Pocket Pal

Team 70
Jack Dai (jackdai2)
Arkajit Dutta (arkajit2)
Alexandria Valdez (avaldez4)
TA: AJ Schroeder

ECE 445 Design Document

March 4, 2021

1 Introduction

1.1 Objective

Cash payments can be a hassle, especially in crowded businesses like coffee shops or fast-food restaurants. People want to get in and out of these places as quickly as possible, so the additional time that people spend calculating and accumulating the correct combination of coins can be inconvenient and tedious.

With Pocket Pal, this is a problem of the past. Instead of frantically fishing around for the perfect amount of change, users can input the coin amount of their purchase into the device and Pocket Pal will dispense. This compact, high-tech wallet keeps track of how many coins it has at any given time, and it will automatically update these values whenever the user inserts more coins into the device. Pocket Pal is designed to calculate and dispense the correct combination of coins needed for a purchase, overall making cash purchases quicker.

1.2 Background

Even though credit and debit usage is a solution for some people, cash purchases are still prevalent in both the United States and overseas. The 2019 Diary of Consumer Payment Choice found that cash purchases in the United States are common for small payments; cash is used for about half of all purchases under $10 and 42% of all payments under $25 [1]. Additionally, the German Association of Money and Bond Services found that 75% of Germany's purchases in 2020 were made through cash, even with the rise in card payments attributed to the pandemic [2].

Pocket Pal is a solution for those who prefer to use cash for small payments. Scrambling for the right coins at a cash register is often time-consuming, but users can hasten this process with Pocket Pal. Our innovative wallet features the capability to determine the exact combination of coins to match the amount input by the user. It will then dispense the coins, sparing users from a frantic look through their wallets. Reloading the device involves inserting coins into the coin loader, where they will be automatically sorted by coin type. The coin compartment will keep track of all the coins in the wallet using IR sensors, as they will detect whenever a coin is inserted into the device. The microcontroller will compute how many coins to distribute based on the user's input amount, and keep track of the counts of each coin type whenever loading or dispensing processes occur. Our primary focus is the tracking and dispensing of coins, as this is the most time-consuming aspect of cash transactions when compared to bills. Even though there are similar products on the market that can sort coins [3], most cannot do this automatically. Instead, coins must be placed into their respective compartments by the user. Only a few wallets can automatically sort coins [4], but the user must still calculate and grab the correct combination of coins to make a cash purchase. Unlike these other products, Pocket Pal's unique design handles both the sorting and dispensing of coins so that the user does not have to.

1.3 High-Level Requirements

- The coin loader must sort a mix of pennies, nickels, dimes, and quarters into their proper compartments with 95% accuracy.
- The coin dispenser must dispense the correct coin amount in a maximum of 15 seconds.
- The overall design of Pocket Pal must be compact, with a maximum size of 6" x 4" x 1.5".
- The battery must be able to remain powered for eight hours, which is the length of a standard work day.

2 Design

2.1 Block Diagram

Figure 2.1 shows the Pocket Pal's block diagram. The design will consist of four distinct subsystems: a power supply, a control unit, a coin loader module, and a coin dispenser module. The power supply for Pocket Pal is a portable 12-volt battery. It will power the microcontroller, user input, IR sensors, IR emitters, coin dispenser, and solenoids. The control unit handles the users' input whenever they want to make a purchase and sends this information to the microcontroller. The microcontroller uses this data to calculate the correct combination of coins for disbursement, decrements all relevant coin counts, and signals to the coin dispenser to power the solenoids corresponding to the necessary coin types. When the user inserts coins into the coin loader, the IR sensors detect this change, and the microcontroller increments the coin counts accordingly.

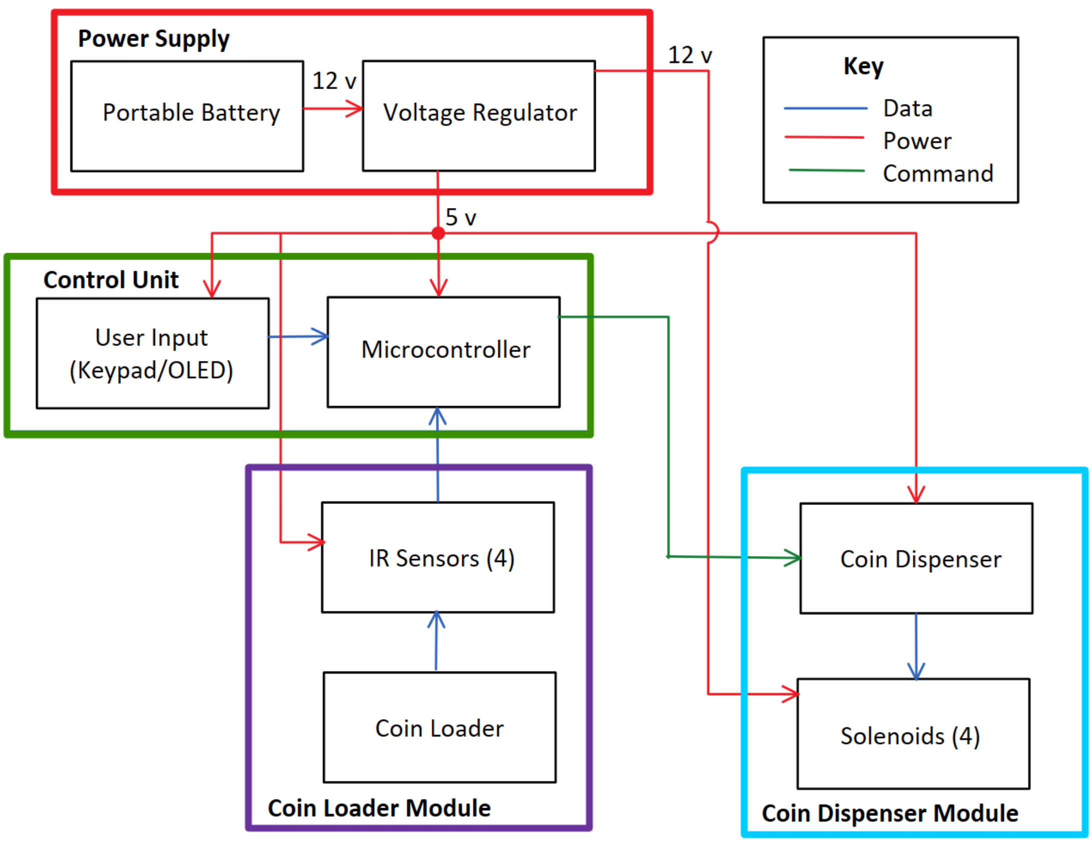

Figure 2.1. Block Diagram of Pocket Pal

2.2 Physical Design

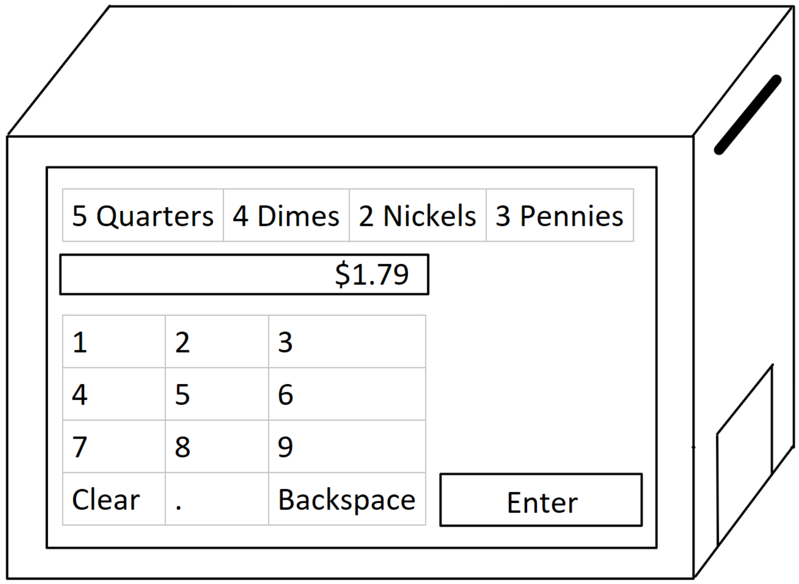

Figure 2.2. Pocket Pal Visual Aid

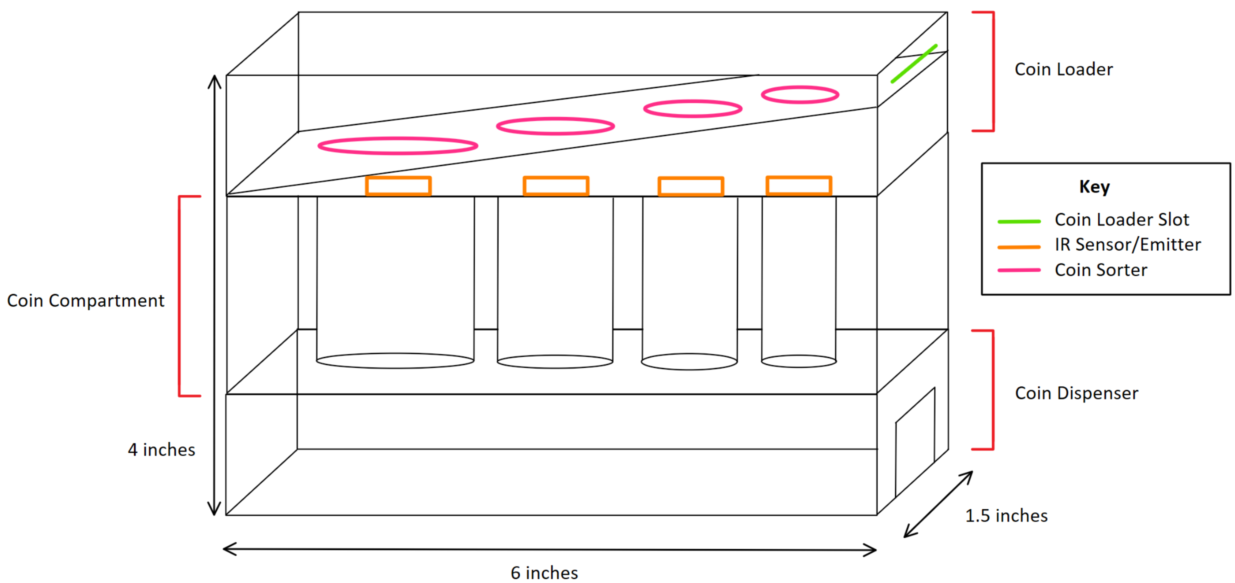

Figure 2.3. Pocket Pal Physical Design of Coin Loader

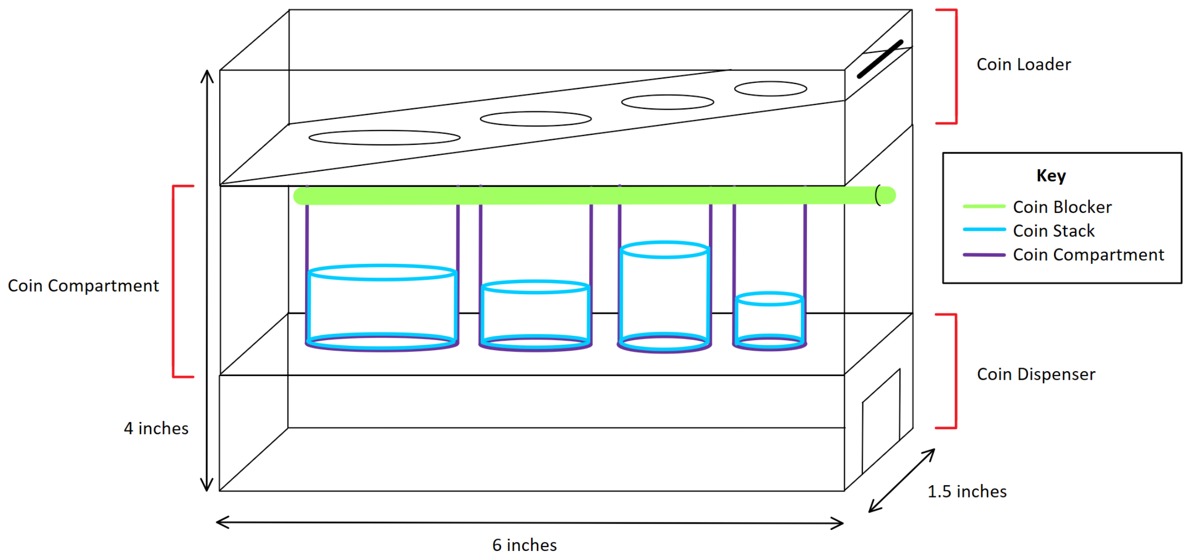

Figure 2.4. Pocket Pal Physical Design of Coin Compartment

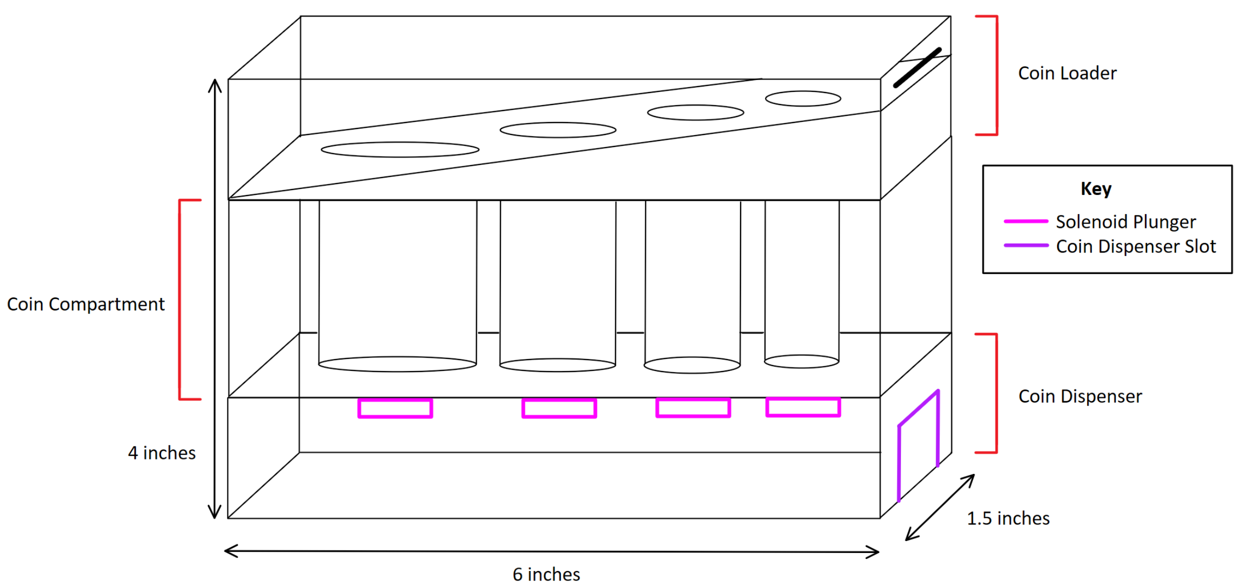

Figure 2.5. Pocket Pal Physical Design of Coin Dispenser

As shown in Figure 2.2, Pocket Pal will display the counts of each coin type that is currently in the device. The user can input a monetary amount that will be shown on screen. When the user selects “Enter,” the microcontroller will calculate the combination of coins needed to make a purchase and begin the dispensing process.

Figure 2.3 depicts the physical design of the coin loader. Coins are inserted individually through the coin loader slot. As each coin slides down the ramp, it will fall into its respective coin compartment through the coin sorter. Notice that each coin sorter is sized according to the coin type (quarter, nickel, penny, dime), so that larger coins easily pass over a smaller coin's sorter. When a coin falls into its compartment, the IR sensor will trigger and the microcontroller will increment the coin counts accordingly. The coin compartment (Figure 2.4) is where Pocket Pal stores all of its coins. We also have a coin blocker, which ensures that no coins will travel back up through the coin sorter during everyday use. Figure 2.5 shows our physical design of the coin dispenser. The solenoids keep the coins from unintentional distribution. To dispense a coin, we will toggle the push and pull states of its corresponding solenoid so that the coin can fall into the collecting tray. We will repeat this process for every necessary coin. Once complete, the user can empty the device through its coin dispenser slot.

2.3 Power Supply

The power supply (portable battery) should be able to power the microcontroller, IR sensors, IR emitters, solenoids, coin dispenser, and user input.

2.3.1 Portable Battery

The portable battery outputs 12 volts, which will supply power to the four solenoids. We will use a voltage regulator to power the microcontroller, four IR sensors, four IR emitters, and the OLED screen at 5V. Table 2.1 shows our requirements and verification table.

Table 2.1. Portable Battery Requirements Table

Requirement	Verification
Charging the battery at maximum current and voltage can be sustained below 85°C to ensure the user's safety.	1. Connect the battery to all relevant Pocket Pal components, as specified in our block diagram (Figure 2.1). 2. Power the battery at full capacity (12V) without limiting current. 3. With a voltmeter, measure the voltage across the battery and ensure it is running at 12 volts. 4. Remove the voltmeter. 5. Without modifying the current setup, leave the battery running for 8 hours without human intervention. 6. Using an IR thermometer, check that the battery temperature after eight hours is less than 85°C.

Battery must be able to last for eight hours, which is the typical length of a work day.	1. Connect the battery to all relevant Pocket Pal components, as specified in our block diagram (Figure 2.1). 2. Power the battery at full capacity (12V) without limiting current. 3. With a voltmeter, measure the voltage across the battery and ensure it is running at 12 volts. 4. Leave the battery running for 8 hours without any human intervention. 5. After eight hours, use a voltmeter to measure the voltage across the battery and make sure it is still running at 12 volts.
Battery must be able to supply 8A of current to the required solenoid at all times.	1. Connect a solenoid parallel to the diode and n-channel MOSFET powered by a 12V battery. 2. Measure current around the terminals of solenoid using an ammeter/oscilloscope for 10 minutes. 3. Current should be >8 A at all instances.

2.4 Control Unit

The control unit handles all of the computation that occurs when a user inputs a cash amount, tells the coin dispenser how many coins in each compartment must be dispensed, and updates coin counts accordingly. It also increases coin counts when the IR sensors and emitters detect a change in the coin compartments.

2.4.1 User Input

When the user wants to make a purchase, the user will enter the necessary monetary amount into this subsystem through our keypad. The user input will output this amount to the microcontroller. The user input consists of the numbers 0-9, as well as “Enter,” “Clear,” and “Backspace” buttons. The OLED screen will also display the current counts of each coin type. Table 2.2 depicts the requirements and verification table for the OLED screen.

Table 2.2. OLED Screen Requirements Table

Requirement	Verification

The OLED screen must communicate with the microcontroller and begin the dispensing process within 3 seconds after the user inputs the desired cash amount.	1. Using the keypad, enter a cash value into the user input and start a timer. 2. Have your partner write down the time in which they hear the first coin drop into the collecting tray. 3. Stop the timer when the last coin has been dispensed. 4. Subtract the time recorded in step two from the time recorded in step three, as this is the time it takes for the OLED screen to communicate with microcontroller and begin the dispensing process.

2.4.2 Microcontroller

The microcontroller's function is to keep track of how many of each coin are in the system and update these values when necessary. It receives data from the user input and IR sensor subsystems, and outputs commands to the coin dispenser. The microcontroller increments values when the IR sensors detect a specific coin type is being loaded into the device. It decrements values automatically when it signals to the coin dispenser that it should begin its dispensing process. When the microcontroller receives data from the user input, it calculates the combination of coins that needs to be dispensed to reach the user's input price. Once the microcontroller calculates the number of coins, it tells the coin dispenser which of the four solenoids should switch between its push and pull states and how many times this should occur to dispense the correct number of coins. We will be using the ATMEGA328P Microcontroller [6] for this subsystem. See Table 2.3 for the requirements and verification table for the microcontroller.

Table 2.3. Microcontroller Requirements Table

Requirement	Verification
Microcontroller (16Hz internal) should consume less than 22mW with maximum current supplied.	1. Remove loads and connect the 12V power supply to the microcontroller. 2. Using an ammeter, measure the current across the microcontroller terminals and observe whether current is < 0.2mA for 8V supply.
Microcontroller should supply current to the required solenoid for dispensing when the user inputs an amount.	1. Enter some monetary value ($0.01, $0.05, $0.10, $0.25 and other combinations) into the keypad.

	2. Wait until all coins are dispensed, and verify that the correct number of coins were released by writing calculations on paper. 3. If the exact monetary value was released from Pocket Pal, we know that the microcontroller correctly supplied current to the required solenoids.

2.5 Coin Module

The coin module communicates with the microcontroller whenever coins are inserted into the coin loader to know when to update the coin counts. The coin module is also signaled by the microcontroller when the coin dispenser needs to control certain solenoids.

2.5.1 Coin Dispenser

The purpose of the coin dispenser is to control the four solenoids which will dispense the correct number of coins. The coin dispenser's input is a signal from the microcontroller designating how many times each solenoid should switch between its push and pull states. With this information, the coin dispenser's output will control each solenoid so that it can dispense individual coins. In Figure 2.6, we have a side-view of the coin dispenser and solenoid plunger. When the solenoid is in its push state, no coins will be able to fall into the coin collecting tray. When the solenoid is in its pull state, it will allow one coin from that compartment to dispense. See Table 2.4 for our coin dispenser requirements and verification table.

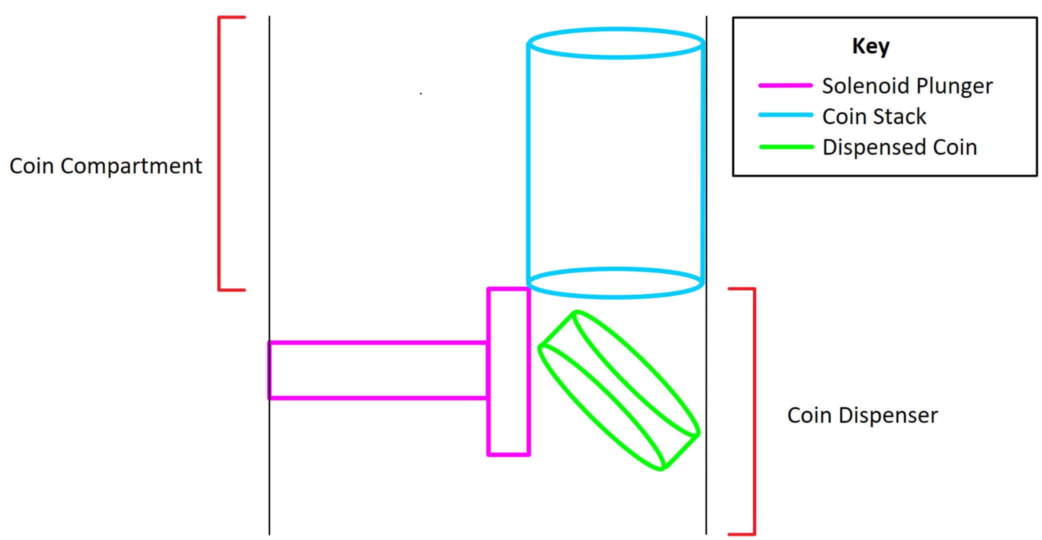

Figure 2.6. Side-view of Coin Dispenser

Table 2.4. Coin Dispenser Requirements Table

Requirement	Verification
Dispenser releases exactly one coin in one push and pull motion.	1. For one coin compartment, check that it has two or more coins inside of it. 2. Enter the value of one coin (from the same coin compartment in step one) into the keypad (ie: $0.05 for a single Nickel) 3. After Pocket Pal dispenses, ensure that only one coin of that type was released from the device when there were multiple coins in the compartment. 4. If there is more than one coin in this compartment, enter the value of the excess coins so that only one coin will be left in the compartment. 5. After Pocket Pal dispenses, ensure that only one coin of that type was released from the device when only a single coin was in the compartment. 6. Repeat steps one through 5 for each coin compartment.

2.5.2 Solenoids

The solenoid plungers are required to dispense coins individually. They receive input from the coin dispenser, which determines which solenoid will need to switch between its push and pull states. The solenoid's states can be toggled multiple times to dispense more than one coin of the same type. Table 2.5 below shows our requirements and verification table for our solenoids.

Table 2.5. Solenoid Requirements Table

Requirement	Verification
Push-pull solenoids' plunger should "push," at minimum, the diameter of the coin for each compartment.	1. Supply minimum current (to a solenoid ($F \propto 1/x^2$) to determine maximum stroke for each solenoid. 2. Maximum stroke needs to be greater than the maximum diameter of coins (24.5mm - Quarters).

2.5.3 Coin Loader

The coin loader is a mechanical roof that will sort coins into their respective compartment when the user inserts a coin into the device. Coins will slide down a ramp with four different sized holes on its floor. Each hole corresponds to the size of a coin type (quarter, dime, nickel, penny), and only coins of matching type will be able to fall into each hole. Table 2.6 below depicts our requirements and verification table for the coin loader.

Table 2.6. Coin Loader Requirements Table

Requirement	Verification
Loader sorts a mix including all four types of coins (quarter, nickel, dime, penny) with 95% accuracy.	1. Collect four coins of each coin type (16 total) and put them in a pile. 2. Grab a single coin at random, and insert it through the coin loader slot. 3. Repeat step 2 for the other 15 coins. 4. Count how many coins were successfully sorted into the correct coin compartment, and record the ratio of correct coins to total coins (e.g. 14 coins were sorted correctly, so the ratio is 14/16). 5. Remove all 16 coins from the device and put them in a pile. 6. Repeat steps two through four two more times. 7. Calculate the average percentage of all three runs, which is the value we will use to determine our coin loader's accuracy.

2.5.4 IR Sensors

IR sensors and emitters are being used to detect when a coin has been loaded into the compartment. They are being fitted onto the top of each coin compartment, and will detect whenever a coin falls from a hole in the loader. As seen in Figure 2.7, the IR sensors communicate directly with the ATMega328P controller. Table 2.7 shows our requirements and verification table for the IR sensors.

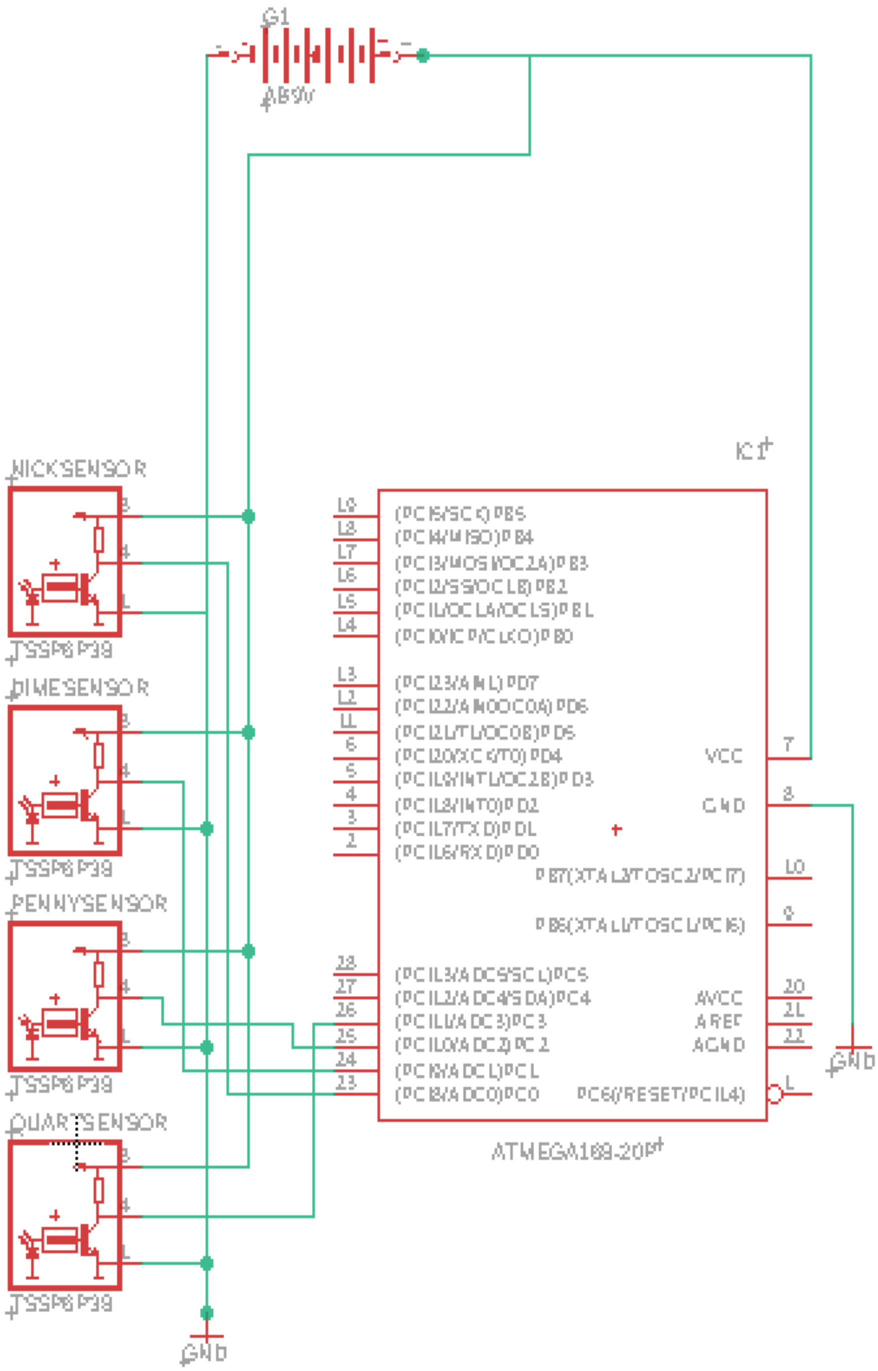

Figure 2.7 IR Sensor Schematic Diagram

Table 2.7. IR Sensor Requirements Table

Requirement	Verification
IR receivers output 5.3 volts whenever a coin passes over the IR emitter. For the IR receiver datasheet, see [7]. The IR emitter is found at [8].	1. With a voltmeter, measure the voltage of the IR emitter for each compartment, and ensure it is five volts and powered on. 2. With a voltmeter, measure the voltage of the IR receiver for each compartment, and make sure it is -0.3 volts and powered on. 3. For one coin type, insert a single coin. 4. Using a voltmeter, measure the voltage output of the IR receiver for that coin type's compartment, and make sure the receiver outputs 5.3 volts when the coin passes over the IR emitter. 5. Repeat steps three and four for each coin type.
IR receivers output -0.3 volts at all times, unless a coin is passing over the IR emitter. The IR receiver never outputs 5.3 volts unintentionally due to the push-pull mechanism of the solenoid moving coins up and down.	1. Using a voltmeter, measure the voltage of the IR emitter for each compartment, and ensure it is five volts and powered on. 2. With a voltmeter, measure the voltage of the IR receiver for each compartment, and make sure it is -0.3 volts and powered on. 3. For one coin compartment, supply maximum power to its solenoid at maximum voltage (12V). 4. Measure the voltage output of the IR receiver for the same coin compartment, and ensure the receiver always outputs -0.3 volts. 5. Repeat steps three and four for each coin type.

2.6 Tolerance Analysis

Our project primarily relies on accurate dispensing of coins through their respective coin compartments, i.e for an amount of $0.58 we would need two quarters, one nickel and three pennies. This accurate dispensing of coins is dependent on the force exerted by the solenoid on the coins. The throw (linear motion) of the solenoid has to be varied within the range **[0 mm, 35**

mm] using current regulators to ensure only a single coin is dispensed through the coin compartment. Table 2.8 shows the dimensions of coins and solenoids used in our project.

Table 2.8. Coin and Solenoid Dimension Table [9]

Solenoid Lead Length: 57 mm = 0.057 m *Solenoid Stroke/Throw : 35 mm = 0.035 m [10]*

Coins	Pennies	Nickels	Dimes	Quarters
Radius (m)	0.009525	0.010605	0.008955	0.01213
Width (m)	0.00152	0.00195	0.00135	0.00175
Surface Area (m^2)	0.00028502	0.0003533	0.0002519	0.0004622

The force exerted by a solenoid is directly proportional to the second power of current flowing through it, or $F \propto I^2$ as shown in Eq. 1. As linear motion of the solenoid is inversely proportional to the force it exerts, we need to regulate the current to hold and dispense coins of different width. The solenoids are connected to a microcontroller via a transistor and resistor, to enable us to regulate the current according to the distance from tip of solenoid plunger (g), number of turns on the solenoid coil (N), and current flowing through it (I).

$$F = (N * I)^2 * \mu o * A/(2 * g^2) \quad (1)$$

The rated current for the solenoid is 8A, therefore the $(N * I)^2$ the term for each solenoid would be the same. The distance for each solenoid will be modified depending on diameter and surface area of coins it is dispensing.

For the instance we need all 4 solenoids for coin dispensing, we would need each MOSFET to amplify current to a minimum of 8A for proper functioning of our device. As the collector of the MOSFET will be connected to the solenoids we need Ic to be a minimum of 8A. Considering a voltage output of 5V from the ATMega328P pins and a βvalue of 100, using the equation (2) and (3)

$$Ic = \beta * I_b \quad (2)$$

$$I_b = \frac{Vbe}{Rb} \qquad V_{be} = V_b - V_e \quad (3)$$

we determine the base resistor (R_b) to be less than 200Ω for collector current (I_c) to be 8A.

This design helps us achieve maximum accuracy (> 95%) and efficiency (< 15 seconds) in accordance with our project requirements. However, a significant tradeoff with this design is the increased size of the device. Our initial design of Pocket Pal was 6" x 4" x 1.5". However, considering a solenoid with a minimum stroke of 25 mm (0.984 inches) will be attached to the side of each compartment, our device will almost certainly exceed 6" in length.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor Costs

Each person in the group is expecting to work at least 10 hours a week. We are using $20/hour to estimate our hourly wage. The development time of our project is approximately 12 weeks. The total labor cost for our team is shown in Eq. 2.

$$2.5\,(3\,students\ *\ \$50/hour\ *\ 10\,hours/week\ *\ 12\,weeks)\ =\ \$45,267 \tag{2}$$

We are also expecting the machine shop to spend about three hours on constructing our design, with an estimated $50/hour wage. Their labor cost is shown in Eq. 3 below.

$$\$50/hour\ *\ 3\,hours\ =\ \$150 \tag{3}$$

3.1.2 Parts

Table 3.1. Pocket Pal's Parts Table

Parts	Description	Quantity	Unit Cost
Microcontroller (Microchip Technology/ATMEGA328P-PU)	This will be an arduino compatible microcontroller that we use to program our dispensing and tracking of the coins.	1	$5.95
IR sensors (Vishay Semiconductor Opto Division/TSOP34838)	IR sensors will be installed at the top of every coin compartment to monitor the number of coins in each tube.	4	$1.59

IR emitters (Everlight Electronics Co Ltd/IR333-A)	IR emitters will be installed opposite from the sensors to detect when a coin is inserted into its compartment.	4	$0.32
Solenoids (Abletop Push Pull Solenoid DC 12V 35mm Long)	Push and pull solenoids that will act as dispensing mechanisms for the coins.	4	$19.95
OLED screen (Orient Display/AOM12864A0-0.96WW)	Act as the output display of the Button Keypad - display the amount required by the user.	1	$7.49
Button Keypad (Sparkfun Electronics/COM14662)	This will act as a component for the user interface, allowing users to enter the required monetary amount.	1	$4.50
Portable battery (Duracell - CopperTop 12V Alkaline Batteries)	Powers the circuit with approximately 12V.	3	$4.00
Voltage Regulator (STMicroelectronics/L78S05CV)	Outputs 12V to solenoids and 5V to the other components	1	$0.69

According to Table 3.1, the total cost of our parts is around $117.94 We used four batteries to calculate this total since it is an upper bound.

3.1.3 Total Costs

The total cost of production of this project is calculated by adding the student labor cost, machine shop labor cost, and total parts cost as shown in Eq. 4 below.

$$\$45,000 \; + \; \$150 \; + \; \$117.94 \; = \; \$45,267.94 \tag{4}$$

3.2 Schedule

Table 3.2. Pocket Pal's Development Schedule

Week	Tasks

3/8	Finalize Pocket Pal design and hand it in to the machine shop. Our entire team will complete the physical design and PCB layout during this week.
3/15	Materials arrive. Test IR sensors/emitters and solenoids to ensure they satisfy the high-level requirements. This will mainly be done by Arkajit because he is on campus.
3/22	Receive the wallet prototype. Test the coin loader, coin compartment, and dispenser functionalities. Complete our PCB. Alexandria will test the coin loader, coin compartment, and dispenser functionalities.
3/29	Setup Arduino software that will track and store the correct number of coins and reduce that number when signaled. Jack and Arkajit will write and debug the software for the Arduino.
4/5	Complete the UI of the OLED display. Program the entire software into the microcontroller. Our entire team will finish the UI software and design here.
4/12	Test for complete functionality (eg. battery duration). Arkajit and Alexandria will perform most of the final functionality testing as they are on campus.
4/19	Practice mock demo.
4/26	Demo final product.

As shown in Table 3.2, our project assignments are very flexible. Each member will participate and remain up-to-date every week, as we communicate and collaborate multiple times a week through Discord.

4 Discussion of Ethics and Safety

There are a few safety concerns with our product. The first concern is comfort. Wallets stay in the pocket for extended periods of time, so rough edges can hurt the user. To mitigate this risk, we will file the edges and corners of Pocket Pal. The material (cardboard) for the design is soft and poses little harm when carried.

Secondly, there is a slight risk of the power supply malfunctioning in the user's pocket due to an overheated battery, especially during hot months when the batteries are exposed to more heat. To mitigate this risk, we are looking at efficient microcontrollers and other components that work accurately and operate with a low power consumption. This will also help reduce the risk of a fire hazard caused by the battery, as lower power consumption and usage will limit the battery's potential to overheat.

Portability of the wallet is also important to us. Users can carry Pocket Pal anywhere their standard wallet can. The batteries used in the wallet will be dry cell alkaline batteries, which are safe to carry on flights [11].

As stated in the IEEE code of ethics, we have a responsibility to "improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems" [12]. Through Pocket Pal, we will be able to showcase these capabilities by increasing the flexibility of coin usage for smaller payments.

Pocket Pal will not distribute any collected information as per IEEE Code of Ethics II.9., stating that we have the responsibility "to avoid injuring others, their property, reputation, or employment" [12]. We take privacy very seriously and will never disclose the amount of coins that a user is currently carrying in their wallet.

References

[1] R. K. and S. O'Brien, "2019 Findings from the Diary of Consumer Payment Choice," Federal Reserve Bank of San Francisco, 26-Jun-2019. [Online]. Available: https://www.frbsf.org/cash/publications/fed-notes/2019/june/2019-findings-from-the-diary-of-consumer-payment-choice/. [Accessed: 19-Feb-2021].

[2] E. Nicholson, "Pandemic Or Not, Germans Still Prefer Cash," *NPR*, 11-Aug-2020. [Online].Available:https://www.npr.org/2020/08/11/899343670/pandemic-or-not-germans-still-prefer-cash#:~:text=%22With%20cash%2C%20it's%20easier%20to,stores%20and%20restaurants%20do%20not. [Accessed: 19-Feb-2021].

[3] *Coin Sorter Wallet*. [Online]. Available: http://coinsorterwallet.com/. [Accessed: 03-Mar-2021].

[4] *Kickstarter*. [Online]. Available: https://www.kickstarter.com/projects/marlin/radical-wallets-that-sort-your-coins-and-notes-ins/comments. [Accessed: 03-Mar-2021].

[5] A. Industries, "Adafruit 1.3' 240x240 Wide Angle IPS TFT Display," *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/4520?gclid=Cj0KCQiA1KiBBhCcARIsAPWqoSr-hpHWuJ6RiEIfG2Irb1eh5HtXINgDQqBhKe70U1xXZQJxBRZH6WoaAkY0EALw_wcB. [Accessed: 16-Feb-2021].

[6] "ATMEGA328P-PU," DigiKey. [Online]. Available: https://www.digikey.com/en/products/detail/microchip-technology/ATMEGA328P-PU/1914589. [Accessed: 03-Mar-2021].

[7] "TSOP75338WTT," *DigiKey*. [Online]. Available: https://www.digikey.com/en/products/detail/vishay-semiconductor-opto-division/TSOP75338WTT/4695865. [Accessed: 01-Mar-2021].

[8] "IR333-A," *DigiKey*. [Online]. Available: https://www.digikey.com/en/products/detail/everlight-electronics-co-ltd/IR333-A/2675571. [Accessed: 01-Mar-2021].

[9] "Coin Specifications," *United States Mint*. [Online]. Available: https://www.usmint.gov/learn/coin-and-medal-programs/coin-specifications. [Accessed: 01-Mar-2021].

[10] A. Industries, "Mini Push-Pull Solenoid - 5V," *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/2776. [Accessed: 01-Mar-2021].

[11] "Travel Tips Tuesday: Safely Packing Batteries for Your Trip," *Travel Tips Tuesday: Safely Packing Batteries for Your Trip | Transportation Security Administration*, 11-Jun-2013. [Online]. Available: https://www.tsa.gov/blog/2013/06/11/travel-tips-tuesday-safely-packing-batteries-your-trip. [Accessed: 04-Mar-2021].

[12] "IEEE Code of Ethics," *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 19-Feb-2021].