

---

---

# Lock and Key Group

**Final Presentation**

— Alan Ngo —

John Vetus

Nicolas Lowman

---

---

# Project Overview

Problem: Incorrectly slotted keys and foreign metal objects may cause shorts and misreads. In the current design when one slot shorts the entire drawer becomes unusable. This is a significant issue that makes up approximately one third of their daily support calls.



# Project Overview

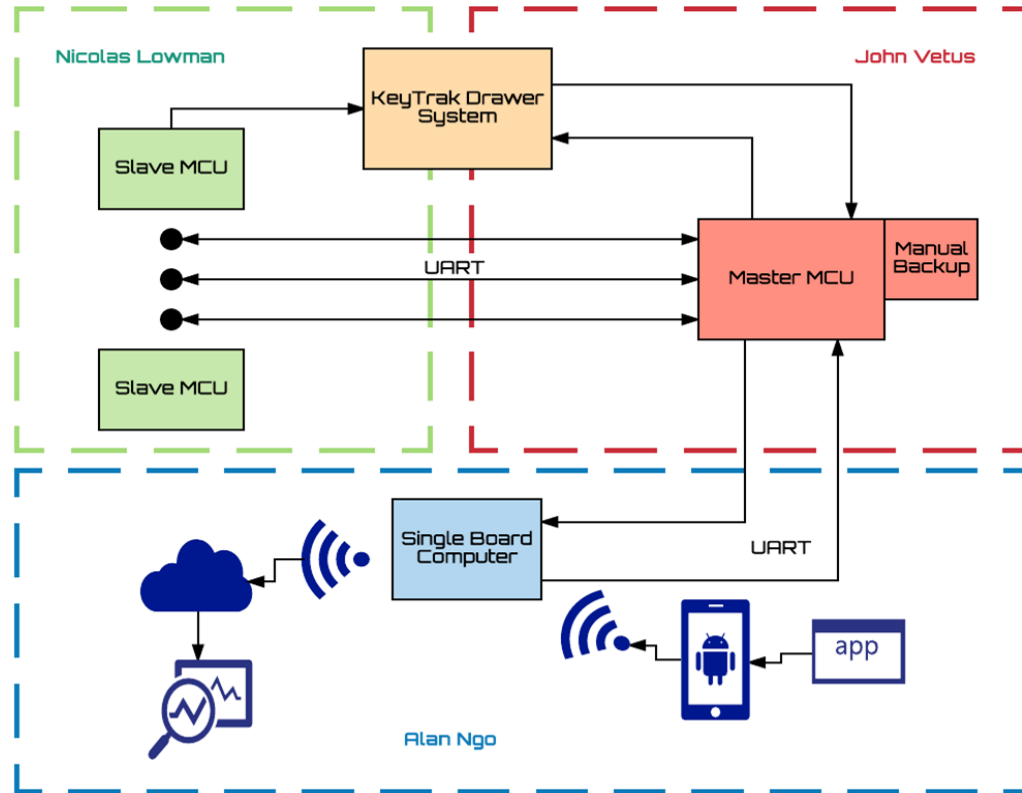
Problem: Incorrectly slotted keys and foreign metal objects may cause shorts and misreads. In the current design when one slot shorts the entire drawer becomes unusable. This is a significant issue that makes up approximately one third of their daily support calls.



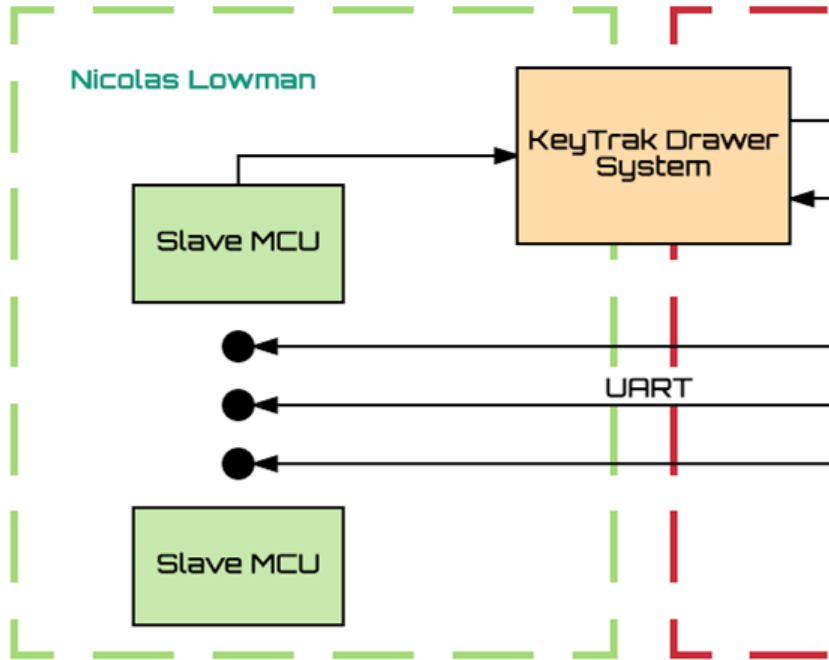
# Proposed Solution

- Solve the shorting issue with the use of microcontrollers to isolate each iButton read line
- Upgrade current communication protocol (LPT) with more modern ones (USB)
- Implement backwards compatibility for legacy devices
- Introduce new wireless communications and features for “next generation” key drawer

# System Overview



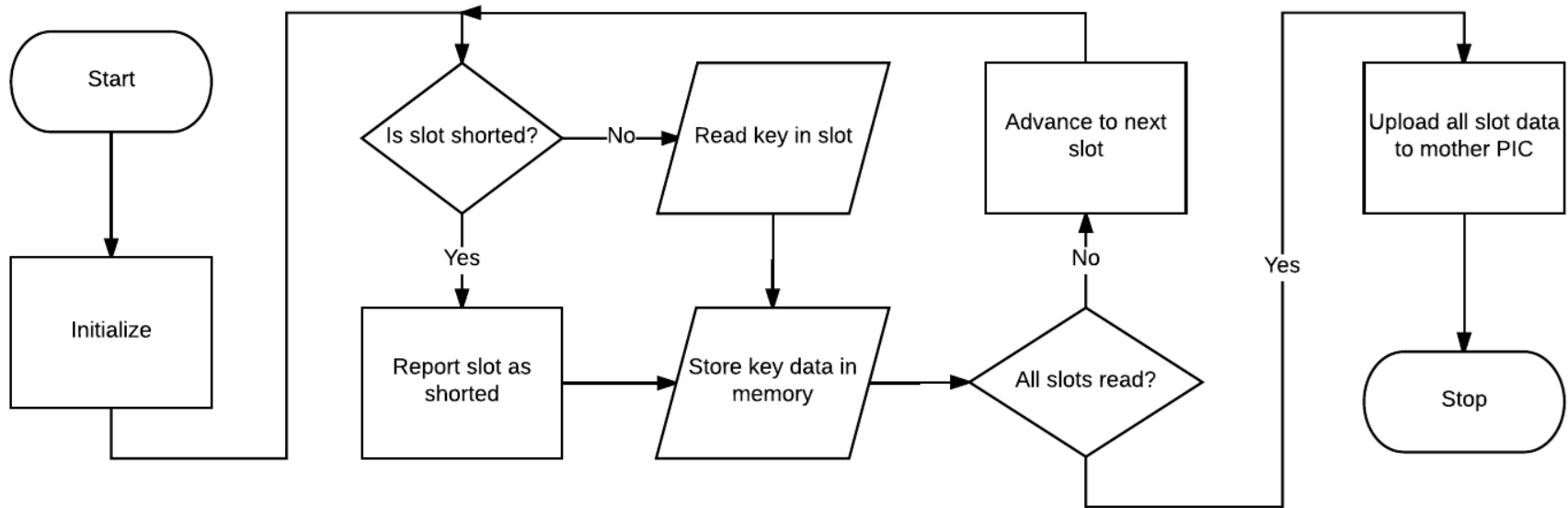
# Slave MCU Subsystem Overview



## 403 Deliverables

- Read iButton
- Store iButton Serial Number
- Solenoid drive system
- Power supply of entire system

# Slave MCU Subsystem Flowchart

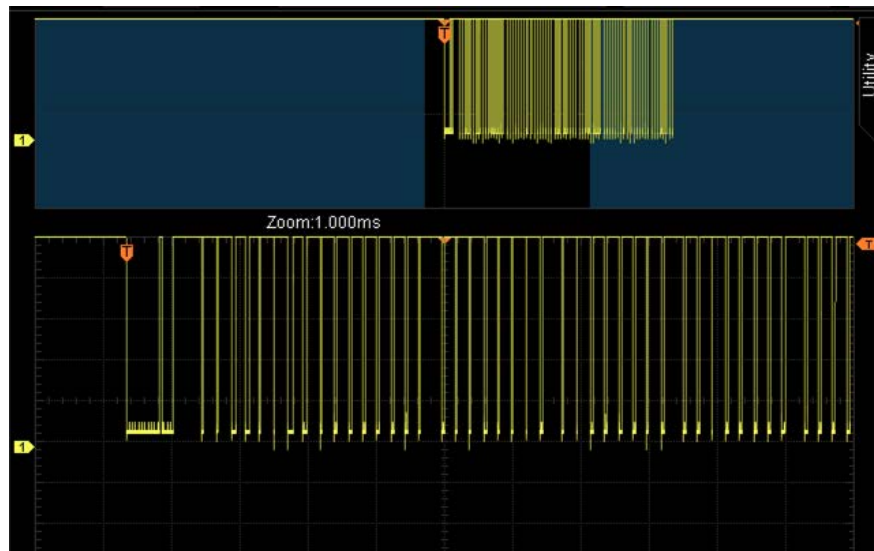


# Slave MCU Challenges

- ICD (In Circuit Debugger) no debugging operation with initial IDE software release
- Reading ibuttons
- Learning the IDE

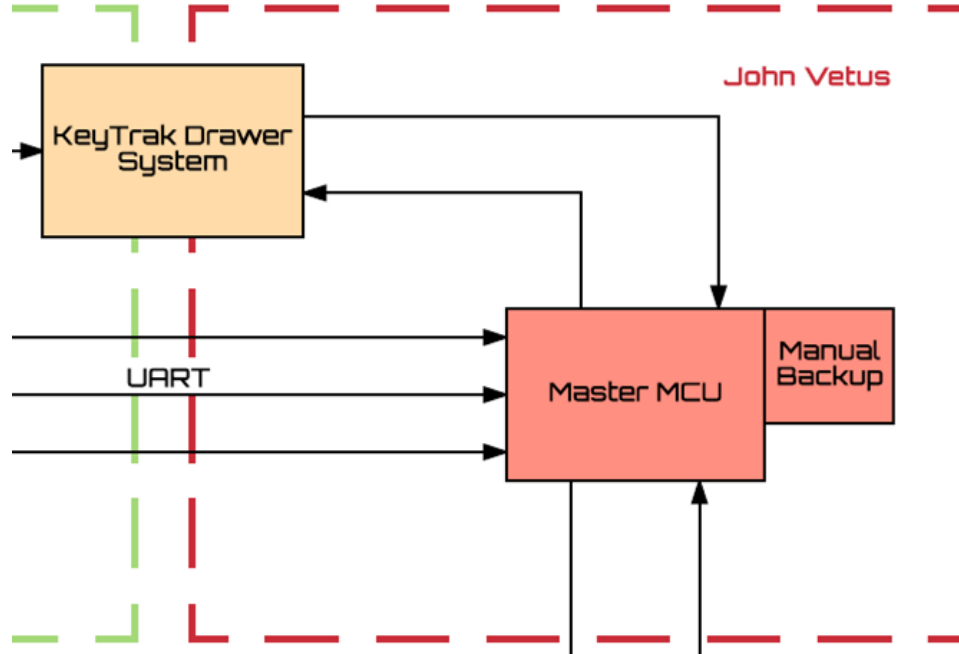


# Slave MCU Subsystem Test/Validation Results



```
The iButton serial number is 37 03 70 10 04 43 76 81
The iButton serial number is 21 03 70 10 02 f0 97 81
The iButton serial number is fb 03 70 10 02 f4 c2 81
The iButton serial number is f6 03 70 10 02 e4 a4 81
The iButton serial number is 25 03 70 10 03 1c 74 81
The iButton serial number is 1f 03 70 10 00 73 e8 81
```

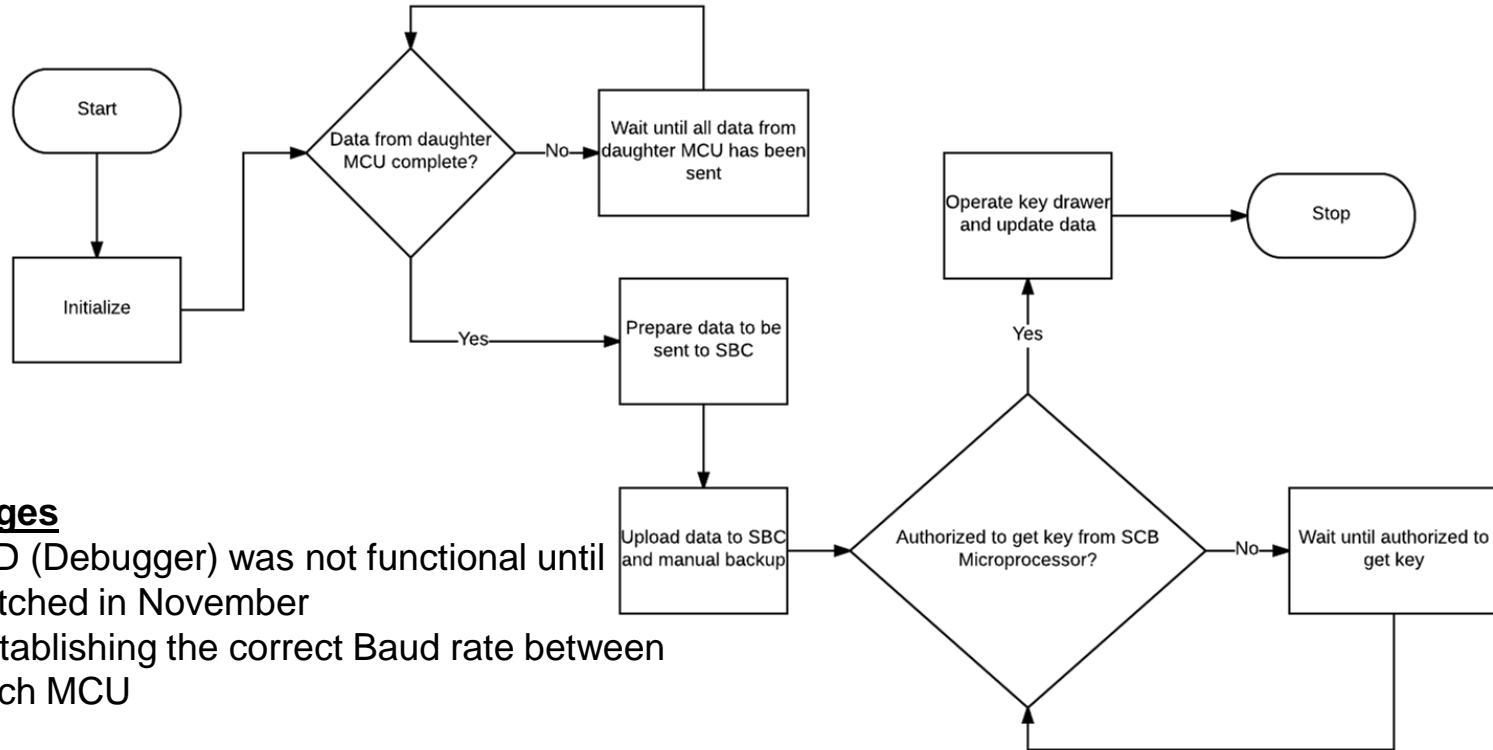
# Master MCU Subsystem Overview



## 403 Deliverables

- Establish communication protocols between MCUs and SBC through UART
- Create software to organize and track iButton data
- Receive data from SBC to operate drawer and indicate iButton slots
- Create a manual backup file that has iButton information and a timelog of operation

# Master MCU Subsystem Flowchart



## Challenges

- ICD (Debugger) was not functional until patched in November
- Establishing the correct Baud rate between each MCU

# Master MCU Subsystem Test/Validation Results

Row/Column	Column 1	Column 2	Column 3	Column 4
Row A	ID: E60370	ID: None	ID: None	ID: None
Row B	ID: None	ID: R90320	ID: None	ID: None
Row C	ID: None	ID: None	ID: 9E0010	ID: None
Row D	ID: 9E0010	ID: None	ID: None	ID: 1F0370

Figure 1. Table showing where iButtons are located on the drawer (Partial screenshot)

Action	User	Timelog
Drawer Open	Alan Ngo	11/16/2017 17:34
ID: E03220928	Alan Ngo	11/16/2017 17:34
Drawer Closed	Alan Ngo	11/16/2017 17:36
Drawer Open	John Vetus	11/16/2017 17:39
ID: F92321200	John Vetus	11/16/2017 17:39
Drawer Closed	John Vetus	11/16/2017 17:40

Figure 2. Table showing timelog of user activity

# Master MCU Subsystem Test/Validation Results

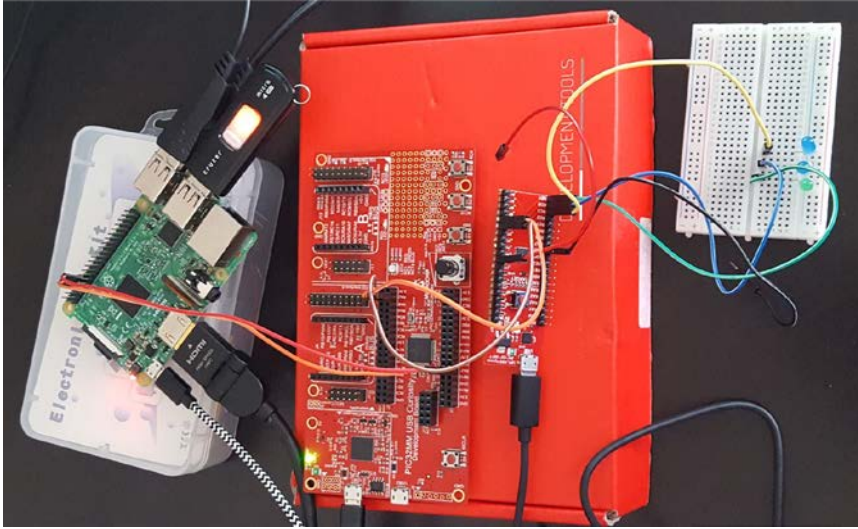
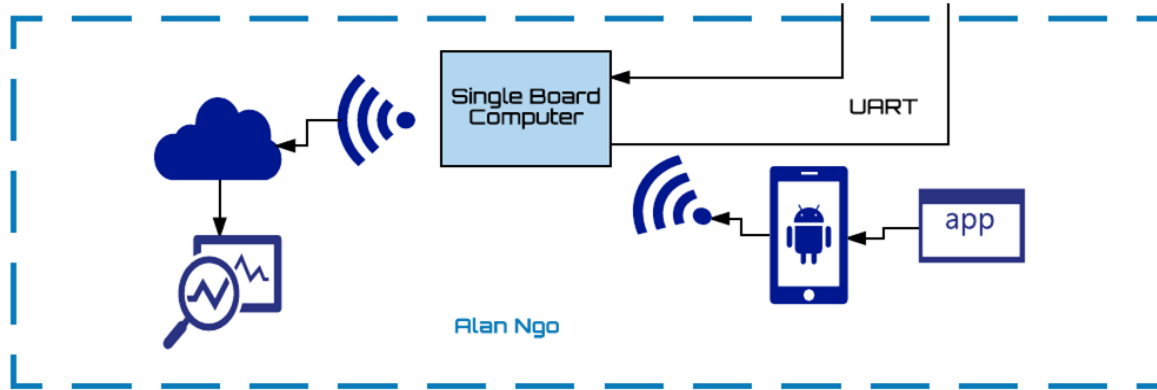


Figure 3. Master MCU connected to slave MCU and SBC

Test	Actual	Met
Validate communication between Master MCU and Slave MCU	Can read/write up to 16 bytes of data at a time through UART	✓
Validate communication between Master MCU and SBC	Can read/write up to 16 bytes of data at a time through UART	✓
Drawer operation and key retrieval/deposit	Can send a signal to operate drawer and flash an LED for visualization of where to take/place key	✓

# Single Board Computer Subsystem Overview



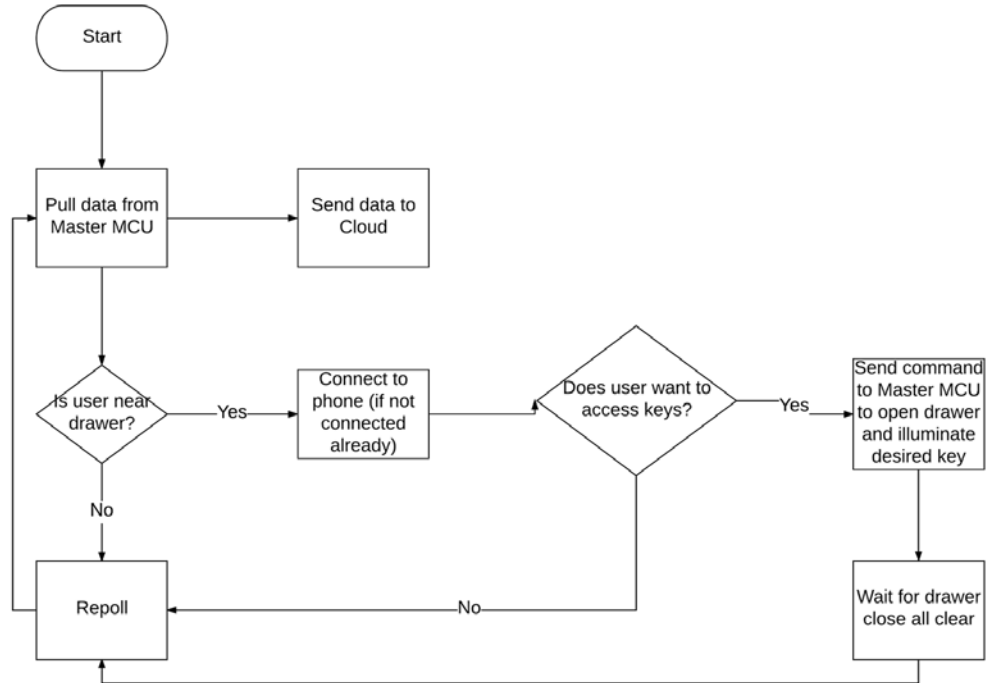
## 403 Deliverables

- Create a cloud backup system for the status of the keys and drawer
- Create an interface to allow the user to open the drawer from an android phone
- Create a website to view relevant information about the drawer and keys

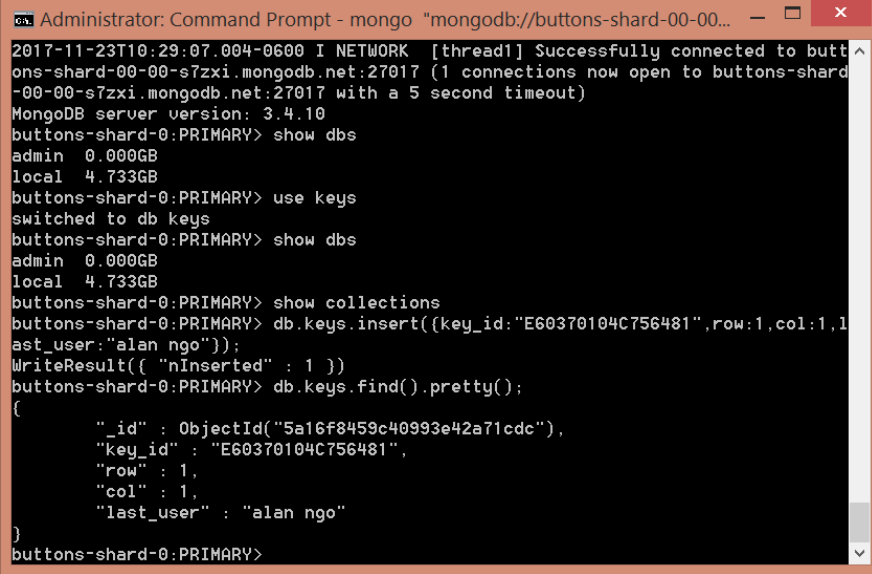
## Challenges

- Learning web dev and database manipulation

# Single Board Computer Subsystem Flowchart



# Single Board Computer Test/Validation Results



```
Administrator: Command Prompt - mongo "mongodb://buttons-shard-00-00-00-s7zxi.mongodb.net:27017 (1 connections now open to buttons-shard-00-00-s7zxi.mongodb.net:27017 with a 5 second timeout)
MongoDB server version: 3.4.10
buttons-shard-0:PRIMARY> show dbs
admin 0.000GB
local 4.733GB
buttons-shard-0:PRIMARY> use keys
switched to db keys
buttons-shard-0:PRIMARY> show dbs
admin 0.000GB
local 4.733GB
buttons-shard-0:PRIMARY> show collections
buttons-shard-0:PRIMARY> db.keys.insert({key_id:"E60370104C756481",row:1,col:1,last_user:"alan ngo"});
WriteResult({ "nInserted" : 1 })
buttons-shard-0:PRIMARY> db.keys.find().pretty();
{
  "_id" : ObjectId("5a16f8459c40993e42a71cdc"),
  "key_id" : "E60370104C756481",
  "row" : 1,
  "col" : 1,
  "last_user" : "alan ngo"
}
buttons-shard-0:PRIMARY>
```

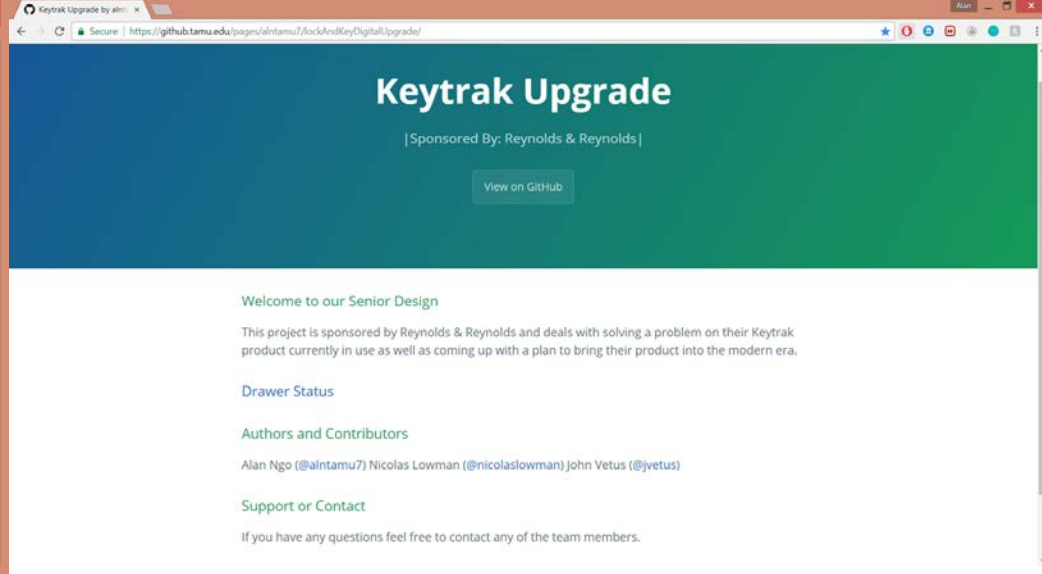


Figure 1: Terminal showing a key being added to the database.

Figure 2: Main project web page.



# Single Board Computer Test/Validation Results

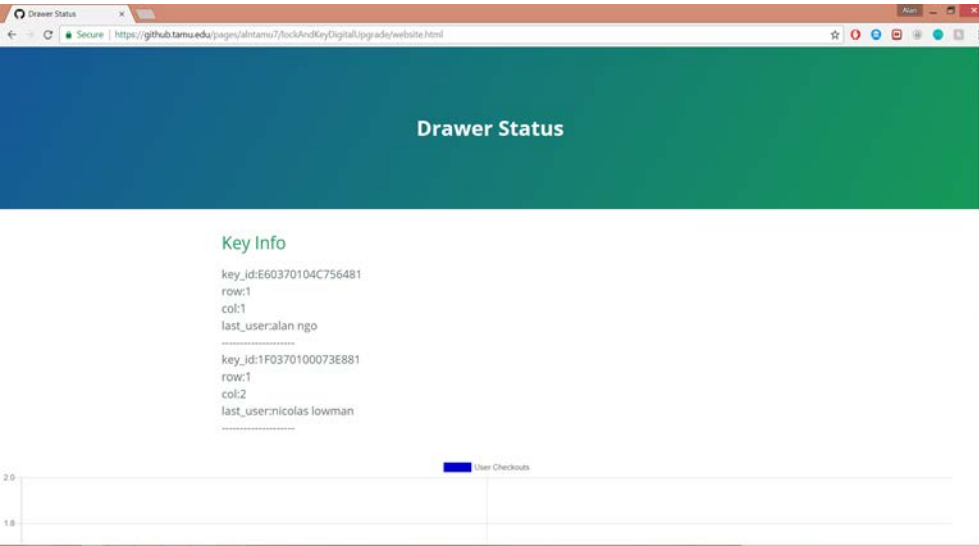
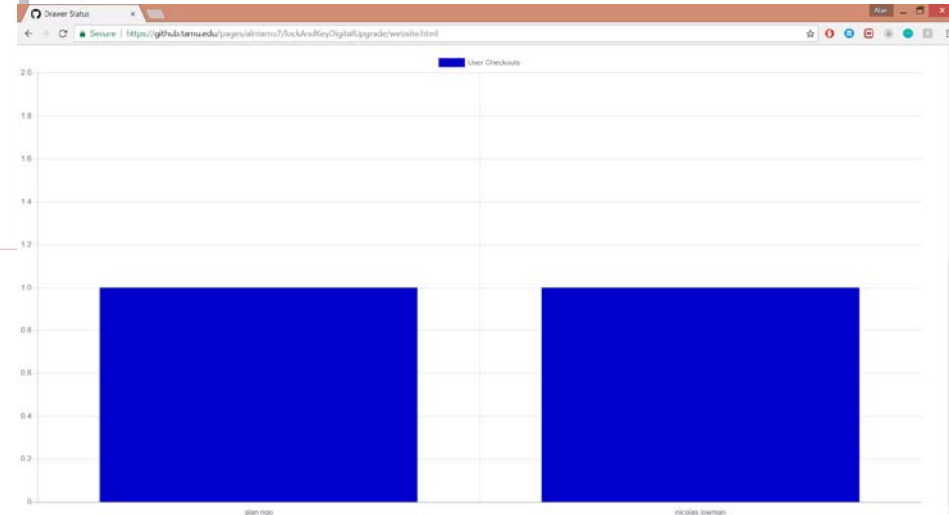
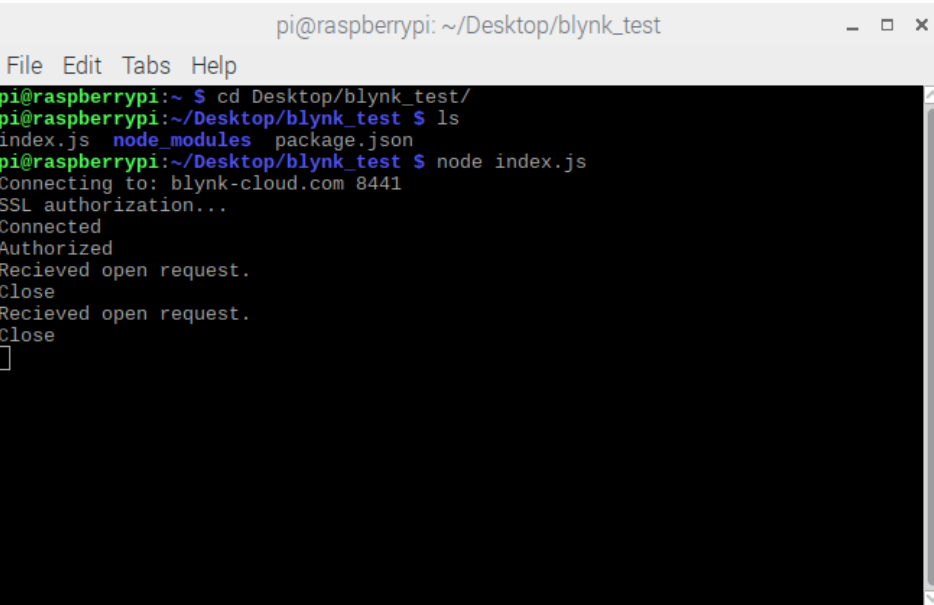


Figure 3: Web page showing current keys in the database/drawer.

Figure 4: Graph showing how many times a certain person has taken keys.



# Single Board Compute Test/Validation Results



```
pi@raspberrypi: ~/Desktop/blynk_test
File Edit Tabs Help
pi@raspberrypi:~ $ cd Desktop/blynk_test/
pi@raspberrypi:~/Desktop/blynk_test $ ls
index.js  node_modules  package.json
pi@raspberrypi:~/Desktop/blynk_test $ node index.js
Connecting to: blynk-cloud.com 8441
SSL authorization...
Connected
Authorized
Recieved open request.
Close
Recieved open request.
Close
█
```

Figure 5: Successful drawer open signal sent to Raspberry Pi.

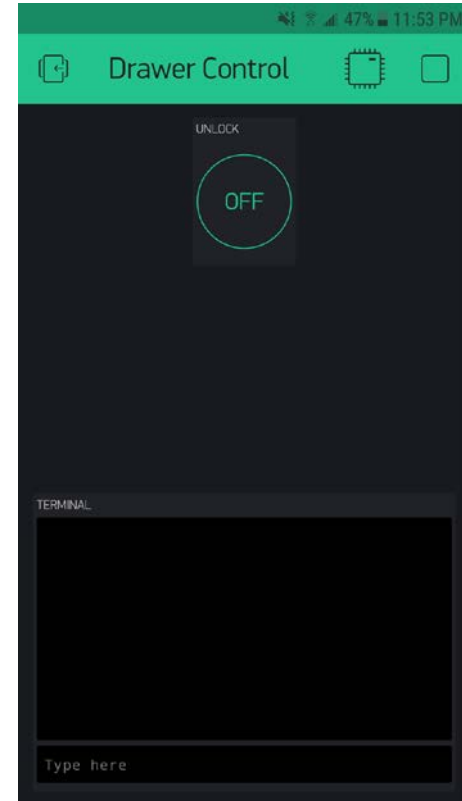


Figure 6: App interface for drawer control.

# Execution Plan Conclusion

- Slave MCU read and store iButton Serial numbers
- Master MCU accepts serial numbers over UART
- Master communicates with SBC over UART
- Website with database created to indicate iButtons in drawer

# 404 Goals

- Fabricate full size drawer PCB
- Validate and test communication between subsystems
- Validate and test overall system performance

# Questions?