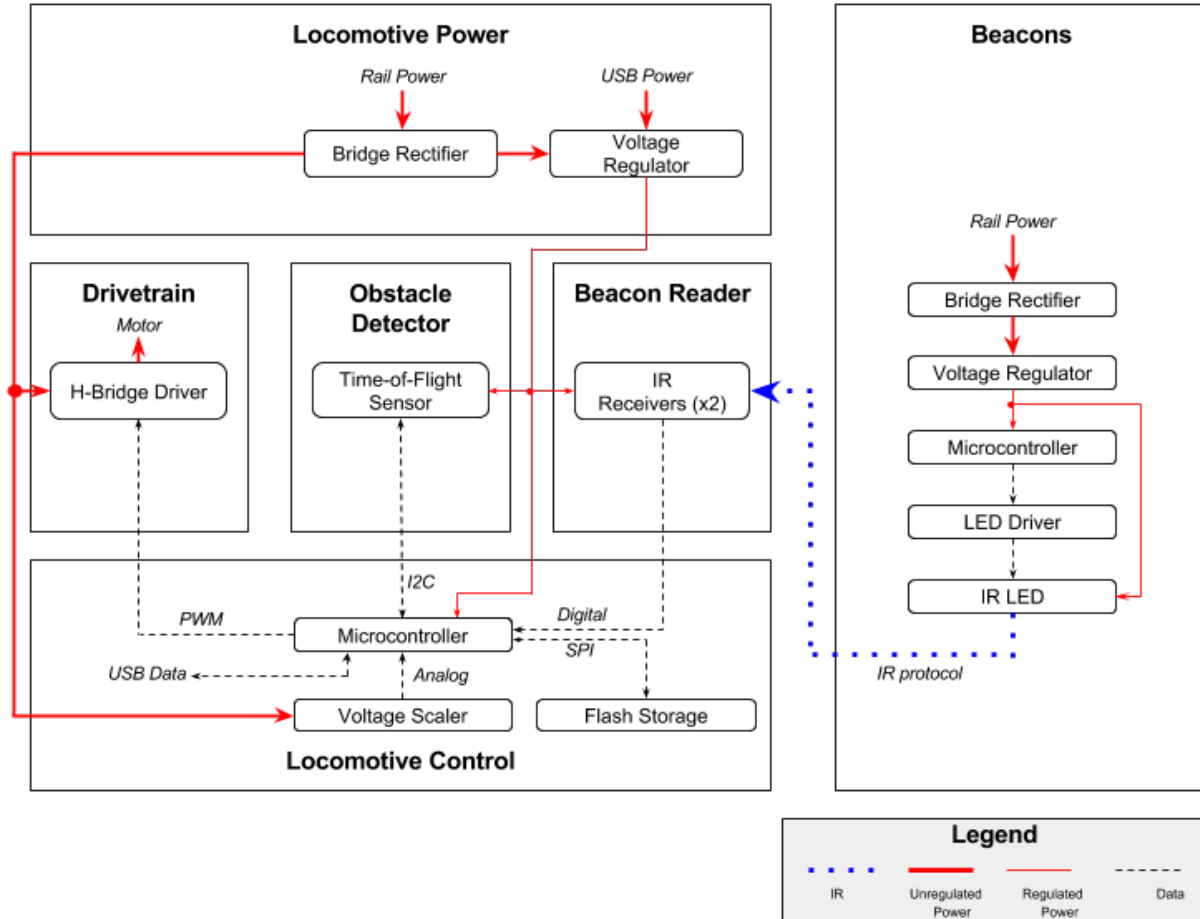


Figure 1: Block Diagram.

2.1 Beacon

The beacon is small board that sits between two track ties as shown in Figure 2. The image on the left shows the placement of a beacon on the track, and the image on the right shows three individual beacons with their ID labels. Its function is to broadcast a unique ID to the IR receiver board on the locomotive. Each ID is mapped to a specific function that can be programmed on the locomotive control board. A bridge rectifier and a voltage regulator are used to provide 2.8V power to the microcontroller, LED driver, and IR LED. The microcontroller controls the IR LED to continuously broadcast the boards ID for the IR receiver on the locomotive board to read.

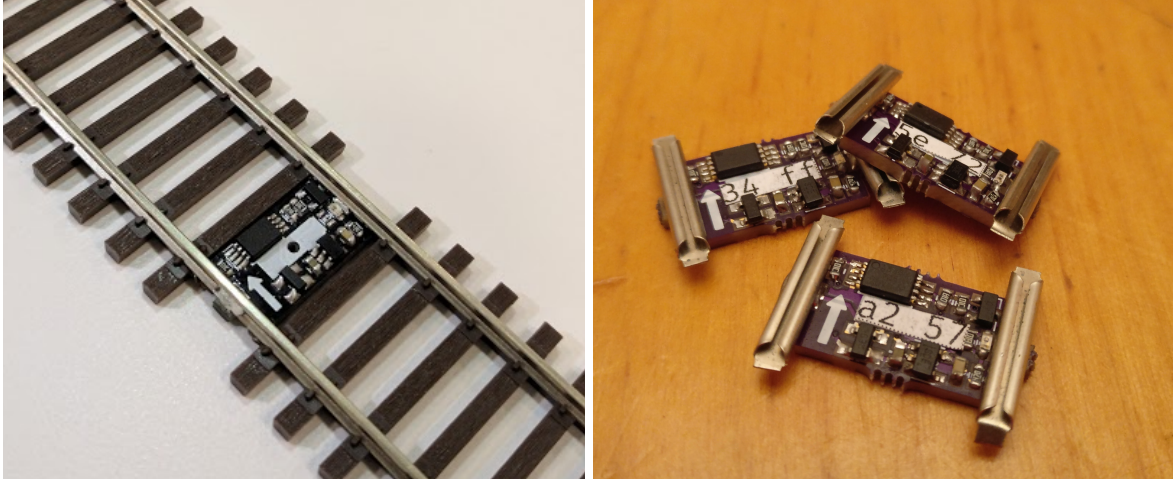


Figure 2: Beacon Boards.

2.1.1 Schematic

The schematics for the beacon board IR LED and power circuits are shown in Figure 3 and Figure 4, respectively. The shunt resistor R5 is used to create a clean 38kHz modulated signal since the LED is unable to switch fast enough. The calculations for R2 and R4 are shown in Calculations 1 and 2 in Appendix B.

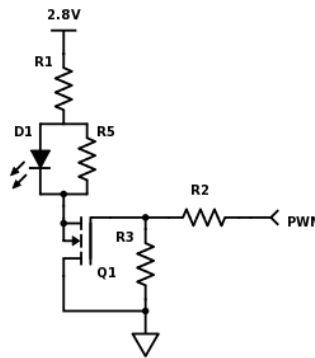


Figure 3: Beacon IR LED Circuit.

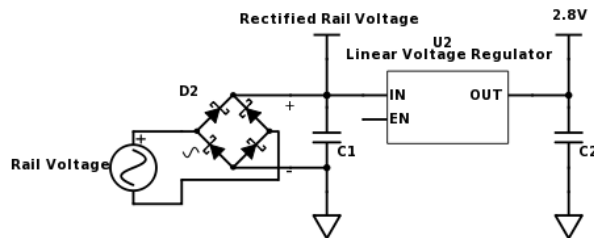


Figure 4: Beacon Power Circuit.

2.1.2 Bridge Rectifier

Since the voltage regulator only takes positive voltage, the full-bridge rectifier is needed rectify the power signals from the rails. It consists of four schottky diodes in a bridge configuration and rectify rail voltages ranging from -16V to +16V.

2.1.3 Voltage Regulator

A linear voltage regulator takes an input from the rectified rail power and bring the voltage down to 2.8V from an input voltage of up to 16V. It provides the power needed for the MCU and its peripherals.

The TPS709 linear voltage regulator data sheet specifies a minimum input voltage of 2.7V[2]. Testing showed that an input voltage of 3.5V would produce an output voltage of 2.63V, while input voltages lower than 3.5V would produce an output voltage that is below our requirements. Therefore, a input voltage of 3.5V would be the minimum threshold for our project to function properly.

2.1.4 Microcontroller

The microcontroller used is an ATtiny45 8-bit AVR processor. It uses a single GPIO pin to control the LED driver circuit and continuously broadcast the boards unique ID using the IR protocol.

2.1.5 LED Driver

The LED driver circuit consists of an N-channel MOSFET, a gate resistor, and a pull down resistor. The MOSFET conducts current when the I/O pin from the microcontroller is set to high. The gate resistor is used to prevent damage due to overcurrent to the microcontroller and the pull down resistor is used to ensure the MOSFET is turned off when the voltage at the gate is left floating.

2.1.6 IR LED

The 940nm IR LED is used to broadcast a unique ID that the receiver on the locomotive can read. It is controlled by the LED driver circuit using PWM. The viewing half-angle for the LED can be calculated using Equation 1. The calculation for the viewing half angle can is shown in Calculation 3 in Appendix B.

$$\theta_{\frac{1}{2}} = \arctan\left(\frac{V_{max}T}{2H_{min}}\right) \quad (1)$$

Vmax is the max speed of the train, T is the length of one ID message, and Hmin is the minimum distance from the beacon board LED to the IR receivers on the train.

2.2 Locomotive Power

The locomotive power circuit takes an input voltage ranging from -16V to +16V from the rail or 5V from the USB. It then rectifies and regulates that voltage down to 2.8V in order to power the microcontroller and its peripherals. A full bridge rectifier is used to rectify the AC voltage from the rails. The rectified voltage is then inputted to a linear voltage regulator. The SPX3918 adjustable linear voltage regulator has a minimum input voltage of 2.5V and a maximum dropout voltage of 550mV with a 500mA current draw. This means the minimum input threshold voltage in order for the locomotive components to be powered would be 3.35V, allowing for the microcontroller to be able to wait for the input voltage to reach the 3.5V

minimum threshold limit for the beacon board to be powered. The circuit for the locomotive power block is shown in Figure 5.

Other designs were considered for the locomotive power circuit. A switching circuit to switch between the USB power source and the rail power source was considered. However, it was simpler to add a protection diode after the USB voltage source. There were also attempts to regulate the input voltage through the full range of rail voltages with switching circuits, switch-mode power supplies, or batteries. The switching circuits were complex and risked shorting the input and output voltages if the timing was not correct. There were no switch-mode power supplies on the market that had a low enough input voltage for the output that was required. Batteries would have defeated the purpose of drawing power from a wall outlet.

2.2.1 Schematic

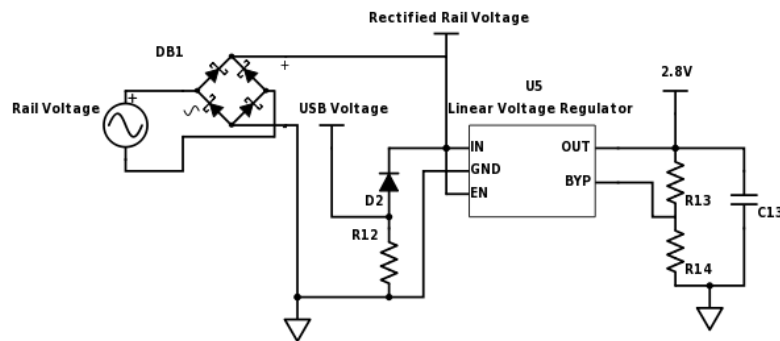


Figure 5: Locomotive Power Schematic.

2.2.2 Bridge Rectifier

The full-bridge rectifier is needed to rectify power signals from the rails before they are routed to the voltage regulator, voltage scaler, and motor driver. It consists of four schottky diodes in a bridge configuration and rectifies voltages ranging from -16V to +16V.

2.2.3 Voltage Regulator

The rectified voltage is fed into a linear voltage regulator in order to bring the voltage down to 2.8V. The 2.8V source is used to power the microcontroller, time-of-flight sensor, IR receiver, and the voltage scale circuit. Calculation 4 in appendix B shows how the adjustment resistors are calculated for the SPX3819 linear regulator for an output voltage of 2.8V using Equation 2.

$$V_{out} = 1.235(1 + \frac{R_1}{R_2}) \quad (2)$$

2.3 Locomotive Control

The locomotive control unit consists of a microcontroller and a USB data connection. If there is a USB connection the control unit will enter a programming mode, otherwise it will run the normal track operation.

This allows operators to map the beacon IDs to one of 128 different speeds[3]. The microcontroller receives a voltage input via the ADC pin from the voltage scale circuit and outputs a PWM signal to the drivetrain to control the speed. The IR receiver sends beacon IDs to the microcontroller in order to determine what speed the train should go. It also stops the train if the time-of-flight sensor detects an obstacle in its path.

2.3.1 Schematic

The main part of the control circuit is the STM32F4 microcontroller. It is connected to a M95512-RDW6TP EEPROM for storage of the beacon mappings. The control schematic is shown in Figure 6.

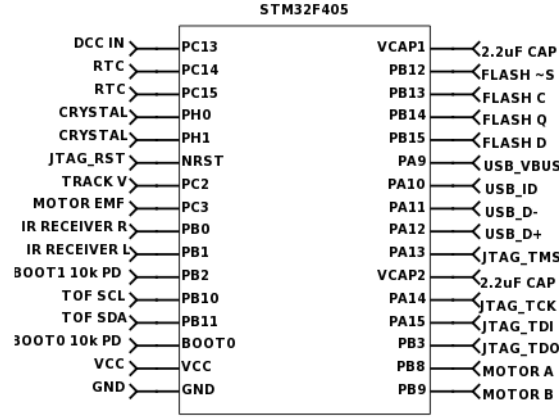


Figure 6: Locomotive Control Schematic.

2.3.2 Microcontroller

The microcontroller used is an STM32F405 32-bit ARM processor. It communicates with the time-of-flight sensor via I2C, the IR receiver digitally, and control the motor with PWM signals. It also stores configuration data for the beacons in external memory.

2.3.3 Flash Storage

An EEPROM flash chip is used to store the beacon mappings. It communicates with the microcontroller over SPI. EEPROM was chosen since we can address individual bytes, which makes performing binary search through the dataset for the beacon mappings more efficient.

2.3.4 Voltage Scaler

The voltage scaler circuit originally included an operational amplifier buffer to improve the accuracy of the voltage readings. However, we were unable to find an operation amplifier that could be powered by a 2.8V source, so errors were corrected in software instead. The voltage scaler circuit consists of a resistive voltage divider, shown in Figure 7, that scales the rectified rail voltage down to a lower voltage that can be read by the microcontrollers ADC. The rectified rail voltage ranges from 0-16V and the microcontroller ADC can read voltages from 0-2.8V. Two resistors divide the voltage to output a range of 0-2.8V, which will give 4096 different values for a 12-bit precision ADC. The calculations using Equation 3 of the voltage divider's resistor values are shown in Calculation 5 in appendix B.

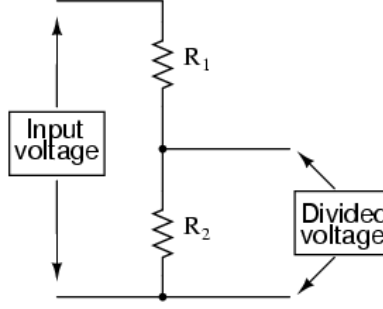


Figure 7: Voltage Divider Circuit.

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2} \quad (3)$$

2.4 Locomotive Drivetrain

The locomotive drivetrain unit consists of the motor controller circuit that interfaces the PWM signal from the microcontroller with the rectified rail power in order to power the motor. The circuit consists of a MOSFET H-bridge that is gate-driven by NPN transistors. The drivetrain circuit is shown in Figure 8.

In order to drive the MOSFETs in the H-bridge, optoisolators were also considered. However, we were unable to find an optoisolator that would support driving the LED with our maximum rectified rail voltage.

2.4.1 Schematic

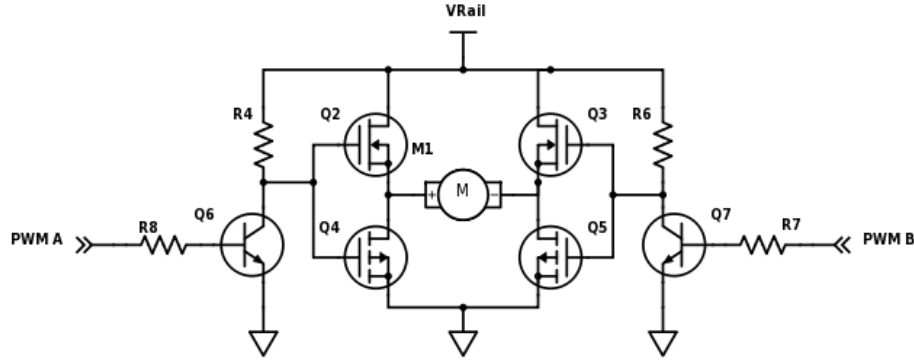


Figure 8: Locomotive Drivetrain Schematic.

2.4.2 Motor Controller

The motor driver circuit consists of two PWM signals driving two NPN transistors which each control an NFET and a PFET. The MOSFETs are configured as an H-bridge and allow the motor to run forwards or backwards at any speed determined by the PWM signals. The frequency of the PWM signals is 10kHz to prevent resonance and noise. The motor current range is $\pm 1.5\text{A}$ and the motor voltage range is $\pm 16\text{V}$.

2.5 Obstacle Detector

The obstacle detector consists of a time of flight (TOF) sensor mounted to the front of the train. It detects any obstruction on the tracks that could cause the train to derail and cause an emergency stop. The TOF sensor uses a laser to measure whether the path in front of the train is clear or if the train is approaching an unexpected obstacle. The range of the TOF sensor is approximately 50-200mm[4].

2.6 Beacon Reader

The beacon reader consists of two IR receivers mounted underneath the train on either side. They read beacons as the train passes over them. With two receivers, the microcontroller can infer the direction the train is moving and execute different actions accordingly.

2.7 IR Protocol

The IR protocol uses 38kHz modulation present in many televisions and other consumer devices. We chose this frequency because sensors for it are widely available. Our protocol uses Manchester encoding according to the G.E. Thomas convention with a bit period of 960s and data length of 16 bits[5]. The timing diagram for the protocol is shown in Figure 9.

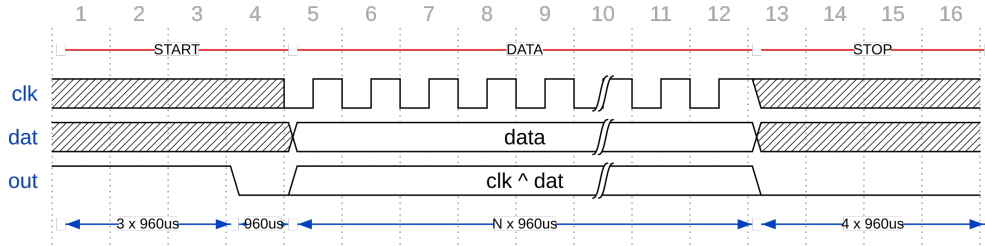


Figure 9: IR Protocol Timing Diagram.

2.8 Beacon Firmware

The beacon firmware is a simple loop that continuously broadcasts the beacon ID using the IR protocol. Every 480us it determines what the next edge is (rising or falling) and sets the GPIO pin accordingly.

2.9 Locomotive Firmware

The locomotive has two modes of operation: track mode and programming mode. It also reacts to pin change interrupts from the IR receiver. At boot, the microcontroller checks to see if USB is connected. If so, it enters programming mode and presents itself as a USB mass storage device. When the user starts to write a configuration file to the locomotive, the microcontroller will buffer each beacon mapping and verify they are being written in ascending order. This is required since at look-up time, it will perform a binary search through the mappings to find the correct one. If an errored mapping comes in, the microcontroller will stop storing more mappings.

In track mode, the microcontroller will continuously check the obstacle detector, stopping the train if there is one. Then it checks whether a beacon ID is read and looks up the appropriate mapping before executing it. Finally, it makes any motor adjustments as necessary.

The IR receiver also works by invoking a pin change interrupt every time its value changes. Using our IR protocol and taking advantage the repetitive pattern of the data we are sending, we need only store one message length worth of pin changes to calculate the beacon ID. When we have the requisite data, we set a ready flag which will trigger the main event loop to look up and execute the mapping.

2.9.1 Feature List

- Mount as USB mass storage device if USB plugged on boot
- Stores the beacon mappings on flash memory
- Reads beacons through pin change interrupts
- Can stop the train if an obstacle is detected
- Can continuously adjust the motor speed and direction based on track conditions

2.9.2 Flow Chart

The flowchart for the locomotive operation is shown in Figure 10.

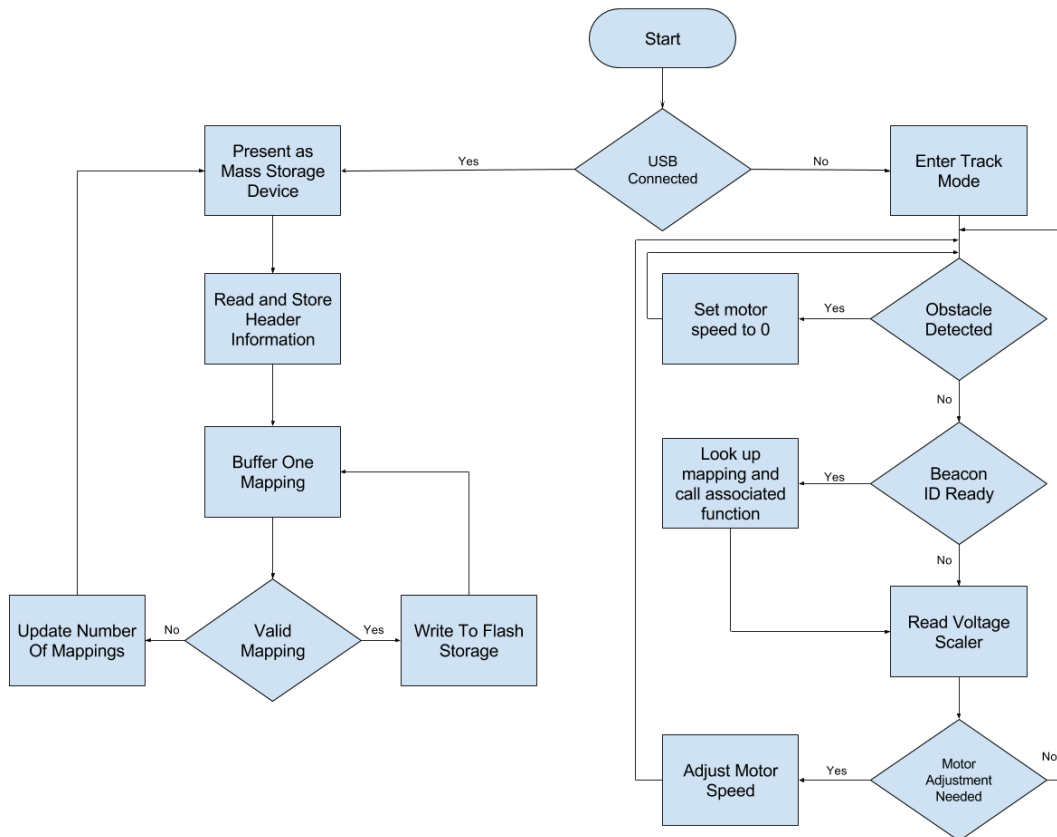
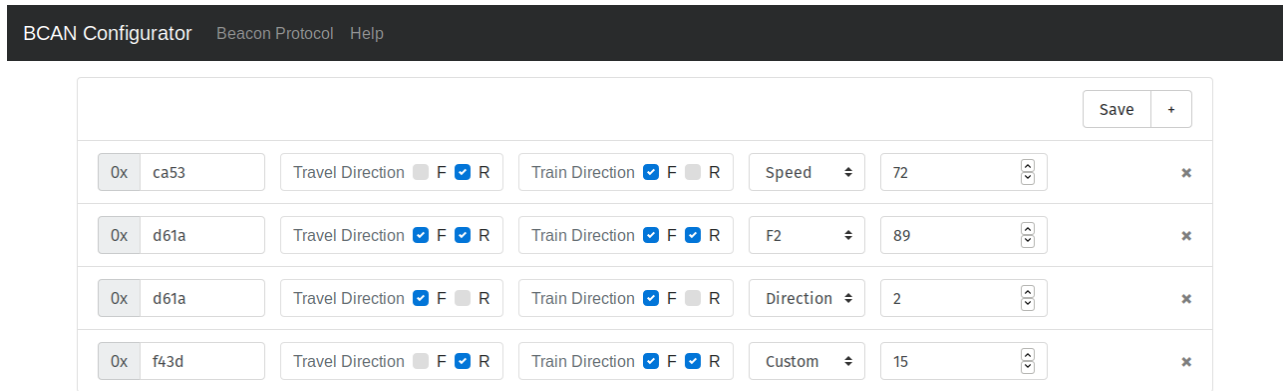


Figure 10: Locomotive Firmware Control Flow.

2.10 PC Software

2.10.1 Beacon Configurator

Figure 11 shows the web interface (hosted online [here](#)) to program the beacons. Once specifying the ID, the user can select the instances where the mapping applies and the action the beacon corresponds to.



The screenshot shows the BCAN Configurator web interface. At the top is a dark header with the text "BCAN Configurator", "Beacon Protocol", and "Help". Below the header is a table with four rows of beacon mappings. Each row has a "Save" button and a "+" button in the top right corner. The table columns are: Beacon ID (e.g., 0x ca53), Travel Direction (checkboxes for F and R), Train Direction (checkboxes for F and R), and a parameter (e.g., Speed, F2, Direction, Custom) with a value and a dropdown arrow. Each row ends with a delete "x" button.

Beacon ID	Travel Direction	Train Direction	Parameter	Value	Action
0x ca53	<input type="checkbox"/> F <input checked="" type="checkbox"/> R	<input checked="" type="checkbox"/> F <input type="checkbox"/> R	Speed	72	x
0x d61a	<input checked="" type="checkbox"/> F <input checked="" type="checkbox"/> R	<input checked="" type="checkbox"/> F <input checked="" type="checkbox"/> R	F2	89	x
0x d61a	<input checked="" type="checkbox"/> F <input type="checkbox"/> R	<input checked="" type="checkbox"/> F <input type="checkbox"/> R	Direction	2	x
0x f43d	<input type="checkbox"/> F <input checked="" type="checkbox"/> R	<input checked="" type="checkbox"/> F <input checked="" type="checkbox"/> R	Custom	15	x

Figure 11: Beacon Configurator Website.

2.10.2 Feature List

- Allows user to add any number of beacon mappings
- Each mapping consists of a beacon ID, description, direction of travel, direction of drain, function, and additional parameters for that function
- Clicking Save downloads a file that can be copied to the USB mass storage device to program the mappings

3 Cost

In order to keep costs low, we chose a locomotive microcontroller that can be switched out with a cheaper one in case the user did not require as many peripherals. We also kept the locomotive board limited to 2 layers as a 4 layer board would have significantly increased the price. On the beacon side, we kept the BOM to under \$5 in single quantities, and have already designed a v2.0 with a cheaper microcontroller that has a \$1 BOM at quantity 1000.

3.1 Labor

Table 1: Labor Costs

Name	Total Hours	Hourly Rate (\$)	Total (\$)	Total * 2.5 (\$)
Susan Chen	170	36.00	6,120.00	15,300.00
Prithvi Garimalla	170	36.00	6,120.00	15,300.00
Jordi Pakey-Rodriguez	170	36.00	6,120.00	15,300.00
Total	510	108.00	18,360.00	45,900.00

3.2 Parts

We spent \$200 on parts which, combined with our \$45,900 labor costs, gives a total project cost of \$46,100.

4 Conclusion

4.1 Accomplishments

While our project definitely had some issues that kept it from being a complete success, the most critical and innovative part, beacon reading, worked. It was demonstrated to be versatile and the speed at which the IDs could be read exceeded our expectations. Furthermore, we designed the project to be as economical as possible, so the total cost of the materials for a beacon was quite cheap. Should there be demand for this product in the model train community, we believe there would be a legitimate path to market.

4.2 Uncertainties

We still have a few lingering questions about our project, namely the nature of the eleventh hour failure we experienced. We had soldered the locomotive board and it had been working for two weeks until a short developed two days before the final demo. After thinking we had identified the cause and replaced the suspect component, another short, this time at the output of the 2.8V voltage regulator appeared. This was approximately 18 hours before our final demo and the only solution we saw was to solder together another board. Unfortunately, even though this board had never even been powered, the short was still present. Even after thoroughly examining the board under a microscope, the only theory we have for the second board being faulty is a faulty component. At this point, even though we were able to demonstrate the individual aspects of the project worked, integrating them was not successful.

4.3 Ethical considerations

There are two potential safety concerns with our project, however we believe they are both within the acceptable tolerances for a general consumer product. The first of these is that the rails of the train track are electrically charged so one could potentially shock themselves. However, we are not changing this characteristic in our design and therefore are not creating any additional safety hazards. This is a well understood risk of model trains so responsible usage dictates they should be kept out of reach of small children or pets. The second is that we are using a time-of-flight sensor that uses a laser. The laser used is Class 1, which means that it is safe for use under normal conditions and while operating within the manufacturer's specifications[4].

In accordance with the IEEE code of ethics, we endeavored to take on technical tasks we feel we are qualified for and seek advice from our mentors for the skills we lack. We sought feedback and criticism for our work and continuously worked to correct our mistakes. We strove for honesty and properly credited individual contributions to our project. Most importantly, we supported and encouraged each other to follow this code of ethics.

4.4 Future work

The software, firmware, and schematics for our project are fully open source and there is no reason why the work on our project has to stop here. Our design choices and decisions are well documented, so should anyone choose to continue our work, they can easily pick up where we left off. We have already designed a v2.0 beacon, and have made a list of changes for the next locomotive version.

References

- [1] “Electrical Standards for Digital Command Control,” Datasheet, National Model Railroad Association, 2006, accessed February 2017. [Online]. Available: <http://download.micron.com/pdf/datasheets/dram/ddr/512MBDDR4x8x16.pdf>
- [2] “TPS709,” Datasheet, Texas Instruments, 2015, accessed February 2017. [Online]. Available: <http://www.ti.com/lit/ds/symlink/tps709.pdf>
- [3] “NMRA Standard,” Datasheet, National Model Railroad Association, 2012, accessed February 2017. [Online]. Available: http://www.nmra.org/sites/default/files/s-9.2.1_2012_07.pdf
- [4] “VL6180X,” Datasheet, Texas Instruments, 2016, accessed February 2017. [Online]. Available: <http://www.st.com/content/ccc/resource/technical/document/datasheet/c4/11/28/86/e6/26/44/b3/DM00112632.pdf/files/DM00112632.pdf/jcr:content/translations/en.DM00112632.pdf>
- [5] “Manchester Encoding Basics,” Datasheet, Atmel, 2012, accessed February 2017. [Online]. Available: http://www.atmel.com/images/atmel-9164-manchester-coding-basics_application-note.pdf
- [6] “IR Receiver Modules for Remote Control Systems,” Datasheet, Vishay, 2017, accessed February 2017. [Online]. Available: <http://www.vishay.com/docs/82434/tsop572.pdf>

Appendix A Requirement and Verification Tables

Table 2: Beacon Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must be able to be powered from rail power with a voltage range of $\pm 16V$ and be able to draw a minimum of 60mA.	1. Connect the board to a bench power supply set at +16V. Verify the LED is lit using a camera. Repeat with -16V.	
1. Must be able to modulate the LED between 38kHz - 40kHz.	1. Connect IR receiver output to an oscilloscope and point it at the IR LED[6]. Verify that the signal is received.	
1. Must have a view angle given by Equation 1 on the axis parallel to the track.	1. With the datasheet value for θ and the distance H, solve for VT. Place the receiver at a horizontal distance VT / 2 and a height H from the LED and verify with the oscilloscope that the ID is read.	

Table 3: Locomotive Power Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must be able to accept voltages of 3.35 to 16V or -3.35 to -16V from the rail supply and output a rectified rail power of $2.8 \pm 0.2V$ and supply up to 280mA.	1. Connect the locomotive to maximum rail power, verify the 2.8V output is active with a multimeter. Reverse the orientation of the train on the track and verify the 2.8V output again. Then put a 10 1 Watt resistor across the 2.8V output and measure the current.	
Continued on next page		

Table 3 – continued from previous page

Requirement	Verification	Verification status (Y or N)
1. Must be able to accept power from the USB supply and output $2.8 \pm 0.2V$ power and supply up to 280mA.	1. Remove the train from the track and connect USB power. Verify the 2.8V output is active with a multimeter. Then put a 101 Watt resistor across the 2.8V output and measure the current.	

Table 4: Locomotive Control Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must be able to act as a USB Mass storage device when connected to a PC.	1. With the final microcontroller firmware, connect it to a PC to put it into programming mode. Copy a valid mapping file to the Mass Storage Device the microcontroller presents itself as. Then without unplugging the microcontroller, verify the file can be read.	
1. Must be able to store beacon ID mappings when not powered externally.	1. Repeat Verification 1, with the beacon mappings set between two IDs for which there are physical beacons that correspond to turning the lights on and the turning the lights off. Then unplug and reconnect the USB. Power the beacons, then hold the locomotive over the beacons and verify the correct actions occur.	
1. Must be able to scale a voltage range of 0-16V down to 0-2.8V and be read by the microcontrollers ADC with at least 10 bit precision.	1. Write and flash a test program that displays the current current ADC reading over the serial connection to the computer. Power the board over USB while connecting the rail power input to a bench power supply and sweep between 0V-27V and ensure the microcontroller is accurately displaying at least 128 distinct speed steps.	

Table 5: Locomotive Drivetrain Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must be able to take a PWM signal from the microcontroller and control the motor accordingly.	1. Write and flash a test program to the locomotive microcontroller that will cycle through 0%, 25%, 50%, 75%, and 100% duty cycles on the PWM line. Verify the output of the motor controller with an oscilloscope.	

Table 6: Obstacle Detector Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must detect obstacles at least 50mm in front of the train.	1. Write and flash a test program to the microcontroller that will display over the PC serial connection the values read from the sensor. Place an object at a distance of 50mm in front of the sensor and 8.25mm off the center axis of the train and perpendicular to the track. Move to 8.25mm off the center axis in the opposite direction. Verify both times the object is detected.	

Table 7: Beacon Reader Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Must be able to sense a 38kHz modulated IR beam and output a demodulated digital signal.	1. Connect a beacon to rail power. Write and flash a test program that will display over the PC serial connection the value read from the IR receiver. Verify that the value read matches the beacons ID.	

Appendix B Calculations

B.1 Beacon MOSFET Gate Resistor

Ohm's Law: $R = \frac{V}{I}$

I/O input voltage: $V_o = 2.8V$

Max current per I/O pin: $40mA$

Choose $I_1 = 28mA$

$$R_1 = \frac{2.8}{0.028} = 100\Omega$$

B.2 Beacon LED Current-Limiting Resistor

Forward voltage of IR LED: $V_f = 1.2V$

Supply voltage: $V_d = 2.8V$

Continuous DC current: $I_3 = 34mA$

Using Ohm's Law:

$$I_3 = \frac{V_d - V_f}{R_3} = \frac{2.8 - 1.2}{R_3} = 34mA$$

$$R_3 = 47\Omega$$

B.3 Beacon LED Half Viewing Angle

$$V_{max} = 1 \frac{m}{s}$$

$$T = 23.04ms$$

$$H_{min} = 6mm$$

$$\theta_{\frac{1}{2}} = \arctan\left(\frac{V_{max}T}{2H_{min}}\right) = \arctan\left(\frac{0.02304}{0.006}\right) = 75.4^\circ$$

B.4 Locomotive Voltage Regulator Resistors

$$V_{out} = 1.235\left(1 + \frac{R_1}{R_2}\right)$$

$$R_1 = 33.2k\Omega$$

$$R_2 = 26.1k\Omega$$

B.5 Locomotive Voltage Scaler Resistors

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2}$$

$$V_{in,max} = 16V$$

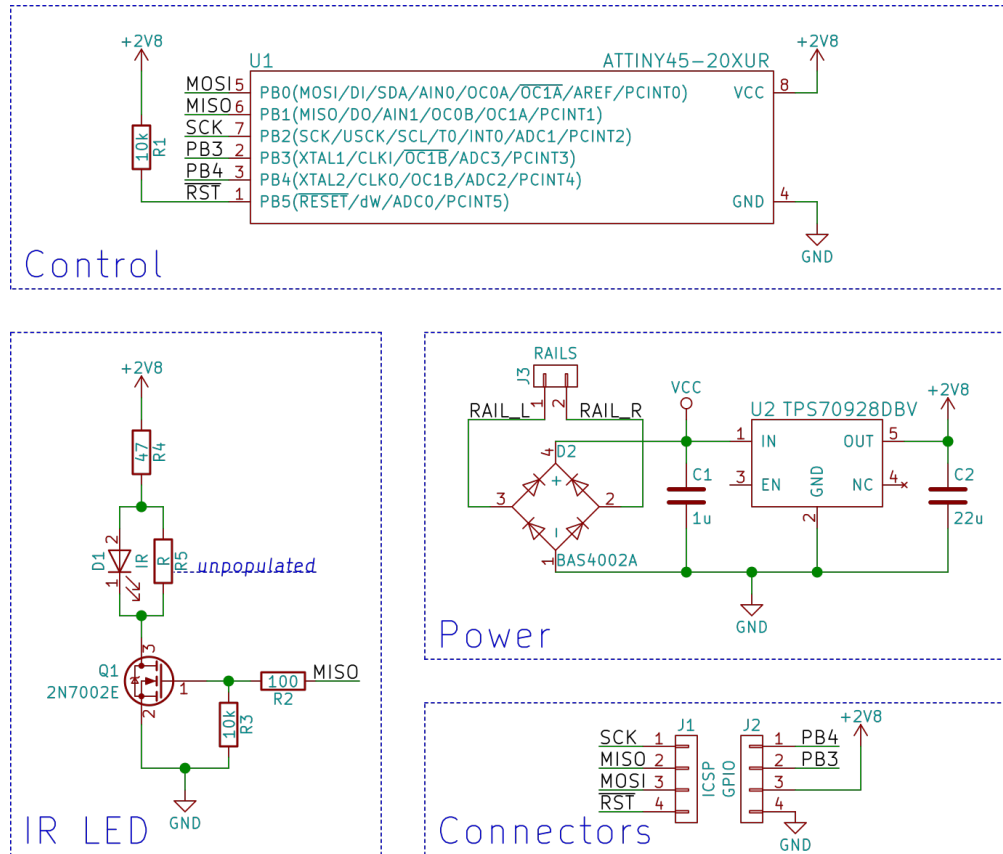
$$V_{out,max} = 2.8V$$

$$R_1 = 13.3k\Omega$$

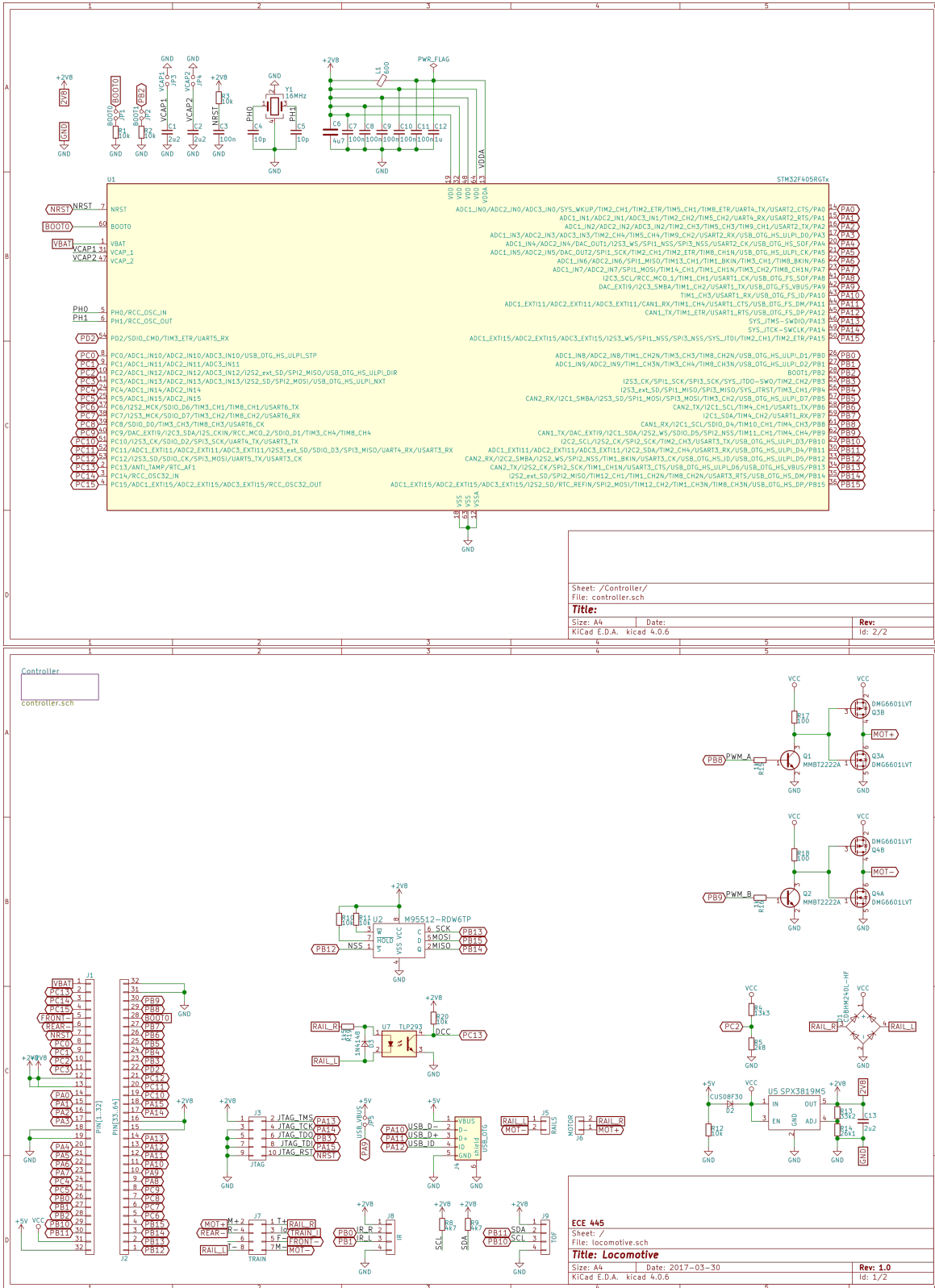
$$R_2 = 2.8k\Omega$$

Appendix C Schematics

C.1 Beacon



C.2 Locomotive



+2V8



Appendix D Pinouts

Appendix E Software

Appendix F Cost

F.1 Beacon BOM

F.2 Locomotive BOM

F.3 Obstacle Detector BOM

F.4 Beacon Reader BOM