RFID Door Lock
Team 12

Miguel Higgins Moy, Noah Page
Miles Breslin, John Sharpe
December 8, 2020

Need Statement

- RFID readers are ubiquitous in commercial environments. This is because of how convenient they are to use, low cost of RFID cards, ease of programing new users, and integration into secure locking mechanisms.
- Most residential smart locks lack RFID/NFC systems to actuate the lock.
- Most residential smart locks require monthly subscriptions, need internet connections to function, can be expensive, and don't always work with standard door locks.
- A new door lock system is needed that is affordable, can read RFID/NFC tags, functions with normal deadbolt lock assemblies, and does not require a monthly subscription or other hosted service to operate.

Concept of Operation

Our project is an RFID smart door lock mechanism that will work with standard deadbolt locks. This would offer residential users the versatility of entering their homes using a normal key, or an RFID enabled phone, implant, or keycard.

By tapping the RFID card against the RFID reader on the outside of the door, the user can easily toggle between locking and unlocking the door.

The user maintains their current functionality and has two methods to bypass the system

- Use the already existing key from outside of the door
- Use a new Lock/unlock button from the inside of the door

The user can easily assign additional tags using the web interface

Motivation

What do we benefit from making this project?

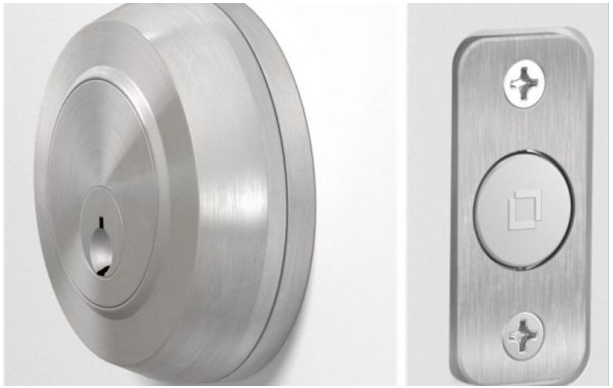
- Convenient keyless entry to house, just slam your tag at the door
- Improve accessibility by lowering the amount of “key fumbling”
- Great for people with RFID/NFC implants or wearables. Never be locked out!
- Low cost at about $75 (actual cost per person)
- Smart home integration with other smart devices

General Market Trends

- Replace entire lock assembly
- Uses pin-pad for access (ugly and not discrete)
- Many cannot use the existing lock
- Most home use smart locks lack RFID/NFC support
- Less expensive alternatives are available, but usually still cost above $100

Popular Alternatives

Level Bolt



- Great Battery Life
- Low Profile Look
- Works with standard locking hardware
- Bluetooth support
- Lacks RFID/NFC sensor
- Depends on Bolt app and subscription
- Expensive ($229)

August Wi-Fi Smart Lock

- Easy Installation
- Great Battery life
- Relatively low profile design
- Bluetooth and Wifi support
- Lacks RFID sensor
- Requires on August service subscription
- Expensive ($249.99)

Requirements

Functionality

- Must be able to work with existing doors
- Must lock and unlock a door
- Must be able to work without an internet connection
- Should be able to work with existing locks
- Should be able to lock/unlock from the inside by physical means
- Should still allow for locking/unlocking with a normal key
- May have smart home integration

Performance

- Should reliably sense RFID on first try > 90% of the time

Economic

- Should be cheaper than most market offerings (<$130)

Energy

- Should be able to work during a power outage
- May have enough battery capacity for regular use that will last a month

Maintainability

- Should be installed with 2mm and 2.5mm hex wrench and phillips head screwdriver by homeowner

Reliability and Availability

- Should be durable enough to last more than a year with regular use

Usability

- Should be easy to assemble

Our Approach

We knew this was a big project for a 2 credit class so we wanted to take advantage of open source work, so we looked to GitHub.

Software

Forked from github.com/esprfid/esp-rfid

Existing Capabilities

- WiFi connectivity with WebUI
- Configuration tool
- Debug Tools
- Lock Toggle action

Our Contributions

- Servo Motor locking mechanism
- Configuration Fields for Servo

Hardware

Forked from github.com/adafruit/Adafruit-Feather-ESP8266-HUZZAH-PCB

Existing Capabilities

- Power Management
- USB to Serial
- Auto-Reset

Our Contributions

- Full board layout
- Found better micro-usb implementation
- Sensors and actuators

Intellectual Property

Our Team's Contributions

Team Wiki: MIT (Full Ownership)

Code: MIT (Inherited from *esp-rfid Community*)

PCB: Creative Commons ShareAlike (inherited from Adafruit) (CC-BY-SA-3.0)

Unmodified Dependencies
Bootstrap: MIT
FooTable: GPL-3.0-or-later
jQuery: MIT
ArduinoJson: MIT
ESPAsyncTCP: LGPL-3.0-or-later
ESPAsyncUDP: Not licensed!!!
ESPAsyncWebServer: LGPL-3.0-only
AsyncMqttClient: MIT
MFRC522 (rfid): Unlicense
Wiegand-Protocol-Library-for-Arduino: LGPL-2.1-or-later
Time: LGPL-2.1-or-later
Bounce2: MIT
ESPtool: GPL-2.0-or-later

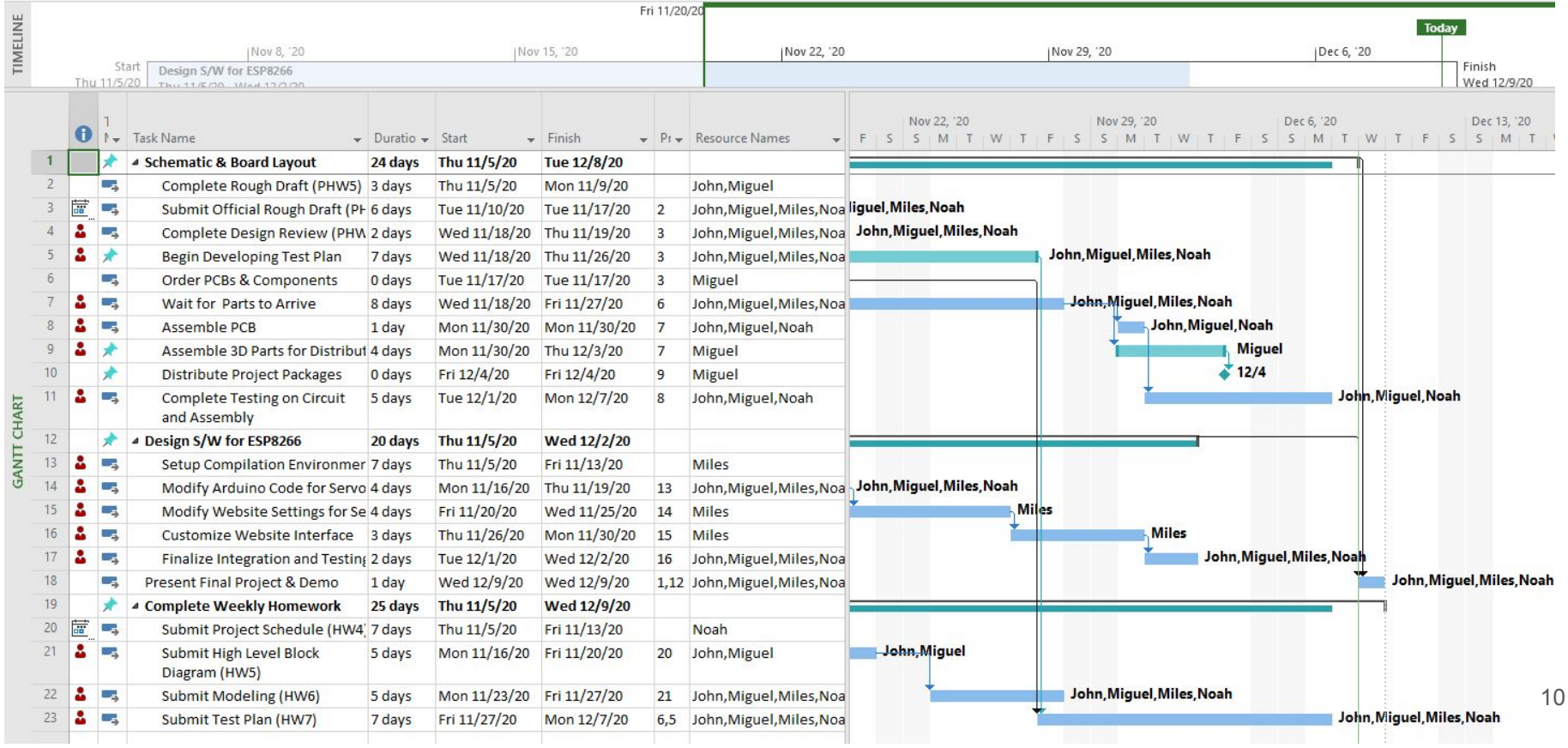
Some dependencies are being used in violation of their existing licenses (our interpretation). These are dependencies that were inherited from the original source code.

Violations

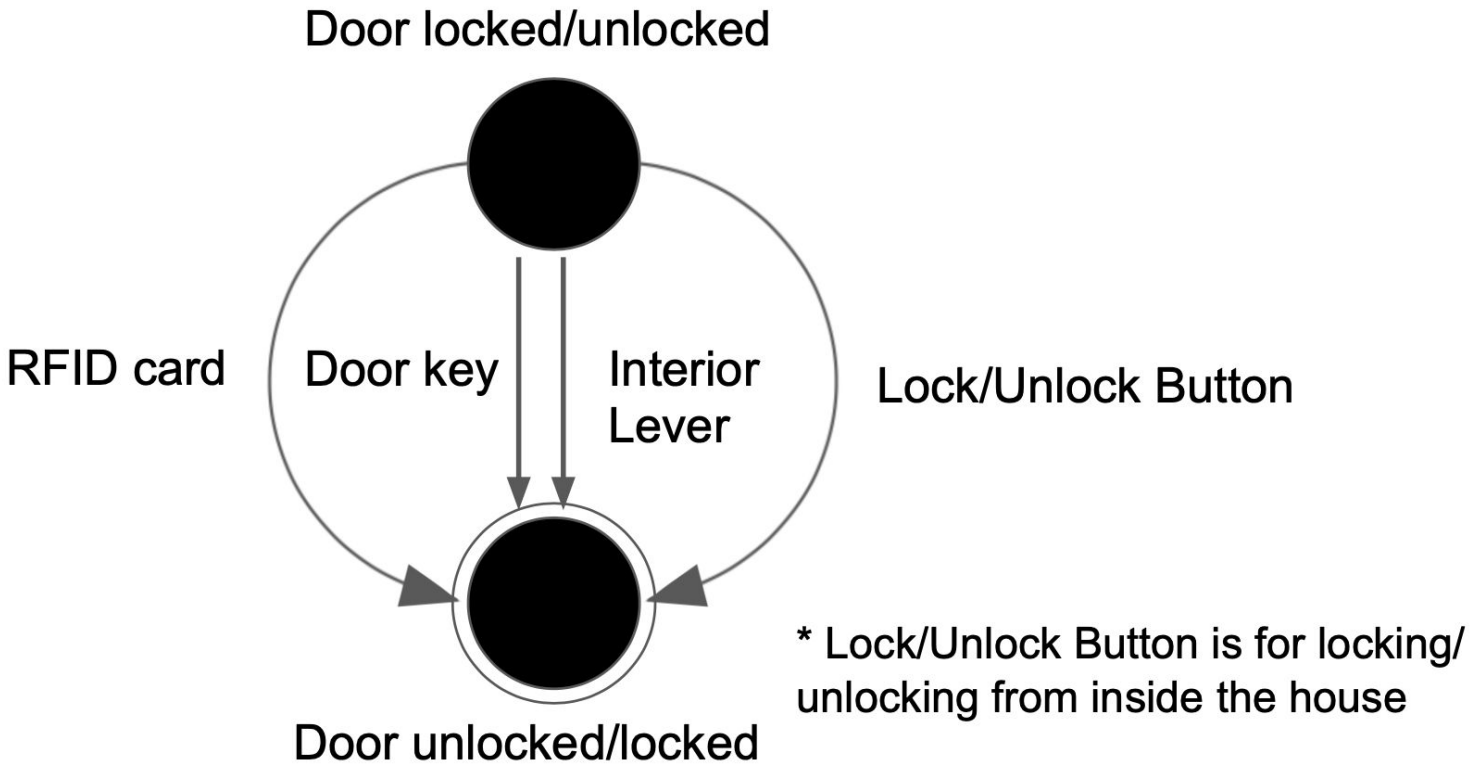
- FooTable (GPL Web Library)
- ESPAsyncUDP (No license networking tool)
- ESPtool (GPL deployment tool)

Since the software repository is licensed under MIT, we are unable to distribute with GPL dependencies, and a project with no license means no rights.

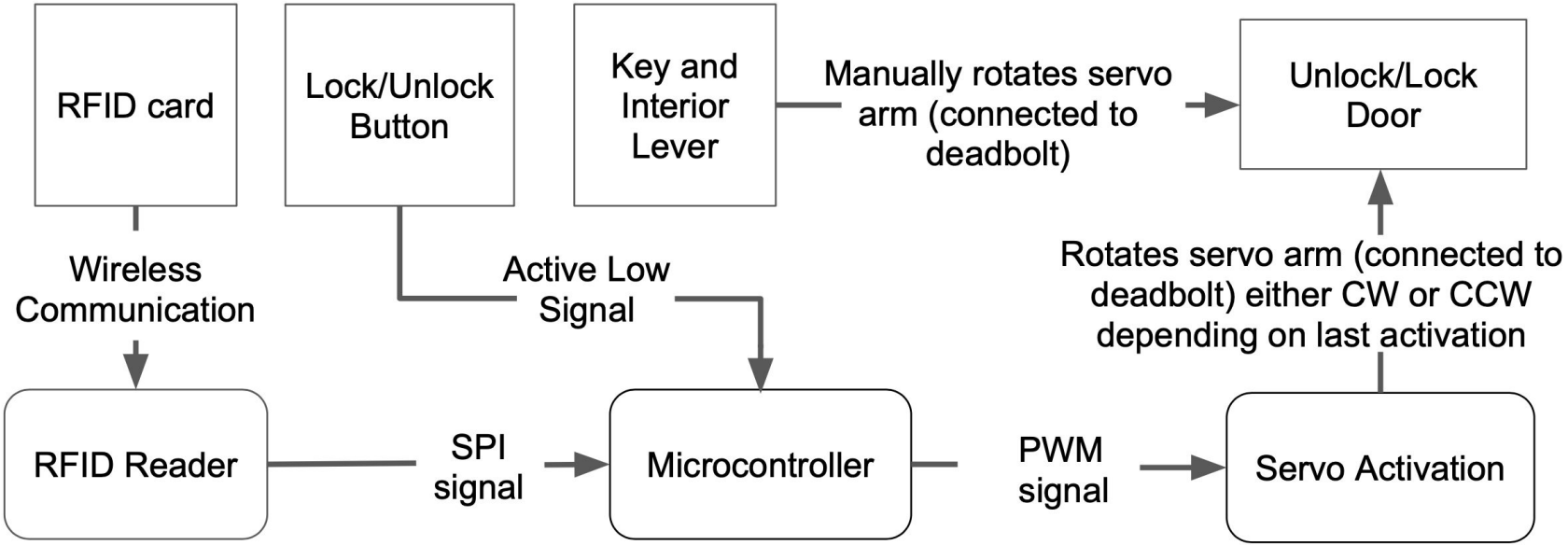
Project Schedule



Models: State Diagram

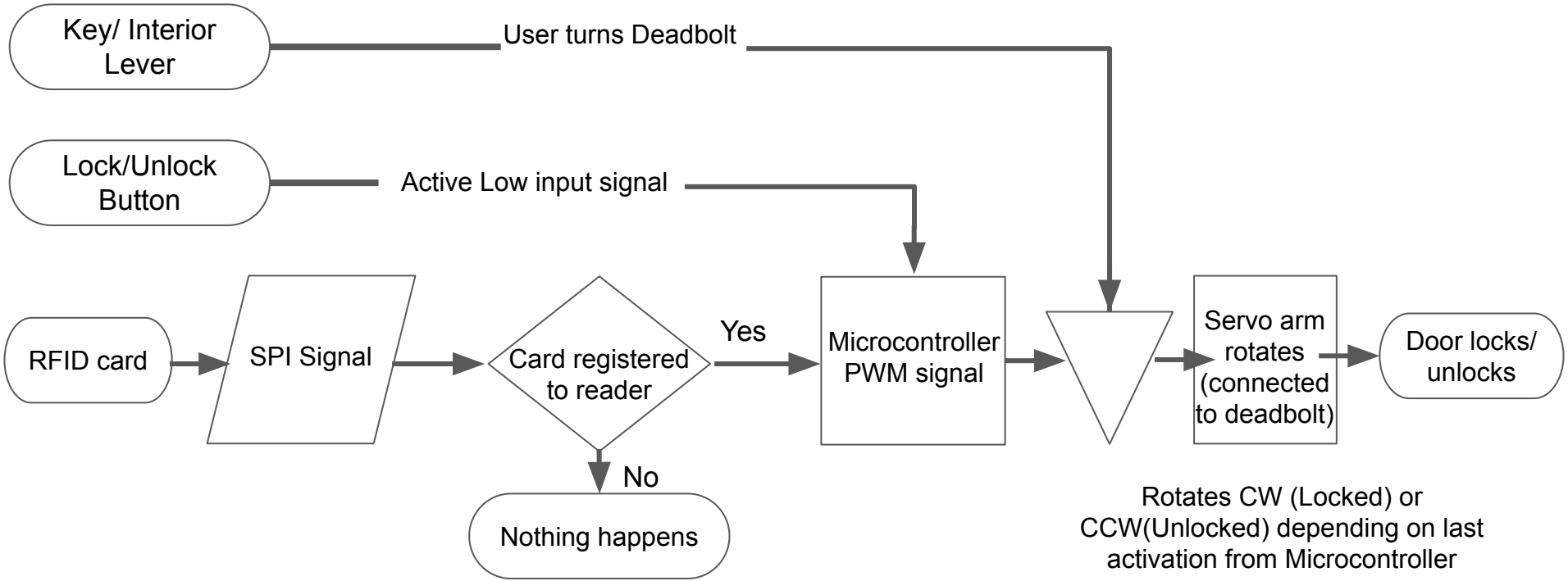


Data Flow Diagram

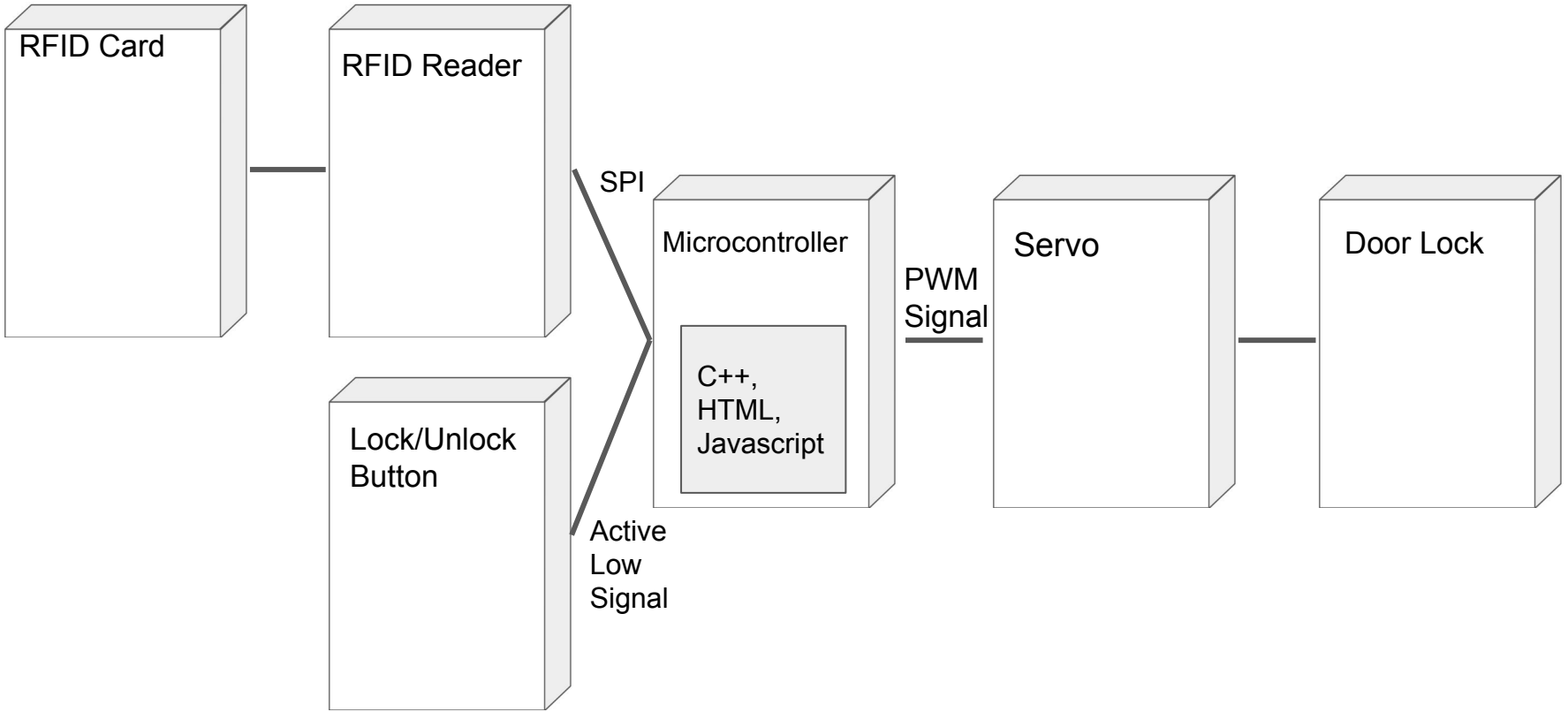


* Lock/Unlock Button is for locking/unlocking door from inside the house

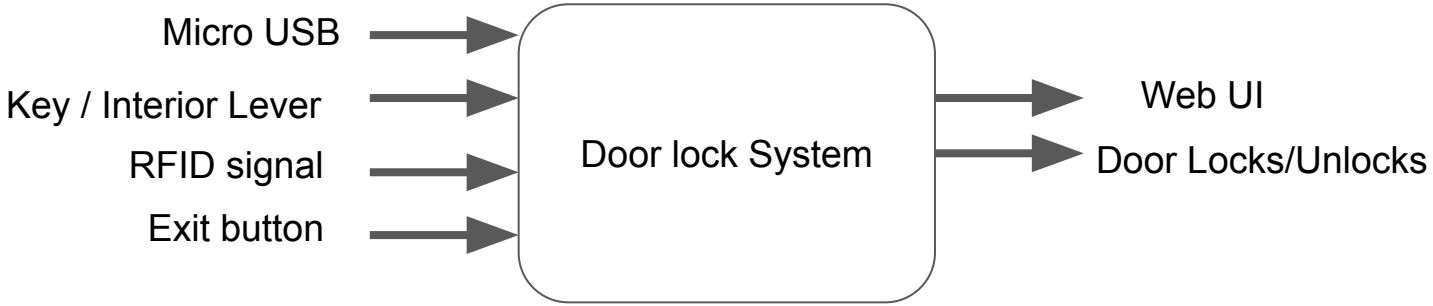
Flowcharts


Key/ Interior Lever
User turns Deadbolt
Lock/Unlock Button
Active Low input signal
RFID card
SPI Signal
Card registered to reader
Yes
No
Nothing happens
Microcontroller PWM signal
Servo arm rotates (connected to deadbolt)
Door locks/ unlocks
Rotates CW (Locked) or CCW(Unlocked) depending on last activation from Microcontroller

UML Physical View

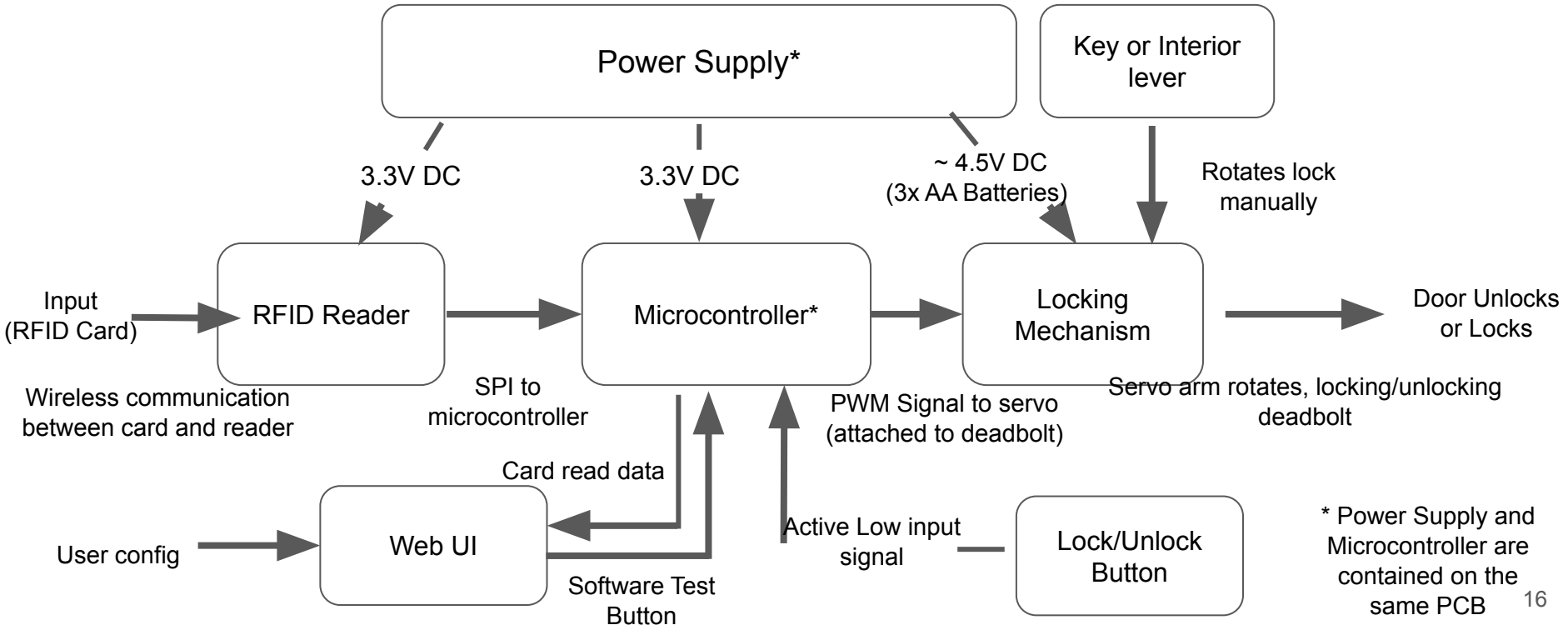


RFID Door Lock: Level 0

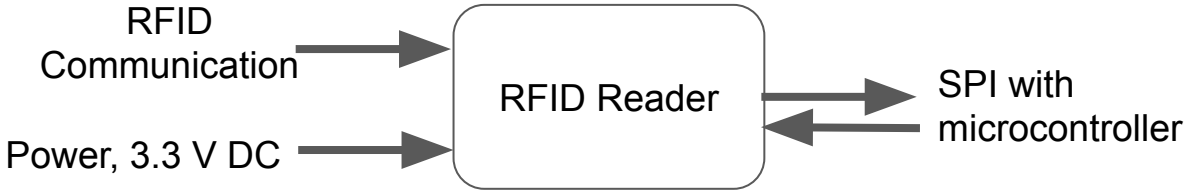


Module	RFID Door Lock System
Inputs:	Key (Can only be input from outside of house) to lock and unlock Interior Lever (Used on the inside of the house) to lock and unlock RFID signal read (Can only be used as input from outside house) Exit button (Can only be used as input from inside house) Micro USB (Used to flash code) WebUI for adding user credentials
Outputs:	Servo actuation that locks/unlocks the door WebUI
Functionality:	Door lock mechanism that can be locked/unlocked using a standard key, overridden from the inside, RFID card from the outside, or a button on the inside. Firmware is flashed through the Micro USB port, and settings are accessed through a WebUI

RFID Door Lock: Level 1

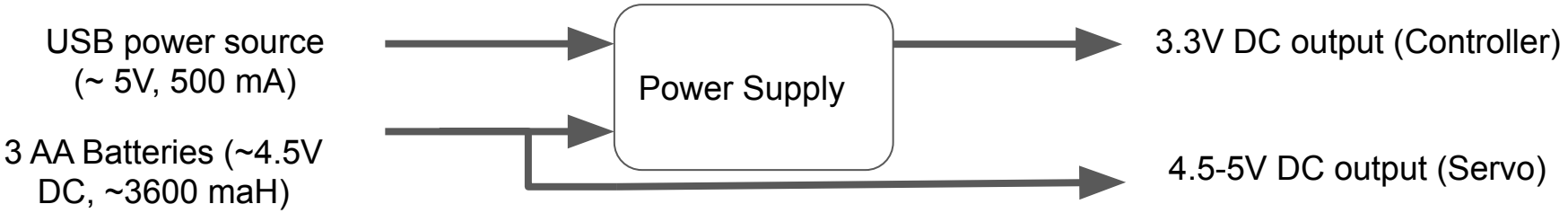


RFID Reader: Level 0 (Pre-built Assembly)



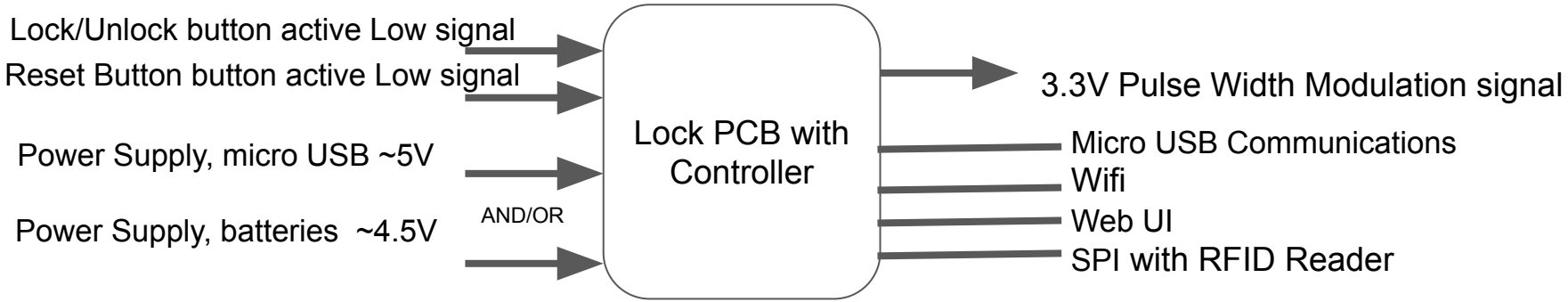
Module	RFID Sensor
Inputs:	Power, 3.3 V ~15mA DC from Lock PCB Wireless communication with RFID Card
I/O:	SPI signal to communicate with Lock PCB Microcontroller with data from RFID/NFC tag
Functionality:	Used to take UID from RFID/NFC tags

Power Supply: Level 0



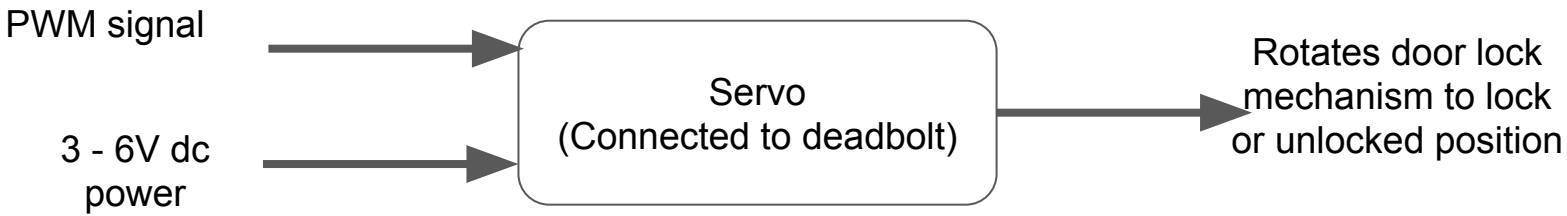
Module	Power Supply
Inputs:	USB power source Battery Power (3 AA batteries)
Outputs:	3.3V dc, ~ 4.5-5V dc
Functionality:	Power Supply, regulates higher voltage inputs down to 3.3V as needed for the microcontroller and RFID sensor. Can take input from either USB or batteries. Servo 3-5V supply bypasses voltage regulator and goes directly to the servo

Lock PCB with Microcontroller: Level 0



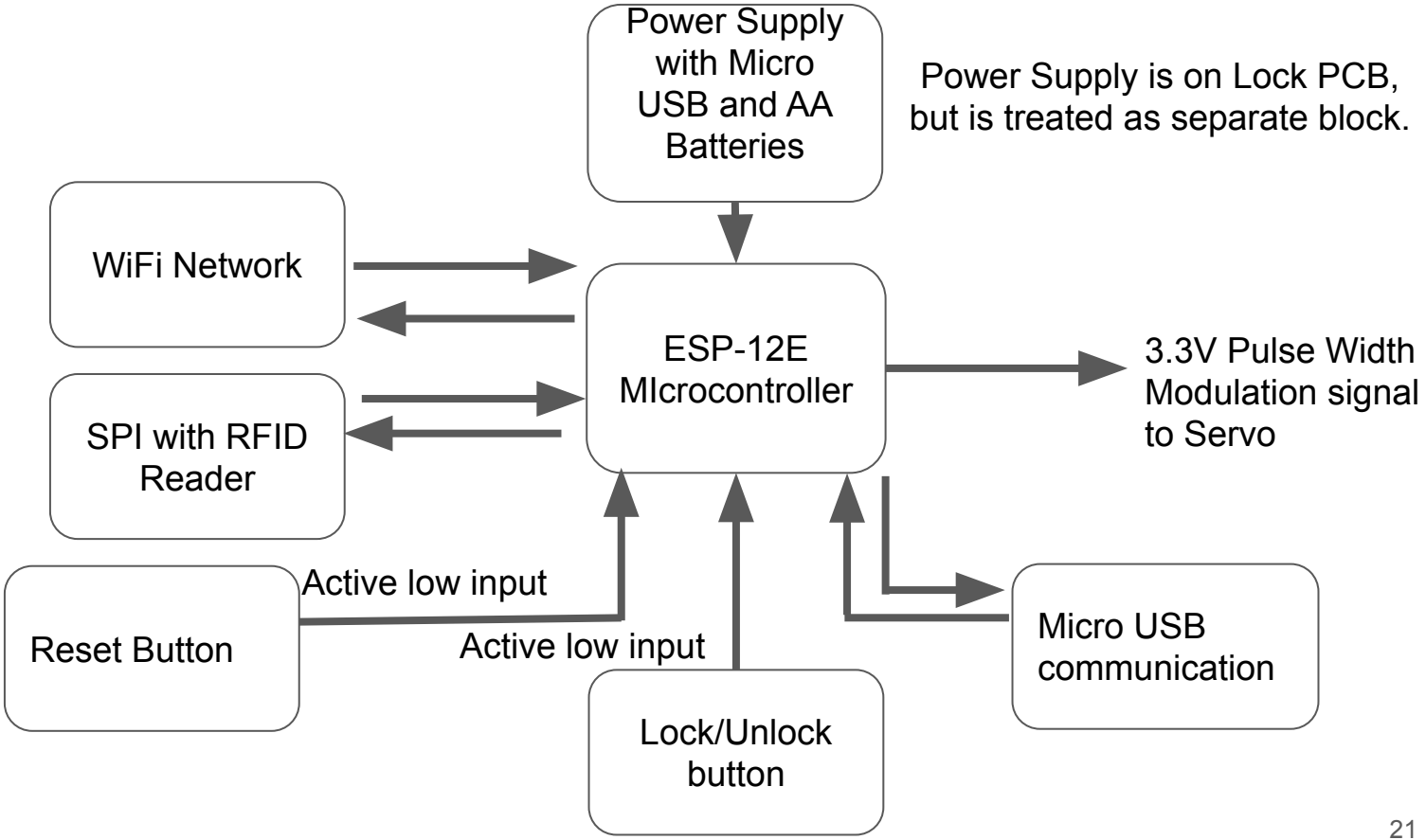
Module	Development Board with ESP 12E processor
Inputs:	From Power Supply: ~5V DC (Micro USB) AND/OR ~4.5 V DC (3 AA Batteries) Reset Button input signal (Resets processor) Lock/Unlock button active low signal to send lock/unlock signal
Outputs:	3.3V Pulse Width Modulation signal to servo to lock or unlocked position Web interface (Used to PWM signal output to servo and view access logs)
I/O	SPI data from RFID reader with tag data WiFi: User configuration, and allow user to store new RFID card data through Web UI Micro USB Communication for programming
Functionality:	Interpretes read data from RFID reader or input from exit button into a PWM signal used to activate servo either to the lock or unlocked position depending on it's last activation. Stores registered RFID card data, checks if RFID card is registered, and time/date of RFID scans.

Servo (Actuator): Level 0 (Pre-Assembled)

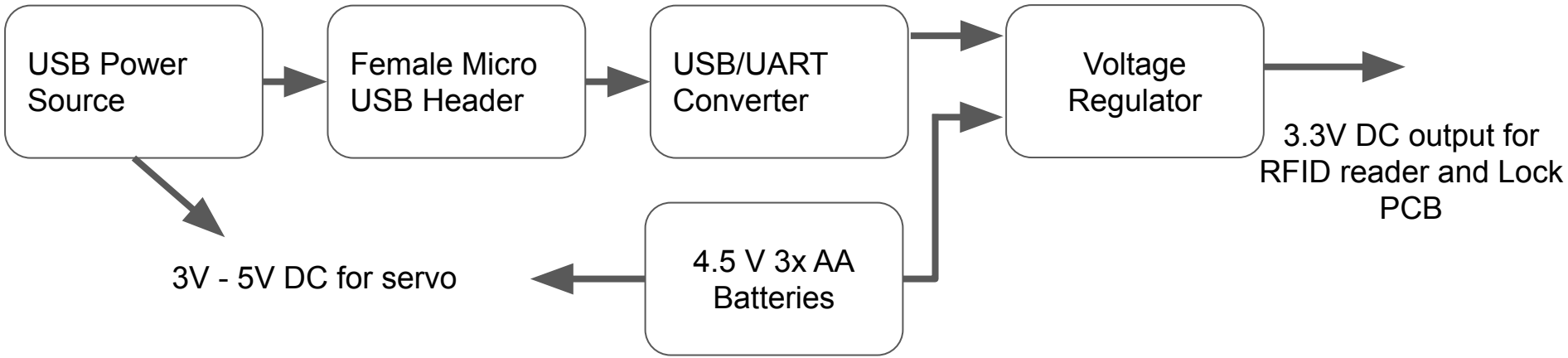


Module	Servo
Inputs:	Pulse Width Modulation signal 3 - 6V dc power
Outputs:	Rotates door lock mechanism
Functionality:	Micro Servo with 180 degree arm rotation. Rotation direction is controlled by Pulse Width Modulation input. Servo arm is connected to deadbolt assembly, which locks/unlocks door.

PCB with Microcontroller: Level 1



Power Supply: Level 1



* USB Power will disable the power from the AA batteries. A charge regulator should also have been included to accommodate rechargeables

Hardware Choices

- ESP-12E/F Microcontroller Module
  - WiFi connected module (no needing to design antennas)
  - Well proven in other IoT devices
  - Already has an RFID software implementation! (ESP-RFID)
- MFRC522 RFID/NFC Board
  - Easy to implement with ESP-RFID
  - Cheap and reliable
  - NFC compatible - meaning NFC implant compatible!
- MG90S Micro servo
  - Metal gears for reliability when using an override system
  - Small and easy to drive using battery voltage
  - Easy to implement in software
- Adafruit Feather Huzzah as a base for our schematic
  - Major DIY brand
  - Good low quiescent current LDO implementation
  - Auto-reset
  - Usb to serial circuitry
  - Reverse input protection
  - USB/Battery power protection/disconnect
  - Open source! (licensing discussed earlier)

Hardware Alternatives

- Power design
  - Wall power from power supply was not chosen due to having to manage the wires to the door would have proven obtrusive and ugly
  - LiPo batteries were not chosen due to more complexity required and to charge it, you would have to have multiple batteries to charge elsewhere and put in

- Actuator design
  - Stepper motor was not chosen due to its size and power requirements. It would have been great because of it’s high power. Added complexity to drive.
  - DC motor was not chosen because of needing a driver and gearing to actuate the lock.

Software Choice

- ESP-RFID
  - Open source! (licensing discussed earlier)
  - Web App
    - Can double as hardware diagnostic tool
    - Native apps limit platform compatibility
    - Simple management interface
  - WiFi
    - Can serve the Web App as a standalone service
    - May be able to extend to add cloud support
    - Does not depend on any additional hardware/services
  - Hardware compatibility
    - Compatible with a multitude of different RFID protocols and reader boards

Bill Of Materials for: Team 12 RFID Door Lock
Last modified: 12/8/20
PCB version: E
BOM revision: 4.0

P/NP Place/Not Place (components marked NP are not stuffed on the board)

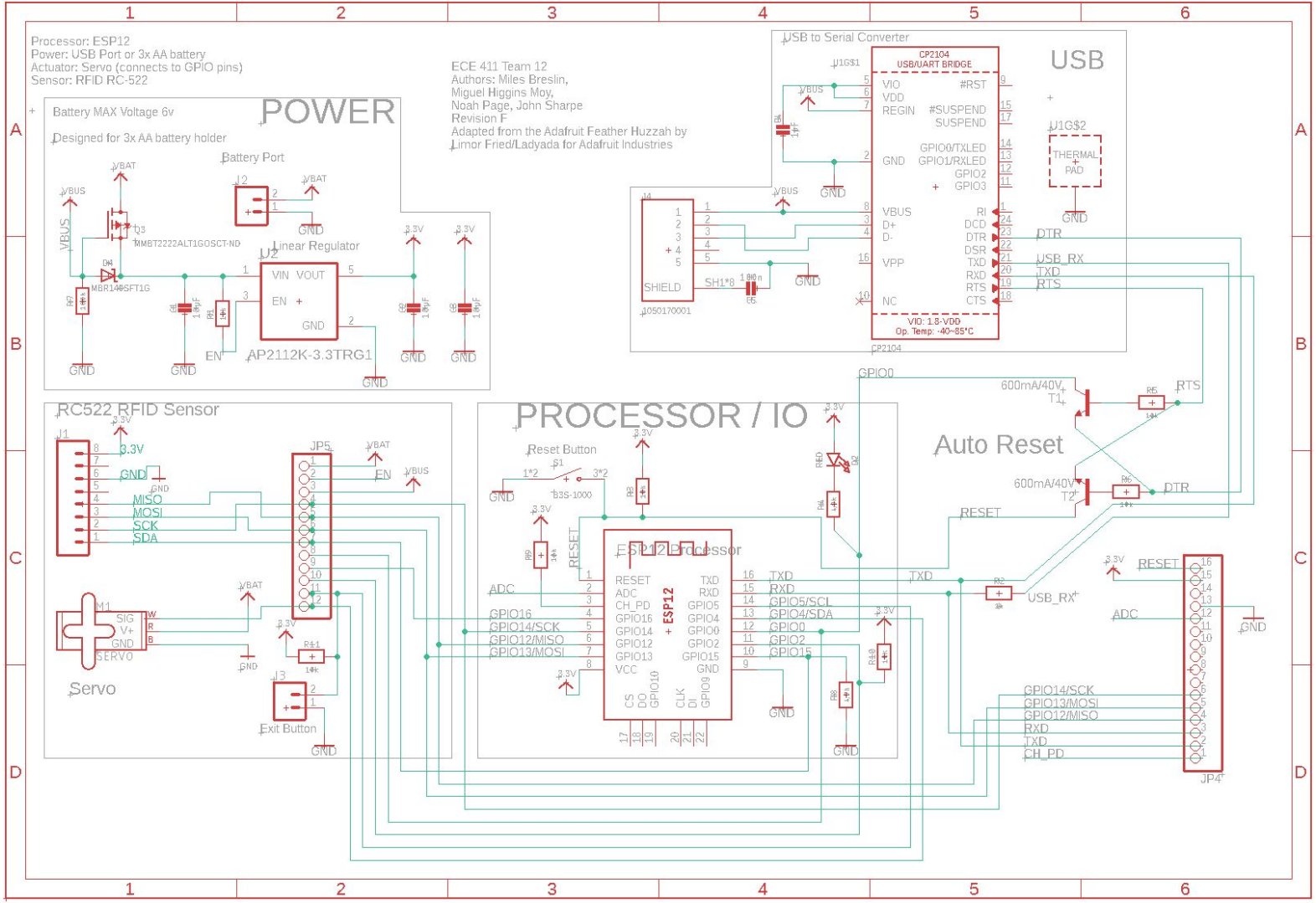
Cnt	Part References	P/NP	Mfg	Value	Package	Mfg PN	Description	Dist	Dist Part Number	Link	Cost Ea.	Cost Total
1	-	NP	OshPark	-	-	-	OshPark PCB	OshPark	-	-	$6.92	$6.92
1	X1	P	Espressif Systems	-	-	ESP-12E	Microcontroller	Digikey	1528-1438-ND	https://www.digik	$6.95	$6.95
1	M1	NP	Tower Pro	-	-	MG90S	Micro servo	Smart Prototyping		https://www.sma	$4.99	$4.99
1	J1	NP	NXP	-	-	MFRC522	RFID Sensor	Amazon		https://www.ama	$1.99	$1.99
1	X4	P	Molex	1050170001	MOLEX_1050170001	1050170001	USB Micro port	Digikey	WM1399CT-ND	https://www.digik	$0.69	$0.69
1	U1G$1	P	Silicon Labs	CP2104	QFN24_4MM_SMSC	CP2104-F03-GN	USB to UART Bridge Controller	Digikey	336-4146-1-ND	https://www.digik	$1.65	$1.65
1	Q3	P	Alpha & Omega Semicond	AO3403	SOT23-R	AO3403	SOT-23 P-Channel MOSFET	Digikey	785-1003-1-ND	https://www.digik	$0.28	$0.28
1	AP2112-3.3	P	MicroChip Technology	AP2112-3.3	SOT25	AP2112K-3.3TR	Voltage Regulator	Digikey	AP2112K-3.3TRG1DICT-ND	https://www.digik	$0.36	$0.36
2	JP2, JP3	NP	Samtec Inc.	-	-	SSM-102-L-SV-E	1x2 Female Pin Header	Digikey		https://www.digik	$0.93	$1.86
1	JP4	NP	Samtec Inc.	-	-	SSM-116-L-SV-E	1x16 Female Pin Header	Digikey		https://www.digik	$2.95	$2.95
1	JP5	NP	Samtec Inc.	-	-	SSM-112-L-SV-E	1x12 Female Header	Digikey		https://www.digik	$2.38	$2.38
1	R7	P	Vishay Dale	100k	0603-MINI	CRCW0603100I	100kΩ Resistor	Digikey	541-100KSCT-ND	https://www.digik	$0.05	$0.05
6	R1, R3, R5, R6, R9, R10	P	Vishay Dale	10k	0603-MINI	CRCW060310K(	10kΩ Resistor	Digikey	541-10.0KSCT-ND	https://www.digik	$0.05	$0.30
1	R2	P	Vishay Dale	1k	0603-MINI	CRCW06031K0I	1kΩ Resistor	Digikey	541-1.00KSCT-ND	https://www.digik	$0.05	$0.05
2	R4, R8	P	Vishay Dale	4.7k	0603-MINI	RCS06034K70F	4.7kΩ Resistor	Digikey	541-2792-1-ND	https://www.digik	$0.04	$0.08
3	C1, C2, C3	P	Samsung EM	10u	0603-MINI	CL10X106MO8N	10µF Ceramic Capacitor	Digikey	1276-6769-1-ND	https://www.digik	$0.11	$0.33
1	C4,	P	Samsung EM	1u	0603-MINI	CL10A105KO8N	.1µF Ceramic Capacitor	Digikey	1276-1034-1-ND	https://www.digik	$0.02	$0.02
1	C5	P	Samsung EM	100n	0603-MINI	CL10B104KO8N	1µF Ceramic Capacitor	Digikey	1276-1005-2-ND	https://www.digik	$0.01	$0.01
2	T1, T2	P	ON Semiconductor	600mA/40V	SOT23-3 (Version 1)	MMBT2222ALT1	NPN BJT	Digikey	MMBT2222ALT1GOSCT-ND	https://www.digik	$0.12	$0.24
1	D4	P	ON Semiconductor	1A/40V	SOD-123	MBR140SFT1G	SOD-123 Schottky diode	Digikey	MBR140SFT1G	https://www.digik	$0.29	$0.29
1	D2	P	Kingbright	RED	CHIPLED_0603	APT1608SURCI	0603 Red LED	Digikey	754-1123-1-ND	https://www.digik	$0.21	$0.21
1	SW1	P	Omron Electronics Inc-EM	BB3S-1000		BB3S-1000	Tactile Switch	Digikey	SW415-ND	https://www.digik	$0.63	$0.63
3	-	NP	Eneloop	AA - 1200mah	-	BK-3MCCA16FA	AA Batteries	Amazon	B00JHKSN4O	https://www.ama	$2.59	$7.77
										SUBTOTAL:		$41.00
					Physical Items							
1	-	-	Miguel (3D printed)	-	PLA+	-	Inner Door Mount	-			$0.26	$0.26
1	-	-	Miguel (3D printed)	-	PLA+	-	Lock Plate Long	-			$0.32	$0.32
1	-	-	Miguel (3D printed)	-	PLA+	-	Lock Cover	-			$0.32	$0.32
1	-	-	Miguel (3D printed)	-	PLA+	-	Frame	-			$1.25	$1.25
1	-	-	Miguel (3D printed)	-	PLA+	-	Battery PCB Frame	-			$1.22	$1.22
1	-	-	Miguel (3D printed)	-	PLA+	-	Battery Cover	-			$0.21	$0.21
1	-	-	Miguel (3D printed)	-	PLA+	-	Battery	-			$0.90	$0.90
1	-	-	Miguel (3D printed)	-	PLA+	-	Cover	-			$0.76	$0.76
1	-	-	Miguel (3D printed)	-	PLA+	-	Override	-			$0.28	$0.28
12	-	-	-	M3 Threaded Heatset Inserts	-	-	-	Amazon		https://www.ama	$0.11	$1.27
8				M3x6	Flat Head Screw			Amazon		https://www.ama	$0.09	$0.72
4				M3x8	Flat Head Screw			Amazon		https://www.ama	$0.09	$0.36
4				M3x30	Flat Head Screw			Amazon		https://www.ama	$0.09	$0.36

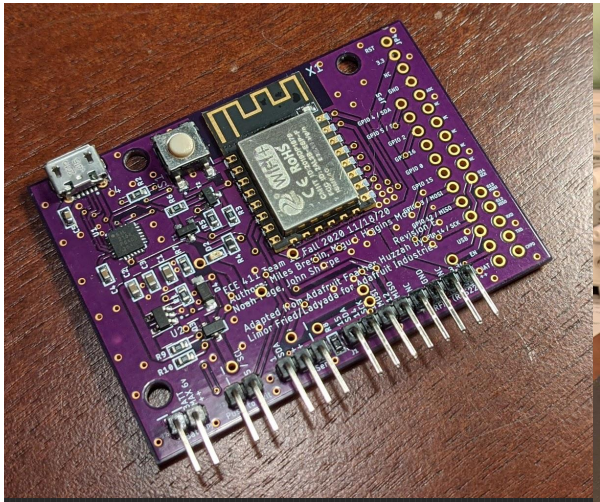
SUBTOTAL: **$8.23**

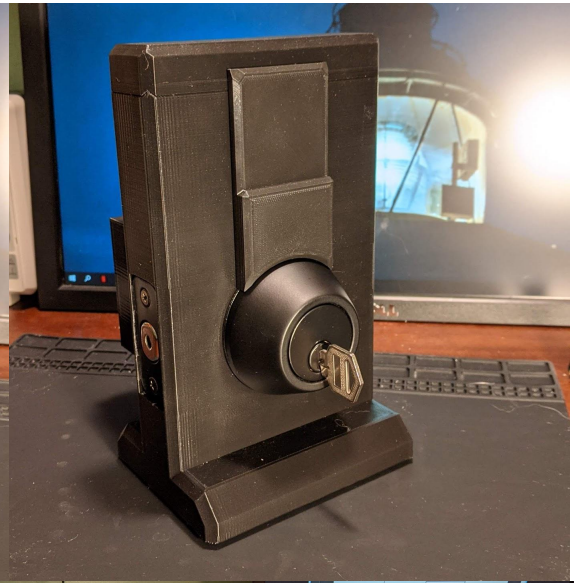
TOTAL COST **$49.23**

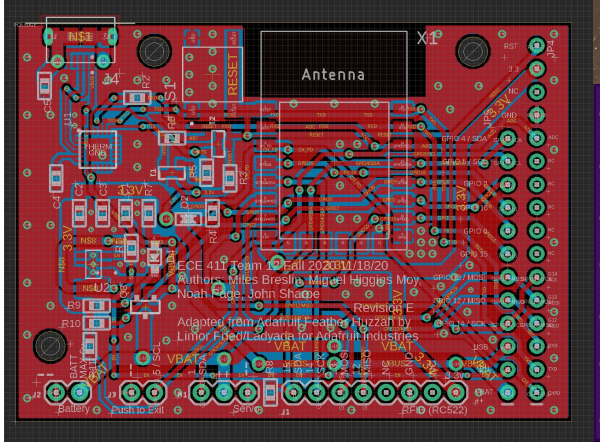
VERSION INFO

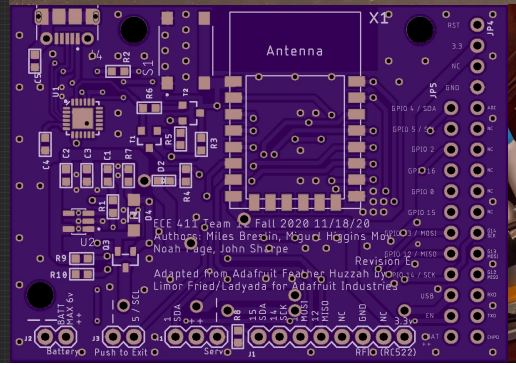
Rev	Date	Notes
1.0r0	2020-11-11	Initial BOM. Imported from Eagle and John's google sheet
1.0r1	2020-11-17	Verifying component parts and footprints to match distributor sizes. Replaced Eagle button with button spec'd here.
1.0r3	2020-12-01	Added 100nF Capacitor missing
1.0r4	2020-12-08	Final Revision including Physical items and batteries

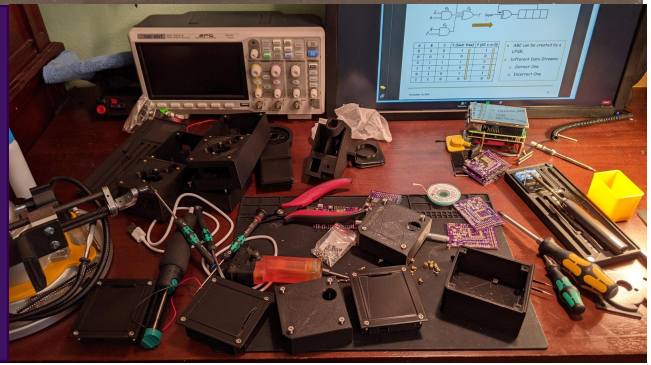

Processor: ESP12
Power: USB Port or 3x AA battery
Actuator: Servo (connects to GPIO pins)
Sensor: RFID RC-522
ECE 411 Team 12
Authors: Miles Breslin,
Miguel Higgins Moy,
Noah Page, John Sharpe
Revision F
Adapted from the Adafruit Feather Huzzah by
Limor Fried/Ladyada for Adafruit Industries
POWER
Battery MAX Voltage 6v
Designed for 3x AA battery holder
Battery Port
Linear Regulator
AP2112K-3.3TRG1
USB to Serial Converter
USB
CP2104
USB/UART BRIDGE
VIO: 1.8-VDD
Op. Temp: -40~85°C
RC522 RFID Sensor
Servo
Exit Button
PROCESSOR / IO
Reset Button
ESP12 Processor
Auto Reset
600mA/40V
GPIO0
RTS
DTR
RESET
TXD
RXD
USB_RX
ADC
GPIO14/SCK
GPIO13/MOSI
GPIO12/MISO
CH_PD
JP4






Antenna
X1
ECE 411 Team 12 Fall 2020 11/18/20
Authors: Miles Bresin, Miguel Higgins Moy
Noah Page, John Sharpe
Revision E
Adapted from Adafruit Feather Huzzah by
Limor Fried/Ladyada for Adafruit Industries
Battery
Push to Exit
Servo
RFID (RC522)


Antenna
X1
ECE 411 Team 12 Fall 2020 11/18/20
Authors: Miles Bresin, Miguel Higgins Moy
Noah Page, John Sharpe
Revision E
Adapted from Adafruit Feather Huzzah by
Limor Fried/Ladyada for Adafruit Industries
Battery
Push to Exit
Servo
RFID (RC522)



Tools Employed

Software

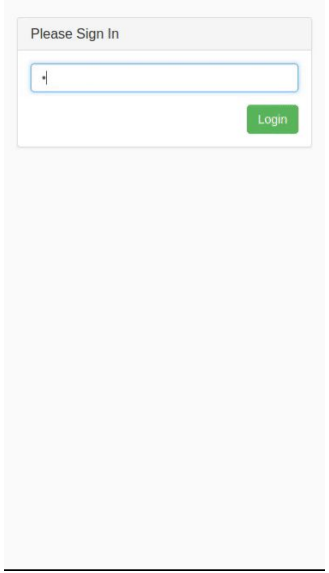
- GitHub
- VS Code
- Sublime Text
- PlatformIO toolchain
- Bootstrap Web Framework
- Gulp
- NPM package manager
- Autodesk Fusion 360
- Autodesk EagleCAD

Hardware

- 3D Printer
- Soldering Iron
- Reflow oven (it didn’t work)
- Hot air tool
- Screwdrivers (#1 Phillips, 2 and 2.5 hex)
- Misc soldering items

Frontend

The microcontroller will either host an ad-hoc WiFi access point or connect to an existing configured access point. As a host on the network, the device will host a web server on port 80 that contains a bootstrap based Single Page Application and a websocket. The single page application contains javascript code to authorize, connect, and issue commands to the hosted websocket. All data is sent via a plain-text HTTP/2 transport protocol and JSON formatted messages for the websocket. The client computer can then upload a new config file, view logged activities, view device state information, and register a new RFID card.



Backend

All web communications are handled asynchronously and hardware communications are handled using a synchronous idle loop. Initialization will read and parse a JSON config file on SPIFFS (a flash filesystem) if available, or assume first setup defaults. Given a servo activation (via Websocket command, RFID match, or physical button press), a static boolean will toggle to determine the frequency to send a PWM signal at. That signal will be at the configured duty cycle (as a integer, typically ranging from ~1ms to ~2ms depending on hardware) at 50hz using the Arduino *analogWrite* implementation and will last for the configured time (typically 500ms). The RFID chip is polled each loop via its driver, connecting to it over the Arduino SPY bus. This is all compiled to a generic ESP8266 firmware using PlatformIO.

Testing Outline

1 Integration Tests

- 1.1 Check components power usage with batteries on prototype
- 1.2 Check interface between RFID module and processor on prototype
- 1.3 Microcontroller outputs PWM signal when exit button is pressed
- 1.4 Check interface between processor and servo on prototype
- 1.5 Fully assembled PCB integration test

2 Functional Test

- 2.1 Servo arm 3D printed enclosure adapts to standard deadbolt
- 2.2 Test servo strength against real deadbolt
- 2.3 Test base code works by flashing forked binary onto processor
- 2.4 Test 3D parts fit together
- 2.5 Test fully assembled PCB USB power connection

3 Stress Testing

- 3.1 Battery life test

Test By Module

Referencing System Block Diagrams, revision 1.0

Power Supply

- 1.1 Check components power usage with batteries on prototype
- 2.5 Test fully assembled PCB USB power connection
- 3.1 Battery life test

RFID Sensor

- 1.2 Check interface between RFID module and processor on prototype
- 1.5 Fully assembled PCB integration test

Development Board

- 1.3 Microcontroller outputs PWM signal when exit button is pressed
- 1.4 Check interface between Processor and Servo on prototype
- 2.3 Test base code works by flashing forked binary onto processor

Servo

- 2.1 Servo arm has has an adapter allowing it to connect to deadbolt
- 2.2 Test Servo strength against real deadbolt

Locking Mechanism

- 2.4 3D Printed Items Fitment

Test Case Name:	Fully assembled PCB integration test		
Description:	Confirm functionality of assembled PCB with RFID sensor, exit button, and servo	Type:	White box
Tester Information			
Name of Tester:	John Sharpe	Date:	December 6, 2020
Hardware Version:	E	Time:	6:02 PM
Setup:	Connect RFID sensor and servo to assembled PCB, connect USB power to board		

Step	Actions:	Expected Result	Pass	Fail	N/A	Comments
1	Connect USB to board	Power light turns on		✓		LEDs flash on connection, but do not stay lit.
2	Press board reset button	Power light flashes for a few seconds, then remains a solid color		✓		LEDs flash on connection, but do not stay lit.
3	Measure voltage at 3.3 V pinouts with multimeter relative to board ground pin	Multimeter reads 3.3 V	✓			
4	Hold RFID card up to card reader	Servo arm rotates to open position	✓			
5	Swipe RFID card to reader again	Servo arm rotates to closed position	✓			
6	Press exit button	Servo arm rotates to open position	✓			
7	Press exit button again	Servo arm rotate to closed position	✓			
8	Toggle lock state from WebUI test button	Servo arm toggles between open/closed positions	✓			
Overall Test result:			✓			

Test Case Name:	3D Printed Items Fitment					
Description:	After each major version of CAD, check for fitment against other existing parts				**Type:**	White box
Tester Information						
Name of Tester:	Miguel				**Date:**	November 21, 2020
Hardware Version:	D				**Time:**	10:00 PM
Setup:	All parts are disassembled					

Step	Actions:	Expected Result	Pass	Fail	N/A	Comments
1	Assemble front plate with RFID	Plastic fits well together with no gaps		✓		RFID reader does not fit and bulges the plastic apart
2	Assemble rear lock portion (turn knob, override, plate, servo)	All portions fit together correctly	✓			
3	Connect both sides of the lock together while managing wire from the RFID	These fit snugly, but keep little overlap. The RFID wires should not interfere with the lock mechanism.		✓		Wires interfere with the locking mechanism and cause it to bind. Reassembling with interference in mind solved this, but possible to add a cable channel to prevent this.
4	Integrate electronics (manage the wires from the button and insert PCB)	The wires should have enough room to move when pulled, but will not move when the lock is actuated. PCB should fit the mounting holes.	✓			
5	Add covers (battery housing, button housing, battery cover)	Screw holes should line up and little gaps should be present. The battery cover should fit snugly, but be easy to remove.	✓			Trimming support material correctly is required for the battery cover to work correctly
6	Inspect entire product	Product should fit well together	✓			
Overall Test result:			✓			

Test Case Name:						
Description:	Doorlock all input/output tests				Type:	White box
Tester Information						
Name of Tester:					Date:	
Hardware Version:					Time:	
Setup:	Constructed doorlock system					
Step	**Actions:**	**Expected Result**	**Pass**	**Fail**	**N/A**	**Comments**
1	Lock door with key starting from unlocked position	Door locks	✓			
2	Unlock door with key starting from unlocked position	Door unlocks	✓			
3	Lock door with RFID card starting from unlocked position	Door locks	✓			
4	Unlock door with RFID card starting from locked position	Door unlocks	✓			
5	Lock door with exit button starting from locked position	Door locks	✓			
6	Unlock door with exit button starting from locked position	Door unlocks	✓			
Overall Test result:			✓			

Test Case Name:	3D Printed Items Fitment					
Description:	After each major version of CAD, check for fitment against other existing parts			Type:	White box	
Tester Information						
Name of Tester:	Miguel			Date:		
Hardware Version:	E			Time:		
Setup:	All parts are dissassembled					
Step	**Actions:**	**Expected Result**	**Pass**	**Fail**	**N/A**	**Comments**
1	Assemble front plate with RFID	Plastic fits well together with no gaps	✓			
2	Assemble rear lock portion (turn knob, override, plate, servo)	All portions fit together correctly	✓			
3	Connect both sides of the lock together while managing wire from the RFID	These fit snugly, but keep little overlap. The RFID wires should not interfere with the lock mechanism.	✓			
4	Integrate electronics (Manage the wires from the button and insert PCB)	The wires should have enough room to move when pulled, but will not move when the lock is accuated. PCB should fit the mounting holes.	✓			
5	Add covers (Battery housing, button housing, battery cover)	Screw holes should line up and little gaps be present. The battery cover should fit snugly, but be easy to remove.	✓			
6	Inspect entire product	Product should fit well together	✓			
Overall Test result:			✓			

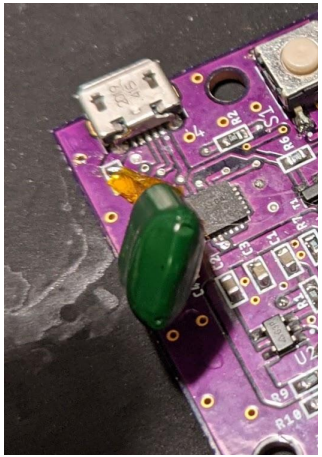
Test Case Name:	Servo Strength Test					
Description:	Ensure smooth operation of the servo and lock mechanism			Type:		
Tester Information						
Name of Tester:	Miguel			Date:		
Hardware Version:	D			Time:		
Setup:	Fully assembled hardware					
Step	Actions:	Expected Result	Pass	Fail	N/A	Comments
1	Use test button to actuate the lock	The lock actuates and does not hang up on anything		✓		Servo does not have enough power to move the lock
Overall Test result:						

Test Case Name:	Servo Strength Test					
Description:	Ensure smooth operation of the servo and lock mechanism			Type:		
Tester Information						
Name of Tester:	Miguel			Date:		
Hardware Version:	E			Time:		
Setup:	Fully assembled hardware					
Step	Actions:	Expected Result	Pass	Fail	N/A	Comments
1	Use test button to actuate the lock	The lock actuates and does not hang up on anything	✓			Removal of flat spring in lock mechanism allows the servo to accuate the lock without problem
Overall Test result:						

Results

What worked

- The product functions! Quite reliably.
- Software only required minor modifications
- The PCB worked first try! (sorta)



What's that green tumor doing there?
- Miguel forgot to add the right cap to the BOM

What didn't work

- Breadboard prototype kept on having brownouts when powering the servo due to the built in LDO overheating, or inadequate usb power supply
- Forgot 100nF capacitor on BOM and didn't order with the rest of the parts
- Battery life is non-existent at < 2 days.
- Compiling the code proved cumbersome making sure all dependencies were installed onto the virtual machine we used to run our builds.

Contributions

All of the group heavily participated in all of the homework assignments, making sure we all had a chance to look things over

John - Schematic design, model designer, homework starter/finisher

Miguel - Fusion 360 designer, Eagle, PCB solderer, 3D printing, parts distribution, Cyborg

Miles - Git resource (we ~~annoyed him with~~ asked him questions), setting up code building environment, code developer, IP rights management

Noah - Board layout, project management, scheduler (making sure we didn’t miss deadlines)

Lessons Learned

Even after quadruple checking the BOM, we didn’t realize the version we used for ordering did not match our schematic and missed the 100nF USB capacitor from our redesign of the USB port. Make sure to verify board/schematic version against BOM version when placing the order.

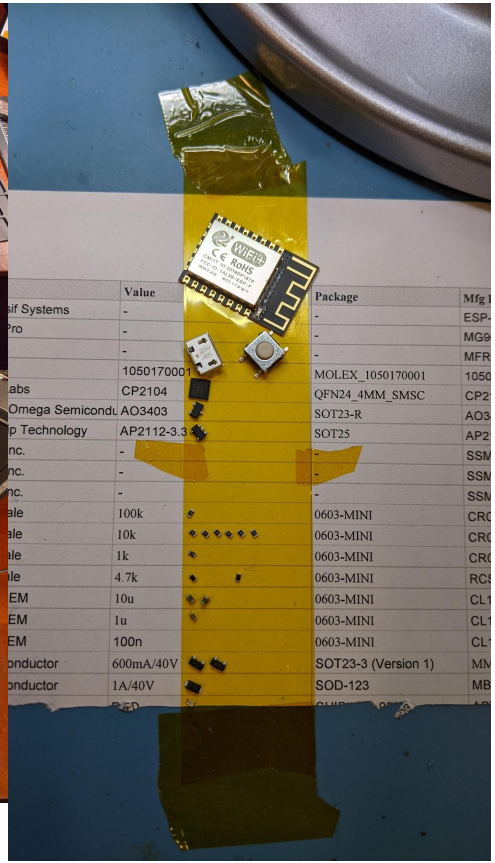
Inherited software comes with baggage. Several implementation decisions were poorly thought out. Development toolchains were difficult to install and sometimes failed without any error. Some dependencies were unmaintained. The open source community doesn’t always understand copyright law. Make sure to audit the codebase thoroughly before committing to using it.

COVID made collaboration fairly difficult requiring someone with the tools to manufacture and distribute. Everyone should have a nice homelab and/or the murder hornets could leave and never come back.

Live Demo

Backup video:
https://photos.app.goo.gl/rCKA4LcqwzgJmUdEA

Many versions… much distribute



Team Links

Team Organization Wiki

Frontend, Backend, Microcontroller Software

Board Layout and Schematic

Photos and Videos - Ordered Chronologically with Assembly Photos