# Smart Portable Key

## Design Review

## ECE 445

**TA: Igor Fedorov**


**Aashay Shah**
**Akshay Chanana**

**Table of Contents**

# 1.0 Introduction

## 1.1 Statement of Purpose

This project was chosen because there are no devices presently available in the market that allows the user to wirelessly access multiple locks used for general purposes. There is a high demand for secure portable locks and this project aims to fulfill that need. The main focus will be to provide a portable and secure lock system with a smart key that will be used to unlock multiple locks wirelessly.

## 1.2 Objectives

### 1.2.1 Goals:

- Develop a portable secure key that can unlock multiple locks
- Secure the key with a fingerprint scanner and enable encryption so as to prevent physical hacking by just sending a high signal
- Allow the smart key to access locks wirelessly
- Develop multiple locks with electric and mechanical components

### 1.2.2 Functions:

- Panel of switches to choose between different locks
- Wireless communication between transmitter and receiver
- Fingerprint scanner to validate smart key
- Microcontroller to validate encryption

### 1.2.3 Benefits:

- Instantaneous access to any of the locks
- Saves the hassle of carrying multiple keys or keys at all
- Physical hacking of the device not possible
- Portable and secure locking system

### 1.2.4 Features:

- Easy to use and carry around
- RFID communication modules for transmitter and receiver
- Reliable, secure and hack proof
- Long system life as it basically runs on batteries
- Cheap and fast access to the locked system

# 2.0 Design

## 2.1 Block Diagrams

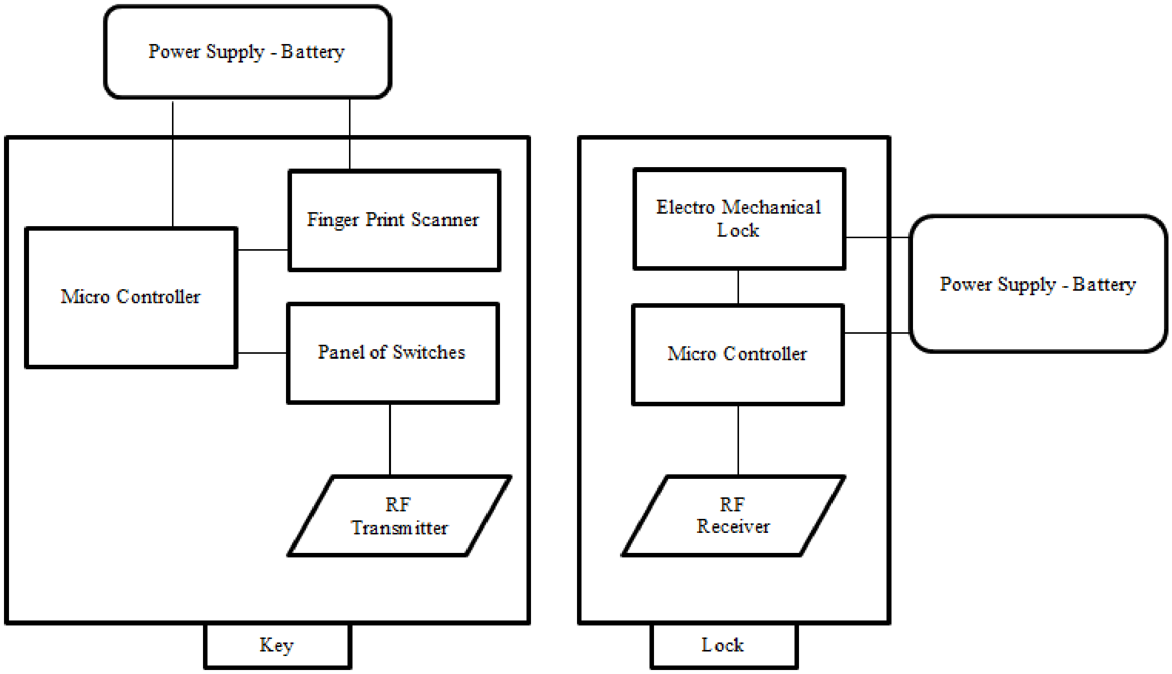


**Figure 1: Detailed Block Diagram of the key and lock system to be implemented**

## 2.2 Block Description

The lock and key system shown above in Figure 1 will be implemented in this project with its components described below.

*Power Supply (Key):*
This will be responsible for powering the entire circuit on the key side of the design which includes the fingerprint scanner, microcontroller and the RF transmitter. We are planning on using a 9V Energizer battery with the MIC5219 voltage regulator to provide an overall constant 3.3V source.

*Power Supply (Lock):*
This will be just like the supply on the key side. It will be used to power the controller and the RF receiver. The microcontroller should be able to send a signal to the electromechanical component which will then implement the unlocking mechanism. The battery can be sufficiently bulky in contrast to the power supply on the key as it will not be carried around. Exact details are being worked out with the Machine Shop professionals. We will be using the same 9V Energizer battery as being used in the key.

*Fingerprint Scanner:*

This is the security measure used on the design of the key module. It is used to record and verify multiple fingerprints from users. Upon validation of the correct user it will send data serially over to the microcontroller connected to it. The controller will then check this data and further activate the panel of switches for further unlocking the locks.

We are using the 3.3V LEM 100 scanner from Integrated Biometrics. The module can perform storing, identification and deletion of fingerprints using one of the best algorithms one can find in the market today. It is quite a compact module and is very easy to integrate in our system. It comes with an in-built memory system as shown in the Block diagram for LEM 100 in Figure 2. The scanner comes with a main board and a sensor board as shown in Figure 3.

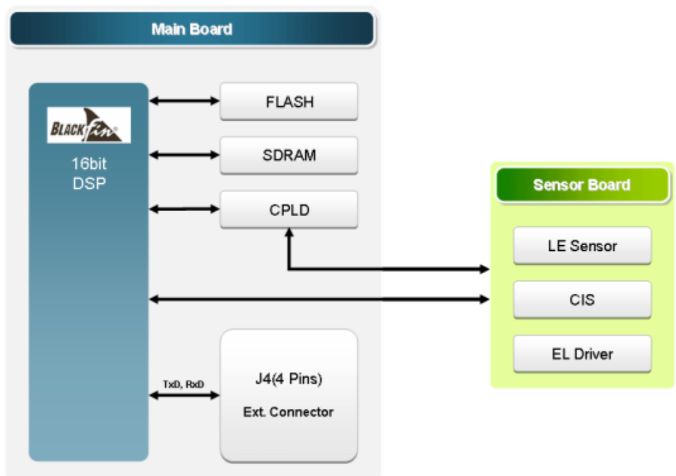


**Figure 2: LEM 100 Block Diagram [2]**

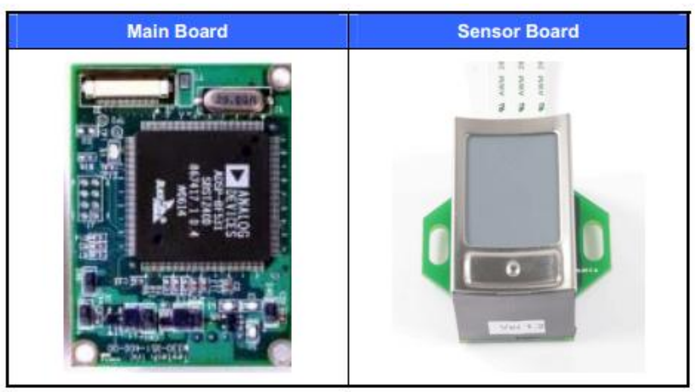


**Figure 3: LEM 100 Module Board [2]**

The sensor board is connected to the main board through the J2 connector. We will be mainly using the J3 4-pin connector to interact between the controller and scanner. J2 basically contains pins for: VCC, GND, TXD, RXD as shown in Figure 4.

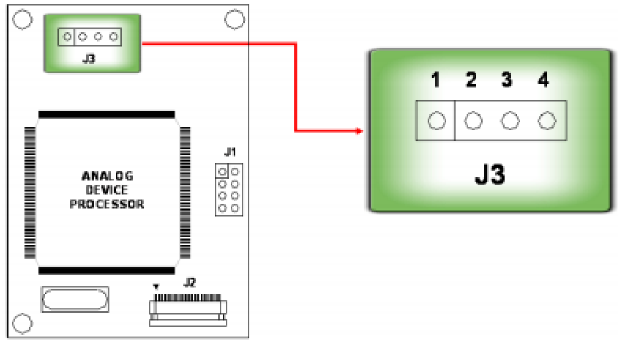


**Figure 4: J3 connector LEM 100 Module Board [2]**

The TXD will be used to transmit the data from the scanner to the controller and the RXD will be used to receive data from the controller. The communication protocol for the LEM 100 is described below in Figure 5. The packet data has a start of packet byte followed by 2 bytes of which command, some reserved bytes, error code and finally 1 byte to signify end of packet. Some of the basic commands and a summary of commands are also shown in Figure 6.

**- Packet Data**

| ① | ② | ③ | ④ | ⑤ | ④ | ⑥ | ⑦ | ⑧ |
|---|---|---|---|---|---|---|---|---|
| STX | Command | Address | , | NG | , | Error Code | CS | ETX |

① Equals to 0xF1 and means beginning of transmission packet. (1 byte)

② Property of code for a specific execution. (2 byte)

③ Code for a specific execution. (2 byte)

④ Comma ( , ) differentiate Command/Address from its parameter division. (1 byte)

⑤ Parameter applied by transmission Command/Address policy.

⑥ Value verifies integrity of data. (2 byte)

⑦ Equals to 0xF2 and means ending of transmission packet. (1 byte)

**Figure 5: LEM100 Communication protocol [2]**

**- Command**

| Item | Binary | Char | Comment |
|---|---|---|---|
| MD_READ | 0x00 | '00' | Read a value from Module |
| MD_WRITE | 0x01 | '01' | Save a value to Module |
| MD_EXEC | 0x02 | '02' | Execute an instruction |

**- Command Summary**

| Item | Binary | Char | Comment |
|---|---|---|---|
| MD_START_CAPTURE | 0x03 | '03' | Command for starting capture |
| MD_ENROLL_FP | 0x04 | '04' | Command for registering User |
| MD_DELETE_FP | 0x05 | '05' | Command for deleting user |
| MD_DELETE_ALLFP | 0x06 | '06' | Command for deleting all users |
| MD_IDENTIFY_USER | 0x08 | '08' | Command for identifying a fingerprint (1 : N) |
| MD_VERIFY_USER | 0x09 | '09' | Command for verifying a fingerprint (1 : 1) |
| MD_IDENTIFY_USER_EX | 0x19 | '19' | Command for identifying a fingerprint( 1 : N ) |
| MD_VERIFY_USER_EX | 0x1A | '1A' | Command for verifying a fingerprint( 1 : 1 ) |

**Figure 6: LEM100 Communication commands [2]**

*Panel of Switches:*

These will consist of three switches that are directly used for unlocking the same respective number of locks. This panel will get activated by the microcontroller after successful verification by the fingerprint scanner. Once activated, this panel will allow the user the option of unlocking any of the three switches. We plan on having button switches for each of the locks, which on pressing will tell the controller to transmit a RF signal with some encryption to the receive side on the lock. These three switches will be connected to the controller digital ports and whenever the controller witnesses a high signal, it will send the signal to the required lock using the RF transmitter.

*Microcontroller:*

The microcontroller used in the key module will be the Texas Instruments MSP430. This will be the main control unit of the module. The microcontroller will be powered by the power supply (key) and is a good option as it has low power consumption as well as a low cost. It will process the data from the fingerprint scanner to validate that it has read the right fingerprint and only then check for any activated switches on the panel after which an RF signal will be transmitted to the particular lock needed unlocking. Another microcontroller will also be on the receive side to process the incoming signal at the receiver and then initiating the unlocking mechanism.

The controller being the main control unit in our design basically has to send and receive data, to and from, respectively, all other devices. The basic flow of information is captured in the flowcharts shown in **Section 2.2**.

*RF Transceiver:*

The key will have a RF transmitter and the lock will have a RF receiver to implement the wireless communication between the two components of the design. Thus making use of RF will also let us unlock the device from a distance. This communication module will be able to relay data to and from the microcontroller on both, transmit and receive side.

The RF transceiver we will be using is the XBee 1mW Trace Antenna - Series 1 (802.15.4) that works on the Zigbee protocol. Once the button is pressed on the panel of switches, the RF transmitter will send the corresponding frequency. The frequency would be the one matched with the lock the user wants to open. For the RF receiver that is placed on the locks, the microcontroller would tell it to activate only when it receives its frequency and not any other.

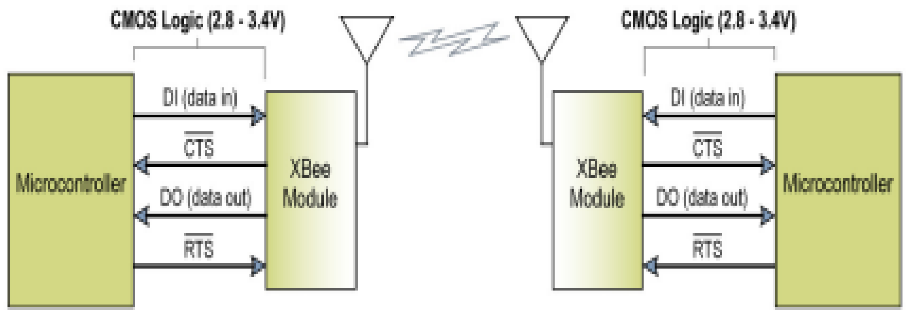


**Figure 5: Microcontroller interface with XBee Transceiver [1]**

An asynchronous signal is introduced in the module through the Data In (DI) pin and is idle high when no signal is transmitted through it. The data transmits as 8 bits where the first bit is the Start/Stop bit. When the signal is low, the data starts transmitting into pin DI and when it becomes high it stops. The process is shown in Figure 6.

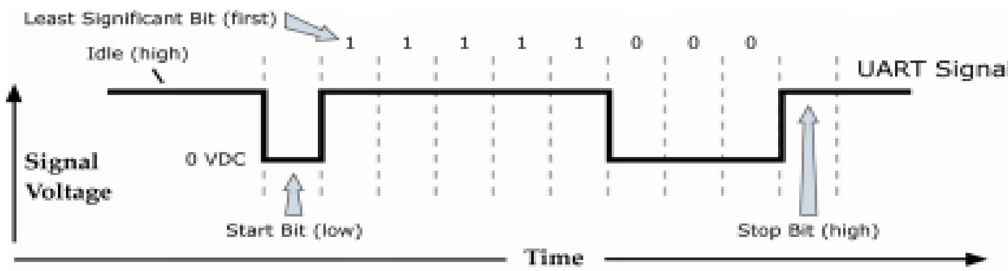


**Figure 6: UART data packet as transmitted through RF module [1]**

During the time that the RF transceiver doesn't operate, it works in the idle mode. Otherwise it supports four different modes of operation as shown in the Figure 7 below. The modes that would be used in this project are the - transmit mode, receive mode and idle mode.

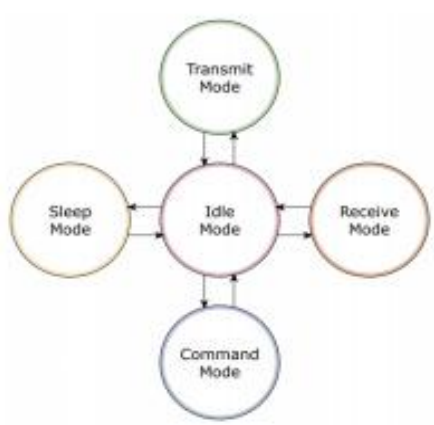


**Figure 7: Modes of operation for the RF Transceiver [1]**

*Electromechanical Lock:*

The electromechanical component on the lock will be the module implemented to click it open on receiving the correct signal from the key. The controller will send a signal to the electric part which will in turn transform to a mechanical procedure that will help to unlock the device.

***Key Process Flowchart***

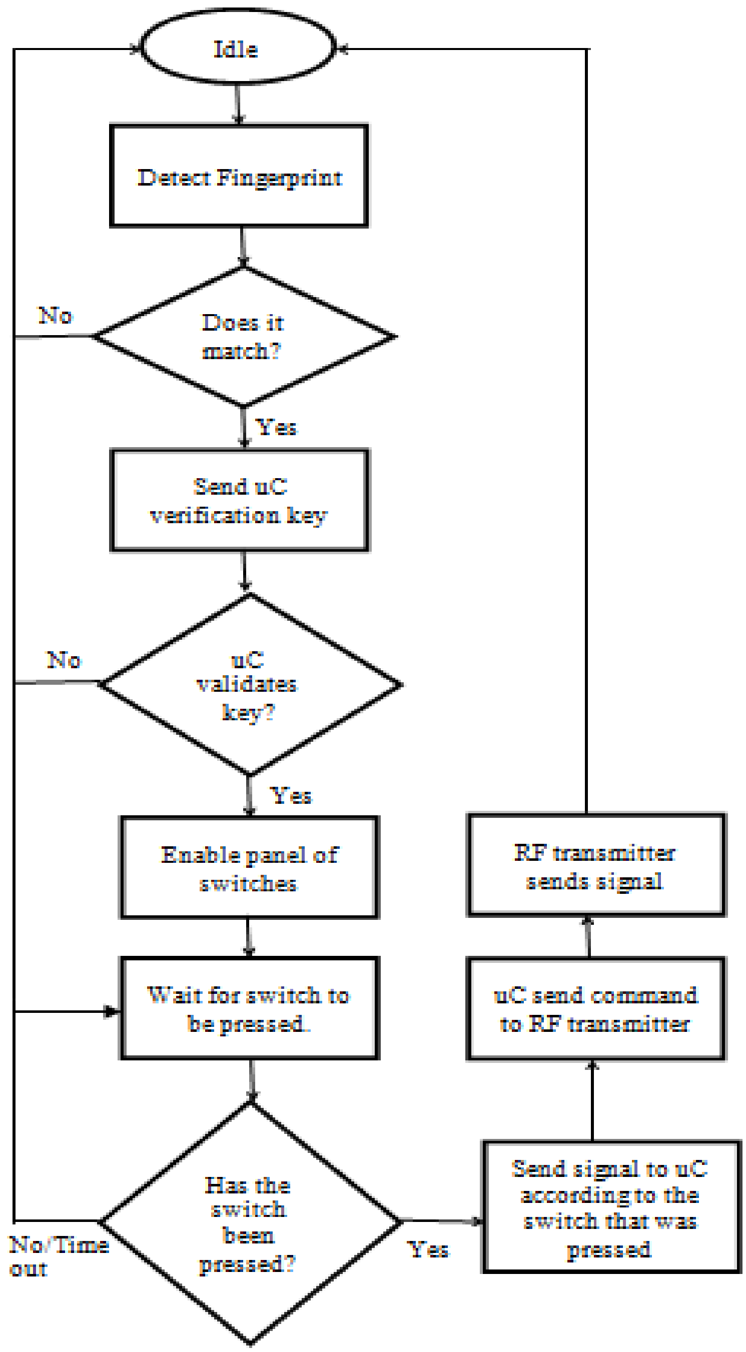


**Figure 8: Detailed flow of the key system**

*Lock Process Flowchart*

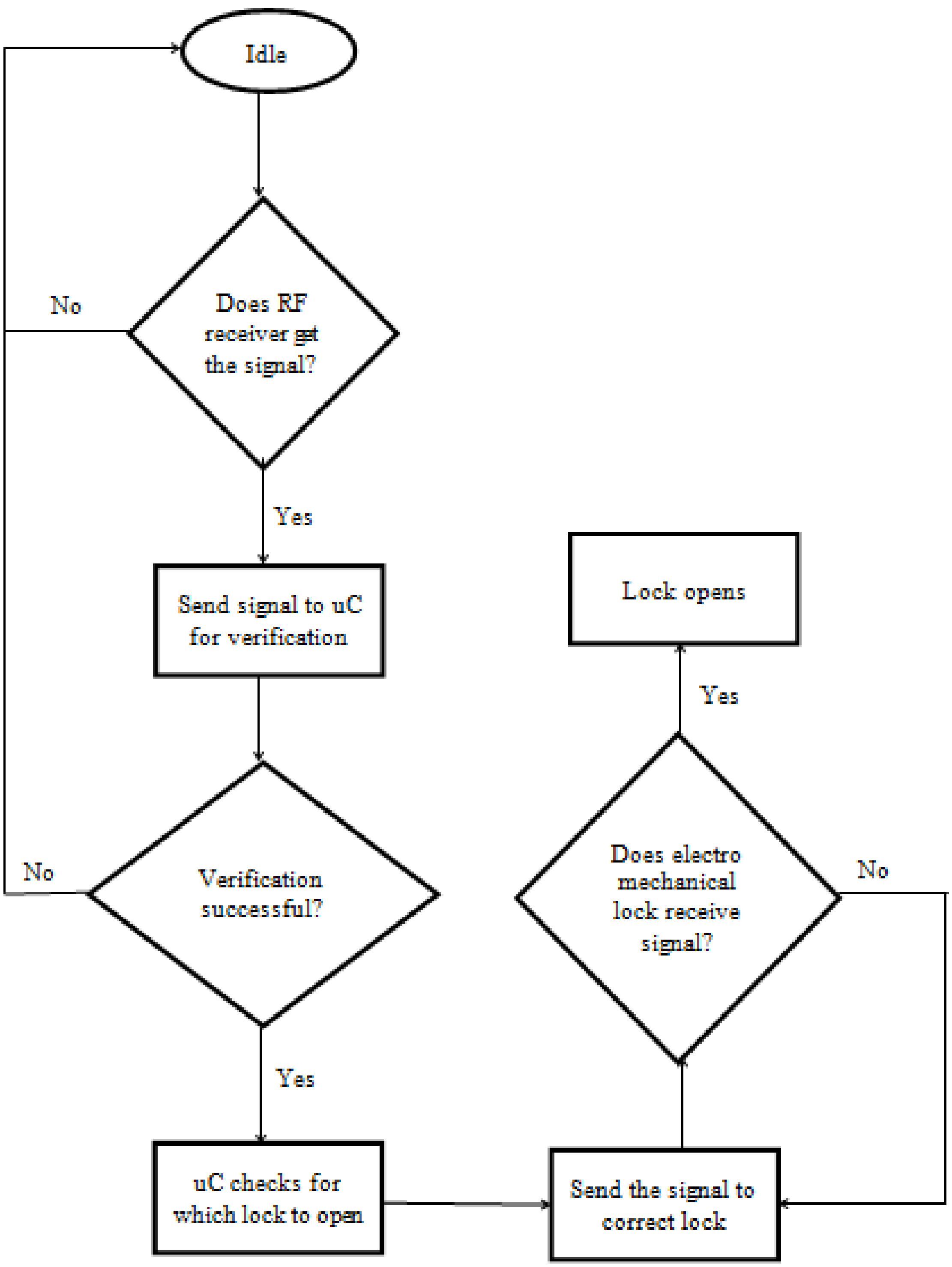


**Figure 9: Detailed flow of the lock system**

## 2.3 Schematics of system

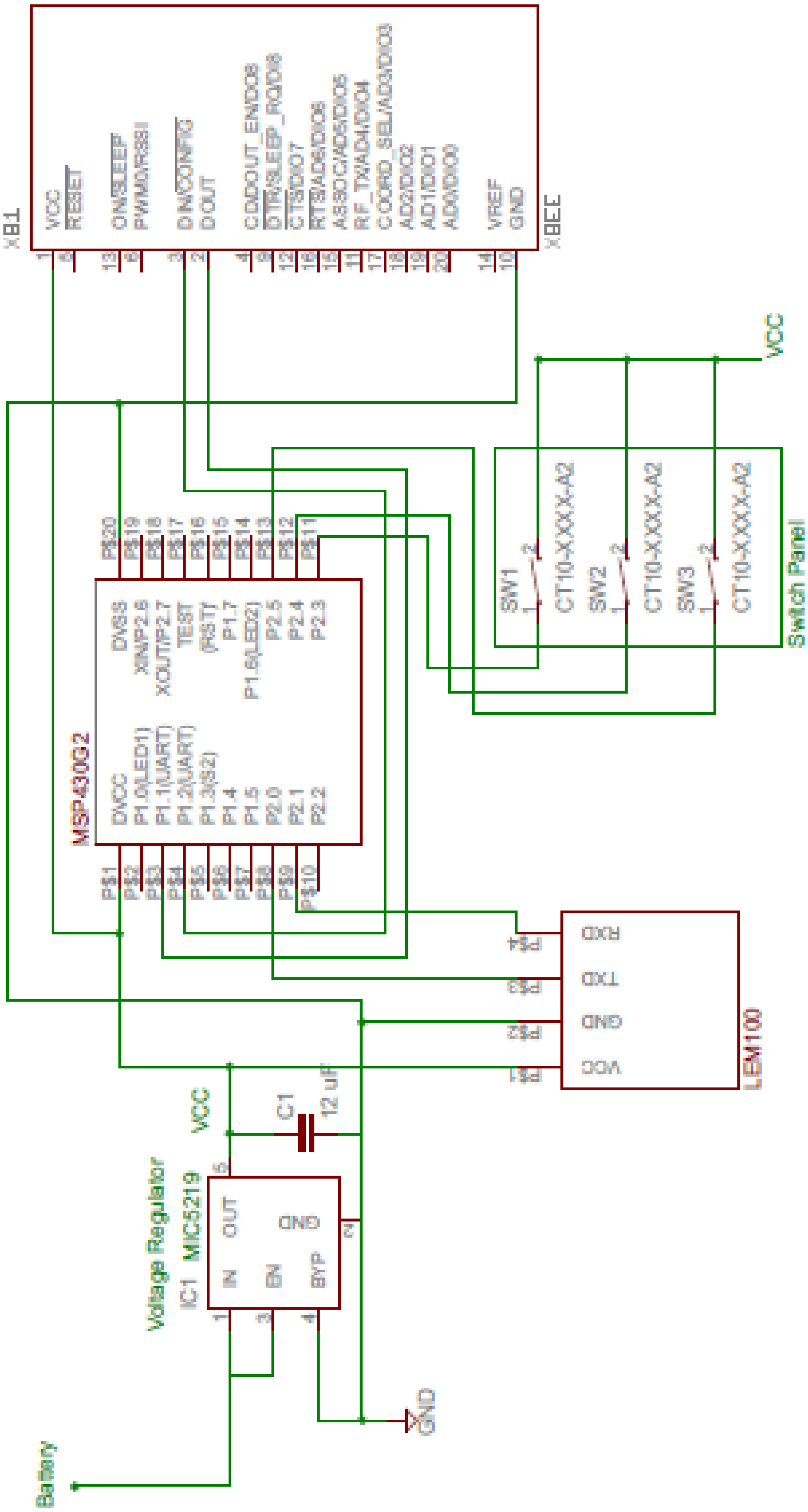

**Figure 10: Schematic of the key system to be implemented**

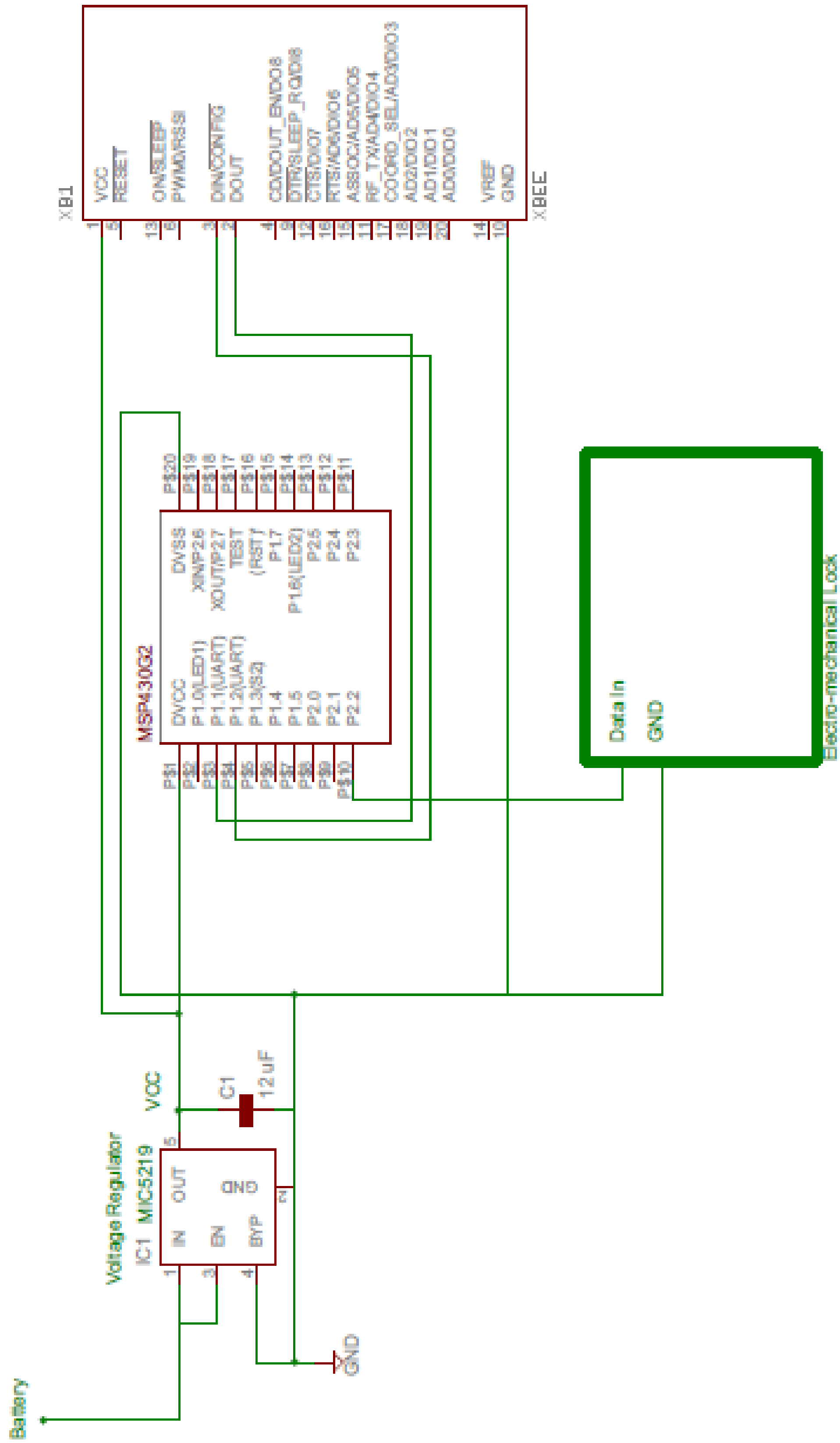


**Figure 11: Schematic of the lock system to be implemented**

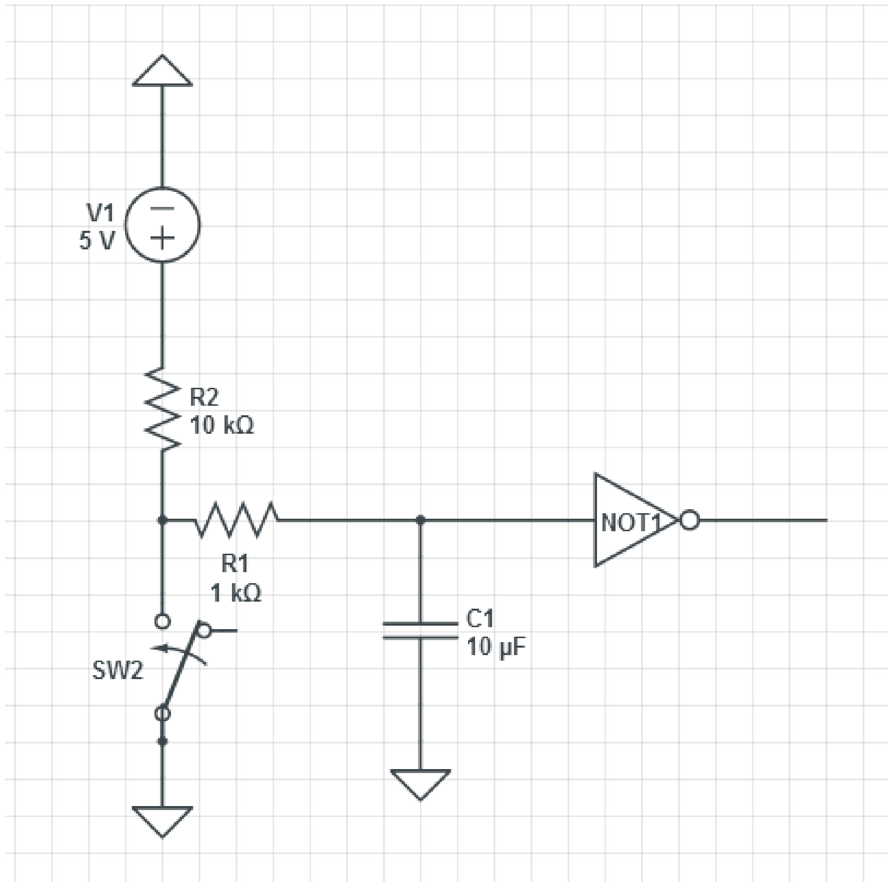


**Figure 12 : Debouncing Circuit**

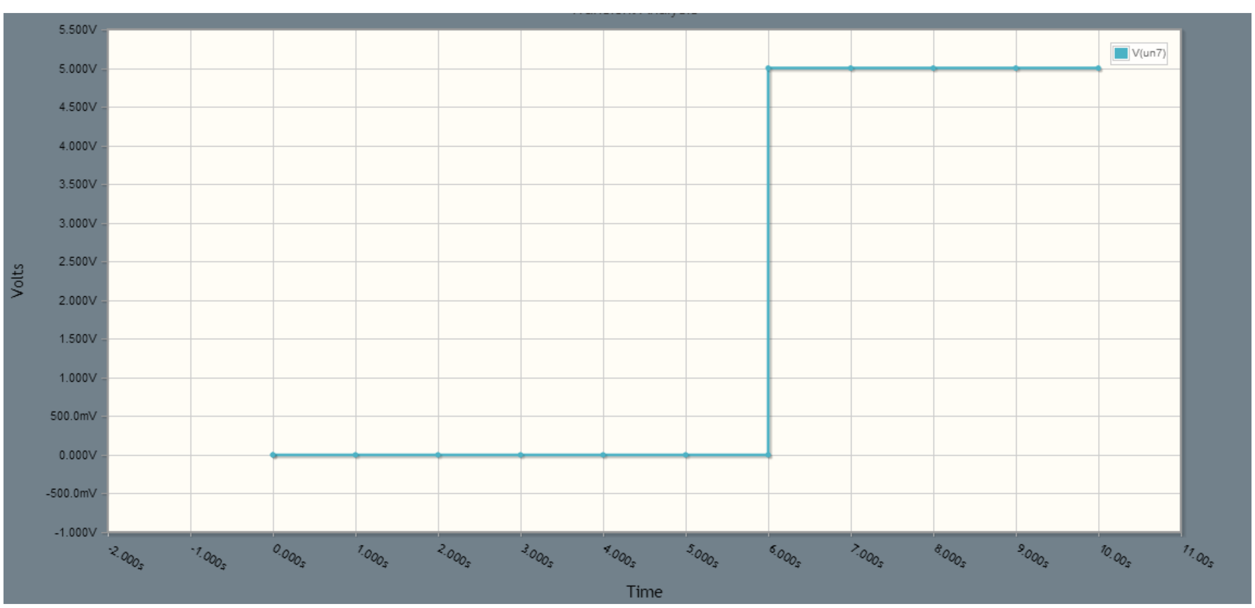


**Figure 13: Simulated Circuit with Debouncer(Output)**

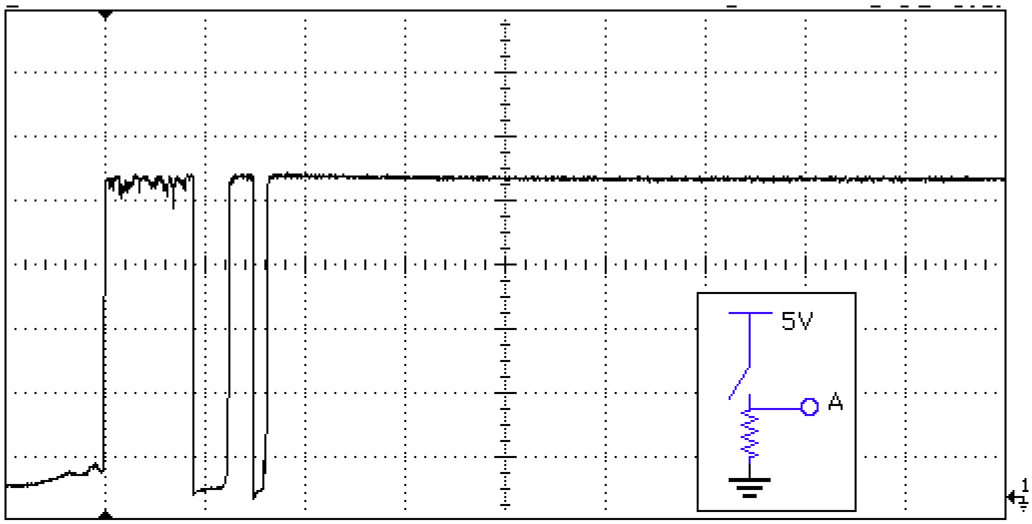


**Figure 14: Simulation Without Debounce Circuit(Output) [6]**

### 2.4 Calculations

XBee RF Transceiver (1mW) [1]: TX peak Current 45 mA @ 3.3 V
RX Current 50 mA @ 3.3 V
Power down Current < 10 uA
Max Power needed = 3.3*0.05 = 0.165 W

LEM100 Fingerprint Scanner [2]: Typical voltage = 3.3V, maximum voltage = 3.8 V
Supply current - Idle state = 118 mA
Enrollment state = 178 mA
Identification state = 170 mA
Deletion state = 178 mA
UART baud rate = 9600 bps
Max Power needed = 3.8*0.178= 0.6764 W

TI MSP430G2 Microcontroller [5]: Max current = 60 mA
Voltage = between 1.8 V and 3.6 V
Max Power needed = 0.06*3.6 = 0.216 W

Max Power Used by any component = 0.6764 W
Total Power Used = 0.216 + 0.6764 + 0.165 = 1.0574 W
Total Power that we have = 9 * 0.178 = 1.6 W
This shows that the power that is less than the power supplied. In the above calculation a 9V battery is used.

## 3.0 Requirements & Verification

### 3.1 Requirements

| Module | Requirements | Verification Procedure |
|---|---|---|
| 1. Fingerprint Scanner | 1.1. Scanner management system should be able to record, detect existing and delete stored fingerprints. | 1.1 The scanner will be tested using the J3 UART connection on the main board. The scanner will interface with the controller using this connection. |
| | a. Communication interfaces should be initially properly working | a. Check to ensure connection between module and device are well connected.<br>• Check input power to module ~ 3.3 V<br>• Check LED blinking of module after turning on the module<br>• Check serial communication interface LED of module is blinking. |
| | b. Serial communication speed should be set optimally for synchronization between scanner and controller. | b. Send one kB of data and check the rate of transfer. |
| | c. Store new users that will be able to validate access to the key | c. To start scanning MD_START_CAPTURE is sent to the module. Multiple users (~5-10) will be added by using the MD_ENROLL_FP command and then checking for these registered users in the in-built memory as shown in Figure 2. |
| | d. Unauthorized users should be denied access | d. Users will be identified using the MD_IDENTIFY_USER command and verified by sending MD_VERIFY_USER. |

| | | |
|---|---|---|
| | e. Existing users should be removed on request<br><br>f. On a successful scan scanner should be able to send a valid data key to controller for verification | e. Users will be deleted using the MD_DELETE_FP command and checking if the digital output associated with it receives a high signal.<br>f. The scanner will send the signal to the controller via the UART interface. This will be done about 10-20 times, to check if the controller validates the data sent by the scanner, using the LED on the controller to signal for valid data received. |
| 2. Power Supply (Battery and Voltage Regulator) | 2.1 Check if voltage regulator gives the correct output that is needed for the fingerprint scanner.<br><br>2.2 Supply rated voltage and current.<br>a. 3.3 ± 0.5 V<br><br>b. Current doesn't exceed 118mA | 2.1 To test the voltage regulator (MIC5219) a small circuit would be made and different resistor values would be used to see if the given output is the required one (3.3± 0.5 V). These results would be checked using an oscilloscope. This voltage regulator would be used for the microcontroller and the fingerprint scanner.<br><br>2.2<br>a. Using an oscilloscope, the voltage will be tested on the power supply. This is very important to check every time as the fingerprint scanner cannot take voltage above 3.6V or less than 2.7 V.<br><br>b. This precaution is very important because the fingerprint scanner is really expensive. A fuse will be used here so that it prevents it from |

| | | |
|---|---|---|
| | | getting damaged. Also, a diode will be kept such that the scanner doesn’t get damaged in case the battery is reversed. |
| 3. Microcontroller (Key) | 3.1 Interface with the fingerprint scanner works with synchronization<br>a. Serial communication speed set at 9600 bps at all times<br>b. Verification key received on a successful fingerprint scan | 3.1 The scanner will be connected to the controller serially.<br>a. Controller will be set to a default serial communication speed of 9600 bps using the Code Composer software. It will be verified as described in 1.1 (b).<br>b. A test program will be written to ensure that the correct verification key is sent. This will be tested 20 times just to see if it works correctly. When the correct key is received an LED will blink. |
| | 3.2 Detects the change of state of any of the switches on the switch panel<br>a. Panel should be debounced | 3.2 Panel switch be connected and tested through the digital ports of the controller<br>a. We will test the switch interface by testing the pins they are connected to on the controller. Every time we switch one on, we will have a LED signal that verifies this interface. |
| | 3.3 Should be able to send a correct signal via the XBee module to the respective lock after verification from scanner completes and one of the switches is on. | 3.3 The XBee DIN pin will receive information from the controller necessary to signal the respective lock<br>a. This waiting period will be done by programming the |

| | | |
|---|---|---|
| | a. Wait for the user to pick which lock to be open<br><br>b. Once a lock is picked, sent signal through the XBee using the UART interface should be to the right lock | controller with a wait function till one of the pins connected to the controller sees a high signal. We will also incorporate a time out so as to not keep it always on once the scanner is successful. Time out will be checked by leaving the system alone and checking if one of the LEDs starts to blink.<br>b. After receiving the signal from one of the switches, send packet to XBee controller based on the XBee datasheet so as to transfer to the right lock. Tested fifty times and will check for right lock by using the controller on the receive side. We will store the message on the controller and read its memory using software for validation of the message received.<br><br>These tests will ensure that the controller works well with each component on the key. |
| 4. Microcontroller (Lock) | 4.1 Receives a valid signal from the XBee receiver and then signals the electro-mechanical lock | 4.1 The RF receiver will have to behave the same as the transmitter interfaces with the key controller. This is tested the same way as described in section 3.3.<br><br>Once the signal is validated, we can output a high initially to test an LED so as to know that the signal passes the verification and the lock can now be unlocked. Multiple tests will be done to check the accuracy of this unlocking mechanism. |
| 5 Panel of | 5.1 The switches need to be | 5.1 This will be done using a simple |

| | | | |
|---|---|---|---|
| | Switches | debounced such that it resets after 1 second.<br><br>5.2 Ensure that the correct lock receives signal when the button is pressed. | debouncing circuit and will be checked using an oscilloscope by pressing the button thirty or so times. The oscilloscope should show that the button was hit thirty times with perfect accuracy. Also, the oscilloscope will give a high output when the button is pressed and it resets after a particular time so that the user can open another lock by scanning the fingerprint and activating the panel of switches.<br><br>5.2 When the button is pressed three LED's will be put near the receiver and whenever the corresponding button is pressed the correct LED should light up. |
| 6 | RF Transceiver | 6.1 Serial connection works perfectly between XBee and the controller<br><br>6.2 To ensure that it activates when the correct frequency is received.<br><br>6.3 Range test. Should be able to | 6.1 This will be verified by connecting the transmitting side and receiving side to different computers (using UART) and checking by sending 2kB data and then receiving it to ensure it works 90% of the time. It will be tested 30 times and error calculation will be done.<br><br>6.2 To ensure this, a test signal will be received. LED's will be placed such that when the correct signal is received, they would turn on and when an incorrect signal is sent it stays off. This whole process will be handled by the controller and signals will be sent manually to check on its verification.<br><br>6.3 The range would be checked by |

| | | |
|---|---|---|
| | communicate up to a 30m distance between the transmitter and receiver.<br><br>6.4 All interfaces work at a communication speed of 9600 bps. The transceivers should do the same. | sending the signal in different situations where the distance between the transmitted and received side vary. We will send sample test data using the controller and we should be able to receive data with 100% integrity up to a distance of 30m.<br><br>6.4 This will be done the same way as described in Section 1.1 (b) by connecting the controller and XBee together to exchange data at the defined rate. |
| 7 Lock | 7.1 Ensure enough power is there so that the lock opens when it has to.<br><br>7.2 It snaps the lock when signal is received. | 7.1 Using an oscilloscope the amount of power received by the lock would be checked. Multiple tests will be done to check for how long the battery will last.<br><br>7.2 Different voltages from the range of 5V-20V would be given to the lock and would be checked for which ones it opens smoothly. |

### 3.2 Tolerance Analysis

The tolerance analysis will be based on the sustainability of our communication module i.e. the RF transceiver. This is the most important part of our design. If this communication doesn’t work, we will not be able to send any kind of signal to the unlocking mechanism. In order to test this system we will send multiple signals from the transmit side to the receive side and test for connectivity, data validation and transfer speed at different range of distances between transmit and receive side. We want an accuracy of 100% for data transfer and also a fast transfer rate so as to not have a huge delay between the pressing of the key and actually getting the lock unlocked.

## 4.0 Ethical Issues

The purpose of this project is to develop a safe portable key and lock system that can be used in daily life. We are well aware of Code of Ethics published by the IEEE to which all EEs must adhere. We will strictly follow these rules and guidelines provided to us in the development of our project and will not violate any of them. The table below summarizes the IEEE code of ethics relevant to us.

| IEEE Code of Ethics [3] | Relevance in Design |
|---|---|
| "1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;" | The purpose of the project is to make a secure lock and key system such that the user can keep their important documents safe. This feature of the device helps the user feel safe about their belongings. |
| "3. to be honest and realistic in stating claims or estimates based on available data" | Only the authorized user should be able to open the lock and hence the accuracy of the device is important such that no one else tries to open the lock. While doing the project every data taken will be reported honestly even if it isn't used. |
| "5. to improve the understanding of technology; its appropriate application, and potential consequences" | Once the project is done, the knowledge about MSP430G2, UART, LEM 100 scanner and the XBee RF transceiver would be gained. This knowledge would help understand these devices and would also help gain technical competence. |
| "6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;" | |
| "9. to avoid injuring others, their property, reputation, or employment by false or malicious action;" | It should be made sure that the authorized user is the only one allowed to use the key and no one else would be able to. This would help avoid any kind of theft or loss of property. |
| "10. to assist colleagues and co-workers in their professional development and to support them in following this code of ethics." | We will readily provide assistance to colleagues with their professional development and support them in their code of ethics. |

## 5.0 Safety Analysis

There are very limited safety concerns while doing this project. This project requires relatively less power and it should not be an issue when testing or lab work is being done.

## 6.0 Cost & Schedule

### 6.1 Cost Analysis

#### 6.1.1 Labor

| Group Member | $/Hour | Hours/Week | Number of Weeks | Multiplier | Total/Person |
|---|---|---|---|---|---|
| Aashay | 30 | 15 | 12 | 2.5 | $13500 |
| Akshay | 30 | 15 | 12 | 2.5 | $13500 |

Total Labor Cost = $27000

#### 6.1.2 Parts

| Part | Manufacturer | Price/Unit | Quantity | Total | Part Status |
|---|---|---|---|---|---|
| MSP430 Microcontroller | TI-430G2 | $5.89 | 4 | $23.56 | Received 1 |
| Fingerprint Scanner | Integrated Biometrics-LEM100 | $130 | 1 | $130 | Ordered |
| Electromechanical Lock | Machine Shop | $125 | 3 | $375 | To be ordered |
| XBee RF Transmitter/Receiver | Digi- 1mW Trace Antenna - Series 1 (802.15.4) | $20 | 4 | $80 | Received |
| MIC5219 Voltage Regulator | Micrel | 1.43 | 2 | 2.86 | To be ordered |
| AA Battery Pack | Energizer | $2.95 | 2 | $5.9 | Ordered |
| 3V Lithium Button Cell Pack | Energizer | $2.00 | 1 | $2.00 | Ordered |
| PCB and circuit elements | Machine Shop | $25 | 1 | $25 | To be ordered |

Total Parts Cost = $644.32

#### 6.1.3 Grand Total

| | |
|---|---|
| Labor Cost | $27000 |
| Parts Cost | $644.32 |
| **Total Cost** | **$27644.32** |

### 6.2 Detailed Schedule

| Week | Akshay | Aashay |
|---|---|---|
| 2/4 | Proposal and figuring out the parts that would be needed | Proposal and figure out the power supply components. |
| 2/11 | Look up how the fingerprint scanner would work with the microcontroller and the RF transmitter. | Figure out the design and the working of the lock and the panel of switches. Order parts for the project. |
| 2/18 | Schematics, ethical issues, block descriptions and tolerance analysis for the design review | Test verifications table, parts table and flowchart design review. |
| 2/25 | Design the working of the microcontroller and the finger print scanner and make sure it takes the correct fingerprint. | Go to the machine shop and explain what kind of locks and the panel of switches would be needed. |
| 3/4 | Connect the microcontroller to the panel of switches and make sure it works when the correct finger print is inputted. | Test the voltage regulator for the power supply. |
| 3/11 | Send the signal from the RF transmitter to the receiver and make sure it works when correct signal is given. | Research on how the panel of switches would send the correct signal to the RF transmitter to transmit to the receiver. |
| 3/18 | Spring Break | Spring Break |
| 3/25 | Design the PCB layout and make sure it matches the schematic. | Make sure that the RF transmitter transmits the correct signal and the receiver receives it. |
| 4/1 | Adjust the PCB after what was found once debugging was complete. | Test for the locks to see what voltage is needed to unlock and that the correct lock opens when the signal is sent. |
| 4/8 | Confirm that the correct lock receives the signal once the panel switch is activated. | Implement the RF receiver and the lock opening system for all the locks. |
| 4/15 | Connect everything together such that everything is working is sync. | Connect everything together such that everything is working is sync. |
| 4/22 | Demo week. | Demo week. |
| 4/29 | Final Presentation and Final Paper | Final Presentation and Final Paper |

## 7.0 References


[1]
"XBee®/XBee-PRO® RF Modules - 802.15.4 - v1.xEx [2009.09.23]."*Http://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Datasheet.pdf*. Digi International Inc, 23 Sept. 2009. Web

[2]
"LEM100 Hardware Development Manual."*Https://integratedbiometrics.zendesk.com/attachments/token/d0btaf2e6bsrvwt/?name=LEM100+Module+Hardware+Development+Manual+_En__Rev1.4.pdf*. Integrated Biometrics, n.d. Web.

[3]
"IEEE Code of Ethics." Available: http://www.ieee.org/about/corporate/governance/p7-8.html. Web.

[4]
"TI MSP430G2 Data Sheet." Available: http://www.ti.com/lit/ug/slau318c/slau318c.pdf. Web.

[5]
"MSP430 Hardware tools Users Guide." http://www.ti.com/lit/ug/slau278l/slau278l.pdf. Web.

[6]
"A non-debounced Circuit"http://www.utdallas.edu/~poras/courses/ee3320/xilinx/upenn/lab7-LabDigitalClock.htm Web