Hands-Free Following Cart

Fall 2021 - ECE 445

Team 16:
Anudeep Ekkurthi (anudeep2),
Vincent Sorrentino (vsc2),
Matthew Mo (msmo2)

TA: Stasiu Chyczewski

Final Report

Electrical and Computer Engineering
University of Illinois at Urbana Champaign
08 December 2021

1 Abstract

The primary purpose of this project is to allow users to carry heavy objects in a time efficient manner and improve workplace safety for small construction and landscape business owners.

Time and labor are scarce resources when it comes to landscaping businesses. Therefore to reduce the amount of time in carrying heavy construction material and landscaping material to a project site, our project aims to carry heavy load to the site autonomously by following a user's phone.

This final product of this project is a 24" x 24" inch flat cart that can carry up to 300 pounds and respond to Bluetooth commands. The GPS portion of this project is uncompleted due to issues with phone privacy and setting up external applications to receive GPS data from a phone. However the project high level functionality is intact and the user can continue to use the cart by sending driving commands via Bluetooth and also receive data on GPS and ultrasonic sensors on board.

Contents

2 Introduction

In landscaping and construction sites, workers are often involved in repetitive tasks, especially moving heavy loads from one point to another. Not only is there an inefficient use of labor and time, such tasks also expose workers to injuries. A study found that adding one robot to a geographic area reduces the need for six workers [1].

Our solution is the Hands-free Following Cart, which aims to relieve workers from carrying these heavy loads and reduce manpower. Additionally, it improves safety in the workplace through eliminating the need to manually move heavy loads. Lifting heavy objects and manually handling materials puts workers at risk of back injuries. More than 111,000 related injuries in the United States requiring days away from work were recorded in 2017 [2].

Our automated cart will track and follow the user's path while simultaneously carrying a heavy load. The cart would be able to navigate wherever a person is able to go, including ramps and elevators with enough. We also originally intended to implement additional functionality to the cart, in that the cart would be able to follow a predetermined route.

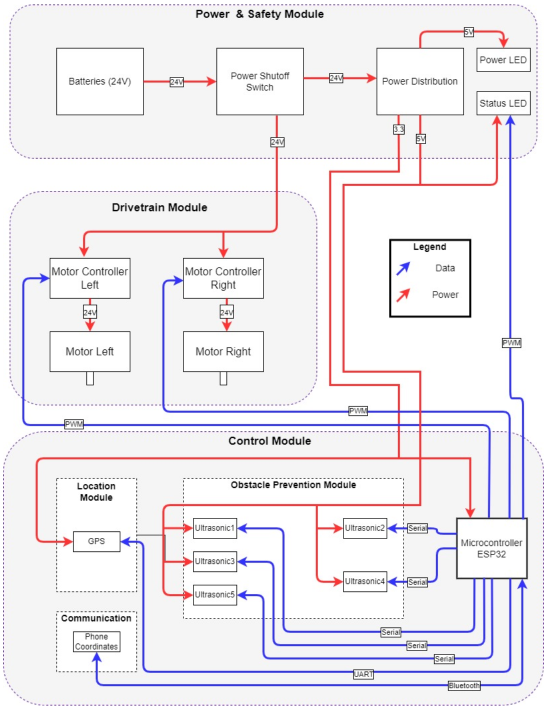

Figure 1: Block Diagram of Hands-free Following Cart

3 Design Procedure and Details

3.1 Overview

Our design can be broken down into three main subsystems: the power module, drivetrain module and the control module.

The main components of the power module are two 12V batteries connected in series to provide a total of 24V to the cart. A power shutoff switch is implemented to kill the power in case of emergency or when the cart is not in use. Regulators are used to distribute the correct voltages to each subsystem.

The drivetrain module is comprised of two motor drivers that will control the motors. The power module powers the drivetrain with 24V, and it receives PWM signals from the microcontroller.

The control module can be split into three main components, namely the GPS, location and obstacle prevention subsystem, all of which relay or receive data from the ESP32 microcontroller. The GPS subsystem consists of a NEO-6M module that communicates with the microcontroller via

UART protocol. The GPS module provides the latitude and longitude coordinates, which are used to determine the location of the cart. The location module consists of the user's mobile device which communicates with the microcontroller via Bluetooth, providing its own GPS coordinates needed for the path following functionality of the cart. Finally, the obstacle prevention module consists of five ultrasonic sensors that will send data via serial communication.

3.2 Power and Safety Module

The power and safety module of our project is designed to sufficiently power all devices at 3.3V or 5V respectively. It also is capable of supporting additional low power components using external voltage rails that have been added to our final PCB, such as LEDs. Lastly, the power system is designed to take an input of 24V from the batteries and protect the PCB against reverse polarity and short circuit. In choosing the different regulators, trace widths and components for our project, we rated them to handle more than the required current for each rail and external wiring.

Part Identifier	Part Description	Communication Protocol	Number Needed per board	Supply current per device [A]	Supply Current per board [A]	Supply Voltage Min [V]	Supply Voltage Typ [V]	Supply Voltage Max [V]	Estimated Power Consumption [W]
U2	ESP32-WROOM-32UE	Multiple available	1	0.5	0.5	3	3.3	3.6	1.65
U8	GPS Module	UART	1	0.025	0.025	2.8	3.3	4.3	0.0825
U4	Compass	SPI	1	0.00045	0.00045	2.8	3.3	3.6	0.001485
J4,J5,J6,J7,J8,J9	Ultrasonic Sensors	SERIAL	6	0.015	0.09		5		0.45
J1 & Cytron 30A DC Drivers	Motor Drivers	PWM	2	0.1	0.2	5	5	30	1
WS2812b	LEDs	PWM	60	0.025	1.5	4.8	5	5.1	7.5
									0
	Motors (Max RPM)	PWM	2	18.4	36.8	24	24	24	883.2
					0				0
					0				0
					0				0
					0				0
Max/Total	Max/Total		60	18.4	36.8	24	24	30	893.883985

	Current	Power		Board Power Consumption	Need to Rate for this
Total 3.3V	1.733985	0.52545		0.81545	1.223175
Total 5V	1.45	0.29			
Total 24V	883.2	36.8			
Battery Rating	886.383985	531.830391			

Figure 2: Power and current calculations for each voltage rail

We selected lead acid batteries as they are easier to charge and maintain for the user in the long run and can sufficiently power our cart and drivetrain without any issues. We chose to connect two 12V batteries in series to double our voltage to 24V. All of this power is run through a main shut-off switch to which the key can be removed to prevent accidentally turning the power on to the system. This power is then distributed to the motor drivers and the PCB.

Each voltage rail has its own debug LED to update the user about any issues with the power system and to let them know that the power is on. The input diode prevents the user from damaging the PCB when polarity is

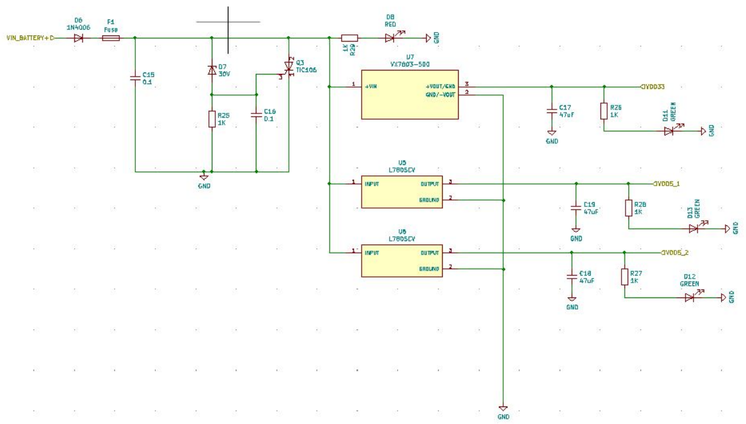

Figure 3: KiCad schematic of our power and safety module

reversed, and the following fuse protects the PCB when a short occurs which causes the system to draw over 7 amps. The status led is integrated into our system to let the user know visually that the system is malfunctioning or has run into obstacles.

3.3 Drive-train Module

The drive-train module consists of motor controllers and motors. The motor controllers take 24V power input, PWM signal to control speed and DIR signal to control the rotational direction of the motors. We decided to use 350W DC motors as they are easy to work with, replace and cost efficient. The motor controllers are capable of handling 30A continuous and 80A peak current for 3 seconds. The motors operate at 18.4A continuous, so our controllers are quite enough for our system. The calculation below shows us the speed of the cart and torque. The cart can travel at 7.1 mph with a 1:1 gear reduction between the motor and wheel, however due to some machining issues, we had to use a bigger sprocket for the wheels which reduced our speed to 5mph. This is still above the average walking speed and we are content with it. Finally, 1.7Nm of torque gives the cart enough power to move the weight from a standstill without any issues.

$$GearRatio = MotorSprocket/WheelSprocket \tag{1}$$

$$CartSpeed = GearRatio * MotorSpeed * WheelSize * \pi * 60/63360 \tag{2}$$

$$MotorOutputPower = 0.75 * MotorVoltage * NominalCurrent \tag{3}$$

Parameter	Value	Units
Motor Speed	300	RPM
Motor Sprocket	1	teeth
Wheel Sprocket	1	teeth
Wheel Size	8	in
Motor Voltage	24	VDC
Nominal Current	18.4	A
Torque	1.7	Nm
Cart Speed	7.139983304	MPH
Gear Ratio	1	:1
Motor Output Power	331.2	W
Output Torque	1.7	Nm

Figure 4: Drive-train calculations for speed, power and torque

3.4 Location Module

For our project, we chose to use a Quectel L26-M33 GPS module as it was the only one available at the time of our design in the market. We designed the module by following the hardware specifications data sheet from Quectel, however we were unable to initiate any contact with the module and were not able to process any input requests. The power to the module was proper and all the pins were correctly mapped out to the specification requirement. Therefore we abandoned this module and used a NEO-6M GPS module, and we used the pads from the depopulated Quectel module on our PCB. We were finally able to get the module to work after facing the antenna towards the sky when we had a clear line of sight to the sky.

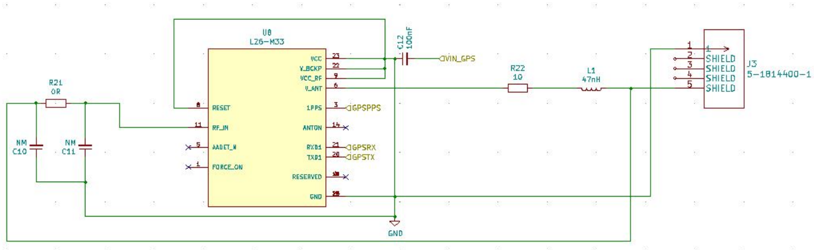

Figure 5: Schematic for Quectel L26-M33 with antenna

3.5 Obstacle Prevention Module

For our project, we used 5 ultrasonic sensors to read any distance changes in our path. This was the cheapest and most effective way to detect any obstacles in our path. We used HC-SR04 sensors which had an issue with timeout. When the trig pin sent a sound way, and was out of range of operation, then the sensors would be locked in a timeout loop until it was power cycled, which was an issue in an open field. Therefore we decided to implement a software solution to change echo pin to an input, pulse a high signal and then revert it to an output. This was a simple fix to solve the timeout issue, we also tested all the sensors we got to make sure we did not have any other issues. By detecting sudden terrain changes from our forward facing ultrasonic sensor, we can recognize any terrain we cannot have the cart travel. By detecting any distances that are low, we know we are running into an obstacle, this was a much better solution than using a Lidar sensor, which takes up cargo space and camera based system which would would be inoperable in different lighting conditions.

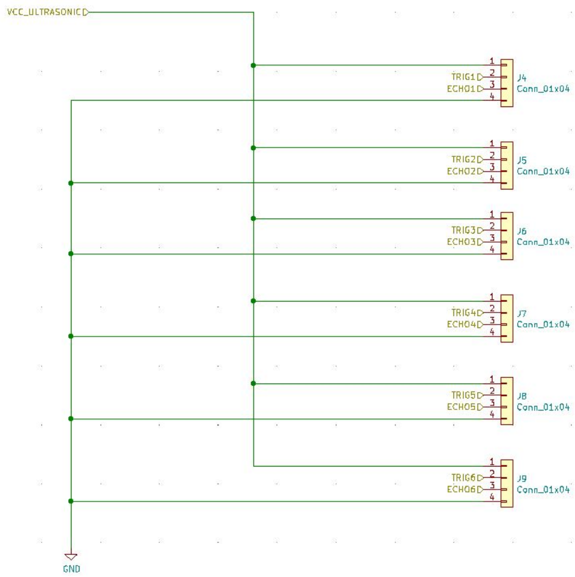

Figure 6: Schematic for Quectel L26-M33 with antenna

T = Time between emission and reception (echo and trig pulse)
C = Speed of sound

$$Distance = 1/2 * T * C \tag{4}$$

3.6 Microcontroller

For our project, we decided to use an ESP32-WROOM-32UE due to its WiFi and Bluetooth capabilities as well as the number pins and communication protocols supported by the module. All of our sensors required different communication protocols ranging from UART, I2C, Serial and PWM, so we wanted to map our signal lines to a brain that was able to support these protocols. The Bluetooth low energy from the micro-controller module meant that our design has a lower footprint, as we eliminated the need for a separate Bluetooth transceiver. While we mapped the pins to the specification, we missed a few important details on some pins about their

IO, which required us to cut a few traces and solder jumper wires to the appropriate signal pad. The microcontroller worked flawlessly in managing the sensor data, however uploading the code required us to use a USB to TTL converter as we could not find a CP2102 USB to UART bridge controller to communicate to the ESP32 via USB-micro. This was not a difficult issue as we resorted to having an RX0 and TX0 lines to communicate with the module.

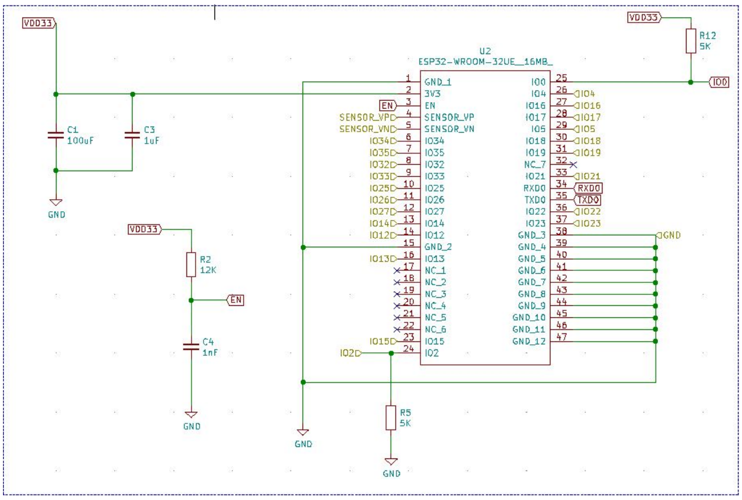

Figure 7: KiCad ESP32 schematic

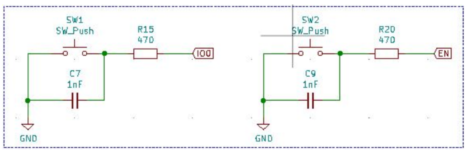

Figure 8: KiCad ESP32 schematic for reset and IO0 buttons

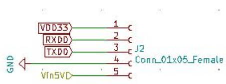

Figure 9: KiCad ESP32 schematic for RX and TX communication lines

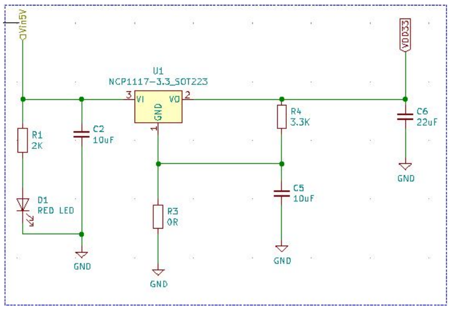

Figure 10: KiCad ESP32 schematic for power regulation

3.7 Software

For many of the subsystems, there were certain libraries that required external libraries to be able to communicate with the micro-controller.

For the drive-train module, we used Cytron motor drivers, for which a library already existed that supported Arduino IDE. In order to configure the software for our cart, we had to set predetermined PWM and DIR pins correctly. However, the library had functions that were not compatible with the ESP32 micro-controller, namely the analogWrite() function which was used to update the status of analog pins and in our case, address the PWM pins, was not supported by the ESP32. We did note however, that the ledcWrite() function, which was part of the Arduino Core library, provided similar functionality. Thus, our solution was to provide an analogWrite() polyfill for the ESP32 framework by wrapping the ledc library.

As previously mentioned, the control subsystem can be divided into three main components, the GPS, location, and obstacle prevention modules. While we originally intended to use a Quectel GPS chip, we were unsuccessfully in getting the chip to function, and so resorted to a NEO-6M module, which required that we use the Arduino TinyGPS++ library and also the SoftwareSerial library which the ESP32 did not support, but we were able to find an implementation of the library for the ESP32.

To communicate with the mobile phone, we intended to use BLE, warranting our choice of the ESP32 with built in BLE functionality. For this, we used the Arduino BluetoothSerial library, which enabled us send and receive data, which would be reflected on the serial monitor. Finally, the obstacle prevention module did not require additionally libraries - we simply had to set the corresponding trig and echo pins accordingly.

To implement path following functionality, we originally planned to use

Blynk, a platform for building interfaces to control and monitor hardware devices. However, we encountered many issues in setting up the datastreams for each subsystem to be integrated as well as problems obtaining GPS coordinates from the phone due to privacy issues.

4 Verification

4.1 Power and Safety

The first subsystem completed was the Power and Safety subsystem. Most of the tests for this subsystem consisted of measuring voltages. We used a voltmeter to measure the voltage of the batteries when in series and measured the voltage rails after stepping the voltage down to the proper values for the sensors. The voltage from the batteries was measured at over 25 Volts, within the range we listed in our Requirements and Verifications table. The voltages we measure along the stepped-down voltage rails were at 5.06 Volts and 3.27 Volts, both within their respective ranges.

Power & Safety System	Batteries	1. Provide ample input power to use the cart under normal conditions for up to 30 minutes. 2. Input voltage between 24 to 27 volts 3. Hardcase, non-spillable	2. Use a voltmeter to measure and confirm battery output is between 24-27 volts.
	Power Shutoff Switch	1. Hard wired to the input power source to cut off power to all subsystems. 2. Rated for 2x the max current of the whole system. 3. Easily accessible by the user	1. Connect power to at least one subsystem through the shutoff switch. Flip switch and determine if the power was cut off from the subsystem.
	Power Distribution Board	1. Step down the input voltages to: a. 5V +/- 0.2V b. 3.6V +/- 0.2V c. 3.3V +/- 0.1V	1. Use a voltmeter to check each output pin of the power distribution board and confirm they are the correct voltage.

Figure 11: Power and Safety Requirements and Verifications

4.2 Drivetrain

The main requirements for the Drivetrain subsystem were that the motor drivers needed to be able to withstand a high enough current and the motors needed to be able to move a maximum weight of 200 pounds.

The current rating requirement was satisfied with the parts we bought. Our motors were rated for a current of 18.4 Amps while the motor controllers were rated for 30 Amps continuously and a peak of 80 Amps. Therefore this requirement was met.

To test the weight requirement of the drivetrain, we purchased cinder blocks, each weighing about 30 pounds, for a total of 120 pounds. The cart itself weighed about 80 pounds. We then put the cinder blocks on the cart for a total of 200 pounds and drove it through a sequence of forward, backward, and turning motions. The cart was able to move easily with the cart without slowing down. We also tested the cart with much more weight, over 300 pounds total, with the same result.

4.3 Control

4.3.1 GPS

When first testing the GPS module, we learned that we needed to go outside to so that the module would have a clear line of sight to the sky and could then establish it's position. To test this, we went outside with the module, then moved it up and down at varying speeds to see if the cart could establish it's position and we could track it's motion.

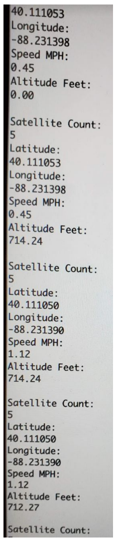

Figure 12: GPS Module Test

In the figure above, the results of the test are shown to be a success. The serial monitor shows the GPS module at an established position. It then moves up with a certain speed, and a new altitude is established. The module then moves downward at a greater speed and the change is reflected in the results.

4.3.2 Bluetooth

An important part of the product is the ability of the cart to communicate with a device via Bluetooth. As seen in our Requirements and Verifications Table (see Appendix), we needed our cart to be able to communicate with a device up to 30 meters away.

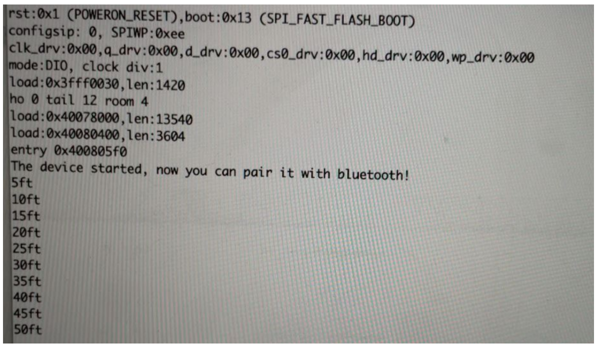

Figure 13: Bluetooth Communication Distance Test

In the figure above, it shows the Bluetooth test performed. We connected a device to the module then stood a certain distance away. We then typed that distance on the phone and it was printed on the serial monitor of the computer connected to the Bluetooth module. The figure shows the distance being increased in 5m increments until we are far beyond the necessary communication distance.

4.3.3 Ultrasonic Sensors

To test the ultrasonic sensors, we simply placed them all face-up on a flat surface then moved an object over them while taking a stream of data points. The test ensured that the sensors were working and that we could indeed see objects in front of them. The test also revealed that the accuracy of our ultrasonic sensors were within 4cm, as stated in our requirements. As multiple sensors were pointed at a surface about 127cm away, they all reported a distance within 1cm of each other. This is also seen with two other sensors placed at a slightly different height, only a few centimeters higher. They saw the same surface and again reported distances within 2cm of each other.

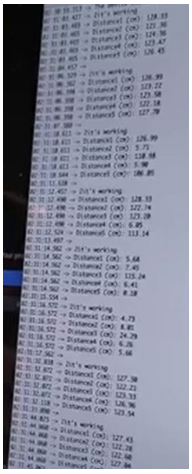

Figure 14: Ultrasonic Sensor Test

4.3.4 Microcontroller

One obviously important requirement for the microcontroller was being able to communicate with all the sensors on the cart, such as the ultrasonic sensors and the GPS module. From the test performed with each sensors, it can be seen that communication was established.

Another requirement for the microcontroller was being able to compute and store way points from the connected device's GPS for the autonomous motion of the cart. However, we were unable to verify this because of problems revolving around receiving actual coordinated from the connected device. We tried using different methods for finding the GPS coordinates of the device, however, each method had problems we were unable to resolve. For example, there were privacy issues when trying to use iOS to obtain the coordinates. Also, when trying to the Blynk app, we were unable to properly connect and receive the coordinates.

5 Cost

After doing a little research, we found that the average hourly salary for a computer/electrical engineer is about $50 per hour. Then we estimated that our project had taken about 300 hours to complete. Given this formula:

$$salary(hourly) * hours * 2.5 = cost/engineer$$

We found that the labor cost per engineer was $37,500.00. With three engineers in our group, the total labor cost comes out to $112,500.00.

For the Machine Shop cost, we estimated an hourly salary of $75 and a working time of 35 hours. Using the same equation as above, we found that the Machine Shop cost comes out to $6,562.50.

The table below shows the parts cost. The project cost is $610.27

Part Name	Manufacturer/Supplier	Part Number	Link	Unit Price	Quantity	Total Price
GPS Module	Quectel	L76-M33	Mouser	11.88	2	$23.76
12V SLA Battery	AJC MotoTec	T9FB1793821	Amazon	25.99	2	$51.98
Battery Isolater Switch	Shin Chin	A23-7B	Mouser	13.8	1	$13.80
LEDs	Amazon	WS2812B	Amazon	15.99	1	$15.99
Ultrasonic Sensors	Adafruit Industries LLC	3942	Digi-Key	3.95	9	$35.55
24V Motors	Unitemotor	MY1016	Amazon	59	2	$118.00
Motor Drivers	Cytron	RB-CYT-133	RobotShop	34.38	2	$68.76
GPS Module	NEO	NEO-6M	Amazon	11.98	1	$11.98
Course Funding	ECE445	-	-	111.48	1	$111.48
8" Pneumatic Wheels	Haul-Master	47638	HarborFreight	8.99	2	$17.98
5" Swivel Caster Wheels	Haul-Master	69852	HarborFreight	7.99	2	$15.98
Mouser Order 1	Mouser	-	Mouser	17.17	1	$17.17
Mouser Order 2	Mouser	-	Mouser	28.72	1	$28.72
Digikey Order 1	Digikey	-	Digi-Key	25.39	1	$25.39
Digikey Order 1	Digikey	-	Digi-Key	53.73	1	$53.73
					Total	$610.27
					Course Funding	$100.00
					Out of Pocket Group	$510.27
					Out of Pocket/Person	$170.09

Figure 15: Parts List and Cost

Adding the total parts cost, the total engineer labor cost, and the Machine Shop cost gives us a total project cost of $119,672.77.

6 Conclusion

6.1 Product Conclusion

Our project was able to meet most of the high-level requirements and all individual subsystems were fully functional. The issue of being unable to obtain a mobile phone's GPS coordinates prevented the complete system integration of all modules, but given additionally time, we believe we could achieve the full scope of our initial proposal.

The cart was able to move with no loss of speed carrying weight well above our listed requirement of 150-200 pounds, and was able to maintain a speed of up to 7mph. The listed safety features such as status LEDs and the safety shutoff switch were successfully implemented, and obstacle prevention was fully functional when tested as an individual subsystem. This was a priority as we wanted to uphold the IEEE code of ethics which will be mentioned in the next section.

6.2 Ethics and Safety

The ultimate goal of this project is to improve productivity and safety in the workplace. With this in mind, we need to take into consideration the possible risks that the automated cart will pose in a work environment, such as what actions will be taken in the event that a human being obstructs its path or when collision with an obstacle could result in injury. Thus, we must uphold #1 of the IEEE code of Ethics, which is 'to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to promptly disclose factors that might endanger the public or the environment,' [3] and also #9 in the code - 'to avoid injuring others, their property, reputation, or employment by false or malicious actions, rumors or any other verbal or physical abuses.' [3]

We aimed to have the cart moving at a top speed of 7-8 miles an hour, double the average walking speed, and so it must be ensured that the cart is able to recognize positive and negative obstacles and react accordingly, so as to eliminate the chance of colliding with a worker. The event of a worker accidentally walking into the cart must also be taken into consideration, and so we cannot have any sharp objects protruding from the cart.

Our cart is powered by a 12V lead acid battery, which can cause serious injury if not handled properly. Firstly, the battery releases highly flammable gases while charging (hydrogen and oxygen) which can result in an explosion. The acid in the battery is also very corrosive and can cause damage to property and injury if it comes into contact with skin. Thus, we must ensure the battery is stored/mounted in a cool, well-ventilated area and to never handle the battery near heat or open flames. [4]

Throughout the course of the project, we were very open-minded in respect to taking criticisms and suggestions to improve upon design and execution. Thus, we adhered to #5 of the IEEE code of Ethics, which is to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic in stating claims or estimates based on available data, and to properly credit the contributions of others.' [3]

7 References

[1] Mit Sloan.mit.edu, (2020) A new study measures the actual impact of robots on jobs. It's significant. [online] Available at https://mitsloan.mit.edu/ideas-made-to-matter [Accessed 29 Sep. 2021].

[2] Injury Facts.nsc.org. (2021) National Safety Council Injury Facts. [online] Available at: https://injuryfacts.nsc.org/work/industry-incidence-rates [Accessed 29 Sep. 2021].

[3] Ieee.org. (2020). IEEE Code of Ethics. [online] Available at: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed 16 Sep. 2021].

[4] Safework.sa.gove.au (2020) Lead Acid Batteries. [online] Available at: https://www.safework.sa.gov.au/workplaces/chemicals-substances-and-explosives/lead-acid-batteries

8 Appendices

8.1 Appendix A Circuit and PCB

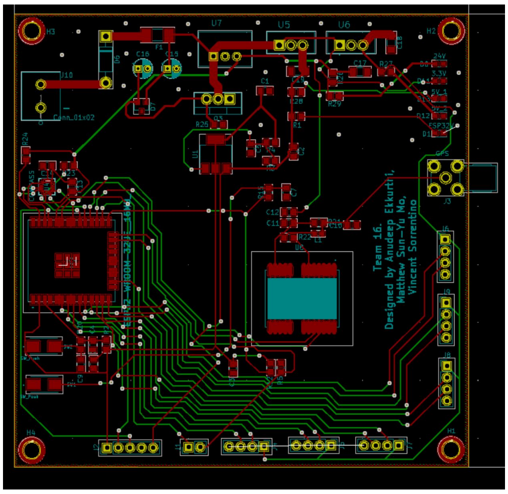

Figure 16: PCB Layout

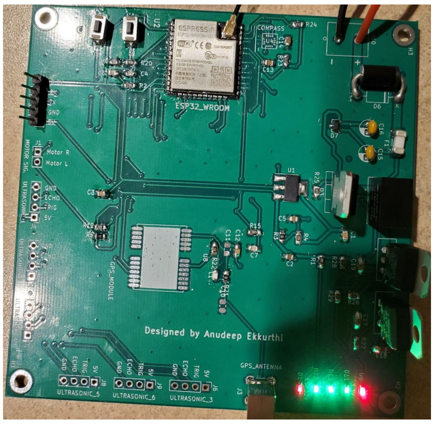

Figure 17: Powered PCB

8.2 Appendix B Requirements and Verifications Table

Module/Subsystem		Requirements	Verification
Power & Safety System	Batteries	1. Provide ample input power to use the cart under normal conditions for up to 30 minutes. 2. Input voltage between 24 to 27 volts 3. Hardcase, non-spillable	2. Use a voltmeter to measure and confirm battery output is between 24-27 volts.
	Power Shutoff Switch	1. Hard wired to the input power source to cut off power to all subsystems. 2. Rated for 2x the max current of the whole system. 3. Easily accessible by the user	1. Connect power to at least one subsystem through the shutoff switch. Flip switch and determine if the power was cut off from the subsystem.
	Power Distribution Board	1. Step down the input voltages to: a. 5V +/- 0.2V b. 3.6V +/- 0.2V c. 3.3V +/- 0.1V	1. Use a voltmeter to check each output pin of the power distribution board and confirm they are the correct voltage.
Drivetrain Subsystem	Motor Controllers	1. Rated for up to 60A under stall conditions 2. Control the rpm with a tolerance of +/- 3 rpm	1. A. Run motor under stall conditions B. Use an ammeter to confirm less than 60 A present.
	Motors	1. DC motor with enough power to start moving a max load of up to 200 pounds. 2. Output shaft compatible with standard wheels hubs without any need for custom adapters 3. Shaft must be able to spin both directions	1. A. Load cart B. Drive motors to confirm they can move 3. A. Use test code to drive in forward direction B. Then switch the backward direction
Control Subsystem	Compass	1. Tells the system the direction of the cart with an accuracy of 1-2 degrees.	1. A. Print cart's compass reading to screen B. turn the module C. measure rotation and

		2. Updates faster than 60Hz 3. I2C communication	determine if it's within 1-2 degrees
	GPS Module	1. Establish the current position of the cart outdoors within a 2 meter accuracy 2. Must update faster than 10Hz 3. UART communication	1. A. Get the value of position from the module B. compare with known accurate reading, i.e. phone application 2. Check speed at which updates happen and determine if they're faster than the required speed 3. Connect through UART
	Bluetooth Transceiver	1. Send and receive data between the cart and a phone within 30ft 2. UART or I2C communication	1. Stand within 30ft and check connection to module 2. Communicate using UART or I2C
	Ultrasonic Sensors	1. Range of up to 7 meters 2. Accuracy greater than 3 cm 3. Wave direction of up 45 degrees or greater	1. Move away from ultrasonic sensors. Measure the location of where the sensor stops reading. Determine if it is greater than 7 meters. 2. A. Move an object in front of an ultrasonic sensor. Mark its location. B. Move it and compare its new location with the sensor's readout. C. Move it in increasingly small increments. Determine if the sensor is accurate up to 3cm. 3. Place the object in front of the sensor, move it to the sides. Until it's no longer visible to the sensor. Measure the angle and determine if it's greater than 45 degrees.
	Microcontroller	1. Must be able to receive data from all the above sensors and modules and	1. A. set up sensors B. print sensor values to screen

		send data to certain modules 2. Must be able to compute a path direction given new points within 2 seconds. 3. Must be able to store gps waypoints in memory for cart to follow 4. Must be able to notify user if it notices an error or be able to change LED color to notify user of cart status	C. change GPS transmitter location and confirm changes D.move objects in front of each US sensor to confirm detection 2. A. Manually input 2 coordinates for calculation B. compute and check time elapsed 3. A. Move GPS transmitter to get waypoints B. print waypoints as received C. after three waypoints, print stored values D. compare with previously printed waypoints 4. A. Create stalled setup with sensors B. confirm LED's change Bluetooth 1. Develop a simple application that sends a signal to the microcontroller, probe the V_{in} using an oscilloscope to determine if a pulse was registered. 2. Repeat the above steps, test for a distance of 10meters.