Poker Chip Counting Companion

ECE 445 Design Document - Fall 2022

Project # 16

Adish Patil, David Hahn, Forrest Hare

TA: Li, Qingyu

1. Introduction

1.1 Problem

The game of Poker is a family comparing card game, in which players wager over the best hand; a hand is an order or arrangement of cards [1]. Since the beginning of the 20th century, this game has increased in popularity, professionally and, more importantly, recreationally. A traditional game of Poker is simple to play regarding the equipment needed; all players need is a case of chips and a deck of cards. However, the setup and conclusion are the greatest barriers to playing a game. Before, during, and after every Poker game, players must hand count different colored chips that equate to different cent/dollar values. Hand counting chips when setting up a poker game is a cumbersome task that is time-consuming and can lead to mistakes. In addition, calculating the exact distribution of chips and buyouts for multiple people can result in errors such as players receiving too little or too many chips or the wrong payout. These errors can be further exacerbated by those playing the game for the first time as they lack experience setting up poker chip stacks.

1.2 Solution

Our project aims to solve all of the inconveniences of setting up and finishing a game of Poker. The Poker Chip Counting Companion has several features. Firstly, it will accurately dispense poker chips based on user inputs about the game, such as the buy-in and big/small blind metrics (Dispensing State). The device will also calculate the appropriate amount of each color chip required, which takes the guesswork out of figuring out the proper chip stacks when starting a game. At the end of a game, the device will switch to an operating buyout mode (Collection State), which will then correctly count the remaining stack sizes of each player. The Power Subsystem will power the entire machine. All user interaction will occur through the phone App subsystem, connected to the device via a Bluetooth connection, where a user can input information, control the machine's state, and receive information about the game.

Ultimately, the Poker Chip Counting Companion reduces the time needed to set up and end a poker game, increasing their willingness to play. Furthermore, it correctly counts chips so the players can guarantee they receive the proper amount based on their buy-in. Finally, the phone application makes it easy for the consumer to interact with the machine.

Many aspects make our product marketable. With a construction cost under $200 and no major existing competitors, this can be a viable product for any poker player in the consumer market. Unfortunately, current technology servicing this problem only exists in high-budget casinos. Nothing of the sort except standard poker chip containers exist on consumer marketplaces. Additionally, the average poker player doesn't understand the rules behind distributing chips or buyouts; the companion can reduce that barrier to entry by doing all the grunt work and allowing players to focus on the gameplay itself.

1.3 Visual Aid

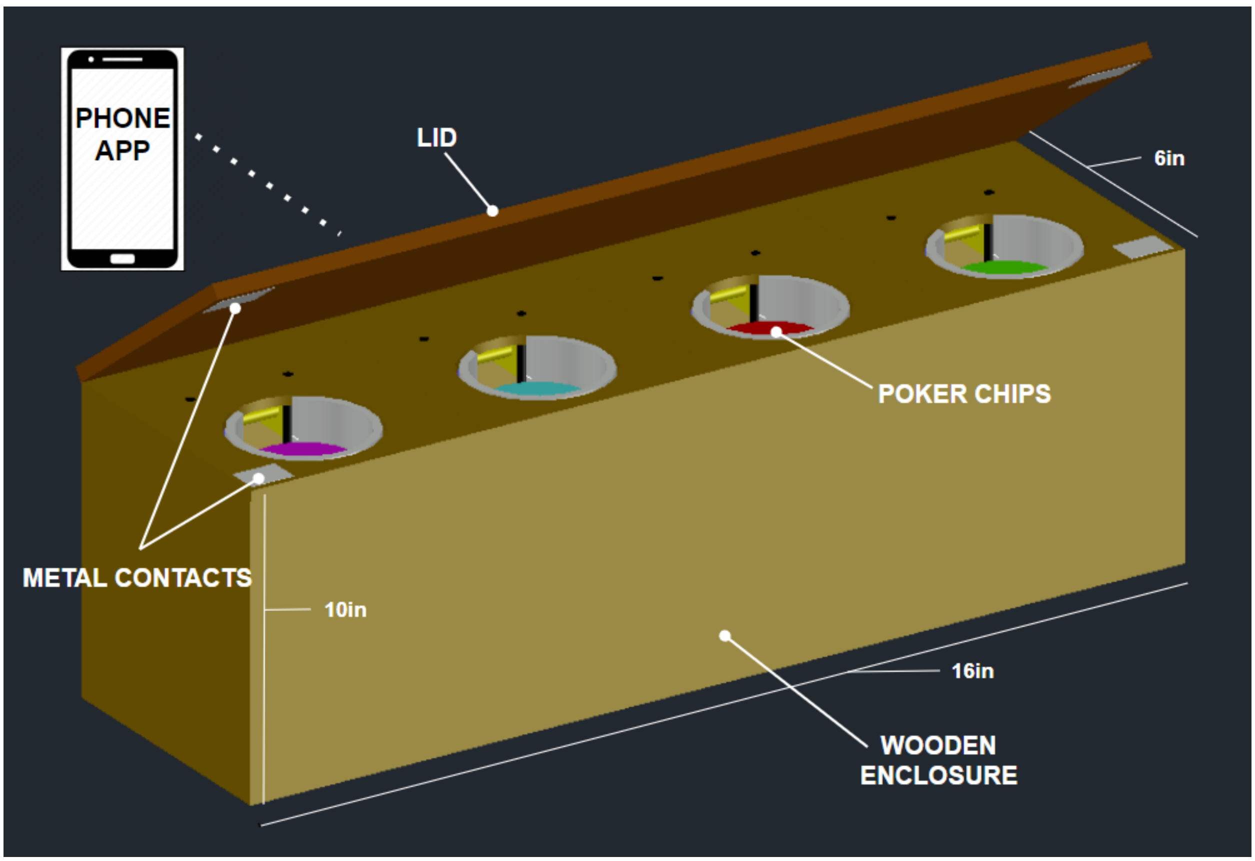

Figure 1: Poker Chip Counting Companion (North-East View)

The chip counting machine will be fully enclosed within a wooden box, with a lid on top attached to a hinge. The overall dimensions of the box will be 16” x 6” x 10”. The machine will be paired to a phone application and will operate based on information sent and received via bluetooth.

1.4 High Level Requirements

To consider our project successful, our device must fulfill the following:

- The device will have a capacity for at least 100 poker chips and will dispense them at a rate of 5 chips per second ∓ 0.5 seconds.
- A phone application that can receive and relay information to the machine with a response time within $800ms \mp 160ms$.
- At the end of a game when the users have deposited their remaining chips into the device, it will count their chips under 20 seconds.

2. Design

2.1 Physical Design

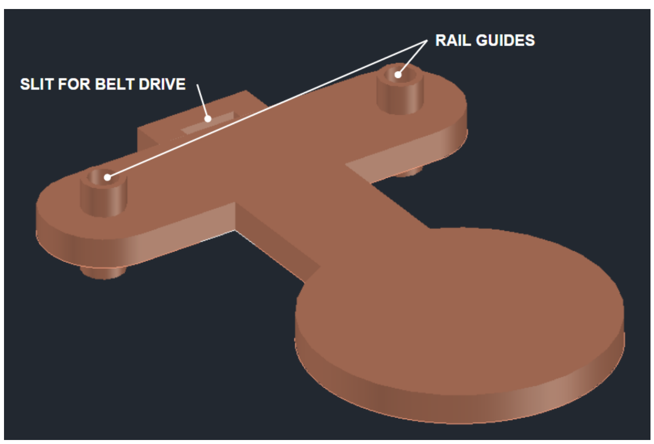

Figure 2: Chip 'Platform' (North-East View)

This platform (Figure 2) will move up and down with the motor belt drive. The platform will be guided with two metal rods (shown in Figure 3) and the chips will be placed on top of the platform.

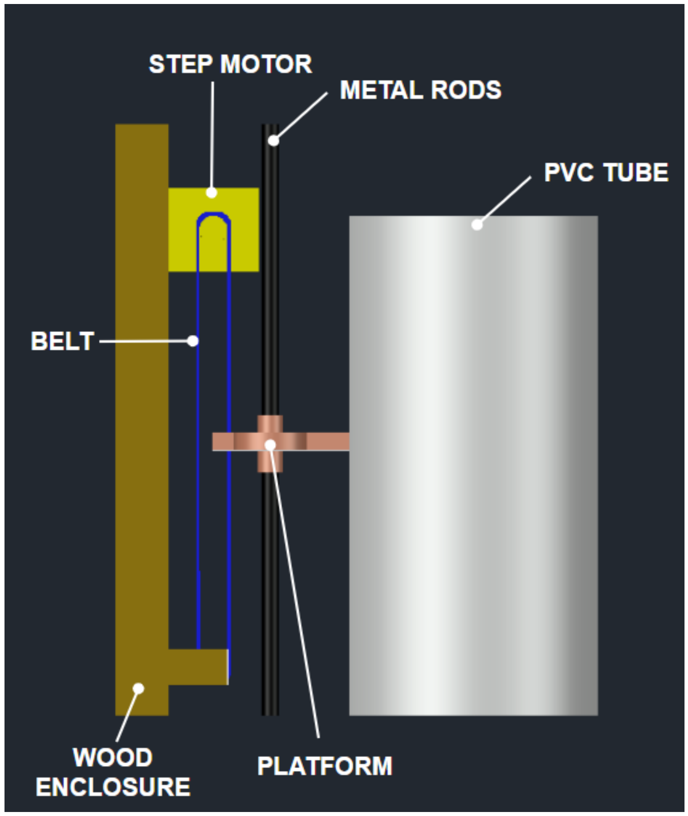

Figure 3: Elevator System (Side View)

This side view (Figure 3) shows the belt driven elevator system for raising and lowering the platform. Part of the platform and chips will be enclosed within a PVC tube such that the chips will be stacked neatly.

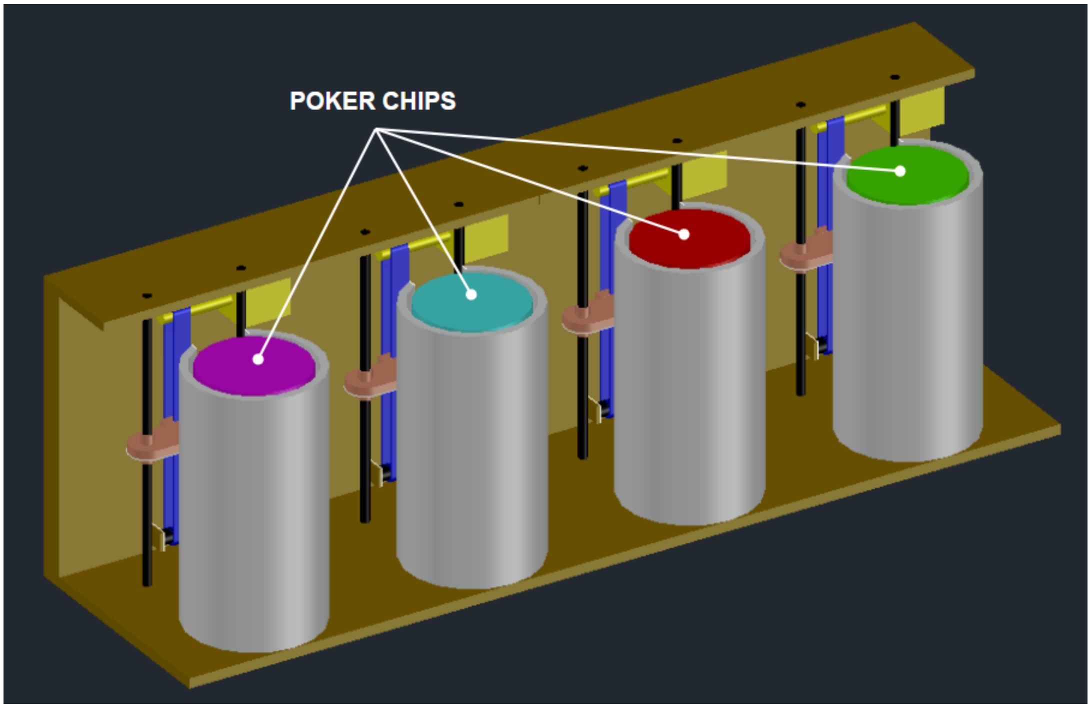

Figure 4: Open Enclosure (North-East View)

Our design consists of 4 identical elevator systems which each correspond to a different stack of colored chips. As shown in Figure 4, there are 4 chip colors which in a game of poker correspond to different cent/dollar denominations.

2.2 Block Diagram

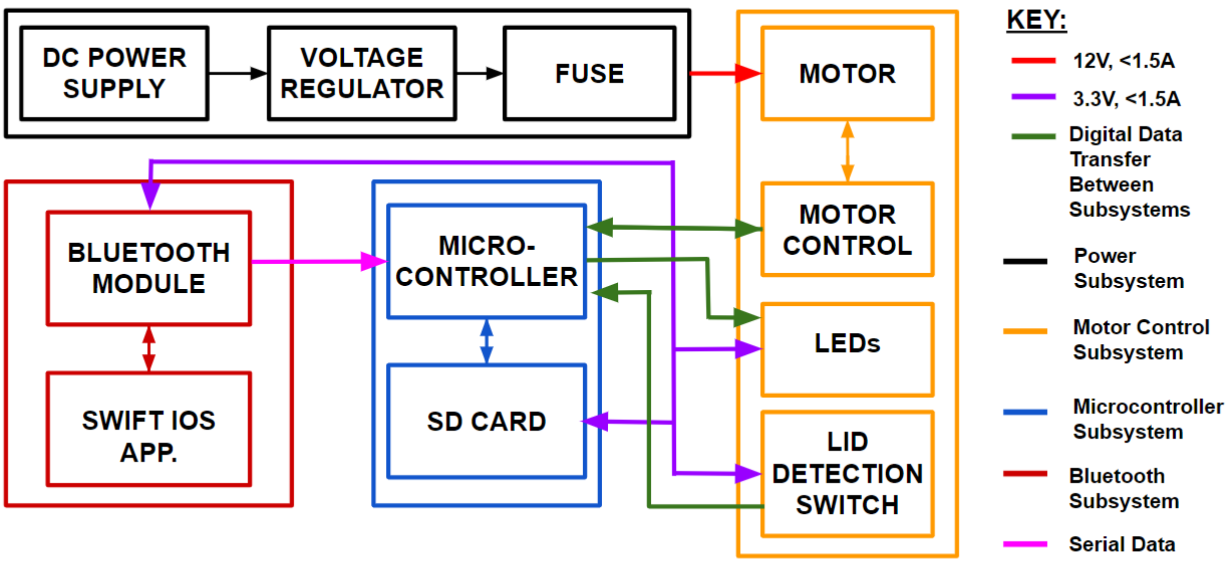

Figure 5: Block Diagram of Poker Chip Counting Companion

2.3 Subsystems

2.3.1 Power

This subsystem will provide the needed power to each component of the circuit. Using a standard 12 V connection. There will need to be voltage and current regulation to prevent damage to components on the PCB. An initial 12 V will need to be reduced to 3.3 V for the digital components but the original 12 V will be needed to drive the motors.

To produce the proper power for the control unit, a combination of a low pass filter, and a voltage regulator should be used. Voltage regulators create a distinct voltage across them making it easy to have an accurate measurement of the output voltage. However they can continue some noise so having a low pass filter prior to the voltage regulators makes the circuit have excellent noise protection.

Some protection diodes should be added on the motor power side to limit the feedback from the motors as this can cause damage to the other components in the circuit. A current limiting fuse will need to be attached as well.

Requirements	**Verification**

• Uses a 12 V connection and regulate voltage and current draw from an outlet	• Make sure the current draw does not exceed 1.5 A measured by multimeter • Variation of the input voltage does not exceed 5% measured by multimeter
• The proper voltages and currents needed for each other sub system will be given.	• The voltage to the Motor subsystem is 12 V ∓ 0.6 V measured by multimeter • The voltage for the digital components is 3.3 V ∓ 0.165 V measured by multimeter

2.3.2 Microcontroller

A microprocessor and SD card will handle the information from the bluetooth module and bring the user input data to the motor controller system. The data from the bluetooth will be processed too quickly for the motor controller system to handle so a microcontroller is needed. A program to run the bluetooth module and interpret the results to control the motors will be stored on the SD card.

The microprocessor will take in the serial data from the bluetooth monitor and provide the proper signals to the motor driver. Inorder for the stepper motor to work properly signals in a specific order have to be given to the motor driver.

Requirements	**Verification**
• Able to communicate with the user's device via bluetooth	• Send numbers over bluetooth and determine error rate.
• Performs these operations within 800ms ∓ 160 ms	• Time each operation using given libraries

2.3.3 Bluetooth

Bluetooth Connection [HiLetgo 2pcs HC-05 Wireless Bluetooth RF Transceiver Master Slave Integrated Bluetooth Module 6 Pin Wireless Serial Port Communication BT] takes the input data from the user and puts the user inputs into registers so that the proper chips can be dispensed. This subsystem will also be used to send information back to the app to tell the user that the machine is done dispensing and to also communicate any errors that occurred.

This will be connected to the user application through a dedicated communication channel. The data will be sent with a baud rate of around 9600. The data being collected by the Bluetooth will be serially loaded into the microcontroller which will then be used to control the different parts of the machine.

Requirements	**Verification**
● Able to communicate with the user's device and with the microprocessor with minimal errors	● Perform communication tests between the user app and the microprocessor ● Test that data rates are fast enough to give instructions.

2.3.4 Motor Control System

A series of pvc pipes will be used to house the different color of chips, a platform will be placed at the bottom of the pipe to move the chips up and down. A gear system controlled by a stepper motor [Twotrees Nema17 Stepper Motor, 4 Lead Stepper Motor Nema] will move the chips in steps of one chip. This subsystem will take the information given by the user and convert it into a signal the linear actuator can understand.

This system will have two settings: collection and distribution. When the user selects a collection the motors will bring the platform all the way to the bottom and the user will place chips into the respective tubes. Once the user is done dropping the chips into each tube they will close the lid. The lid will have a small conductive plate that completes the circuit for a control signal. Once the lid is closed the motors will move the chips up until the lid is lifted and the circuit is no longer complete this will cause the motor to stop in place. One can count the number of steps the motor made and determine the number of chips. Once this occurs the user can then select distribution mode. In distribution mode the chips will be raised by a single chip that is repeated until all the chips the user wants are provided.

The motor will need to be driven by a driver that consists of a H-Bridge and the needed control signals. Stepper Motors work by having a magnetized rotor that can be stepped by different magnetic coils. By providing the coils different signals one can make fairly precise movements of the rotor which allows for fairly high angular precision around 1.8°. The motor control unit will also need to include a safety system to quickly turn the machine off in the case there is an emergency or an unsafe condition (such as a child's finger getting caught).

Requirements	**Verification**
● Motors are able to make small steps in order to rise the chips accurately	● Measure how much the chips are able to move with ruler to the tune of 1 mm
● The control signals from the microcontroller need to be processed quickly	● Determine the amount of time that progresses between the user giving an instruction and the machine is by using timing libraries
● Able to detect when the chips have reached the top during the collection phase	

2.3.5 User Mobile Application

The user interface will come in the form of an IOS application. We will be using Bluetooth Low Energy as the way for the application to communicate with the Poker Chip Companion machine. There are a few ways users interact with the machine…

At the start of the game, in the companion's dispensing state, the user can input the value of each chip, the number of players, the small & big blind, and the buy-in of each player. The application will input these values into an algorithm to calculate how all the poker chips are divided and the amount per player if buy-ins differ. This information will be sent to the companion.

At the end of the game, in the collection state, the user will be instructed to input the chips of each player one by one. As the user completes the input of one player, they will press a "Next Player" button (like a timer lap button) to signify to the machine the collection process of the next player. Once complete, the user presses an "End Collection" button and will be presented with the full scoreboard of the players' buyouts.

We will use a GATT structure to communicate information between the Bluetooth Low Energy central device module and our peripheral device (application). The BLE GATT structure is set up so you can organize the information you want to be sent between devices in a step-down model [2]. The broadest type of structure is a profile, simply a "collection" or services (a subset of the profile). There are lists of characteristics within services that contain data being sent and received. That is how the structure of communication works. The HC-05 Wireless Bluetooth RF Transceiver is a great device using Bluetooth LE. It uses UART to communicate with the microcontroller and is easy to set up.

Regarding the mobile application, we agreed that it would be developed on IOS. This is because all three of us use Apple devices. Hence, we will be using Swift for development; it allows easy access to BLE functionality on IOS devices. Additionally, it allows for more control over performance and BLE nuances, such as data speeds and advertising data. Apple provides a Core Bluetooth framework [3] that provides the classes needed for the Swift mobile application to communicate with the BLE module. For the development of the application, we will be coding Swift on the XCode IDE provided to all Apple ID users. Finally, for easy UI design, Figma will be used.

Requirements	**Verification**
● Bluetooth Module Setup ● Figma for UI designs ● XCode and Swift for IOS Development	● Ensure application is able to be hosted and tested on an IOS device (mobile/iPad) ● Ensuring BLE connection with Core Bluetooth Framework for the

collection and transmission of data between app and machine

User Mobile Application Low Fidelity Wireframes

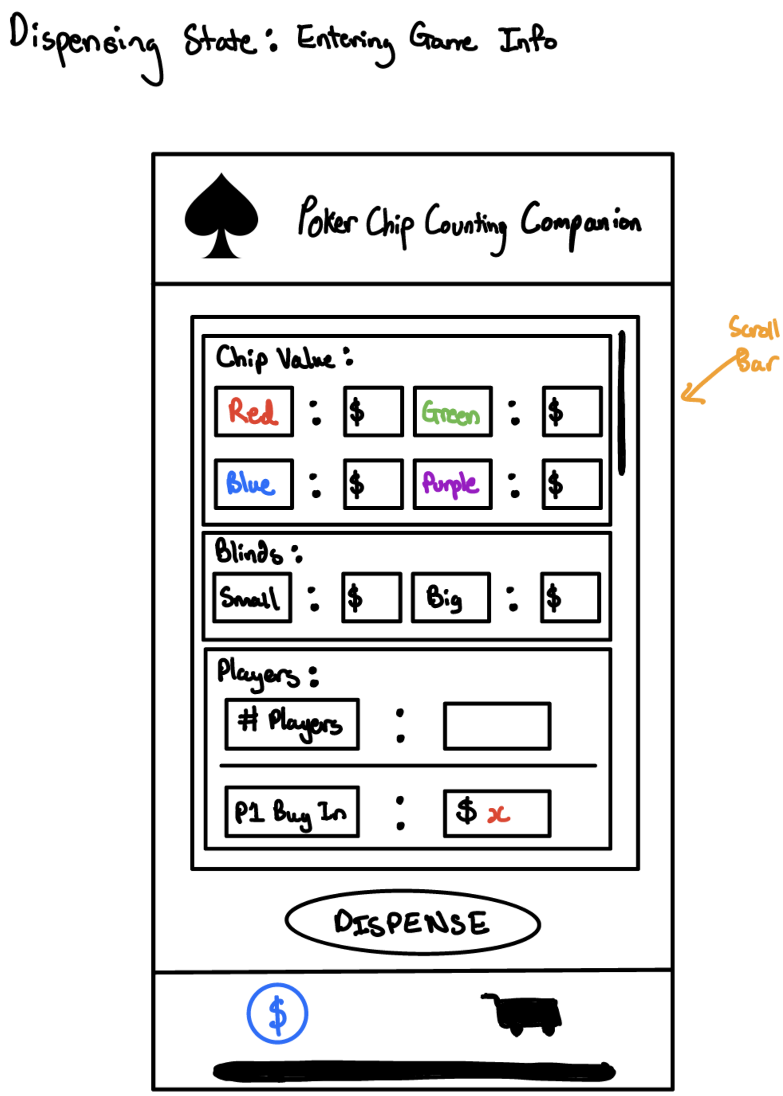

Figure 6: Dispensing State - Entering Game Information

This is the initial screen of the *Dispense* state of our machine. Users will be asked to input game information such as the value of the different chips, blinds (small & big), and player information (# of players, each player's buy-in). Once all this information has been entered, the user can press the "DISPENSE" button to begin the chip dispensing process.

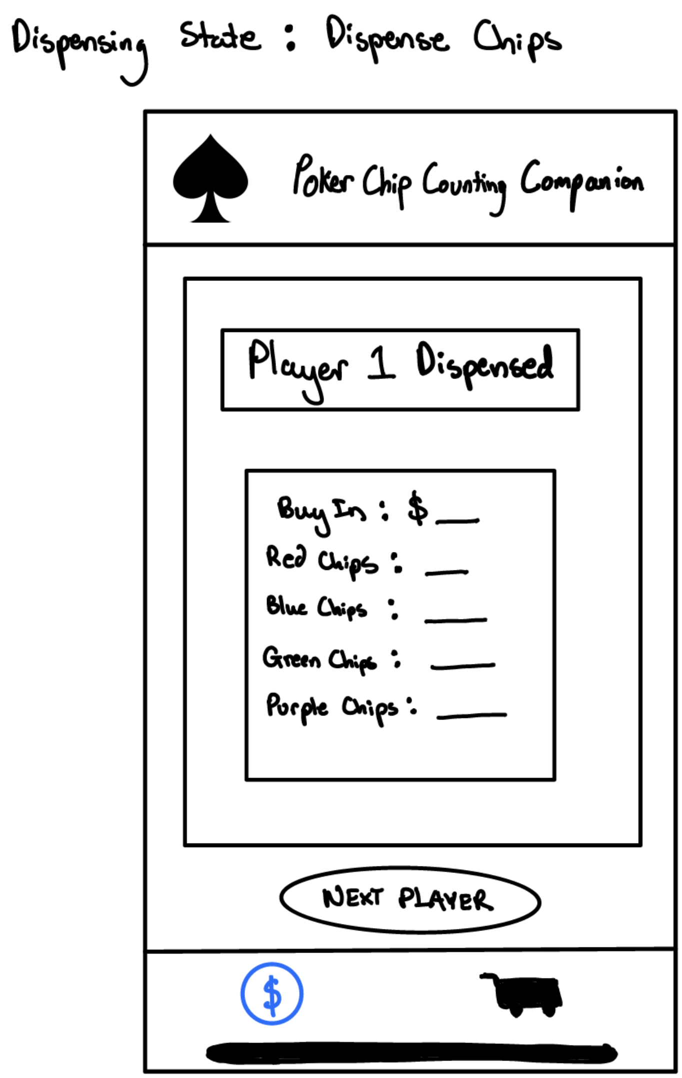

Figure 7: Dispensing State - Dispense Chips

Each player's chips will be dispensed one at a time. The application will present which player’s (#) chips are being distributed, how many colored chips are given, and their buy-in amount. Once the chips have been collected from the machine, they can press the “NEXT PLAYER” button to begin the dispensing process for the following user.

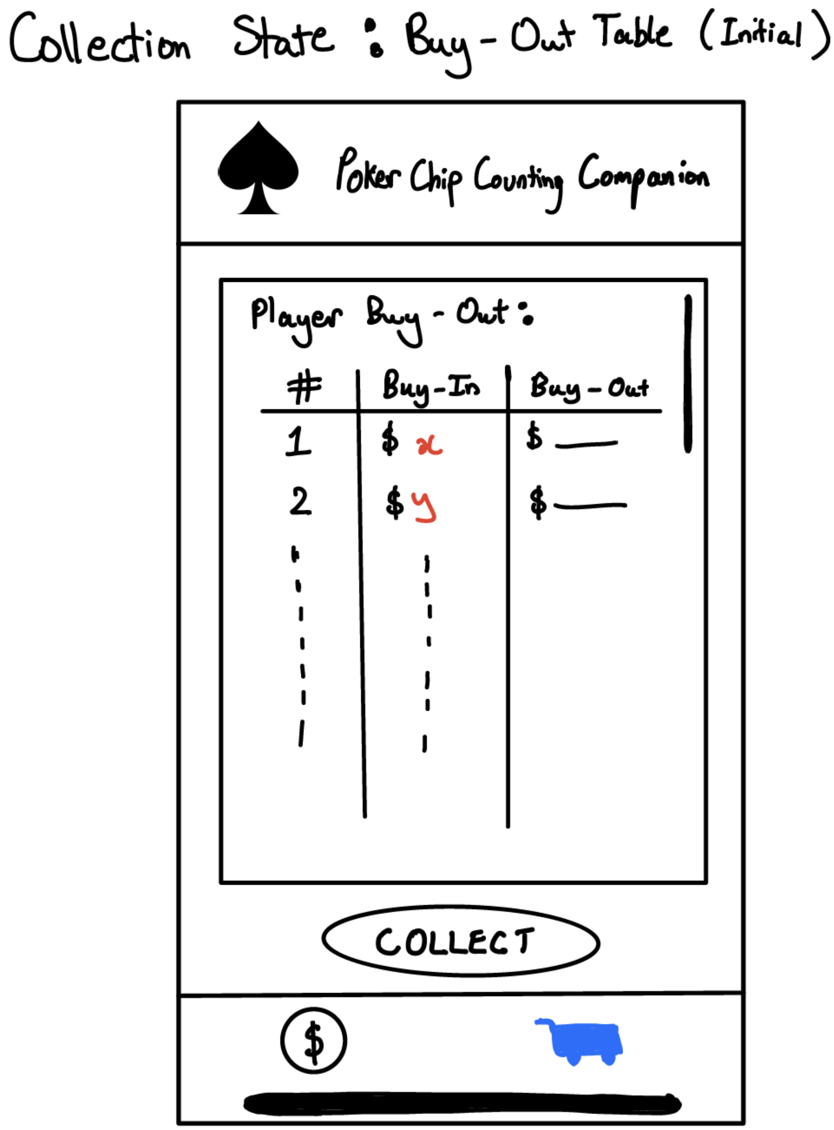

Figure 8: Collection State - Buy-Out Table (Initial)

This is the initial screen for the *Collection* state. For example, a "Player Buy-Out" table has three columns of information: Player #, buy-in amount, and buy-out amount. Once the game's chips have been dispensed, the "Buy-Out" column is set to NULL, while the "Buy-In" column is set to the player information inputted on the initial dispense screen. Once a game ends, the user can press the "COLLECT" button to begin the chip collection process.

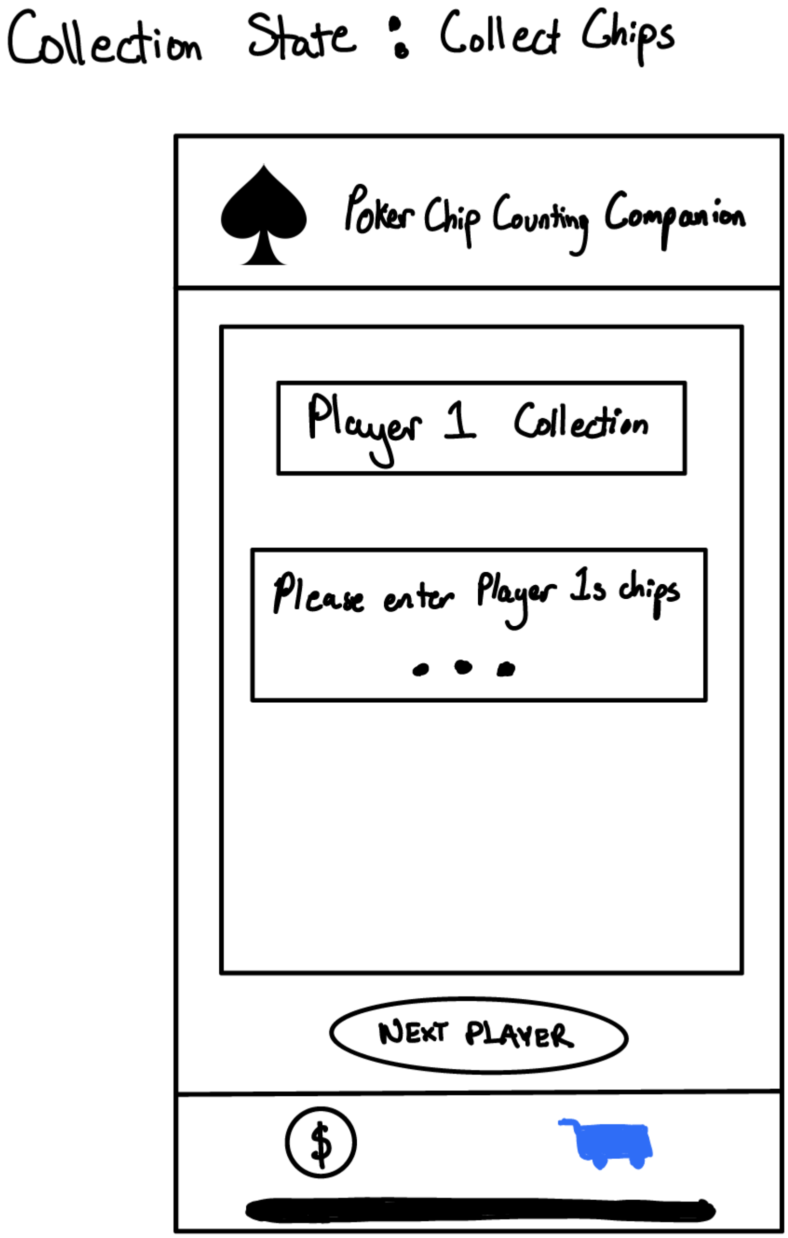

Figure 9: Collection State - Collect Chips

Each player's chips will be collected one at a time. The application will present a screen that alerts users to input chips for a particular player number. Once the chips are put into the machine, the user will click the "NEXT PLAYER" button to indicate to begin the collection process for the next player till all the chips have been collected.

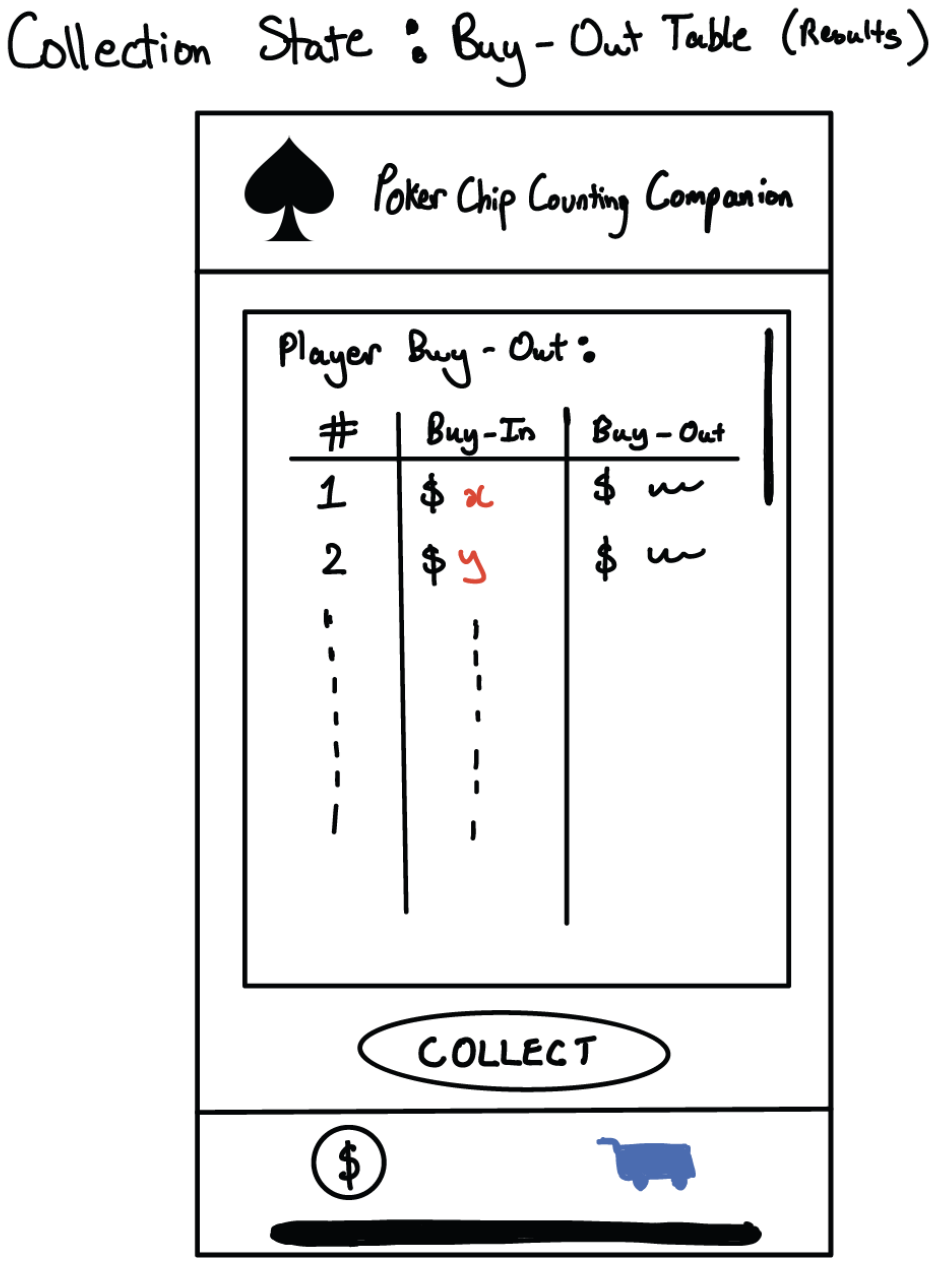

Figure 10: Collection State - Buy-Out Table (Results)

This screen presents the *Collection* state screen once the companion has collected chips. Compared to the initial screen, now the "Buy-Out" table values correctly present the money a user is rewarded based on the colors of the chips entered into the machine.

2.4 Tolerance Analysis

Communications:

Transmitting all user data to the processor via the bluetooth module can take around 200ms. It could take an additional 200ms for the processor to handle the data and then another 400ms for the motor control to communicate with the motors, leading to a 800ms delay with an acceptable tolerance range of 20%.

Control:

The microprocessor is able to perform at 9600 operations per second. This value should not change much, but there can be some more time delay between memory and the microprocessor; this delay will be on the order of microseconds so synchronous read and write operations should be possible.

Motor:

The stepper motor is able to move 1.8° ∓ 5%, which will correspond to a fraction of the height of a poker chip. This will ensure that the motor has enough accuracy to raise the chips within a 1-chip height accuracy.

Furthermore, the motor must be able to lift and lower a stack of 50 poker chips. The weight of 50 chips is about 600grams. According to Newton's second law, the amount of force that each stack will exhibit is 5.88 Newtons. We can then calculate the torque requirement for the motor:

$$Torque = F * r/(n)$$

$$r = Radius\ of\ drive\ pulley + Length\ of\ platform\ arm\ (m) = 4.6\text{cm}$$

$$n = Efficiency\ of\ belt\ system = \sim 90\% = 0.9$$

So, Torque required = 30 N-cm

The stepper motor can provide a maximum torque of 42 N-cm so these motors will work well for our 30 N-cm requirements.

3. Cost and Schedule

3.1 Cost

Labor:

The average starting salary for CE/EE graduates is $80,000/year or $40/hour [2].

We estimate that each team member will dedicate 100 hours to this project.

Using these values, we can calculate an estimate for the total cost of labor.

$$100\ hours \times \frac{\$40}{hour} = \$4{,}000/\text{person}$$

$$\frac{\$4{,}000}{Person} \times 4\ Group\ Members \times 2.5\ Overhead = \$40{,}000\ for\ labor$$

Parts:

Description	Manufacturer	Part Number	Quantity	Unit Price ($)	Cost ($)
1-½" x 2' PVC Tube	Charlotte Pipe	PVC 071000200	2	6.71	13.42
Stepper Motor	TwoTrees	17HS4401S	4	10.99	43.96
1/8 in. x 48 in. Round Rod	Everbilt	801567	2	4.38	8.76
Timing Belt Pulley	Zeelo	ZR-19090203	1	14.99	14.99
1/4 in. x 2 ft. x 4 ft. MDF	Home Depot	1508104	3	10.49	31.47
H Bridge	Bridgold	L293 L293D	4	0.81	3.24
12 V Power Supply	ALITOVE	5050 3528	1	12.99	12.99
BLE Microprocessor	HiLetgo	ESP-WROOM-32	1	10.99	10.99
				PARTS TOTAL:	139.83

Figure 12: Parts List

Grand Total:

$$Total\ cost = Cost\ of\ Labor + Cost\ of\ Parts = \$40{,}000 + \$139.83 = \$40{,}139.83$$

3.2 Schedule

9/25 - 10/1	*9/27: Design Doc Check at 2pm Forrest: • Create 3D models of the device & gather measurements for chip/device sizes Adish: • Finish prototyping application designs, create github repository, learn swift, aid with bluetooth module David: • Work on a basic prototype of the motor and bluetooth modules, work on PCB design
10/2 - 10/8	Forrest: • Print first prototypes with 3D printer • Plan and prepare for first version of PCB Adish: • Front end development of application with Swift • Begin analysis into Core Bluetooths Apple Framework David: • Familiarize with laser cutter, Work on PCB Design • Help make the housing
10/9 - 10/15	*First Round PCBway Orders (10/11) *Teamwork Evaluation I (10/12) Forrest: • Order all necessary parts to begin prototyping Adish: • Finalize Swift application front-end, • Begin BLE connection • Aid with PCB Design David:

	● Finalize PCB Design, Construct and Test prototype
10/16 - 10/22	Forrest: ● Construct a working prototype and test ● Work with first version of PCB and find improvements Adish: ● Finalize BLE connection ● Evaluate state machine diagram needed for companion functionality ● Microcontroller work David: ● Test prototype, make improvements
10/23 - 10/29	Forrest: ● Refine prototype, find ways of making it more efficient and effective Adish: ● Test prototype David: ● Test prototype, make improvements, Finalize PCB design
10/30 - 11/5	*Second Round PCBway Orders (11/1) Forrest: ● Integrate second version of PCB and test/refine Adish: ● Continue to update and test prototype David: ● Continue to update prototype
11/6 - 11/12	Forrest: ● Test and fix any necessary bugs in hardware system Adish:

	● Bug fixing, demo preparation David: ● Test prototype, make improvements
11/13 - 11/19	*Mock demo week Forrest: ● Practice and prepare for mock demo Adish: ● Practice and prepare for mock demo David: ● Test prototype, make improvements
11/20 - 11/26	Forrest: ● Prepare for final demo Adish: ● Finalize design David: ● Finalize design
11/27 - 12/3	*Final demo week Forrest: ● Draft final report paper Adish: ● Draft final report paper David: ● Draft final report paper
12/4 - 12/10	*Final Presentation Week Forrest: ● Rehearse final presentation & perform it Adish: ● Rehearse final presentation & perform it

	David: ● Rehearse final presentation & perform it

4. Ethics

Intentional Misuse - Network Hacking:

Issue - Individuals exploiting the phone application's security to get additional chips. This poses a risk because someone could have access to more poker chips than what they had actually bought in for.

Code of Ethics Breach - Without accounting for this issue, we would be subject to breaking 1.3 of the ACM Code of Ethics that instructs computing professionals to "Be honest and trustworthy"[5]. Without prioritizing security, we run the risk of losing the values of honesty and trustworthiness in the game of poker.

Solution - We will add a passcode to the app so that a 'game master' can initiate the game and have oversight of the buy-ins for each player.

Harmful Implications - Gambling Addiction

Issue - The Poker Chip Counting Companion strives to make the Poker gameplay experience as seamless as possible. With this, we run the risk of creating gambling addictions for those who use the companion to a high extent.

Code of Ethics Breach - With this issue, we are subject to breaking I.1 of the IEEE Code of Ethics that says "to hold paramount the safety, health, and welfare of the public…"[6]. We would also be subject to breaking 1.2 of the ACM Code of Ethics that instructs computing professionals to "Avoid harm"[5] and not inflict physical or mental injury. Gambling addiction is a serious mental safety concern.

Solution - We would put a disclaimer in the phone App that lists the National Problem Gambling Helpline Network hotline (1-800-522-4700). The message would also urge individuals against excessive gambling, and warn them of the mental health issues that come with it.

5. Safety

Accidental Misuse - Child safety:

Issue - A child could accidentally place their hand inside the device and get their hand pinched by the motor and motor stage.

Code of Ethics Breach - With this issue, we are subject to breaking I.1 of the IEEE Code of Ethics that says "to hold paramount the safety, health, and welfare of the public…"[6]. We would also be subject to breaking 1.2 of the ACM Code of Ethics that instructs computing professionals to "Avoid harm" and not inflict physical or mental injury.

Solution - The device & housing will be fully enclosed to avoid pinch spots so that in the event of a child being near it, the device will not harm them.

6. Citations

[1] "History of Poker" in Roya, Will (2021). *Card Night: Classic Games, Classic Decks, and the History Behind Them*. Black Dog & Leventhal Publishers. p. 203. ISBN 9780762473519.

[2] Grainger Engineering2. (2021). *Salary averages*. Electrical & Computer Engineering | UIUC. https://ece.illinois.edu/admissions/why-ece/salary-averages

[3] "Core Bluetooth Framework." *Apple Developer Documentation*, Apple, https://developer.apple.com/documentation/corebluetooth.

[4] Teel, John. "How to Develop a Mobile App That Communicates with Your Product Using Bluetooth." *PREDICTABLE DESIGNS*, Predictable Designs LLC., 12 Jan. 2022, https://predictabledesigns.com/how-to-develop-a-mobile-app-that-communicates-with-your-product-using-bluetooth/.

[5] ACM Code 2018 Task Force. (2018). *ACM Code of Ethics and Professional Conduct*. Code of Ethics. https://www.acm.org/code-of-ethics#:~:text=1.3%20Be%20honest%20and%20trustworthy,problems%20to%20the%20appropriate%20parties.

[6] IEEE Code of Ethics, IEEE, 2020. https://www.ieee.org/about/corporate/governance/p7-8.html