Bill Tech Dollar Identifier

By

Justin Hsieh

Javier Matinez Linera

Pratheek Eravelli

Final Report for ECE 445, Senior Design, SPRING 2022

TA: Jeff Chang

04 May 2022

Project No. 11

Abstract

This device is an all-in-one embedded system that performs image classification on currency. To be more specific, the Bill Tech Dolor Identifier uses a convolution neural network (CNN), a machine learning architecture used to classify pictures, and a camera to recognize paper US currency placed by the user. In the development of the device, core components worked separately but failed when integrated with one another. The biggest issue that we met was the memory size of the microcontroller. The separate sections cost too much memory and when brought together broke the system. Ultimately, the following document will guide you through the motivation of making this device, the specific design challenges that resulted in the device not working, and the successes behind each of the device subsystems.

Contents

1 Introduction

1.1 Problem

In the United States, paper currency cannot be easily identified by those with either visual impairments or blindness. This reality is a uniquely American one, as America is one of the few nations that does not have tactile markers or different sizes in its paper currency [1]. For those who are blind in the United States, many have developed identification systems that require the folding of bills in unique ways [1]. This system works when the identity of the bill is known in advance, but it can become messy when the user is given unknown bills. Without the help of another person this is often a difficult task and for those who live alone, almost impossible. In order to aid those with visual impairments who live alone, we propose a desktop device that can identify any paper legal tender in the United States, the Bill Tech Dollar Identifier.

1.2 Solution

The Bill Tech Dollar Identifier works by taking a picture of. The user puts the piece of currency on a tray and pushes a button on the physical device. After a brief delay, the device will take a picture of the bill and after a few seconds, the microprocessor is able to correctly identify the value of the bill and output its value from the speaker of the device.

This device differs from mobile app solutions because it does not require a phone to use it. The device follows design standards for ease of use that make it more accessible. The Bill Tech Dollar Identifier only uses a button and power switch as inputs from the user, allowing the user to easily become familiar with the controls.

1.3 Final Product

Figure 1 displays the final design of the device. The device uses a white plate for the user to place currency onto. The white plate allows for less background noise to the camera and ultimately to the CNN. The camera is mounted top down in order to extract as many recognizable images as possible from the bill. In addition to this top mounted camera view the device also has light in order illuminate the bill, resulting in a clearer photo for the actual device.

In the end we wanted a device that was able to do the following:

1. Produce different audio Reponses for each bill: $1, $5, $10, $20, and $100
2. Run independently on battery life for at least 1 hour.
3. Perform the model with a classification accuracy of 0.95 +/- 0.02.

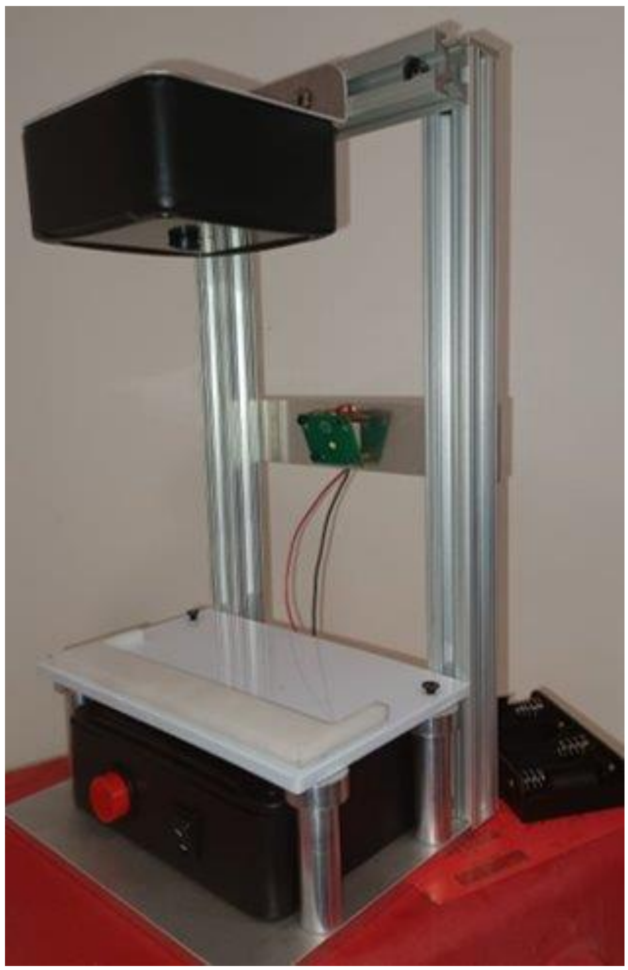

Figure 1. Final Model of the "Bill Tech Dollar Identifier"

2 Design

We will go over the overall design of the “Bill Tech Dollar Identifier” as well as talking about the design details within and between blocks.

2.1 Block Diagram

The final design result in Figure 2 displays how data and power flow through the device. Everything starts with the power subsystem; this subsystem validates the power going into all other blocks allowing for smooth operation of electrical components. This voltage feeds into the microprocessor system that holds the code that runs the actual device while also holding the model that will be processing camera data. The microcontroller sends IO based on the processed image data and tells the buzzer to output a certain number of sounds corresponding to the correctly identified value of the bill. For example, if the user puts a $1 bill the device beeps once or if the user puts down a $5 bill the device beeps twice.

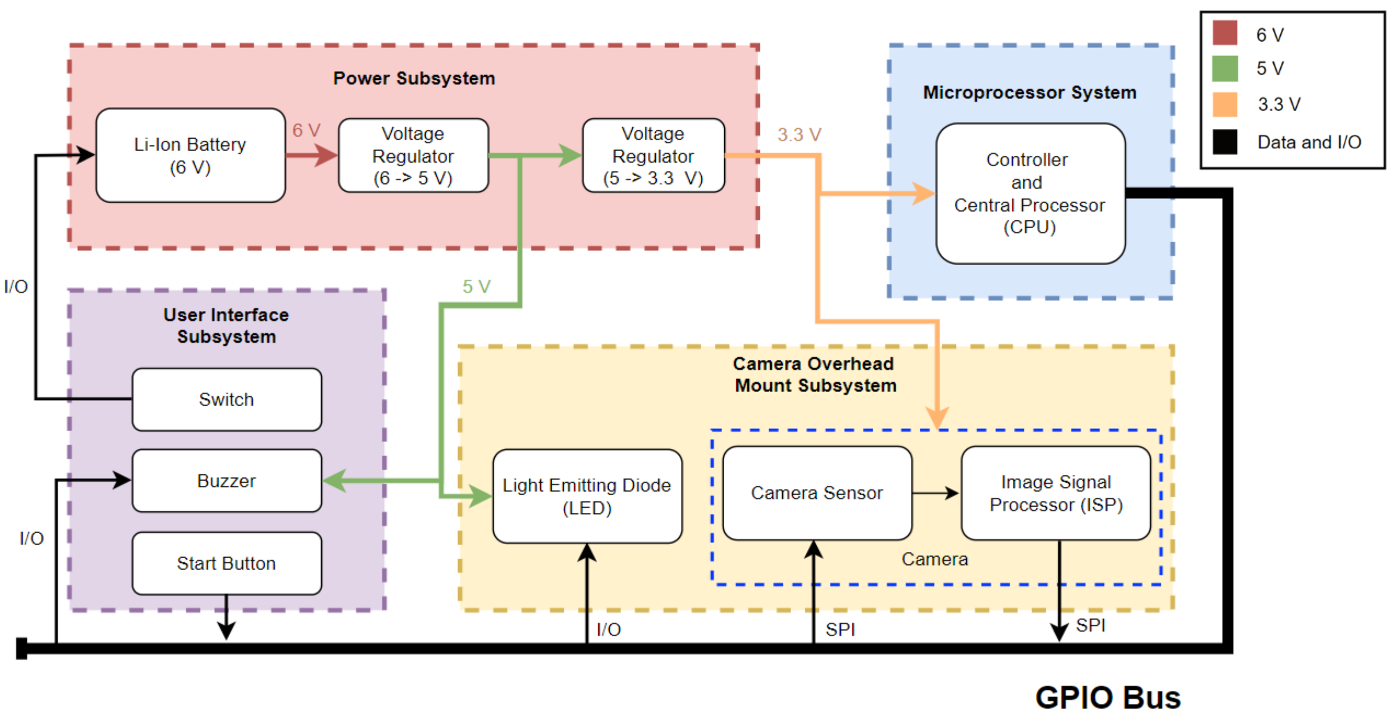

Figure 2. Final Model Block Diagram

2.2 Power Subsystem

The Power Subsystem is responsible for supplying the necessary voltage to the different parts of the circuit. A 6 V source is being used and the voltage is dropped to 5 V and to 3.3 V. The 3.3 V is used to power the microcontroller, the button debouncing circuit and the camera. The 5 V are used for the buzzer and LED circuits. Figure 3 shows the schematic of the power subsystem circuit.

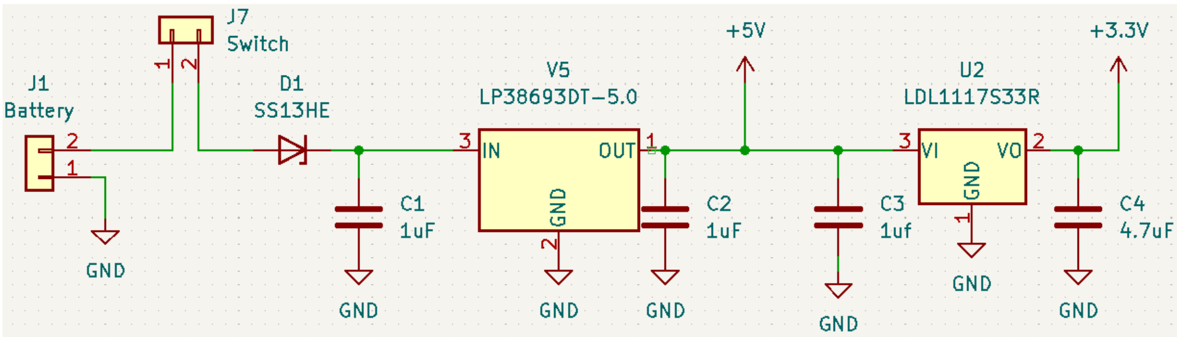

Figure 3. Power Subsystem Schematic

2.2.1 Battery

A 6 V battery has been used to power the device for more than 1 hour. To provide the 6 V to the circuit, 4 1.5 V C-type alkaline batteries are connected in series. The voltage of these batteries is added together to obtain the 6 V needed. 1.5 VC-type alkaline batteries are used so that when the battery runs out the user can easily replace it.

2.2.2 Schottky Diode

A Schottky diode has been used to protect the circuit in case someone connects the battery in the opposite polarity. It has been chosen a diode that allows a high current flow of 1 A, but that its current reverse leakage is small 50 µA at 30 V to get a better protection of the circuit.

2.2.3 Voltage Regulators

We needed 3.3 V and 5 V for our project and to obtain these voltages we are using a 6 V battery and two step-down voltage regulators. We are using LDOs instead of Buck Converters because of the simplicity of their design, the smaller device size and the absence of switching noise. The disadvantage is that linear DC regulators must dissipate power and thus heat.

$$Power\ Dissipation = (Vi - Vo) * Io$$

To minimize the power loss, the LDOs are placed to step down the voltage first from 6 V to 5 V and then from 5 V to 3.3 V instead of directly converting 6 V to 3.3 V. To help dampen noise on the input and output of the LDOs we added the capacitor recommended by the manufacturer.

2.3 User Interface Subsystem

The User Interface block includes the switch, start button and the speaker. It is through these elements that the user physically interacts with our device. The switch and the button detect the user's inputs and send the signal to the microcontroller. The speaker works as an output, producing differentiable audible responses so the user can identify the bill value.

2.3.1 Buzzer

The buzzer used to reproduce the beeps needs a current of 30 mA and a voltage range between 3 V and 7 V. As the maximum output current of the microcontroller is 20 mA, an N-Channel MOSFET it is used working as a switch and allowing to provide the current that the buzzer needs. The designed circuit is shown in Figure 4.

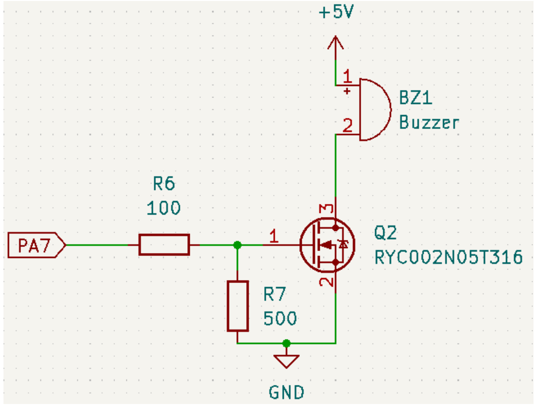

Figure 4. Buzzer Circuit Schematic

The mosfet is operating in triode mode and follows the following calculations:

$$Vgs = V12 = 3.3 * \frac{500}{600} = 2.75\ V \qquad Vds = 0.1\ V$$

$$Io = \frac{3.3}{600} = 5.5\ mA < Imax = 20\ mA \qquad Id = 30\ mA$$

2.3.2 Button

When the button is pressed, the button connects and disconnects to ground before finally settling down. A low pass filter has been designed to eliminate this disturbance [2]. A simple hardware solution using a capacitor and a resistor, RC filter, is used to prevent the button bouncing. The debouncing circuit is shown in Figure 5.

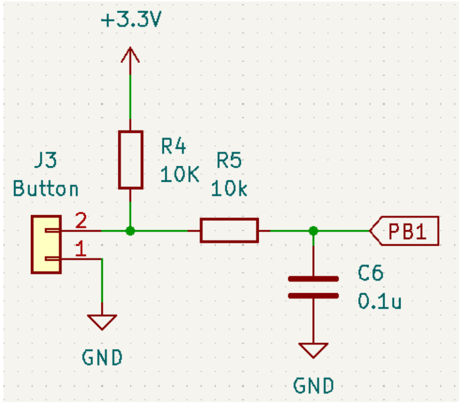

Figure 5. Button Debouncing Schematic

2.4 Camera Subsystem

This system ties directly with the microprocessor and the user interface. When the user presses the start button, with the device turned on, the microprocessors system sends a control signal to the camera and the LED. The LED will then turn on to illuminate the tray, which holds the dollar, and the camera system will take the picture. The ISP on the camera will then process the photo and go through Analog to Digital Conversion and send this data through SPI and I2C to the microprocessor for image analysis.

2.4.1 LED

As the device is designed to assist the visually impaired or blind, there is a possibility that the room where the device is located may be poorly lit. In order to take a good picture of the bill and output its value. An LED is used to ensure that the bill is well illuminated.

The minimum luminance flux required to get a good illumination of the tray is 50 Lm. The LED used in the device provides more than the 50 Lm needed. It needs a current of 150 mA, and a voltage of 3.3 V. As the maximum output current of the microcontroller is 20 mA, an N-Channel MOSFET it is used working as a switch and allowing to provide the current that the LED needs.

Since an LED with sufficient power to illuminate the bill is needed, the LED is surface mount. A second PCB is used to solder the LED. The designed circuit of the LED and the schematic of the second PCB are shown in Figure 6.

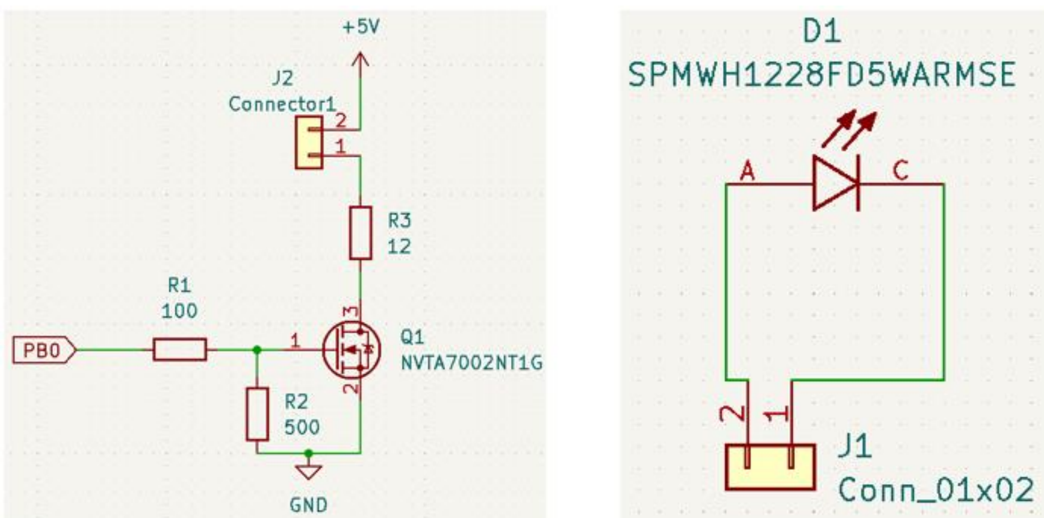

Figure 6. LED Circuit Schematic / LED circuit for PCB 2

The mosfet is operating in triode mode and follows the following calculations:

$$Vgs = V12 = 3.3 * \frac{500}{600} = 2.75\ V$$

$$Vds = 0.3\ V$$

$$Io = \frac{3.3}{600} = 5.5\ mA < Imax = 20\ mA$$

$$Id = 150\ mA$$

2.4.2 Camera

For the camera subsystem we used the ArduCam 2MP Plus. This camera was chosen because of the microcontroller used for the final design. The camera doesn't require complicated communication protocols and was able to communicate with the microcontroller via I2C and SPI[3].

I2C was used to communicate with the physical properties of the camera like image brightness, saturation, and exposure while SPI was used to change photo resolution and data acquisition. With the use of SPI the microcontroller is able to use a relatively simple communication protocol in order to send and receive image data. In addition to simple communication, this camera also handled a lot of the image processing that would take up valuable resources on the main microcontroller of device. The image processor is the big vertical chip as can be seen in Figure 7.

Figure 7. Arducam 2MP Plus

2.5 Microprocessor Subsystem

This block, Figure 8, takes care of both controlling the individual elements like the LED and the speaker but also performs most of the image analysis. This system will have a CNN loaded onto its flash memory while also handling the image buffer loaded from the camera. On the microcontroller that we will be using in this device we have 512Kb of flash as well as 148kb of ram.

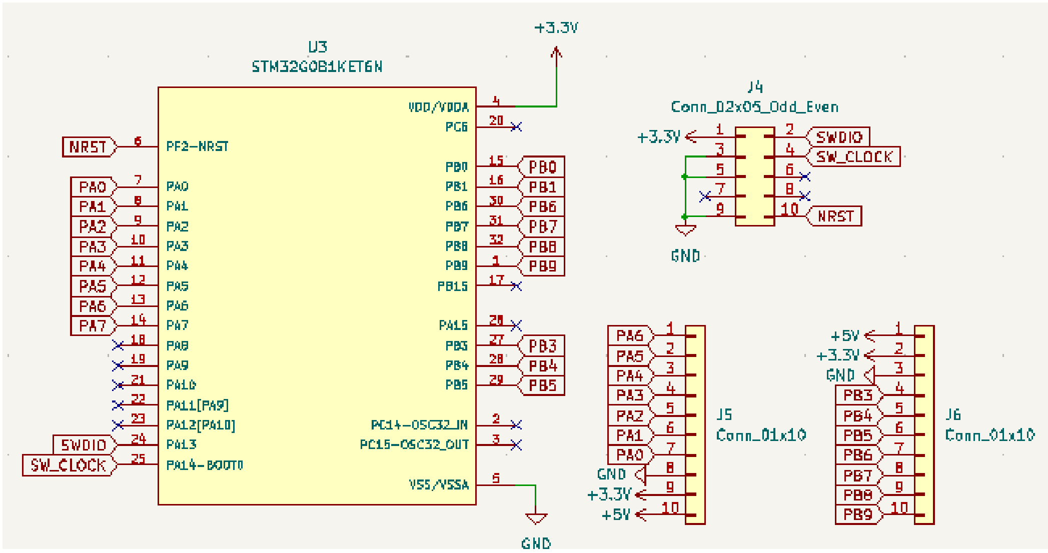

Figure 8. Microcontroller and Connectors Schematic

2.6 Image Classification Model

For classifying the bills on the platform, we used a convolutional neural network (CNN) as our image classification model [4]. We gathered hundreds of training images for our dataset settled on implementing two architectures, ResNet50 and MobileNetv2, using transfer learning, to train the network.

2.6.1 Dataset Collection

In order to begin training on the convolutional neural network, a dataset with the required information needed to be acquired. The goal of this training dataset is to provide the neural network with reference images that replicate the environment of the actual device. These reference images are labeled correspondingly to their respective classes and allow the network to learn and distinguish patterns between these images. Our dataset images were taken using an APS-C crop sensor camera with a 35mm lens 24 inches above standard letter paper as the platform on which the bills were placed upon. Around 40 images were taken for each side of the bill placed in different orientations on the surface. In total, over 80 images were obtained for each of the 5 classes. Figure 9 show how the images are collected.

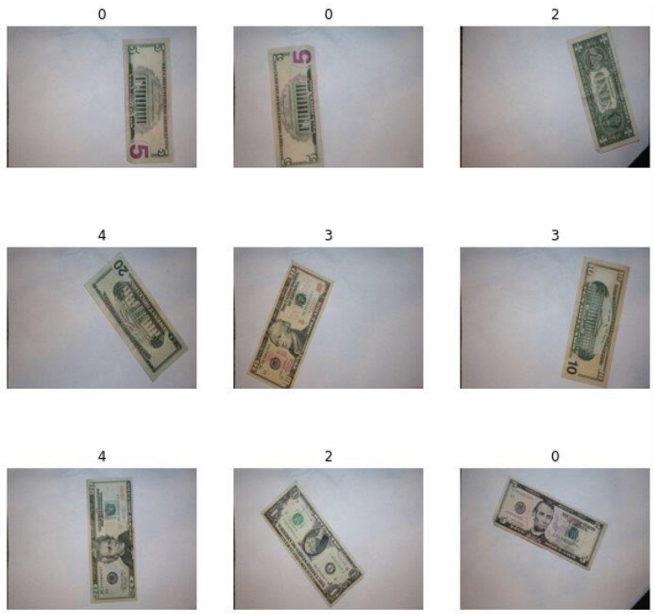

Figure 9. Example Dataset Images with Labels

2.6.2 Dataset Augmentation

For a convolutional neural network to be effective, hundreds if not thousands of training images are required. Even though we obtained hundreds of images by hand, the network needs more images in order to produce our desired accuracy goal [5]. In order to achieve this, we used a technique called data augmentation as seen in Figure 10. Each image from the training dataset was randomly rotated, flipped, and transposed in order to create hundreds of additional images for each hand collected sample. This allowed the training dataset to amass over a thousand images in total.

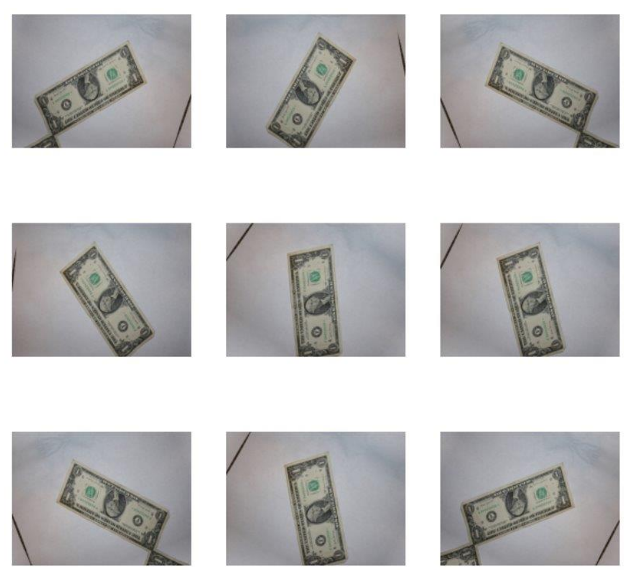

Figure 10. Example Data Augmentation on a Single Image

2.6.3 Model Architecture

Our first attempts on using the LeNet and AlexNet architectures resulted in low validation accuracy. Switching to the more complex ResNet50 allowed us to obtain a validation accuracy reading of 99.9%. However, with the packaged model requiring over 2.37 gigabytes of space, it was impossible to load onto our microprocessor. Even after quantization, the process of compressing the model using smaller data structures, the model still required almost 100 megabytes of storage. We then tried to use the MobileNetV2 architecture to try to downsize the model to fit onto the microprocessor. The advantage of the MobileNetV2 model is very high compression but suffers from worse accuracy. After quantization, we were able to get the model trained on MobileNetv2 to 2.53 megabytes. Even though this presented a huge improvement over the initial two gigabytes, it was still too large to fit onto our microprocessor.

2.6.4 Training and Verification

The validation dataset was created by randomly subsampling the initial training dataset. These images were set aside and not used during the training process. After each iteration of the training process, the validation dataset was run through the model in order to evaluate the accuracy of the network. network. This is shown by the red line in both figure 11 and figure 12. After training both the ResNet50 and MobileNetV2 models, we were able to measure roughly 99.9% validation accuracy on the ResNet50 model and roughly 80% validation accuracy on the MobileNetV2 model.

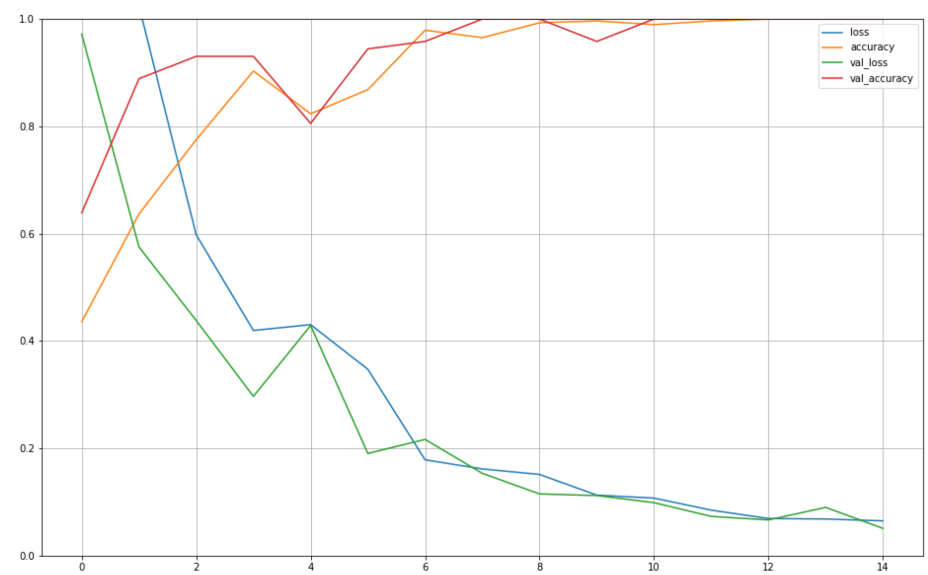

Figure 11. ResNet50 Evaluation Graph

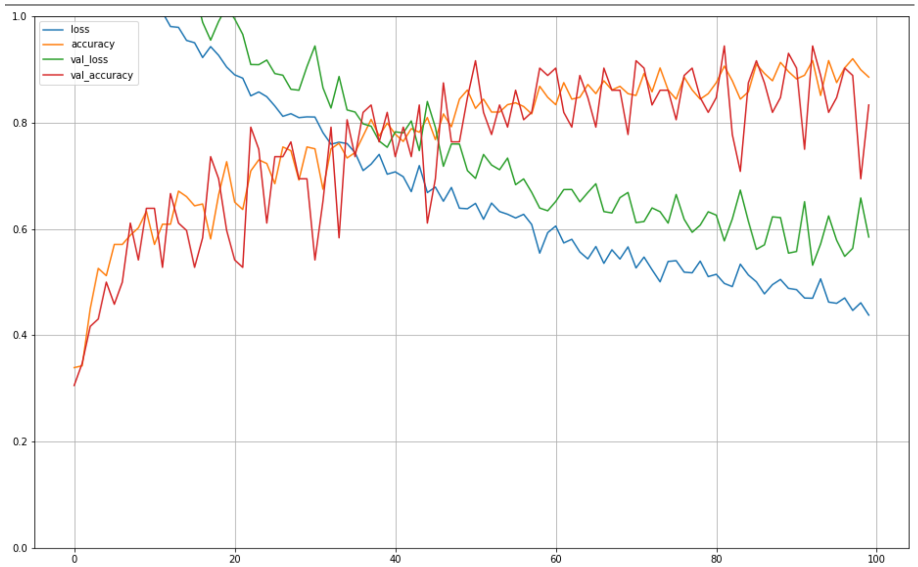

Figure 12. MobileNetV2 Evaluation Graph

3 Design Verification

See Appendix A for a comprehensive list of our requirements and verifications.

3.1 Power Subsystem

3.1.1 Battery

As one of our requirements is that the circuit battery must last for more than 1 hour, a battery with sufficient power has been used.

Batteries when discharged decrease their voltage. The voltage regulator that drops the voltage from 6 V to 5 V needs an input voltage higher than 5.5 V to be able to provide the desired voltage. As we are using 4 batteries of 1.5 V the minimum voltage of each battery for the circuit to work is 1.1 V.

Our device has several processes: turn on the LED, take a picture, running the model and output the bill value with the buzzer. These processes work in series, so they do not occur at the same time. The one that consumes the most current is the LED which consumes 150 mA.

Figure 13 is a graph from the battery datasheet that shows how the battery voltage changes over time depending on the current used. Even if the device is continuously operating at the maximum current 150 mA the battery would last more than 1 hour.

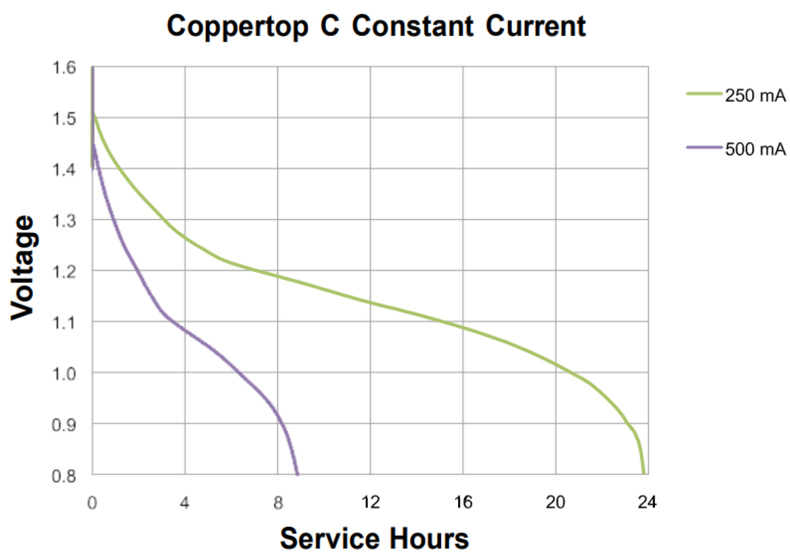

Figure 13. Voltage - Service Hours graph

3.1.2 LED

The minimum luminance flux required to get a good illumination of the tray is 50 Lm. To provide those lumens an LED with sufficient power has been used to obtain a good image of the bill. As can be seen in Figure 14, if 150 mA is supplied to the LED, it provides between 61 and 65 Lm, which is high enough to obtain a good image.

a) Luminous Flux Bins (I_F = 150 mA, T_s= 25ºC)

CRI (R_a) Min	Nominal CCT (K)	Product Code	Flux Bin	Flux Range (Φ_v, lm)
	6500	SPMWH1228FD7WAP☆VG	VG	61.0 ~ 65.0

Figure 14. Voltage - Service Hours graph

To check the current flowing through the LED, we have measured with a multimeter the real value of the resistance R3 shown in Figure 6, and the value of its voltage. It has been obtained that the current I = V/R is 147 mA. Therefore, the actual illumination that we are providing to the banknote should be sufficient.

4. Costs

In the following sections we will be discussing the total costs in both materials and labor while also discussing how work was divided throughout the school year.

4.1 Parts

Table 1 shows a total bill of all parts required for a single run of making the "Bill Tech Dollar Identifier". A lot of the major costs for creating this product are in the one-time purchases that are needed for production. Examples of one-time purchases are, the st-link v2 which is used for programming and can be reused, the TC-2050 Adapter which is also only needed for programming, and the programming cables which are also reusable. Without these three expenses the recurring cost for making one of these devices would be $60.99 which could be brought down even further with bulk ordering.

With the addition of labor from table 2, we can see that the total production cost for this device is $38,616.44.

Table 1. Part Costs for Final Product

Part Name	Part Manufacturer	Quantity	Price per part	Total
08055C105JAT2A 1 uF Capacitor	KYOCERA AVX	3	$0.44	$1.32
CL21A475KAQNNNE 4.7 uF Capacitor	Samsung Electro-Mechanics	1	$0.11	$0.11
RMCF0805FT100R 100 Ω Resistor	Stackpole Electronics	2	$0.10	$0.20
RC0805FR-07499RL 500 Ω Resistor	YAGEO	2	$0.10	$0.20
ERJ-P06J120V 12 Ω Resistor	Panasonic Electronic Components	1	$0.13	$0.13
ERJ-P06J103V 10 kΩ Resistor	Panasonic Electronic Components	2	$0.13	$0.26
LP38690DT-5.0/NOPB 5V LDO	Texas Instruments	1	$2.38	$2.38
TC2117-3.3VDBTR 3.3V LDO	Microchip Technology	1	$1.00	$1.00
SPMWH1228FD5WAPUVG White LED	Samsung Semiconductor	1	$0.17	$0.17
NVTA7002NT1G N-channel Mosfet	onsemi	1	$0.43	$0.43
MBR230LS Schottky Diode	SMC Diode Solutions	1	$0.42	$0.42
IE092505-1 Buzzer	DB Unlimited	1	$2.51	$2.51
Continued on next page				

Table 1. Part Costs for Final Product				
Part Name	**Part Manufacturer**	**Quantity**	**Price per part**	**Total**
RYC002N05T316 N-channel 5V Mosfet	Rohm Semiconductor	1	$0.46	$0.46
C-MN1400 1.5V C type batteries	Duracell Industrial Operations	4	$1.19	$4.76
4X C Cell Battery Holder	SDTC Tech Store	1	$4.24	$4.24
3-640440-2 1x02 Crimp Connectors	TE Connectivity AMP Connectors	4	$0.17	$0.68
0022232021 1x02 PCB Header	Molex	4	$0.22	$0.88
61201021621 2x05 Even Odd Male Header	Wurth Elektronik	1	$0.46	$0.46
0022272101 1x10 Male PCB Header	Molex	2	$0.84	$1.68
PR144C1900 Panel Mount Push Button	E-Switch	1	$2.30	$2.30
A8L-21-11N2 Panel Mount Power Switch	Omron Eletronics	1	$3.72	$3.72
ST-LINK/V2 External Programmer	STMicroelectronics	1	$21.25	$21.25
TC2050-ARM2010 ARM 20-pin to TC2050 Adapter	Tag-Connect	1	$29.95	$29.95
2.54mm Pitch 2 Row 10 Pin Female to Female Wires IDC Ribbon Connector	C & Xanadu	1	$4.00	$4.00
OV2640 2 Megapixels Lens	Arducam	1	$26.00	$26.00
STM32G0B1KET Microcontroller	STMicroelectronics	1	$6.93	$6.93
Total				$116.44

Table 2: Labor Costs for Development

Name	Hourly Rate	Hours	Total	Total x 2.5
Javier Martinez	$40.00	120	$4800.00	$12000.00
Pratheek Eravelli	$40.00	120	$4800.00	$12000.00
Justin Hsieh	$40.00	120	$4800.00	$12000.00
Machine Shop Personal	$50.00	20	$1,000.00	$2,500.00
Total				$38,500.00

4.2 Schedule

Table 2 shows how work was divided during the 10-week given for development time.

Table 2: Schedule of Work Throughout the Project

Week	Task	Pratheek	Justin	Javier
02/21	Design Document	1. Design Document 2. Circuit Schematics 3. Order Parts	1. Design Document 2. PCB Design	1. Design Document 2. PCB Design
02/28	PCB Review	1. Learn SPI protocol to communicate with the camera 2. Tolerance Analysis	1. Research Image Classification Models 2. Tolerance Analysis	1. Research Image Classification Models 2. Tolerance Analysis
03/07	PCB Order #1	1. Prototype with development board	1. Start Training Model	1. Look at parts needed and write down delivery list
03/14	Spring Break	1. Assemble Parts List	1. Assemble Parts List	1. Assemble Parts List
03/21	PCB Order #2	1. Help with PCB design as well as finalizing footprints.	1. Continue training model and working on expanding data set.	1. Primary PCB design and try to integrate a physical buzzer into circuit,
03/28	Individual Progress Report	1. Write Individual Progress Report. 2. Help with soldering components and verifying PCB	1. Write Individual Progress Report. 2. Work with memory issues and improve accuracy of model	1. Write Individual Progress Report. 2. Verifying PCB components.
04/04		1. Test Software Operation 2. Fix Possible problems		
04/11		1. Prepare for Mock Demo		
04/18	Mock Demo	1. Implement Mock Demo Suggestions 2. Final Assembly with device from machine shop		
04/25	Demonstration	1. Demonstrate Project Operation 2. Send final paper to be reviewed 3. Start working on the presentation		
05/02	Presentation Final Papers	1. Writing and formatting the final paper.	1. Creating PowerPoint for presentation.	1. Creating visuals and graphics for both final paper and presentation

5. Conclusion

5.1 Accomplishments

At the end of the day, the group was able to develop a CNN with 99.9% validation accuracy, have a completely validated and working PCB, and were able to program and execute code from the microcontroller. As we have discussed earlier, a lot of the problems with this device revolved around the miss use of memory and being unable to get these blocks to combine.

5.2 Uncertainties

The main reason why our final project wasn't functional was due solely to the lack of flash memory onboard the microprocessor. The microprocessor that we integrated into our system only had a flash memory capacity of 512 kilobytes. This memory is meant to be shared between storing the image classification model, storing the image buffer from the camera, and storing the controller code to operate the system. In addition to these storage tasks, the flash memory would also potentially need to serve as a buffer for the RAM while the processor is performing calculations. This is because the microprocessor only had 148 kilobytes of RAM. Without an additional SRAM chip or storage interface, there was not enough capacity built into the microprocessor to contain all the necessary code required for the full functionality of the device.

5.3 Ethical considerations

The main ethical concerns for this project come from taking pictures of legal currency. The United States government allows for individuals to take pictures of US currency if these pictures are either less than 75% of the original size or greater than 150% of the original size. These pictures also require deletion after use [6]. During the training and designing of this device, the pictures of the bills are never saved for very long. As soon as the images are used for training the model they are no longer needed and are deleted. Because of this practice, the development of the device followed all legal and regulatory standards provided by the United States.

In terms of safety concerns this device is operating with both low voltage and current levels which do not pose a risk to any of the group members. LEDs will not be of a strong enough luminance to pose a risk to vision and speakers will not produce a loud enough sound to cause hearing loss. In order to uphold safety standards, members of the group have complied with all safety precautions in the senior design lab and have been carefully following instructions on proper etiquette and technique when soldering components.

We have followed all codes of ethics from IEEE [7] by both upholding moral and ethical standards among ourselves and others, continually considering ethical solutions to problems that do not introduce conflicts of interest.

5.4 Future work

One possible way to get these components to combine would be to get the microcontroller to take a picture. Send the image data to a computer and run the model. Once the model is done running send the data back to the microcontroller. The microcontroller is then able to output the identity of the bill in a series of short beeps. By leveraging the fact that all the sub systems work separately, we could use a distributed approach to achieve final functionality. Another approach would be to integrate some form of additional storage onto our PCB for the microprocessor to interface with. This could be in the form of an SRAM chip or even a MicroSD expansion slot. By adding this additional storage, the microprocessor would be able to store and compute the model.

References

[1] “How Do People Who Are Blind or Visually Impaired Identify Money?”

chicagolighthouse.org

https://chicagolighthouse.org/sandys-view/identifying-money/ (accessed May. 2. 2022)

[2] HACKADAY. “Debounce your noisy buttons”
hackaday.com
https://hackaday.com/2015/12/09/embed-with-elliot-debounce-your-noisy-buttons-part-i/
(accessed Feb. 12. 2022)

[3] Arducam, “ArduCAM-M-2MP Camera Shield”,

https://www.arducam.com/downloads/shields/ArduCAM_Mini_2MP_Camera_Shield_Hardware_Application_Note.pdf (Revised March. 2015),

[4] NeurIPS “ImageNet Classification with Deep Convolutional Neural Networks”

Proceedings.neurips.cc

https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[5] Proceedings of the IEEE “Gradient-Based Learning Applied to Document Recognition.”

proceedingsoftheieee.ieee.org

http://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf

[6] U.S. Currency Education Program. “Currency Image Use”

uscurrency.gov

https://www.uscurrency.gov/media/currency-image-use (accessed May. 3. 2022)https://www.uscurrency.gov/media/currency-image-use (accessed Jan. 28. 2022)

[7] IEEE. “IEEE Code of Ethics”
IEEE.org

https://www.ieee.org/about/corporate/governance/p7-8.html (accessed May. 10. 2022)https://www.ieee.org/about/corporate/governance/p7-8.html (accessed Feb. 10. 2022)

Appendix A Requirement and Verification Tables

Table 3. Requirements-Verification table for Power Subsystem

Requirement	Verification	Verified Status
Power System 1. The battery provides 5 V +/- 0.5 %.	1. Verification Process for Item 1: (a) Measure the voltage after the voltage regulator using a voltmeter to ensure the voltage is in the range of 5 V +/- 0.5 %.	Yes
2. The battery provides 3.3 V +/- 0.5 % to the microcontroller, speaker, camera, and LED.	2. Verification Process for Item 2: (a) Measure the voltage after the voltage regulator using a voltmeter to ensure the voltage is in the range of 3.3 V +/- 0.5 %.	Yes
3. Protect the circuit if someone connects the battery in opposite polarity.	3. Verification Process for Item 3: (a) Connect the battery in opposite polarity and measure the current using an ammeter	Yes

Table 4. Requirements-Verification table for Microprocessor System Mount Subsystem

Requirement	Verification	Verified Status
Microprocessor 1. Neural Network must be able to fit on a higher end microcontroller with flash memory values of 512Kb.	1. Verification Process for Item 1: (a) Evaluate size of neural net model as well as image size from camera. (b) Scale down both image size as well as model complexity. (c) While scaling down both parameters adjust for accuracy until 0.95 accuracy is attained while also being able to fit on the microprocessor.	No

Table 5. Requirements-Verification table for User Interface Subsystem

Requirement	Verification	Verified Status
Button and Switch 1. Button should be debounced allowing for smooth transition between analog and digital signals.	1. Verification Process for Item 1: (a) Attach our mechanical switch to the debounce circuit. (b) With the circuit powered on, look at the trace once the switch is turned on and off. (c) The image on the oscilloscope should be smooth without any very rampant voltage jumps.	No
Buzzer 1. Speaker should be able to produce at a minimum 7 different beep sequences as to distinguish each US bill denominations	1. Verification Process for Item 1: (a) Attach a 10 Ω resistor in series with the speaker. (b) Set the voltage to 5 V and put 3.3 V in the microcontroller pin. See if the buzzer beeps. (c) Introduce the code to the microcontroller and prove the 7 different beep sequences	Yes
2. Speaker should exceed 60 dB in volume	2. Verification Process Item 2: (a) Hear if the speaker can be comfortably heard in a quiet room. (b) This step can be done during step 1 C.	Yes

Table 6. Requirements-Verification table for Camera Overhead Mount Subsystem

Requirement	Verification	Verified Status
Camera 1. Camera must be mounted high enough as to not hinder loading and unloading the bill by the user but close enough to capture a high-resolution image of the bill, at least 0.3 MP.	1. Verification Process for Item 1: (a) The camera must be of sufficient quality to be able to read the value of the bill at a minimum height of 25 cm. (b) Test the model at different heights. (c) Choose the smallest height that doesn't negatively affect the accuracy of the model.	No
LED 1. LEDs must be capable of outputting 30 Lumens in brightness sustained at 25 cm.	1. Verification Process for Item 1: (a) Use a phone app like Lux Light Meter to check light levels in a dark room.	Yes