SELF CHECKOUT CART

Team 13 — Rohan Khanna, Pooja Kankani and Cherian K Cherian

ECE 445 Project Proposal —Spring 2020

TA: Johan Mufata

1 Introduction

1.1 Objective

Self checkout terminals have become an essential component of any major supermarket chain. They cut labor costs and make the checkout process efficient. However, these terminals are very inefficient when it comes to scanning items that do not have barcodes such as fruits and vegetables. For such items either the customer or the supermarket staff have to manually enter the information of the item. There are many categories and subcategories of fruits and vegetables which make this process very inefficient and time consuming. If the staff member forgets the lookup code of a fruit or vegetable then it can cause a large amount of delays for customers. Gizmido reports that one of the main reasons people hated self checkout carts was that a cashier or supervisor had to be called over to enter some arcane code into the system for every other item.[1]

We propose to solve this problem by designing a display box that will be placed next to each shelf with grocery items likely to be missing a tag. This box will contain a programmable LCD screen that shows the information for the items placed on the shelf. For example, if the shelf stores apples, the LCD screen will show the price per pound for the apples, the type of apples, and other useful information like when the apples were stocked or their specific kind. This can be changed by the store managers if the product on the shelf changes. The box will also have a weighing platform that the user can place the items on, and the LCD screen will show a barcode that stores the corresponding price for that weight of that particular item (e.g., if the user places 1 pound apples on the platform, the barcode will store the price for 1 pound of the apples using the preprogrammed price for that shelf). There will be a detachable scanner attached to each cart at the entrance. The person shopping can use the scanner that they have attached to their carts to scan the barcode. It will keep adding the price for all the items they have shopped. This way the scanner will keep the total bill of the person ready and they can pay using an app, without having to wait in any check out counter lines. This makes it extremely easy to self checkout. The handheld scanner can also be used to scan the barcode of packaged products like bread and will add the cost of those items to the customer bill as well. The display box will also include an option to print out a barcode sticker, whereby the customer can just put the sticker on the plastic wrapper holding the groceries and then proceed to checkout if the customer is uncomfortable doing the labor of self checkout.

1.2 Background

Currently, self checkout scanners expect the customer to manually search for the item that they are buying or enter its reference code. Most customers struggle with finding the exact reference code and end up needing assistance from a store employee. Since there is usually only one employee looking over multiple self checkout stations, there is great delay in receiving assistance and the whole point of the self checkout process is eliminated. Vox reports that a lot of people end up not buying certain items or just outright steal items because they get angry that an item won't scan and figure it's not their job to try that hard. [2] This is a huge loss for the supermarket. Moreover, given that there are multiple sub categories of items such as fruits, customers can enter the wrong information at the

checkout screen and end up paying less or more for their product. The English newspaper Guardian reports that since people can search for items
without a barcode on the self checkout machine, they try to sneak in more expensive items by keying them in as some other cheaper items.

The fall 2014 self checkout project tried to address this issue and develop a solution to this problem. It tries to use computer vision to automatically recognize fruits and vegetables. It uses a camera to take images of an item which will rotate on the scale and then tries to determine what the item is. Such an approach can have many issues and hence we decided to approach this problem in a different manner. The following are three main issues with the solution and how our approach is radically different and free from these issues.

1. Accuracy of computer vision

   ISSUE: Considering no machine learning model has a 100 % accuracy, there can be a lot of false positives. Apples can easily be confused with oranges or pears. Moreover, There are items that have certain subtle differences that cannot be visually seen from the outside. For example, alphonso mangoes look exactly like Atualfo mangoes but alphonso tend to cost more. Having many false positives can hinder customer experience and cause long delays in checkout lines.

   OUR APPROACH : Our solution involves a handheld scanner and an LCD screen with a tag at each fruit and vegetable station. When the scanner scans the tag it will know which item the customer is dealing with and the cost will be calculated on the scanner itself. There is hence no chance of a false positive.

2. Processing power

   ISSUE: Even if the computer vision part of the project can be done with a reasonable accuracy, it requires extreme processing power. The current approach of the group uses expensive raspberry pi's which alone account for 70 dollars in the total cost of the project. These raspberry pi's have a speed of 1.4 GHZ but our microcontroller only has a max speed of 28MHz. The difference between the two is about 1.37 GHZ This high processing power is needed for the image processing component of the original project. Since our project has relatively simple processes we only need the 28MHz microprocessor.

   OUR APPROACH: Our solution has very simple components with no major processing power required. The microcontroller in the hand held scanner just has to be responsible for the scanning. The LED screen will consume

minimal power as it just has to display basic information about the section.

3. Time sink

   ISSUE: Rotating the fruit and continuously taking images will waste customer time and lead to long lines. Also since each fruit item has to be weighed to generate the cost, every fruit has to be individually put on the scale and scanned. According to the original project the aim was to make the entire computer vision project take about 4 seconds for each fruit. This is extremely large as you need to do it for every single fruit and this will cause delays.

   OUR APPROACH: Our approach does not have any component requiring the customer to wait and hence there is no question of a line. Secondly since the cost of the items is calculated on the handheld scanner, and the scanner is synced to an app on the phone. The customer can easily just pay on the phone and leave without having to stand in lines.

   Our solution hence has no computer vision component, no camera taking images and is not located at the checkout counter. Instead, it is a handheld scanner that interacts with static lcd screens at different locations in the store which is a radically different approach from the first project proposal.

1.3 High-Level Requirements

- The display box should automatically sync with an online server every 5 hours and update contents of its LCD display such as the time the produce arrived, quantity left and other relevant information

- When the user places an item on the scale, the LCD screen should show the price barcode for the selected weight for 15 seconds before going back to the price/pound screen

- Once the item is scanned, there must be add/delete options for the item available on the software application cart within 1s and an option to make a payment through the app.

2 Design

2.1 Block Diagram

Our Block diagram is split up into 3 parts. The first is a programmable shelf box. This will be used for items such as fruits and vegetables for which it is difficult to weigh and scan items. Our solution involves using a load sensor to be able to weigh the item. Once we have weighed it, the Shelf box will calculate the cost and other information and display a barcode with the appropriate information. This can then be scanned by the handheld scanner to input the information for the quantity of the items you want to buy. The shelf is programmable so that we can change the item information depending on the item it is near.

The second part of our block diagram is the handheld scanner. This is a detachable scanner which can be used to scan the barcodes of any item. It contains a WiFi chip to connect to the server of the store to retrieve item information from it. Every time an object is scanned, it looks up the price, and updates the list of items and total cost stored in it. For items such as fruits whose price is dependent on weight, it scans the barcode from the programmable shelf box to receive information about the item. The handheld scanner also interacts with the Software app using the WiFi chip to send information about the items to the app.

The software app is used so that customers can view the items they have scanned and the total cost of all the items. It can also be used for payment so that checkout is done quicker.

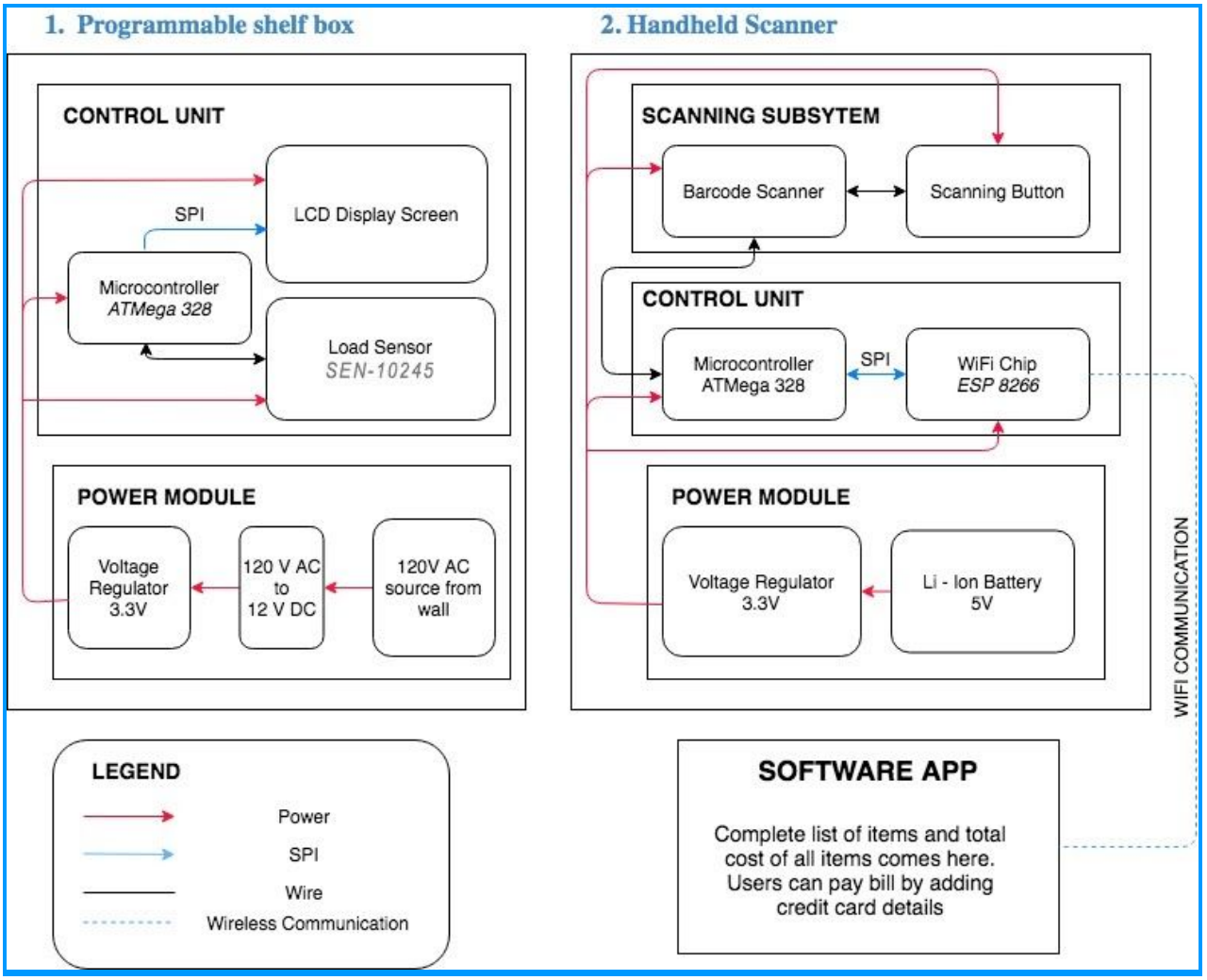

Figure 1. Block Diagram

2.2 Physical Diagram

We chose a scanner because it can easily be attached to the cart on a holder. Most people are familiar with how scanners work and hence the learning curve is minimal

The box is designed to be as small as possible for easy installment in the grocery area. The LED display gives much needed information about the product and has the potential to store a lot of information about the product.

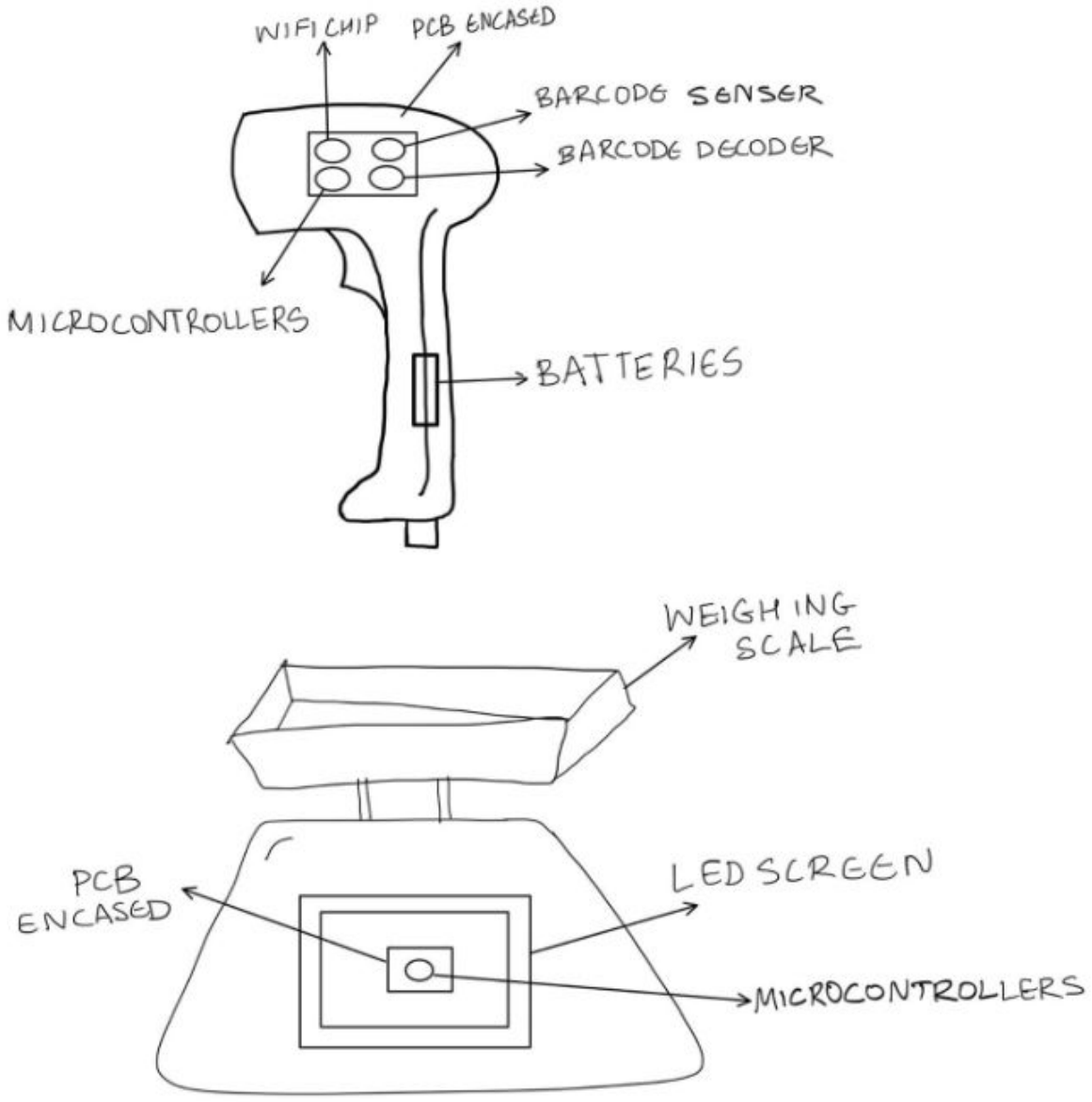

Figure 2. Physical Diagram of the System

2.3 Function Overview and Block Requirements

1) Programmable shelf box

This box will be placed on the shelf of grocery items and will consist of a weighing scale that the users can weigh grocery on. This weighing scale will communicate with the microcontroller to generate a barcode specific to that item and the weight the user selected. The barcode will show on the LCD screen and convey the total price for that item for that user.

I. Power Supply

The power supply will be a 120V AC source provided from the wall. This will be used to power the entire circuit. WE chose this instead of batteries so that they do not need to be moved or charged often and can have a constant source of power all the time.

a) AC to DC converter

This will be used to convert the AC source received from the wall to a DC source which can be sent to the voltage regulator and used by the circuit.

Requirement 1: Should be able to receive the 120V AC and convert to 12V DC power supply.

b) Voltage Regulator

Is used to convert the 12V power supply from the AC to DC converter to the 3.3V power supply requirements of the LCD screen, microcontroller and load sensor. A linear voltage regulator is used. We do not require a low dropout regulator since we are going from 12V to 3.3V.

Requirement 1: Be able to provide a clean 3.3V power supply required for the LED screen, Wifi Chip and Load Sensor.

II. Control Unit

The control unit is the computing center of the device. It is responsible for calculating the weight of the products, finding price based on this and displaying information and barcode to be scanned.

a) Microcontroller:

Specification: ATMega 328

The microcontroller will receive the weight of the components from the load sensor and will allow the agent to program a price for the items on that shelf. Using the weight per pound and the total pounds, it will calculate the total price for that customer and generate a barcode that will be sent to the LCD screen.

Requirement 1: Should be able to receive weight information from the load sensor and use this to calculate the cost of the items.

Requirement 2: Should be able to produce a barcode with the appropriate information and send this to the LCD screen to display.

b) LCD Display

Specification: Adafruit Industries 358

The LCD display screen will first show the price per pound of the item on the shelf. Once it receives a barcode from the microcontroller for a particular price (price/pound * pounds picked by user), it will display that on the screen.

Requirement 1: Should be able to receive barcode from the microcontroller and display it along with any other information needed.

c) Load Sensor:
Specification: SEN 10245

The load sensor will be placed under a platform that the user can put the grocery bag onto. This will help calculate the total weight of the contents and send that to the microcontroller.

Requirement 1: Should be able to calculate the weight of any item upto 110 pounds.

Requirement 2: Should be able to send the calculated weight to the microcontroller.

2) Handheld Scanner

I. Power Supply

The power supply is provided by 5V Lithium Ion rechargeable batteries. This will be passed to a voltage regulator to provide the appropriate power supply required for each of the components in the circuit. As this is rechargeable, the employees can put the hand held scanners on charge when they are running low and it can be used again the next day.

a) Li - Ion Battery:
A rechargeable Li - Ion 5V battery that is used to power all parts of the circuit will be used. It can be recharged at the end when the store closes and should be ready for usage the next day.

Requirement 1: Should be able to provide power to all parts of the circuit with appropriate voltage.

Requirement 2: Should be able to provide power for more than 14Hrs which is the duration the store is open in a day.

b) Voltage Regulator:
Voltage regulator is used to convert the 5V power supply from the Li-Ion battery to the 3.3V power supply requirements of the microcontroller, WiFi chip and barcode scanner. A linear voltage regulator is used. We do not require a low dropout regulator since we are going from 5V to 3.3V which has voltage to prevent too much dropout.

Requirement 1: Should be able to convert the 5V power source from the battery into 3.3V which can be used by the microcontroller, Wifi chip and barcode scanner.

II. Control Unit

The control unit for the handheld scanner contains a system that can read and understand barcodes to figure out the price of the item. This price is sent to the microcontroller that has the user id stored, which sends that information to the WiFi chip for communication to the user application.

a) Microcontroller:
Specification: ATMega 328

The microcontroller will receive the price information from the barcode scanning subsystem and convert that from binary. It will then use the unique user id to send that information to the user mobile application through WiFi communication.

Requirement 1: Should be able receive information from the barcode scanner and use that to calculate total cost of product scanned.

Requirement 2: Should be able to receive and send information to the WiFi chip through SPI protocol which will then be used for communication with the server and app.

b) WiFi chip
Specification: ESP8266

The WiFi chip will send data of purchases made by the user from the hand held scanner to the software application using UART communication. The information will be sent to the user application based on the unique identifier received from the microcontroller.

Requirement 1: Should be able to receive and send information to the microcontroller through SPI protocol since it is used to communicate with the server and app.

Requirement 2: Should be able to communicate with the store server to receive information about items sent to it by microcontroller.

Requirement 3: Should be able to communicate with the software app to send information about all the items scanned to the app.

III. Scanning subsystem

a) Barcode Scanner: The barcode scanner will flash red light on the LCD barcode and an electronic circuit will read the black and white lines to generate

pulses. Black lines would be off-pulses and white lines would be on-pulses. It is then converted to binary 1s or 0s and this number will be sent to the microcontroller for conversion.
Requirement 1: Should be able to receive input from the button and start scanning only when this has been received and stop once the button is not being pressed.

Requirement 2: Should be able to read barcodes of various products and send information received to microcontroller.

b) Button: A push button indicating when to start scanning for a product. This will start the red flash light in the scanning subsystem and start the barcode scanning process.

3) Software App

The software application will get the information of all the grocery purchases made by the user from the hand held scanner and can be used as a personal shopping cart to keep track of items and quantities. It will get all the information through the WiFi chip on the hand held scanner. Once the user is done shopping, the application will allow the user to make payments for the items directly to the store through a secure online payment method. It can also be used by the user to keep track of a buying list, to make the buying process easier for the customer.

Requirement 1: Should be able to receive information about items scanned from the handheld scanner and display this on the app.

Requirement 2: Should allow customers to enter credit card information and use this to pay for items, allowing faster checkout.

2.4 Risk Analysis

Signal processing is an essential component to the success of this project. Although every team member has had basic exposure to signal processing on a theoretical level, no one in the team has experience in applying those theoretical concepts in a practical scenario. The team has at best a basic understanding of the signal processing required for scanners. We must wisely use the help of the course staff personnel that have had experience in practical applications of signal processing.

Secondly, sensitivity errors in the weighing component of the display box or errors in the communication channel could cause great issues in cost computation and lead to inaccurate costs being determined. Since this component needs to be smaller than a traditional weighing machine or one placed near self checkout carts, the size could affect accuracy and hence display incorrect costs for the user. To prevent this we could have a weight average mechanism that quickly computes three readings of the weight of the item and takes average to determine cost. In addition, we could try to use past data of fruits and vegetable weights to determine whether a measured weight is significantly inaccurate from the actual weight.

Finally, a critical component of our project is that the barcode generation, the scanning and the transfer of data to the mobile application must all happen very quickly within a max timeout of 5s. All of these heavily rely on the wifi quality at the supermarket. If the Wifi signal is not strong there could be major lag in this process and hence cause inconvenience to the customer rather than improve the customer experience.

3 Safety and Ethics

Our project has quite a few safety or ethical issues based on our current project design and vision. We will completely abide by the IEEE Code of Ethics [3], throughout the development of our project and its usage. We will follow guideline #1 in the IEEE Code of Ethics, and our main goal is the wellbeing of our users and of the environment, while making the shopping experience easier and more efficient for all. We will be transparent about our design and limitations, so that the project cannot harm people. We will make ethical design choices and hold the user's safety and well-being as the main priority while designing and implementing the project.

Through our scanner system, the grocery stores would get a lot of data about customer buying behavior that could be used by the store for different purposes. We will ensure that we do not store or allow misuse of any personal data of the users. We will make our system reliable to prevent data manipulation and information stealing through strong internal security measures and robust data transfer mechanisms. On the software application, we will allow users to pay online and we have to make sure that the payment data is securely stored and properly encrypted to prevent stealing. The user will also be given preference on whether they want to store that information or enter it everytime. The financial details of customers will not be shared or used in any way.

According to point #3 in the IEEE Code of Ethics, we will be honest and transparent with the user about any data we collect and how it will be used. Our project has a hand held device that allows users to scan barcodes. For user safety, we will make sure that the device is safe and does not, in any situation, cause any harm to the user. The current within the device will be cut off through a fuse if any circuit errors are detected. This will ensure that the user cannot get an electric shock due to exposed parts or other issues. Following guideline #7, we will always be willing to improve and correct any errors and criticism we receive, to improve the product and make it more useful.

The Li-ion batteries in the handheld scanner can pose risks that can be very harmful for human life, due to overheating and overcharging. This can cause the scanner to catch fire, while in the hand of customers. To ensure that this does not happen, the employees will be trained on how to charge the scanners and will be given instructions for proper storage. While working with the batteries, we will be careful and closely follow the instruction manual to ensure our safety and those of the people around us.

4 References

[1]https://gizmodo.com/why-self-checkout-is-and-has-always-been-the-worst-1833106695 [Accessed: 04/01/2020].

[2]https://www.vox.com/the-goods/2018/10/2/17923050/self-checkout-amazon-walmart-automation-jobs-surveillance [Accessed: 04/01/2020].

[3] Institute of Electrical and Electronics Engineers, 'IEEE Code of Ethics', 2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 04/01/2020].