Tinjac Is Not Just A Cart

Shekh Arefen, John McFarland, James Johnson, Edgar Velez

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — This paper presents the design methodology utilized to design, construct, and test an autonomous shopping cart that utilizes motion and proximity sensors so that no physical touching of the cart is involved in order to navigate to a desired location. The motivation for this project stems from inspiration in the wave of "smart" technology. The main goal of this project is to create a device that's accessible for all types of people, mainly for those that have some sort of physical impairment due to an injury while also keeping the overall budget for the unit as cheap as possible.**

***Index Terms* — Autonomous vehicles, infrared sensors, microcontrollers, motors, shortest path problem, sonar detection.**

I. Introduction

With technology advancing faster than people can adapt, the future of retail and grocery shopping will soon be inundated with digital monitors, scan-and-go mobile apps, and cashier less checkout services. Despite some of the negative stigmas this may bring forward, it ultimately will provide shoppers with the swiftest and most efficient shopping experience.

Not only will these technological advancements increase overall effectiveness, it will aid those customers that are physically unable to go shopping for themselves or require some sort of assistance due to a handicap, injury, or any other impediments that they may face on a daily basis. So now we ask: How can we make shopping fast and easy for those that need help while still holding on to some of the more traditional shopping characteristics?

Our project idea is a self-driving shopping cart that would utilize motion and proximity sensors so that no physical touching of the cart is involved in order to navigate to the desired location.

The design would work where the customer chooses which grocery items they want on an LCD screen, and based on this, the cart would navigate to where these items are by using the shortest distance possible. This would utilize proximity and motion sensors to navigate. Proximity would be used to ensure the cart doesn't run into objects, or other customers, and the motion sensors would be used to ensure that the cart driver is following behind, if not, the cart would stop until the driver resumes walking. The cart would also automatically adjust to anyone's walking speed.

The power source for the cart is to be based on solar cells since they're environmentally friendly and would be a quiet power source in comparison to a gas motor. The solar cells will convert the energy from the sun and store them into rechargeable batteries which will power on the cart. Solar energy would work especially well for a shopping cart since when they're not in use, they're most likely outside, sitting in the sun, and gaining energy. The cart is perfect for any shopper while also benefiting the environment, and a great alternative to the classic cart design.

Tinjac could be considered nothing short of an improvement into the shopper's experience. Once fully functional and having implemented the above components into the system as a whole, Tinjac would create a friendly environment where the user could simply walk hands-free to their destination, accident free.

II. Existing Products and Relevant Technologies

There are a number of smart inventions that help to improve the shopping experience for customers. This inspiration was initiated in response to Amazon's new store, Amazon Go, where the most impressive feature of the store is that it's completely cashier-less. The customers can simply just walk out of the store with their items, as the items are already scanned and then automatically billed to the customer through the Amazon app. By using a series of cameras and sensors, the inventory management is able to determine which item is picked up, based on weight, location, image analysis, and a shopper's checkout history [1].

Along with Amazon, Walmart has brought the most interesting advancements to the table. In 2016, Walmart unveiled their plans in development that includes a driverless shopping cart which guides customers to the items they desire, and robots that scan the shelves to check the stock, as well as the ability for customers to be tracked through the usage of wearables [2]. Walmart has also put into place some measures to ensure that every item is in stock by having tower like robots roam the store and take pictures of the shelves and analyze them to check if a product is missing [3].

III. Requirements Specifications

The requirement specification section covers the intended features of the Tinjac. It goes over the core features that the group decided should be implemented into the Tinjac design and that it should be fully functional when it is finished.

- An LCD display that allows a user to select from a given list of grocery items. The cart then takes the customer to each item. Items are able to be added/removed from the grocery list even after the initial list is inputted.
- The shopping cart will have a selected store's display programmed to know where items on the list are located.
- The shopping cart should be able to avoid obstacles using IR and sonar module sensors to calculate the shortest path to gather all the items on the list.
- The cart should be able to know when the user steps away from it and will stop accordingly.
- The battery on the shopping cart is rechargeable and will use a solar panel to supplement this as well.
- Solar panel to supply 12V to a rechargeable battery.
- The weight should be kept under 15lbs, with the weight of the cart being 10 lbs. or less.
- Cart should be able to hold at least 50 lbs. and still function without slowdown.
- Dimensions are 21 by 24.5 inches. A regular sized cart won't be used due to budgeting.

IV. Component Selection and Prototype Design

A. Component Selection

In order for our shopping cart to be as fully autonomous as possible, we must be able to strategically pick out each single component and determine the compatibility it would have on the other parts of the system. This may be determined by numerous factors such as: cost, power usage, force (or rotational force), physical size, and more.

Figure 4.1 and Figure 4.2 show the specific parts that we finalized on our choices. In Figure 2.1 it is apparent which of the main components we chose for the system as a whole, but neglects some other (less impactful) parts such as the cart frame itself. Figure 4.2 is the specific details of the model and functionality of the specific core components we decided to go with. This is listed to help provide insight to why we chose these specific models over some of the competing choices. Once a fully-functioning prototype is designed we may alter these decisions and adapt them to meet realistic restraints that we did not foresee.

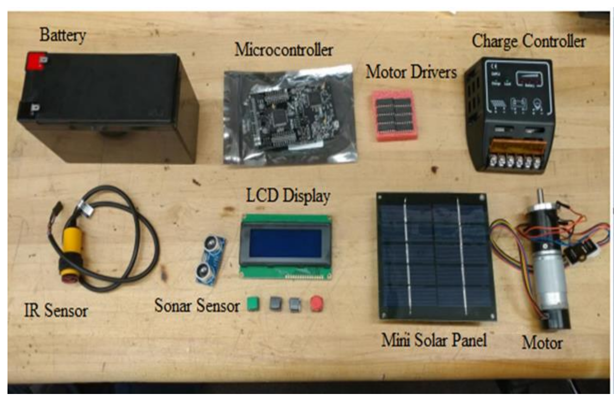


Figure 4.1 Individual Components

Part	Description
ExpertPower EXP1270 Battery	A lead acid battery that is capable of being recharged.
MSP430F5529	The microcontroller being used to control the whole system.
L293D Motor Driver	A 16 pin motor driver that will be used to help regulate the two motors.
Anself Charge Controller	The charge controller that was chosen to be able to regulate the power coming from the solar panel to the battery without overcharging.
Gowoops E18-D80NK	The infrared sensor module to be used to detect objects. There will be 7 of these placed on the cart.
HC-SR04 Sonar	The ultrasonic sonar module to be used to help detect motion and the lack of motion. There will be 2 of these placed on the cart.
Riorand LCD	This display will be used to act as a user interface when paired with some buttons.
Sunnytech Solar Panel	A small solar panel capable of charging the battery chosen.
12V 146RPM DC Motor w/Encoder	This motor was chosen based on the high torque it has and the low rotations per minute to meet walking speed.

Figure 4.2 Table of components selected and their descriptions

B. Prototype Design

Tinjac will be utilizing a microcontroller to run the code which is a large part of this project. Because the map layout of the grocery store will be hard coded, we will be able to accurately measure distance traveled so that it can reach the destinations provided by the users. It should also be able to calculate the shortest path to gather all the items. The primary method in which the user will interact with Tinjac is to use the LCD display which will allow users to add and remove items to their shopping list.

Two different types of sensors will be used to detect obstacles: IR sensors and sonar sensors. Once an obstacle is detected by Tinjac, it will proceed to travel around the object and continue on its path to the next item on the list. The sensors will also be used to detect when the user steps away from Tinjac so that it knows to stop and wait for the user to return.

The power supply that Tinjac will use is a 12V battery that will be recharged using solar panels. In order to prevent overcharging, a solar panel charger control will be used to stop charging the battery once it is completely charged. This battery will be used to power everything on Tinjac including the motors. Displayed below is the raw design of Tinjac and how it will look like once everything has been integrated into place.

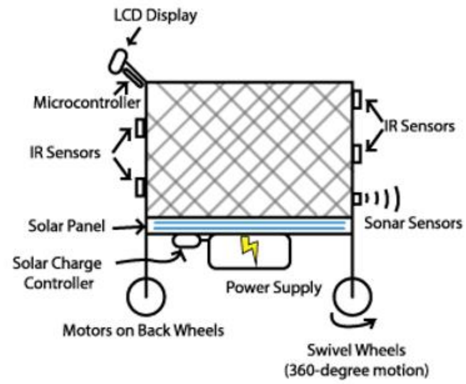


Figure 4.3 Prototype Design Architecture

V. Hardware Design

This section details the hardware schematics of Tinjac. Included is an overall schematic that will be used to create a printed circuit board (PCB), as well as schematics that give more detail on the individual subsystems and how they function individually..

A. Overall Schematic Design

Figure 5.1 below shows our complete schematic design of all the necessary connections needed in order to get the most out of each requirement's functionality.

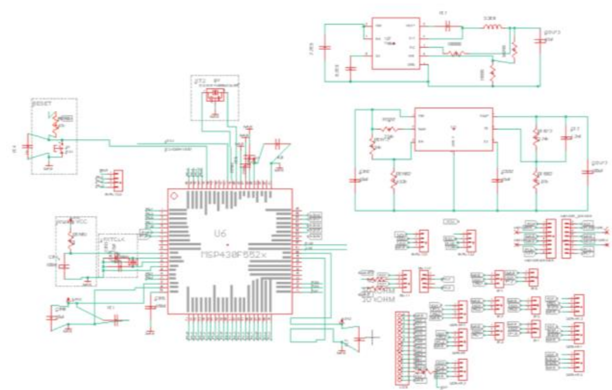


Figure 5.1 Overall Schematic Design

B. Printed Circuit Board Design

Figure 5.2 below shows what the printed circuit board looks like based on the schematic. The footprints in the printed circuit board provides easy access to the pins so that all the devices will be connected to the board with no problems. A ground pour was added onto the printed circuit board so that all the ground connections could be easily connected, and it would cause the board to be a lot more accurate, with less noise. For viewing, the ground pour was left unfilled so that the PCB can be seen easier above. The ground pour uses thermal connections so that soldering onto the board will be easier.

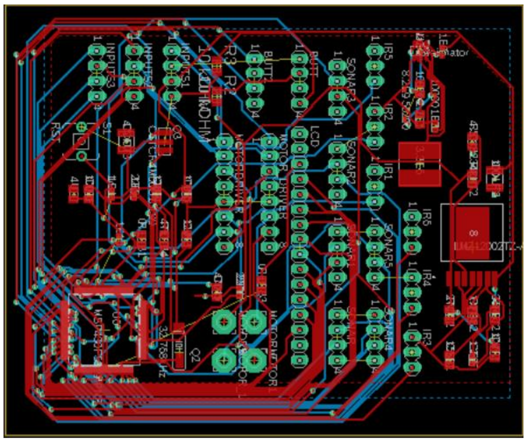


Figure 5.2 PCB Schematic

C. Subsystem I - Detection

The first subsystem of Tinjac is the network of sensory modules on the cart. The sonar and the IR sensors communicate with each other, then back to the microcontroller. In Figure 5.3, we connected the two types of sensory tools Tinjac will be optimizing (they are set up in separate circuits, but displayed here as a single picture). We wired each pin the sensory tool uses to properly test the accuracy and functionality of each part.

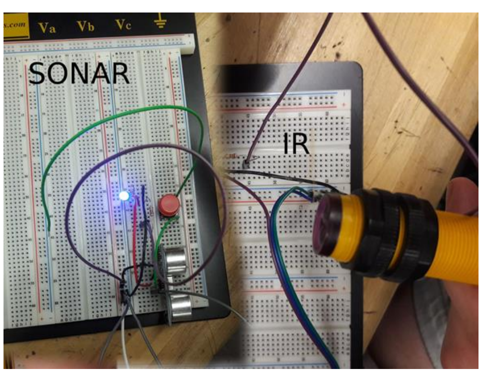


Figure 5.3 Sensor Modules

The code within the microcontroller handles the input taken in from the first subsystem and uses it to produce any necessary outputs to other subsystems depending on the configuration. The main thing this subsystem needs to accomplish is collision detection to prevent the cart from running into obstacles in front of it. Once an obstacle is detected, it should then begin its routine to bypass the obstacle and get around it. Its secondary function is that the sensors on the back of the cart should be able to notice when the user of the cart steps away from the cart and should produce the appropriate response which is to stop and wait for the user to return. In order to accomplish these tasks, there will be multiple sensors placed on the cart.

D. Subsystem II - Motion

Our second subsystem is composed of our system of motors and triggers to enable/disable them. Part of this subsystem also includes the code functions that control the direction of the motor as well as enabling/disabling them through the microcontroller. The motor driver chip acts as the middle-man between the motors and the microcontroller, only turning on the motors when they receive the proper signal from the microcontroller.

While the motor has an internal encoder to manipulate the speed and direction of the motor, we are actually using a motor driver chip instead to be able to control the motor (as it was actually much more simple to integrate than using the encoder itself).

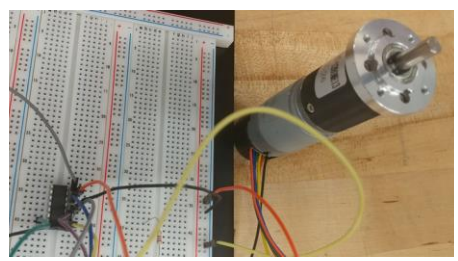


Figure 5.4 Motor and Motor Drives

E. Subsystem III – User Interface

The third subsystem of our project is the user interface. The user interface allows the user to input the items they would like to shop for. It can also communicate any issues with the user during the cart's progress.

Figure 5.5 LCD User Interface with Buttons

The LCD panel itself will connect directly to the microcontroller and will be updated through the pins that it connects to on the microcontroller. It will also get any error or warning notifications pushed to it from the microcontroller as well. The module will be powered through one of the microcontroller pins dedicated to voltage supply.

In the displayed picture (Figure 5.5), we made a simple circuit to connect the LCD display to a few pins to be able to detect that it can receive and input signal. We also put into view the buttons we intend on using for our early prototype designs of Tinjac.

F. Subsystem IV - Power

The fourth subsystem is our powering functionality. Tinjac uses a large battery to allow maximum amount of usage before needing to recharge. The connected solar panel allows the battery to be recharged when the cart is not being used.

The solar charge module is what prevents the battery from being overcharged from the solar panels. It also provides other useful information, like the battery life remaining, when its charging and also when load can be turned on. The battery will supply power to all the other subsystems in Tinjac, including the motors (which require the most amount of power).

To show both the variation in a large and a small solar panel for Tinjac, Figure 5.6 shows the two types of solar panels connected to our charge modules which is then connected to our battery source.

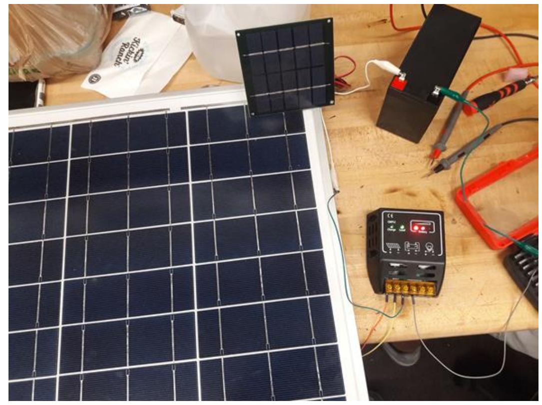


Figure 5.6 Solar panel and Solar charge module connected to the battery

G. Subsystem V - Frame

The last division of our major system is the actual skeleton of the system. Composing of the cart and modified wheels, the physical framing of Tinjac is what holds all the other subsystems together and allows them to connect directly.

The back wheels of the system will have to be custom-made and designed to fit our motors so they can move and turn with pinpoint precision. The front wheels of the cart will have to be modified to allow smooth turning and avoiding alterations to our straight path design. The bottom of the cart will be modified so that it can house the solar panel at the base, but still allow items to be placed on top. This will allow the cart to have an ideal center of gravity for movement, while also allowing the cart to recharge properly when the journey is complete (and the cart has been emptied).

Figure 5.7 shows the base foundation for what we intend to use as Tinjac's skeletal system.

Figure 5.7 Base Frame of the Shopping Cart

VI. SOFTWARE DESIGN

Using the MSP430 microcontroller, our software structure will be solely constructed using the C language and using the standard 'Divide and Conquer' approach that we have decided to utilize in order to break down each objective into smaller necessary functions. After each core objective is broken down into smaller functions, a simple flowchart can map out how the function will work at a basic level. The five main core functions are:

1. User Interface
2. Create Path
3. Run Cart
4. Obstacle Detection
5. Movement Correction

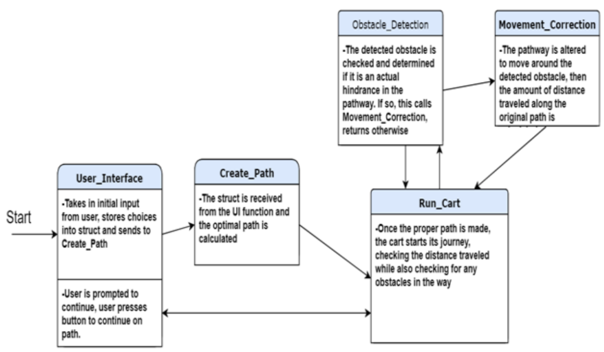


Figure 6.1 Software Function Flowchart

The above flowchart (Figure 6.1) is an illustration on exactly how Tinjac will work in an optimal setting using all of the core functions.

A. User Interface

This function will be tasked with taking in the user input from the physical buttons on the physical panel of Tinjac and update the LCD in accordance with which button was hit. In order to properly display every grocery item available to purchase, we will have an array of grocery type objects (such as "Eggs", "Milk", "Cereal", etc.). The items are programmed beforehand (similarly to how we will already have a predetermined store layout), and we will print it to the LCD screen by traversing the array. Once an item is selected, it will be removed from the array and placed in a separate array of "Chosen Items". This new array of items that the shopper selects will be filled and sent to the Create Path function.

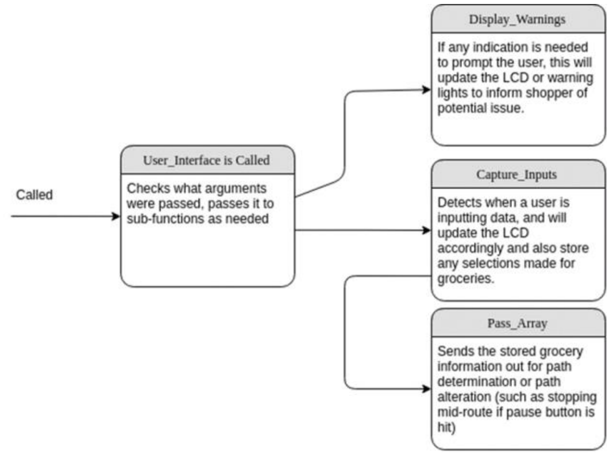


Figure 6.2 User Interface Basic Functionality

As displayed above, the user interface function will be designed to handle several user-related function calls, as well as system management related functions. This will be divided into sub-functions for easier management and error avoidance.

B. Create Path

This function takes in a struct of all the grocery items that the shopper wishes to add to the cart, then calculates the absolute shortest path. It will reference a predetermined layout of the store, with entrance and checkout locations.

Since this would consist of very few possible combinations of possibilities, the type of algorithm we use to find the shortest path will have little real-life impact on performance and calculation. However, for a compromise between efficiency and simplicity, we will implement a basic Dijkstra algorithm to find the optimal path. Once an accurate path is made, it is passed to the Run Cart function, so that continuous progress can be monitored.

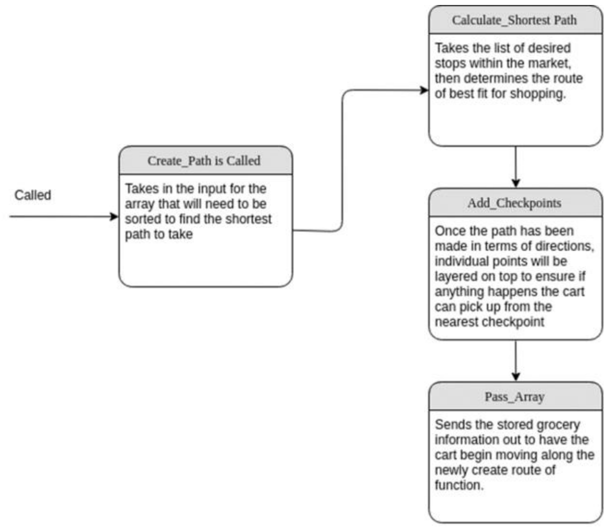


Figure 6.3 Create Path Basic Functionality

Once a list of the items we want to obtain is inputted, the cart will begin to calculate the shortest path and proceed to the locations.

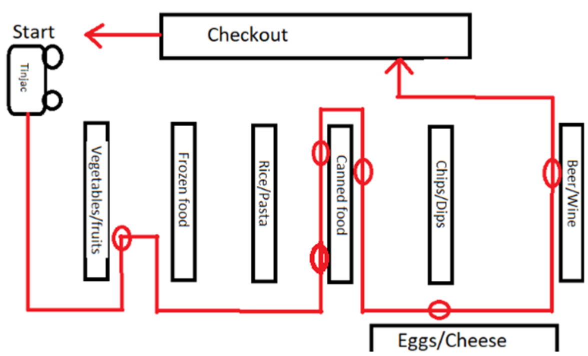


Figure 6.4 Example of store layout and each stop point

Figure 6.4 is a rough sketch of optimal pathing we envision Tinjac to be able to determine on its own. The algorithm that is being implemented is Dijkstra's shortest path algorithm. This algorithm essentially takes a starting point and an adjacency matrix that lays out which point is where in relation to one another, and then calculates the shortest path to each point. We have modified the algorithm to find the most optimal points between just one point and the starting point. This cycle continues as the cart moves to the next item on the user's list.

Each circle represents a "stop" needed by the shopper in order to grab all of the groceries that are needed to complete their trip to the store.

C. Run Cart

As the cart physically moves forward, we will use distance detection along with collision checks to monitor when the cart arrives at each checkpoint. A "checkpoint" is simply a designated point within reach of one of the items the shopper entered. Once the next point in the path is reached, the cart will stall until the user prompts that they are ready for the next stop in the path. If any obstacle occurs during this process, the Obstacle Detection function is called mid-run until the specific collision is resolved.

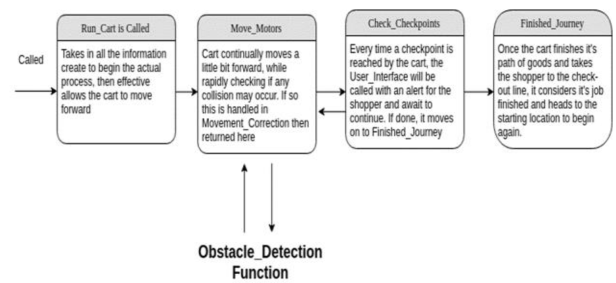


Figure 6.5 Run Cart Basic Functionality

D. Obstacle Detection

The obstacle detection function has a single, yet invaluable, task. We structured the Object Detection function to work in combination with every form of detection on the cart. Since we are using both sonar and IR detecting modules, we have broken this function up into sub-methods so we can accurately check to see that the obstacle being detected is definitively a threat to the path.

When each module senses an object in its field of detection, the microcontroller is then able to call any appropriate sub-function and will use the combination of detection modules to pinpoint where the obstacle is in relation to the cart's position. This information gets passed to the movement correction function. If the object is determined not to be a threat to the path, then the Obstacle Detection simply sends that information back out so the cart can continue its journey.

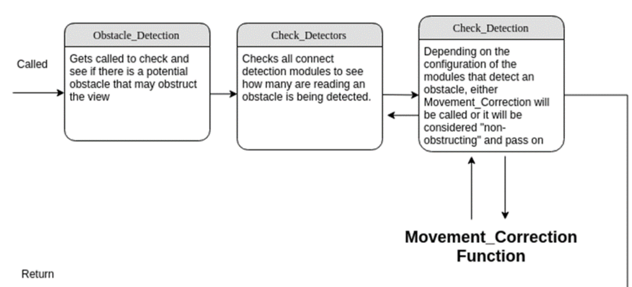


Figure 6.6 Obstacle Detection Basic Functionality

E. Movement Correction

Using the detection modules, the Object Detection function checks if there is an obstacle and where it is in relation to the cart. The Movement Correction function will take this relative position and trigger the motors in the specified way needed to move the cart out of the way of the obstacle, but still on the path towards the objective.

Once this correction has been made, the Movement Correction function will use the distance traveled to calculate how much further along the path the cart has travelled. Then it will exit out and return all the way back to the Run Cart function with the updated distance so the checkpoint can continue to be accurately monitored.

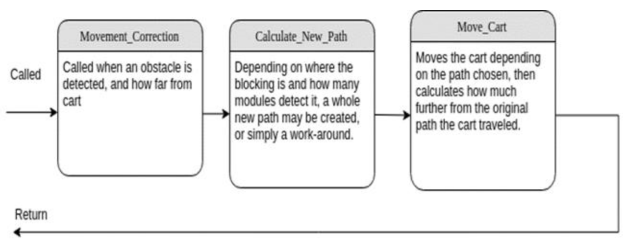


Figure 6.7 Movement Correction Basic Functionality

VII. CONCLUSION

In this ever-changing society where technology affects the way we live, learn, and work, there will always come advancements that aim to facilitate the many processes of human behavior. Our self-driving shopping cart aims to help those that require assistance by removing the physical burden of having to push around a shopping cart and by allowing the customer to simply input the desired location of where it wants the cart to go. We wanted this project to help improve the lives of others while trying to maintain a hold on the evolving technology around us.

ACKNOWLEDGEMENT

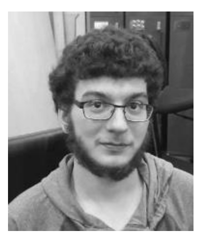
We would like to thank Dr. Lei Wei and Dr. Samuel Richie for their continuous advisement and support throughout the last two semesters, and the University of Central Florida for a fantastic four years filled with successes, failures, knowledge, and most important, love.

BIOGRAPHY

Shekh Arefen is a 22 year old soon to be University of Central Florida Computer Engineering graduate. He has accepted an offer as a Systems Engineer with Lockheed Martin in Orlando. He will be pursuing his Masters in Computer Engineering in the near future.

John McFarland is a 25 year old soon to be University of Central Florida Computer Engineering graduate. Right after graduation, he will be starting his PhD career in Computer Engineering with the University of Central Florida. His focus will be in Computer Architecture, working directly with Dr. Amro Awad.

James Johnson is a 20 year old soon to be University of Central Florida Electrical Engineering graduate. His future dream goals involve working with renewable energy and the environmental as well as hardware design.



Edgar Velez is a 24 year old soon to be University of Central Florida Electrical Engineering graduate. He plans to work in the subfield of signal processing and communications and later pursue both his Masters and PhD in Electrical Engineering

REFERENCES

[1] Bishop, Todd. "How 'Amazon Go' Works: The Technology behind the Online Retailer's Groundbreaking New Grocery Store." *GeekWire*, GeekWire, 14 Jan. 2017, www.geekwire.com/2016/amazon-go-works-technology-behind-online-retailers-groundbreaking-new-grocery-store/.

[2] Menegus, Bryan. "Walmart Considering Smart Carts, Drone Assistants." *Gizmodo*, Gizmodo, 22 Mar. 2018, https://gizmodo.com/walmart-considering-smart-carts-drone-assistants-1823991966

[3] "Why Walmart Is Testing Robots In Stores-and Here's What It Learned." *Fortune*, Fortune, fortune.com/2018/03/26/walmart-robot-bossa-nova/.