Component Analysis

Year: 2025 Semester: Spring Team: 20

Project: Encrypted USB Drive

Creation Date: February 8, 2025 Last Modified: February 8, 2025

Member 1: Abhijay Achukola Email: aachukol@purdue.edu

Member 2: Brandon Liu Email: liu3394@purdue.edu

Member 3: **Stanley So** Email: sos@purdue.edu

Member 4: Joshua Wai Email: waij@purdue.edu

Assignment Evaluation: See Rubric on Brightspace Assignment

1.0 Component Analysis:

Component 1 will be the microcontroller, which will be the processing center for our device, handling all the inputs, outputs, and managing data in/out of the device. Component 2 will be the LCD display, which will inform the user on the status of the device, whether it is locked or not, and display the password as it is entered. Component 3 will be the keypad where passwords will be entered to gain access to the device. Component 4 will be the flash memory IC, where the data for each user will be stored in separate ICs. Finally, Component 5 will be the fingerprint sensor, which will be used to gain access to the device as well.

1.1 Analysis of Component 1:

We will start by comparing the central part of our project, the microcontroller. We decided to stick with an STM32 type of microcontroller due to our familiarity with it in ECE 362 and our familiarity with the STM32Cube IDE. We then had to consider what functionalities our microcontroller had to do. It must be able to communicate via SPI to four flash ICs, communicate via SPI with an LCD, iterate between pins and read pins to utilize the keypad effectively, communicate via UART to the fingerprint sensor, communicate the data through USB at a minimum of 12 Mbps, and be able to handle a menu system and storing passwords to help connect all the individual components and present a good ui. We had an option to disregard the need for USB handling on microcontroller and instead sending data over UART, but we felt that it would be needlessly difficult and could mess up the USB timing requirements, so we instead opted for looking at microcontrollers that handled USB on chip.

This resulted in narrowing down our search to three families of microcontrollers, the STM32L4 family, the STM32F7 family, and the STM32H7 family. All three of these contain enough peripherals for all functions of the base model for our project but vary slightly on other functions it brings to the table. All three handle USB on chip, but the L4 family only has access to full-speed USB transmission, not high-speed. The first microcontroller to be eliminated was the L4 family, as using a low-power chip isn’t particularly useful, as we’ll be powering our device through the USB port itself, so the lower processing power and other downsides aren’t really worth saving a few minutes of battery life of a laptop or having no effect on PCs. That leaves the H7 and F7, both having top-of-the-line processing power. These two microcontrollers were similar, but we ended up choosing the H7 as we believe the high-performance edge it has over the F7 will help us with our responsiveness while managing the 4 GB flash ICs and USB protocol timings.

1.1 Analysis of Component 2:

One of the main components to display directions to the user is via an LCD screen. This LCD screen is to connect with the MCU with ease and ideally be communicated with using the SPI interface; this interface has been used by our group before so the learning curve should be short, but if it is difficult to identify a purchasable component that operates in SPI, then a different custom protocol can be considered. The other main constraint is that the LCD should be two rows, that way users can see a prompt while also seeing what they are typing on the keypad (which will then be displayed on the LCD).

The metrics mentioned above will be used to determine which LCD to obtain, and weights will be assigned to each metric to reflect its importance, and the highest score will be the component our team will pick. Each component will be ranked 1 through 3 on the weighted metrics with 3 being the most ideal and 1 being least ideal. In the case that multiple products fulfill the metric in the same way, they will receive the same rank. Both metrics are weighted the same because the # of rows is an important requirement to streamline the user experience and the learning curve is important because in previous classes, we have operated with LCDs before, so if we can handle this part quickly, we can save more time to focus on more difficult aspects of the project.

|  |  |  |  |
| --- | --- | --- | --- |
| Product | # of rows  (w = 1) | Learning curve  (w = 1) | Total Score |
| (HD44780 LCD) | 2 | 1 | 3 |
| (WEH001602A LCD) | 2 | 3 | 5 |
| (SPI LCD) | 2 | 2 | 4 |

After comparing the 3 devices, we will probably pick the WEH001602A LCD if we can buy it (we couldn’t find a site that lets us purchase the part yet) because it is the same LCD that we can prototype with and have used in ECE 36200 and can be communicated with via the SPI interface. The SPI LCD is the runner up because it is a rather different LCD and has more than 2 rows, but since it uses a protocol that we are familiar with, it is second place. Here’s where the rankings get a little interesting; although the HD44780 is not indicated to be using the SPI interface and has its own little protocol and commands, the pinout is very similar to that of the WEH001602A LCD and the same pins all have the same purpose. The datasheet for this device does not mention SPI, so the learning curve here is a little unknown and we do not know if the two parts are the same. If they are, then the HD44780 can easily be purchased on digikey, but the datasheets presented on the site do not mention SPI anywhere.

1.1 Analysis of Component 3:

One of the main components to allow user interaction with the thumb drive itself is via a keypad, which will request a user identification and password to process authentication. A matrix keypad will be ideal as it is a component that we have used and programmed before. We also don’t need that many buttons because there are only 4 users and each can use a numeric passcode for authentication, so numbers from 0 through 9 are sufficient along with some special keys to help the user indicate what it wants to do. If the keypad can also be more sustainable in various physical environments, that is a plus as well.

The following metrics will be taken into consideration when picking the ideal keypad: being a matrix keypad, having less buttons to yield flexibility in space constraint, and having the ability to operate in different environments. Weights will be assigned to each metric to reflect its importance, and the highest score will be the component our team will pick. In the table below, the cells will be filled with 1 or 0, 1 being meeting the requirement and 0 not meeting the requirement. Matrix keypad and button count are weighted the same because both are constraints that the group is unwilling to be flexible on, and the flexibility in physical environments has a lower weight because it is more of an extra requirement.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product | Is matrix keypad  (w = 3) | Has less than 4x4 buttons  (w = 3) | Flexible in various physical environments  (w = 2) | Total Score |
| (membrane 3x4)  https://www.adafruit.com/product/419?gQT=1 | 1 | 1 | 1 | 8 |
| (phone style 3x4)  https://www.adafruit.com/product/1824?gQT=1 | 1 | 1 | 0 | 6 |

After comparing the two devices based on our metrics, we ended up choosing the membrane 3x4 keypad. This is because although like the phone style keypad both use a matrix architecture and is smaller than 4x4 (essentially less the A, B, C, and D keys that are found in a 4x4), it is flat and rain proof; if water were to hit the surface of the keypad, it won’t seep through. However, the phone keypad has gaps and water can get beneath the keys and damage the product.

1.1 Analysis of Component 4:

A major part of our design is storing data, so to avoid partitioning issues that could lead to security issues, we decided to have four separate ICs for each user of the usb drive. We initially were looking at EEPROMs, but after seeing the small storage size and learning about the differences between flash and EEPROM chips, we decided to go with flash memory to allow use of paged memory, which allows modification of bigger chunks of data at once. Some of the criteria we looked at while selecting flash ICs included the package they came in, the size of the memory they can store, how easily they can be interfaced with, among other things. We did not want to try using a ball grid array packaged ic, as it would be difficult to solder and prototype, and ideally, we wanted one that was through-hole or surface-mount package. We also wanted somewhere between 1-10 GB of storage per IC, as each user should be able to store a fair amount of data for our device to be useful as a USB drive. Finally, flash ICs often were able to be used by communicating with the IC through SPI, so we would want to find ICs such as that, so they can be interfaced with the microcontroller we have.

This ended up narrowing our search down to two different flash ICs. Notably the MT29F4G01ABAFDWB-IT:F from Micron and the MT29F4G08ABADAH4:D TR from Micron. Both were similar, providing 4 GB of flash memory, both being able to be modified using SPI. While both were very useful for our project, we ended up going with the MT29F4G01ABAFDWB-IT:F, as it had an easier package to solder and prototype with. While it wasn’t a through-hole or simple SMD component, it is easier to handle than the BGA package the other IC came in. The IC we chose came in an 8-UDFN package, which is small and consists of 8 pads on the bottom of the IC that act as the connectors. Also, the other IC also outputted data in parallel, which would be hard for us to handle correctly, especially within the microcontroller.

1.1 Analysis of Component 5:

One major part of making this encrypted USB drive is to make sure that it is secure and so our decision was to add a fingerprint sensor in order to access the content stored within the USB drive. We want a fingerprint sensor that is not too bulky or else it won’t fit well onto the device itself which will be unfavorable practically and aesthetically. Also we also wanted it to be a relatively flat sensor so that it can lay on the device comfortably and since our goal is for this device to be carried around in the customer’s pockets or bags it should be easy to store so having protruding parts would not be ideal. Fingerprint sensors come in 4 types: optical, capacitive, ultrasonic, and thermal. We went for the capacitive or optical sensors because they are currently the most reliable on the market. Thermal sensors wouldn’t perform well under variable temperatures and ultrasonic sensors came out relatively recently increasing the risk. Another factor would be the price/ease of obtainability.

The metrics and criteria mentioned above will be used to determine which fingerprint sensor we ultimately decide to use. Weights will be assigned to each metric, and the highest score will be the component our team will pick. In the table below, the sensors will be ranked from 0 to 3, 0 being least ideal, 3 being most ideal.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fingerprint Sensor Product | Flat shaped  (w = 3) | Capacitive/optical sensor  (w = 1) | Ease of obtainability  (w = 2) | Total Score |
| GROW R308 | 0 | 2 | 3 | 8 |
| UART Fingerprint Reader | 0 | 2 | 1 | 3 |
| GROW R502-A | 3 | 2 | 2 | 15 |

2.0 Sources Cited:

Adafruit Industries, "Adafruit 16x2 Character LCD + Keypad for Raspberry Pi," [Online]. Available: <https://www.adafruit.com/product/1824>. [Accessed: 08-Feb-2025].

Adafruit Industries, "Membrane 3x4 Matrix Keypad + extras - 3x4," [Online]. Available: https://www.adafruit.com/product/419. [Accessed: 08-Feb-2025].

^ prices found in each link

HD44780

UNIVERSAL-SOLDER Electronics Ltd, "26076," [Online]. Available: <https://www.digikey.com/en/products/detail/universal-solder-electronics-ltd/26076/16821383>. [Accessed: 08-Feb-2025].

WEHLCD

Winstar Display Co., Ltd., "WEH001602A - 16x2 COB OLED Character Display," [Online]. Available: <https://www.winstar.com.tw/products/oled-module/oled-character-display/weh001602a.html>. [Accessed: 08-Feb-2025].

SPI LCD mouser

Display Visions, "EA TFT024-23ATNN," [Online]. Available: <https://www.mouser.com/ProductDetail/Display-Visions/EA-TFT024-23ATNN?qs=IKkN%2F947nfCa9Z2pp%252BAi%2FA%3D%3D>.

[Accessed: 08-Feb-2025].

“STM32 32-bit Arm Cortex MCUs - STMicroelectronics,” *STMicroelectronics*. <https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html>

‌Digikey, “MT29F4G01ABAFDWB-IT:F,” *Digikey*. <https://www.digikey.com/en/products/detail/micron-technology-inc/MT29F4G01ABAFDWB-IT-F/8509729> (accessed Feb. 08, 2025).

‌Digikey, “MT29F4G08ABADAH4:D TR.” <https://www.digikey.com/en/products/detail/micron-technology-inc/MT29F4G08ABADAH4-D-TR/4315127> (accessed Feb. 08, 2025).

Waveshare, “UART Fingerprint Reader,” Waveshare, [Online]. Available: https://www.waveshare.com/uart-fingerprint-reader.htm. [Accessed: Feb. 8, 2025].

Hubtronics, “GROW R308 Fingerprint Module,” Hubtronics, [Online]. Available: https://hubtronics.in/grow-r308-fingerprint-module. [Accessed: Feb. 8, 2025].

Hubtronics, “GROW R502 A Fingerprint Module,” Hubtronics, [Online]. Available: https://hubtronics.in/grow-r502-a-fingerprint-module. [Accessed: Feb. 8, 2025].