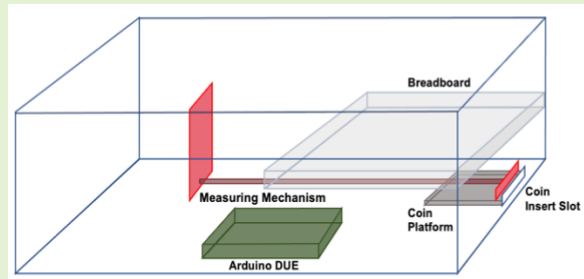
The Coin Reader

Alain Ballen 216341703

***Abstract*—This project identifies the value of coins using the diameter of a coin. This is achieved through the use of an HC-SR04 ultrasonic distance sensor and a movable reference object. A mechanism is used to measure the displacement of the reference object when an inserted coin pushes against the mechanism. The distance displaced is equal to the diameter of the coin. The displacement is then compared to different coins to determine the value of the inserted coin. If there is a match, a green LED will light up and the value of the coin will be added to a balance which is displayed on an LCD screen. If there is no match, a red LED will then light up. A button module will allow the user to reset their balance to the value of zero. For the continuation of the project, improvements and changes are needed to allow for the identification of foreign currencies. Factoring weight or using machine learning to visually identify the coins are examples of potential changes and improvements. Changing the material of the mechanism from cardboard to more sturdy materials such as plastic would be needed in order for smooth movement and more accurate size measurements. Future applications of this product include systems that require coin payments such as parking meters, vending machines, or arcade machines. This project may also be used in banking situations where counting coins is required. This project uses the Arduino DUE microcontroller based on the Atmel SAM3X8E ARM Cortex-M3 CPU.**



***Index Terms*—Coins, Coin Identification, Ultrasonic Sensor, Payments System, Coin Counter, LCD Screen, Arduino DUE, Embedded Systems, C Programming, Microcontroller, Currencies, Engineering, Actuators, Sensors, Hardware Design, Software Design, Circuits, Counterfeit Currency**

I. Introduction

PAYMENT systems are revolutionary technologies that play an important role in today's society. Without them, they are used in many different modern embedded systems such as parking meters, vending machines, and arcade machines. This project, called "The Coin Reader", provides a simple yet effective way to identify physical currency. It achieves this through the use of a mechanism that allows for the measurement of physical currencies and uses the readings to determine what the current value of the currency is. It can also potentially identify counterfeit currencies if it were the case that counterfeit currency is used as a form of payment. This project was created as a requirement for the Winter 2021, Introduction to Embedded Systems course and was developed in the C language, using the Arduino IDE. It was also a result of my interest in physical payments systems used by various devices such as a parking meter, a vending machine, arcade machines, and many other devices that require coin payments. The main function of this project is to identify the value of Canadian coins being inserted into the device. Other functions include displaying the total value on a liquid crystal display (LCD) and having the option to reset the total value. This project is inspired by similar Arduino projects that separate coins based on their size [1]. It uses a mechanism that allows coins to drop into an opening that is equal to their diameter. The coins will fall on an incline where the smaller coins will fall into their respective opening first, then allowing larger coins to pass the smaller openings and drop into their respective openings. Instead of allowing coins to fall into an opening, this project uses a technique that measures coin sizes using a similar mechanism found in a vernier caliper which can then be used to determine the value of the coin. The potential market of this project is its real-world applications as they can be used in parallel with different embedded systems that require physical payments. Banking institutions or government agencies that require the counting of currencies and the identification of counterfeits may have an interest in this project. Other applications can also include environments where intensive cash handling is needed such as in casinos where large payouts can occur. The main challenge faced when developing this project is the lack of in-person shopping due to the COVID-19 pandemic. Components were not received immediately due to the delays in the deliveries which resulted in lost time. There was also the possibility of receiving faulty parts. Another related issue is the difficulty of finding the parts needed and finding the right online vendor. The issue of having a full course load was mentally draining. Trying to balance out the development of the project and other course work was extremely mentally fatiguing and requires exceptional time management and organizational skills.

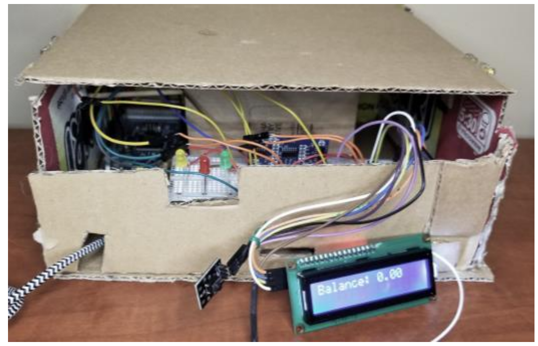


Fig. 1. An outer view of the project.

II. Proposed Project

This section will look at the hardware and software aspects of this project. The software aspect will cover the functions and design techniques/tricks used in the development of the project. The hardware aspect will cover the different components used in the project and how each component relates to each other as well as explain the mechanical mechanisms used in this project.

A. Program

The software aspect of this project is crucial to its functionality. **Figure 2** gives a brief overview of the program

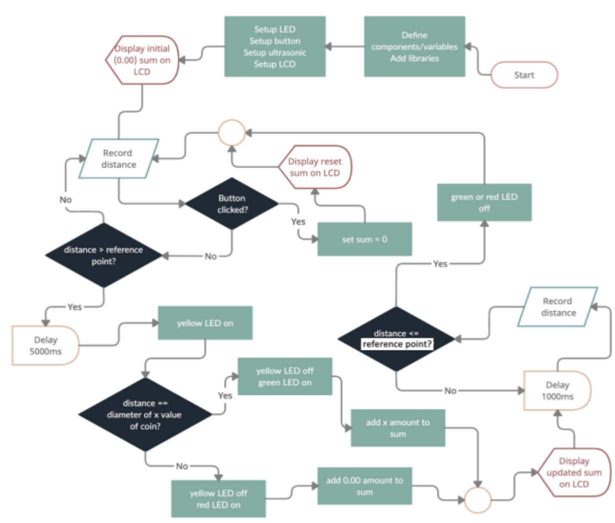


Fig. 2. A flowchart diagram that outlines the structure of the program.

in the form of a flow chart. The program will first define the pins for the *trig* and *echo* pins of the HC-SR04 ultrasonic sensor and will use digital values. Next, the program will define the pins for the three red, green, and yellow light-emitting diodes (LED) will also use digital values. The final component that is defined is the signal pin of the button module and will also use digital values similar to the previous components. In order for the program to use the LCD screen, the program uses the *Wire.h* library to allow for communication with the LCD screen and the *LiquidCrystal_I2C.h* library so that the LCD screen can be controlled. The program can then define the options of the LCD screen. The values that define the options includes the address of the LCD, each pin on the LCD, and the state of the backlight. Variables are created to store the "duration" as a long type variable, "distance" as an integer type variable, "button" (the state of the button) as an integer variable, and "sum" as a double type variable. A function is created to display the "sum" variable on the LCD. This function will set the position and writes the string to the LCD according to the position. In addition, another function is defined to clear a line on an LCD. It works by replacing all characters on the line with a blank space with the use of a loop. In order to get the distance from the ultrasonic sensor, the trigger pin of the ultrasonic sensor will alternate between high and low values which will send out eight ultrasonic pulses. The time at which the ultrasonic pulse between the sensor and the object is measured in microseconds and will be stored in the "duration" variable. The time will be outputted by the Echo pin. The distance is calculated using the following formula [2]:

$$d = \frac{t \times v_{sound}}{2}$$

Where the speed of sound in air is equal to 0.034 centimeters per microsecond. As for the setup function, the program will allow communication to the serial monitor by opening the serial port and setting the data rate to 9600 bytes per second. This is used for testing purposes. The inputs will consist of the signal pin of the button module and the echo pin of the ultrasonic sensor. The outputs will consist of the LED pins and the trigger pin of the ultrasonic sensor. The setup function will also change the state of the backlight in the LCD to a high state and will define cursor options to be sixteen characters long with two lines. For the loop function, it will first call the function which retrieves the distance from the ultrasonic sensor. The loop function will then check if there has been a change in state of the button. If a button is pressed, then it will call a function that will set the cursor to the second line and print the statement that indicates the balance is reset. The function will also set the variable holding the balance to a value of zero. The program will then check if the retrieved distance value is in an acceptable range. An acceptable range means that a coin has been inserted and is ready to be identified. The state of the yellow LED will be switched to a high state and a delay is used to give the user time to position the coin properly. Looking at the structure of the if statement, another if statement is used to determine if the distance value corresponds to a coin size. If there is a match, the state of the green LED will be high and the state of the yellow LED will be switched to a low state. The corresponding value

Fig. 3. Showing the functionality. Successfully inserted a $2 dollar coin

of the coin will then be added to the "sum" variable. If there is no matching distance value, then there will be no change in the "sum" variable and the state of the red LED will high while the rest of the LEDs will have a low state. Another delay will be used to allow time for the user to remove the coin from the insert slot. An if statement is used to check if the coin has been successfully removed and will set the state of all LEDs to low if it was successful. If it was not successful, another delay is

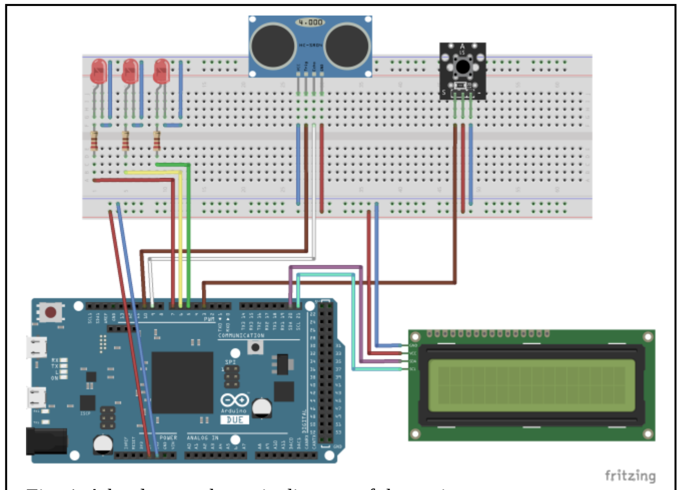


Fig. 4. A hardware schematic diagram of the project

applied and will check if the coin is removed by then. This will continue in a loop. After it sets all the states of the LEDs to a low state, then it will move out of the if statement structure and a delay is used. After the completion of the if statement structure, the whole process of the loop function is repeated.

B. Hardware

The program will not be functional without its hardware counterpart. **Figure 3** visualizes the components used in the project. This project uses an Arduino DUE microcontroller based on the Atmel SAM3X8E ARM Cortex-M3 to control the various components. A green, red, and yellow LEDs were used as the indicators. Each LED is connected to 1kΩ resistors to reduce the amount of current traveling within the LED. The Titri HC-SR04 Ultrasonic Distance Sensor is the vital component in this project. Its trigger pin is connected to digital pin 9 and the echo pin is connected to digital pin 10. The KY-004 Button Module has its signal pin connected to digital pin 47 and grants the ability for users to reset their balance. The SunFounder I2C 1602 Serial LCD Module Display contains an I2C bus that supports I2C protocols. It contains only four input/output pins which are VCC, GND, SDA, and SCL. The

SDA pin connects to communication pin 20 and the SCL pin will be connected to communication pin 21. A potentiometer is included to adjust the contrast between the backlight and the characters.

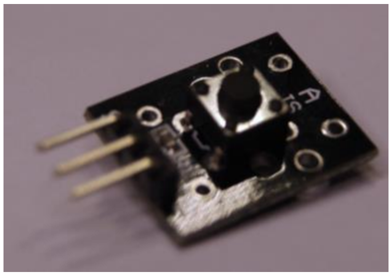


Fig. 5A. An image of the button module.

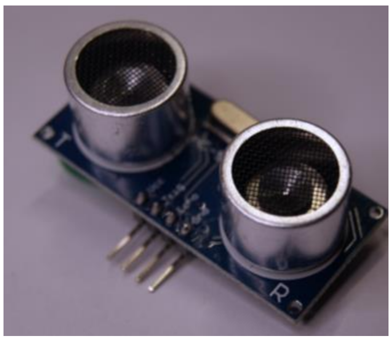


Fig. 5B. An image of the ultrasonic sensor.

The LCD screen allows for sixteen characters to be displayed for up to two lines. The default address of the LCD screen is 0x27. The LCD is used to display the user's balance and indicate if the coin inserted is valid or not. Other components required by the project are a micro-USB cable to supply voltage the Arduino DUE, jumper wires, and a BB-32656 solderless breadboard. The breadboard contains two terminal strips and 830 contact points. This project uses a mechanical mechanism which allow for the measurement of the coins. It works by creating a special platform for the coin rest. A slot is created to allow the users to insert a coin, for which the coin will then rest of the cardboard platform. The platform is being supported by small wooden skewers for robustness and stability of the platform. When the coin is being inserted, the coin will push on a wooden arm which will then displace a piece of cardboard. This piece of cardboard will be referred to as the reference object When the ultrasonic sensor sends a series of ultrasonic waves. These waves will travel through the air and meet the reference object, the wave will then travel back to the ultrasonic sensor for which the echo pin will measure the time. Then the program will calculate the distance based on the time. If the program has successfully determined a matching coin, then the green LED will turn on. The red LED indicates that the inserted coin is unable to be read as a result of the coin being a counterfeit or an error within the system. The yellow LED will indicate that the coin inserted is currently being identified, or there is a coin still inside the coin slot. Plastic sheets were placed on the coin platform to reduce the friction between the coin and the platform. This results in smooth movements of the coin and allows for more accurate measurements. Stabilizers were added to the wooden arm so that it will travel in a straight path.

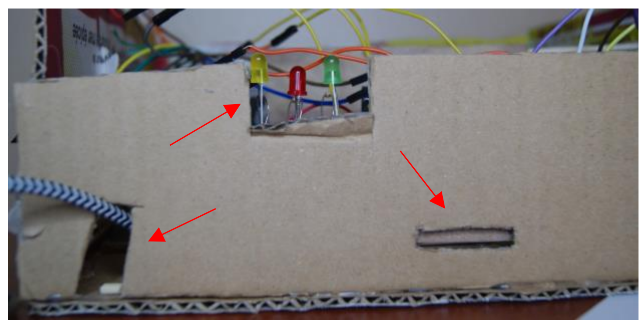


Fig. 6A. Power supply opening, the coin insert slot, and the three LEDs.

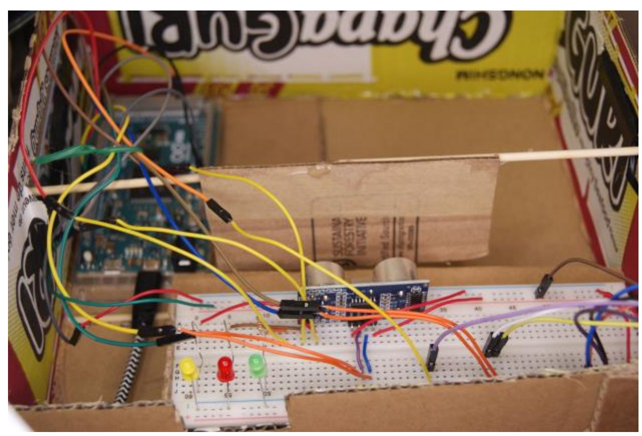


Fig. 6B. An image of the breadboard and the Arduino DUE.

The stabilizers were made in a way to allow the users to push the arm back. When the arm is pushed back, the coin will be ejected out of the coin slot and the user will then be allowed to insert another coin inside the slot. An additional slot was

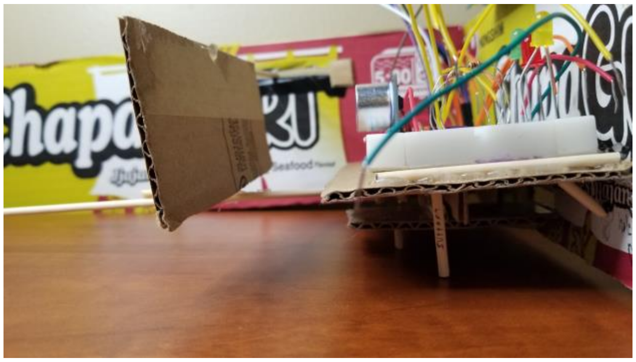


Fig. 7A. The mechanical mechanism is currently measuring a coin.

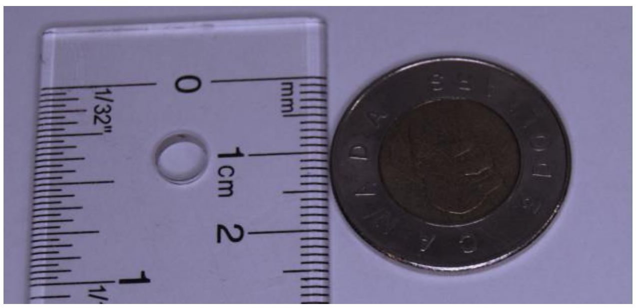


Fig. 7B. The diameter of a $2 Canadian coin is around 2.6 cm

made for the wire that powers the Arduino DUE. Another platform was created to house the breadboard. This platform sits on top of the platform that receives the coin. One thing to note about this platform is that it contains rests that are covered in plastic. These rests hold the wooden arm and require little friction between the rests and the arm and is the reason why plastic covers were fitted on these rests.

III. Results

The development of this project underwent a process of fabrication which first includes researching for project ideas and planning out what materials I needed to complete the project. I would also set deadlines for myself and organize a basic schedule to follow. After the initial planning phase, the next stage will consist of finding the required components online and ordering from the online vendors. Once the components have been delivered, setup and testing of the hardware were done. The hardware was tested for any malfunctions which would determine if any parts needed to be reordered. This was the stage where I would learn about the functionality of each component and how it would help achieve the final objective. Once the hardware setup and testing were complete, the next stage was to complete the software program. Since several programs were made to test the hardware components, the only task needed was to create a structure that would combine the programs of each component. The structure of the program would follow the structure shown in **Figure 2**. The program was tested for any programming errors and small changes were made to improve the functionality of the measurements through the use of delay functions. As a result,

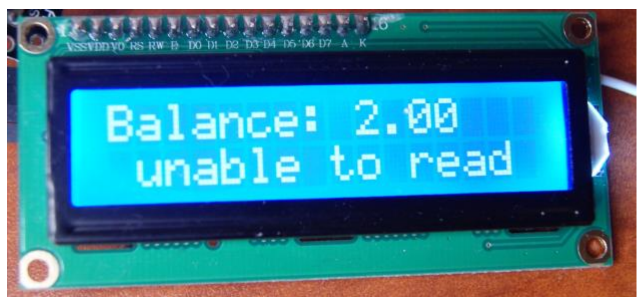


Fig. 8A. Unable to read the coin inserted.

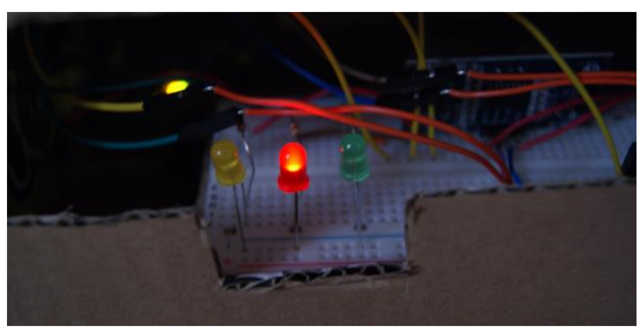


Fig. 8B. The red LED indicator is illuminated to notify the user of an error

the general function of this project is to measure the size of an object. The main application of this project is to identify the value of Canadian coins and keep track of the total value of coins that were identified. **Figure 7A** and **Figure 3** show the successful identification of a toonie ($2 Canadian coin). The coin is inserted into the coin slot which pushed the mechanical mechanism a distance equivalent to the diameter of the toonie. As a result, $2.00 was added to the total balance and is indicated by a confirmation statement displayed on the LCD screen and a green indicator LED being illuminated. **Figure 8A** and **Figure 8B** show the functionality of the project when an invalid coin is inserted. As a result, no changes will be made to the balance shown and the red LED indicator will illuminate.

IV. Discussion

Working on this project presented me with numerous technical difficulties that I needed to overcome. The main technical difficulty experienced when developing this project is having a limited knowledge of C programming. Additional research was required for me to understand the function of different sections of the program and what libraries was needed. I would also have occasional problems with the ultrasonic sensor as it would give slightly inaccurate readings or even give completely wrong readings. By manually calibrating the values by adding or subtracting a single digit from the variable that holds the distance, I was able to fix the slight inaccuracies in the values. In order to fix the major mistakes in the readings, I would position the ultrasonic sensor such that the reference object will never fall within 2 cm of the ultrasonic sensor as it has a theoretical measuring distance of 2 cm to 450 cm. Another solution to this is to use the DHT11 Temperature and Humidity Sensor. There are numerous factors that affect the speed of sound in air. One such factor is the temperature surrounding the ultrasonic sensor. Therefore, the following equation can be used to get the speed of sound factoring in air by factoring the temperature reading [3]:

$$v_{sound} = \frac{331.4 + 0.606T_{°C} + 0.0124H}{10000}$$

Where 331.4 is the unaffected speed of sound in air in meters per second. The humidity is denoted as H and temperature T is measured in degrees Celsius. However, I was not able to implement this solution since the sensor I currently own is malfunctioning and was unable to order a new sensor due to the lack of time. Other technical difficulties include trying to find the correct library for the I2C LCD screen. The original library from the Arduino library manager is incompatible with my current LCD screen. I was required to do additional research to find the correct libraries that is compatible with my LCD screen. Another technical issue with the LCD is that I was having difficulties displaying the values correctly on the screen. An effective solution that I used is to use the function that will clear the screen whenever any characters or variables were changed. This is to keep the formatting of the characters on the LCD screen.

V. Conclusion

Working through the COVID-19 pandemic has proved to be a challenging task as it can be both mentally and physically draining. Despite this issue, I was able to work through the hardships by taking care of my physical and mental wellbeing. A person must maintain a proper sleeping schedule, good eating habits, and regular exercises. A lack of sleep would result in various problems such as feeling tired and being unable to think critically which would affect the quality of the project. Increasing my intake of Vitamin D was also important since staying indoors was becoming more frequent due to lockdowns caused by

COVID-19. Increasing my Vitamin D intake has allowed me to maintain my energy levels which are needed to work on and improve the quality of my project. It is also necessary to dedicate a small amount of time for breaks in order to stretch or perform some exercises. This is also important for good mental health as it reduces stress and gives me time to reflect on my thoughts. Doing these routines has allowed me to dedicate my full attention to the project and by doing this, the final results achieved in this project have successfully met the original objectives set when initiating and planning this project. The two original objectives set were first being able to read different Canadian currencies by differentiating them from their diameter. The second objective set was to count the total value of coins inserted and give the users the ability to reset the balance to a value of zero. The first objective was successfully accomplished through the use of the mechanical mechanism along with the HC-SR04 ultrasonic sensor. The second objective was fully accomplished through the use of the KY-004 button module and a function to sum up the coin values. This will greatly contribute to my future success as I have learned valuable concepts related to embedded systems and have learned the basic foundational knowledge of how payment systems are used in real-world situations such as casinos, banking institutions, and government agencies.

Acknowledgment

I would like to acknowledge the support of my fellow classmates, Yi-Ting Ke and Andy Liang. They have supported me throughout the development of this project by providing me with advice, suggestions, and constructive criticism. I would also like to thank Professor Ebrahim Ghafar-Zadeh and TAs for running the course, guiding me through this project, and helping resolve any problems I encountered. I would also like to acknowledge the people at HowToMechatronics for teaching me the basics of the HC-SR04 ultrasonic sensor as well as the online vendors used to purchase components for my project, Amazon and Digi-key Canada.

References

[1] S. (2020, July 28). *Arduino Coin Sorter and Counter*. The Engineering Minds. https://www.theengineeringminds.com/arduino-coin-sorter-and-counter/

[2] D. (2021, February 5). Ultrasonic Sensor HC-SR04 and Arduino Tutorial. HowToMechatronics. https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

[3] Dubey, A. (2020, August 22). *How to improve readings of Ultrasonic Sensor - Temperature and Humidity Compensation*. NerdyElectronics. https://nerdyelectronics.com/embedded-systems/sensors/ how-to-improve-readings-of-ultrasonic-sensor-temperature-and-humidity-compensation/

[4] Fmalpartida. (2020, November 24). *New*-LiquidCrystal. GitHub Repository, https://github.com/fmalpartida/NewLiquidCrystal

Ballen, Alain. (M'00) currently pursuing a Bachelor of Engineering in Software Engineering at York University, Toronto, ON, Canada. Alain is an aspiring developer looking to further develop his skills in software development. Alain has on multiple software projects both personally and for academically. His most recent project was a Venn Diagram Application which allows users to organize information in the form of a Venn Diagram. It was developed in the Java programming language. One interesting fact about Alain, is that he originally wanted to pursue a degree in medicine. However, Alain decided that Software Engineering is a more appropriate path for him.

ProjectCode

```
#define trigPin 9
#define echoPin 10

#define redLED 6
#define greenLED 7
#define yellowLED 5

#define buttonPin 47

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x3f, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

long duration;
int distance;
int button;
double sum = 0.00;

void displays(){
  // displays the sum
  lcd.setCursor(0,0);
  lcd.print("Balance:");
  lcd.setCursor(9,0);
  lcd.print(sum);
}

void clearLine(int line){
  // clears a line by replacing all characters with a blank space
  String string = "";
  for(int i = 0; i < 16; i++){
    // addes a white space to the string variable
    string += " "; //concatenate a blank space to the string variable
  }
  lcd.setCursor(0, line); //set the position on the LCD
  lcd.print(string); //print the blank spaces on the LCD
}

void resetValue(){
    lcd.setCursor(0,1); //sets the position on the LCD
    lcd.print("Balance reset!");
    sum = 0.00;
    delay(500);
    clearLine(1); //clears the "Balance reset!" statement
}

void getDistance(){
  // trig pin controls when an ultrasonic pulse is sent out
  //will send 8 waves
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin,LOW);

  // reads how long the wave travels to and from the object
```

```
  // reads how long the wave travels to and from the object
  duration = pulseIn(echoPin, HIGH);
  /* measures the distance by multiplying
   * the duration by the speed of soundthen divide by 2
   *  manual calibration by subtracting 1 cm.
   */
  distance = ((duration*0.034)/2 - 1);
}

void setup() {
  Serial.begin(9600); //start communication to the serial monitor.
  //serial monitor is used for testing purposes
  pinMode(trigPin, OUTPUT); //controls the echol pin
  pinMode(echoPin, INPUT); //recieved duration values from the echo pin
  lcd.begin(16,2); //defines the dimensions of the LCD
  lcd.backlight();
  pinMode(greenLED,OUTPUT);
  pinMode(redLED,OUTPUT);
  pinMode(yellowLED,OUTPUT);
  pinMode(buttonPin, INPUT);
}

void loop() {

  getDistance();

  button = digitalRead(buttonPin);
  Serial.print("Distance: ");
  Serial.println(distance);
  Serial.println(sum);

  if(button == LOW){
    // when the button is pressed
    // resets the balance to 0
    resetValue();
  }

  displays();

  if(distance >= 2 && distance < 8){
    // when there is a coin inserted turn on the yellow LED
    digitalWrite(yellowLED, HIGH);
    digitalWrite(redLED, LOW);
    digitalWrite(greenLED, LOW);
    delay(3000); //allows time for the user to adjust the positioning of the coin
    getDistance();

    if(distance == 2){ //can change distance values
        // reading a coin with a value of $0.05 (nickel)
        digitalWrite(greenLED, HIGH);
        digitalWrite(yellowLED, LOW);
        digitalWrite(redLED, LOW);
        Serial.print("Distance: ");
        Serial.println(distance); //testing purposes
        lcd.clear();
        displays();
```

```
      lcd.setCursor(1,1);
      lcd.print("adding 5 cents");
      sum = sum + 0.05;
      delay(5000); //allows time for the user to remove the coin
      clearLine(1); //clear the indication statement

  } else if(distance == 3){ //can change distance values
      // reading a coin with a value of $2.00 (toonie)
      digitalWrite(greenLED, HIGH);
      digitalWrite(yellowLED, LOW);
      digitalWrite(redLED, LOW);
      Serial.print("Distance: ");
      Serial.println(distance); //testing purposes
      lcd.clear();
      displays();
      lcd.setCursor(1,1);
      lcd.print("adding 2 dollars");
      sum = sum + 2.00;
      delay(5000); //allows time for the user to remove the coin
      clearLine(1); //clear the indication statement

  } else {
    // Coin is unable to be read
    digitalWrite(redLED,HIGH);
    digitalWrite(greenLED,LOW);
    digitalWrite(yellowLED, LOW);
    Serial.print("Distance: ");
    Serial.println(distance);
    lcd.clear();
    displays();
    lcd.setCursor(1,1);
    lcd.print("unable to read");
    delay(5000); //allows time for the user to remove the coin
    clearLine(1); //clears the indication statement
  }
  /*distance values can be changed to match other Canadian coins
   * device does not use a DHT11 sensor
   */
 } else {
  // turn off all LEDs when there is no coin being inserted
  digitalWrite(yellowLED, LOW);
  digitalWrite(redLED,LOW);
  digitalWrite(greenLED, LOW);
 }

 delay(100);
}
```