Portal Box: Integration Test and Datasheet

Project Manager: Axel Andrews

Systems & Performance Engineer: James Powick

Design Engineers: Peter Beaudette, Kevin Duong, Aidan Flynn, Colton Jiorle

Client & Supporter: Matt Lamparter

Bucknell Senior Design Phase 4B Check-In

March 3, 2025

**Abstract**

This datasheet contains information related to the integration testing of the Portal Box 4.0. At this stage, our Portal Box replicates existing functionality seen in the MakerE with RPi-based Portal Boxes, but with an ESP32-based implementation. Through a connection with a database and webserver, the Portal Box verifies the identity of a user with an RFID card, then determines whether or not to provide power to the machine they are attempting to use. For machines that require power even when not actively running (i.e. laser cutter), the Portal Box uses a USB interlock to prevent or enable the transmission of data to the machine. The two main new features on the Portal Box 4.0 compared to past Portal Boxes are the ESP32-C6-DevKit running MicroPython, and an LCD screen for a clearer and more informative user experience. The Portal Box 4.0 connects to a MySQL database hosted on AWS RDS, and an Apache2 web server hosted in an AWS EC2 instance. These cloud deployments pave a pathway for broader expansion of Portal Boxes with their easily scalable nature. This integration test represents the point of Portal Box development where the core of the new iteration matches the existing deployments, with a clear path towards further development of new features.

**Architecture**

Our design replicates many elements of the prior Portal Box 2 and 3, with the PCB and code being very similar. The most notable change is the shift to the ESP32-C6 microcontroller, which comes with a change from CircuitPython to MicroPython. This change has significantly decreased the power consumption of our Portal Box, as well as the cost. The ESP32–C6-DevKit that we are using only costs $9, compared to $35 for the Raspberry Pi 4 used in the 3.0. Both of these benefits support our goals of decreasing the power consumption and cost of Portal Boxes, making them more sustainable and affordable for makerspaces to deploy. The other apparent change from prior Portal Boxes is the addition of the 16x2 LCD RGB, which for the purposes of this test will illustrate the active state of the finite state machine operating on the microcontroller. Physical inputs are received through the RFID card reader and the keypad, which we currently have configured to simply serve as the pushbutton on the existing Portal Boxes. As we reiterate, our team will evaluate the utility and user reception of adding a keypad for two-factor authentication with the RFID card, which would seek to prevent the sharing or theft of RFID cards. Shown in Appendix G is our block diagram.

Table 1: Portal Box Inputs and their Descriptions

| **Inputs** | **Description** |
| --- | --- |
| User Card | User insert RFID card into dedicated spot on the Portal Box enclosure |
| User PIN | User inputs their respective personal identification number using the keypad |
| Electrical Receptacle Power | Power coming from the wall outlet in a room, 120 VAC |
| Control Computer | USB data sent via a cable, provides information about how to control certain machines from a corresponding software interface |
| Database | HTTP responses provide information (Appendix D) |

Table 2: Portal Box Outputs and their Descriptions

| **Outputs** | **Description** |
| --- | --- |
| LCD Screen | Displays messages providing user information, time spent on machine, and updates on Portal Box status |
| LEDs | Emits different color light that depends on the status of the Portal Box |
| Buzzer | Emits different sounds that depends on the status of the Portal Box |
| Machine Power | 120 VAC power delivered to the machine |
| Machine Data | USB data from a computer which contains information and commands for how to control a given machine |
| Database | HTTP requests ask for information from database (Appendix D) |

Table 3: Limits to Current System

| **Limits to the System** | **Description** |
| --- | --- |
| RFID Reading Distance | The maximum distance the RFID reader can read from is 5 cm. |
| LCD Screen Size | The LCD screen only has 2 rows of 16 characters. |
| PCB Size (Smallest Possible Box) | The smallest size of the enclosure is limited to the size of the PCB. |
| Local Storage | The ESP32-C6 only has 4-6 MB of flash memory. |

**Behavior**

The Portal Box cycles through different states according to the inputs it receives to safely and efficiently provide access to machines in a makerspace. The Portal Box is designed to balance access and security in makerspaces while ensuring seamless power delivery. The Portal Box system allows only authorized users to enable power for machines while preventing unauthorized access. The system follows a structured flow: an authentication request is processed, validated, and, if successful, power is granted. If authentication fails, access is denied, and the event is logged for security tracking.

**Performance Metrics & Testing Procedures**

**FSM Flow:** The purpose of the Portal Box is to seamlessly flow through different states in order to provide access to a machine. Measuring the performance of its ability to manage each state is imperative to the final portal box.

**Testing Procedure:** Full end to end testing to reach all states. (Full procedure and results in Appendix)

**Power Delivery Efficiency**: Ensuring stable and quick power delivery is crucial for user experience and system reliability. If power activation is delayed or unstable, it may affect machine performance and user trust.

* **Testing Procedure:**
  + Authenticate an authorized user using a valid card.
  + Measure the time taken for power to stabilize from the moment of authorization.
  + Record voltage levels at the machine’s input terminals.
  + Press the button to end the session.
  + Measure the time taken for power to return to zero from the moment of button press.
  + Repeat the test multiple times to identify any inconsistencies.
* **Expected Metrics:**
  + Power stabilization time should be less than 1 second.
  + Voltage should match the expected supply level within ±5%.

**Reliability**: The system must consistently activate power for authorized users while preventing unauthorized access. Any failures must be minimal.

* **Testing Procedure:**
  + Only use authorized user cards.
  + Track activations over a test period.
  + Log and compare successful vs. failed activations.
* **Expected Metrics:**
  + Success rate of at least 99%.

**Power Consumption**: The Portal Box should operate efficiently without excessive energy consumption.

* **Testing Procedure:**
  + Measure the system’s current draw under normal operation.
  + Compare power consumption to previous versions of the system.
* **Expected Metrics:**
  + The system should draw less than 1.2 A of current.
  + Total power consumption should be lower than prior implementations.

**Startup and Recovery Timing**: The system must start up and recover from faults quickly.

* **Testing Procedure:**
  + Observe and log how the system handles faults.
  + Check if the system returns to normal function upon receiving correct inputs.
* **Expected Metrics:**
  + The system should provide clear error messages or logs.
  + Recovery time after a fault should be under 30 seconds.

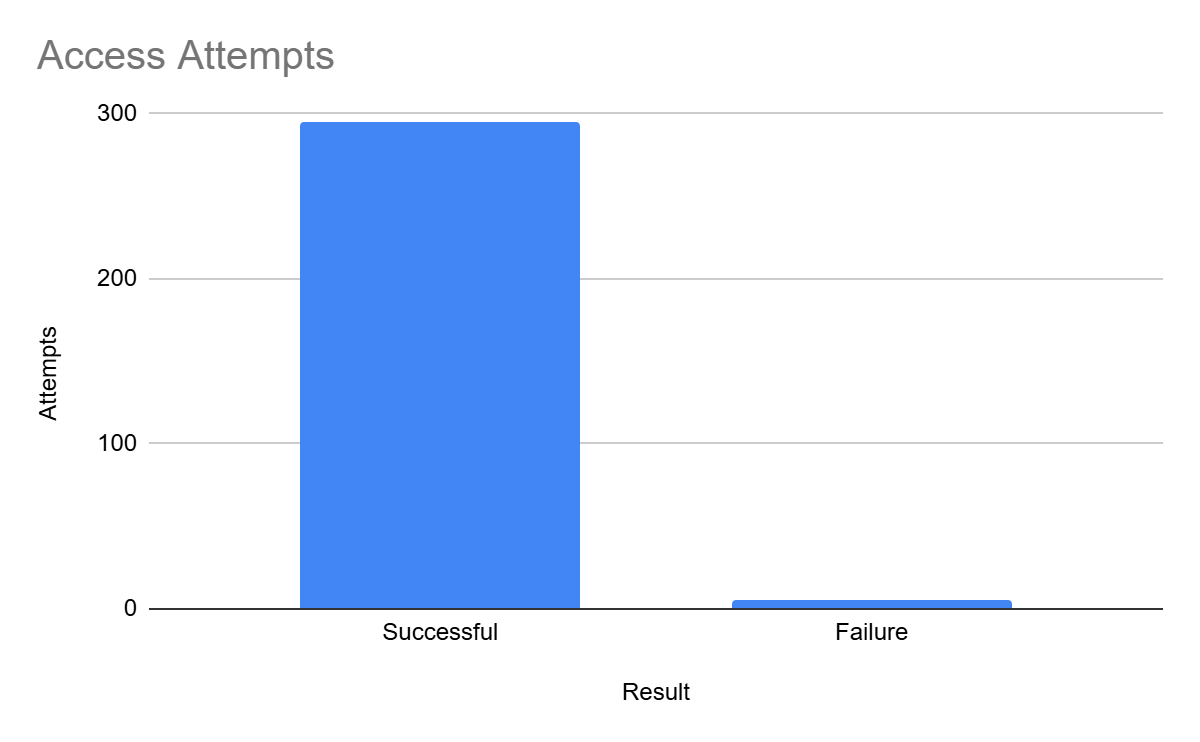
**Performance Data**

**Power Delivery Efficiency**

| **Measurement** | **Value** | **Description** |
| --- | --- | --- |
| RMS Voltage | 125 V | Measured AC voltage |
| Rise Time (0V → 125V) | 560 µs | Time to reach stable voltage |
| Fall Time (125V → 0V) | 65 ms | Time for voltage to decrease to 0 |

Conclusion: The power delivery efficiency test revealed that the Portal Box reaches a stable output of 120V ± 5% in approximately 560 microseconds after being switched on, indicating a rapid response time. Upon switching off, the voltage decay to 0V took approximately 65 milliseconds, suggesting a slower discharge process likely influenced by residual capacitance or inductance in the circuit. These results demonstrate efficient voltage stabilization on power up and a controlled voltage drop on shutdown.

**Reliability**



Conclusion: During the reliability test, we attempted 300 authorization events using the same RFID card over the course of a day. 297 attempts were successful and 3 were unsuccessful. This gives us a success rate of 99%. Failed authorization attempts were due to the RFID card being around 5cm from the RFID scanner.

**Power Consumption**

| **Attempt** | **Current (mA)** | **Power (mW)** |
| --- | --- | --- |
| 1 | 318 | 1590 |
| 2 | 316 | 1580 |
| 3 | 316 | 1580 |
| 4 | 315 | 1575 |
| 5 | 315 | 1575 |
| 6 | 315 | 1575 |
| 7 | 315 | 1575 |
| 8 | 315 | 1575 |
| 9 | 315 | 1575 |
| 10 | 315 | 1575 |

Average value of current: 315.5 mA

Average instantaneous power: 1577.5 mW

Conclusion: Testing the Portal Boxes power consumption revealed that the Portal Box has a very negligible power usage, compared to previous Raspberry Pi based Portal Boxes. This is due to the fact that the ESP32 microcontroller isn’t as power hungry as the RPi. The current draw of Portal Box is less than half that of the RPi version.

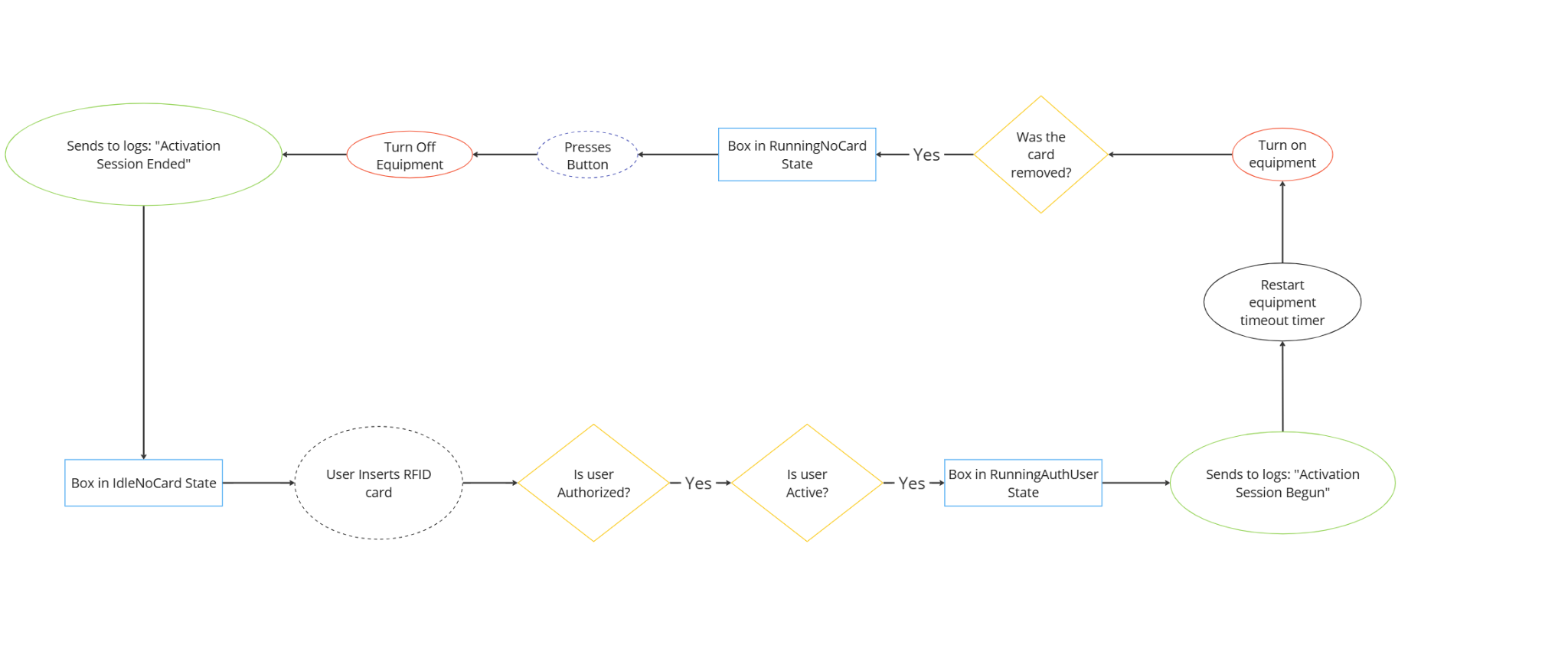
**Startup and Recovery Timing**

| **Fault** | **Expected Recovery** | **Actual Recovery** | **Recovery Time** |
| --- | --- | --- | --- |
| Power Loss (Idle State) | Reboot, PB returns to idle state | Reboot, PB returns to idle state | ~ 13 seconds |
| Power Loss (Running State) | Reboot, PB returns to running state | Reboot, PB returns to running state | ~ 15 seconds |
| Reset Button Pressed | Reboot, PB returns to idle state | Reboot, PB returns to idle state | ~ 15 seconds |
| Shutdown Card Activated | PB enters shutdown state | PB enters shutdown state | < 1 second |

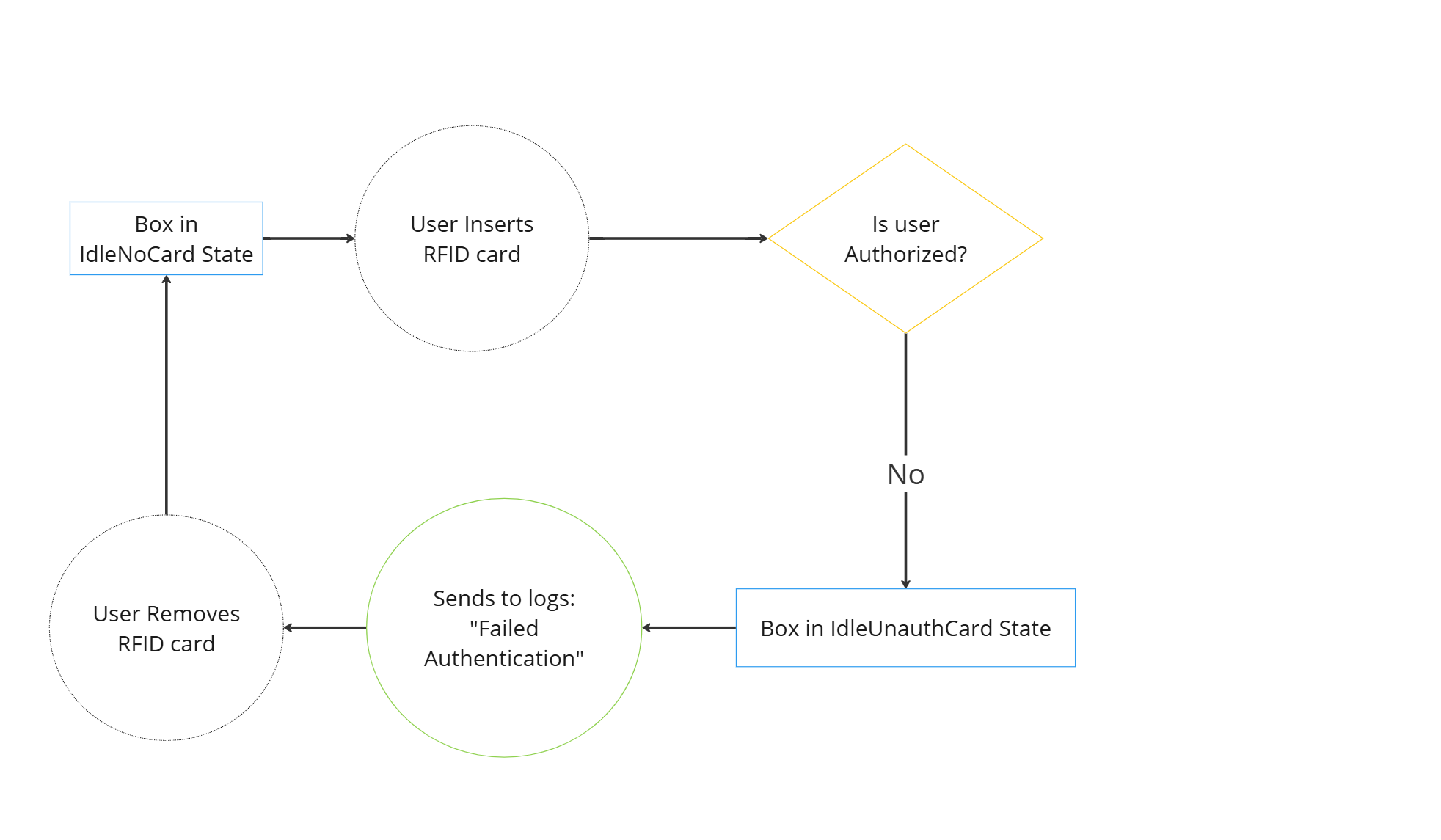
Conclusion: The fault detection and recovery test for the Portal Box demonstrates that the system reliably reboots and returns to its expected state following power loss, reset, or shutdown event. Power loss in both idle and running states resulted in recovery times of 13–15 seconds, while a manual reset required a similar recovery period. Notably, the shutdown card triggered an almost immediate shutdown. All reboot/recovery times are faster than those of RPi based Portal Boxes.

**Appendix**

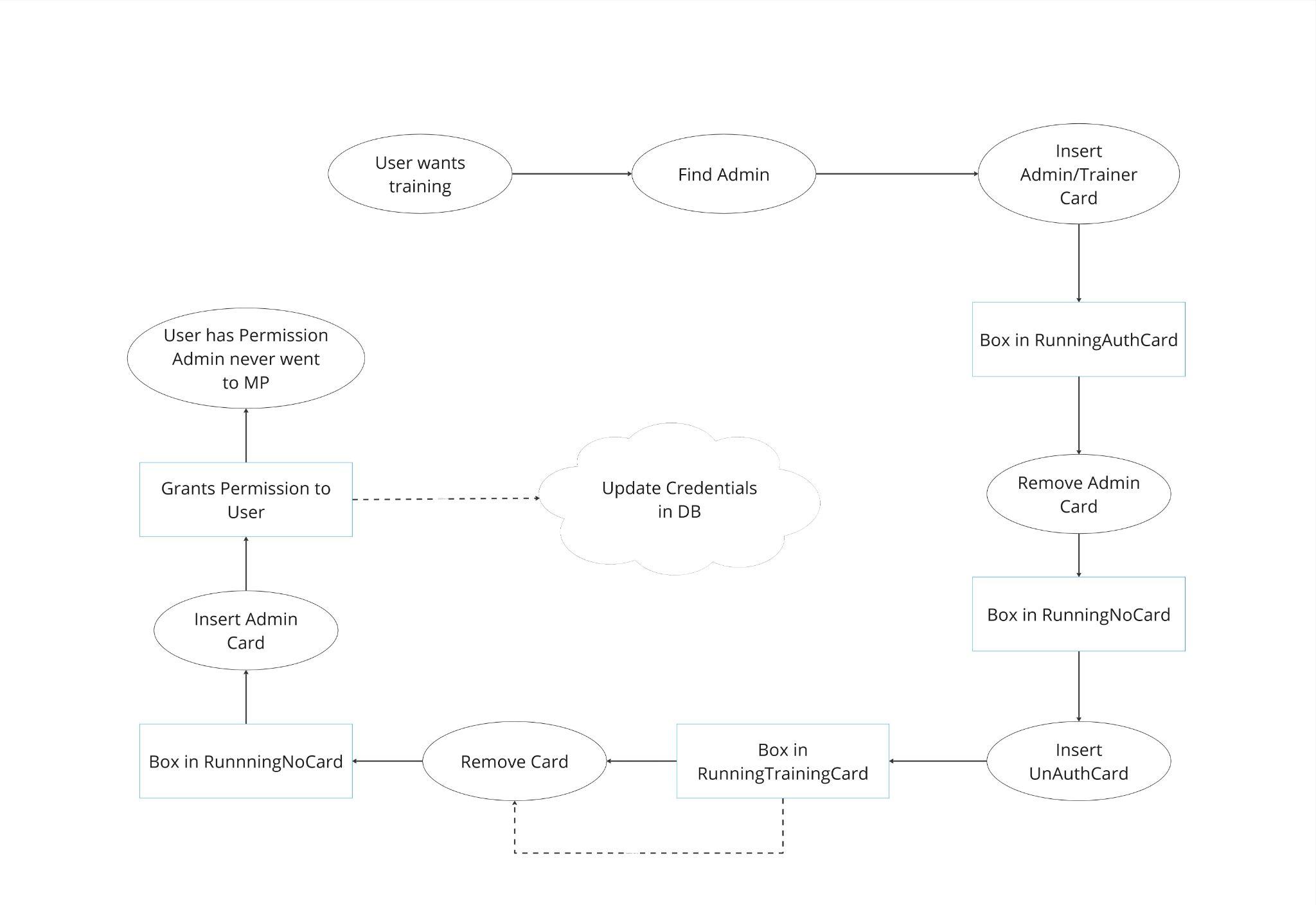
Appendix A: Isolated flow diagram of an authorized user using the Portal Box



Appendix B: Isolated flow diagram of an unauthorized user using the Portal Box



Appendix C: Isolated flow diagram of the new training flow



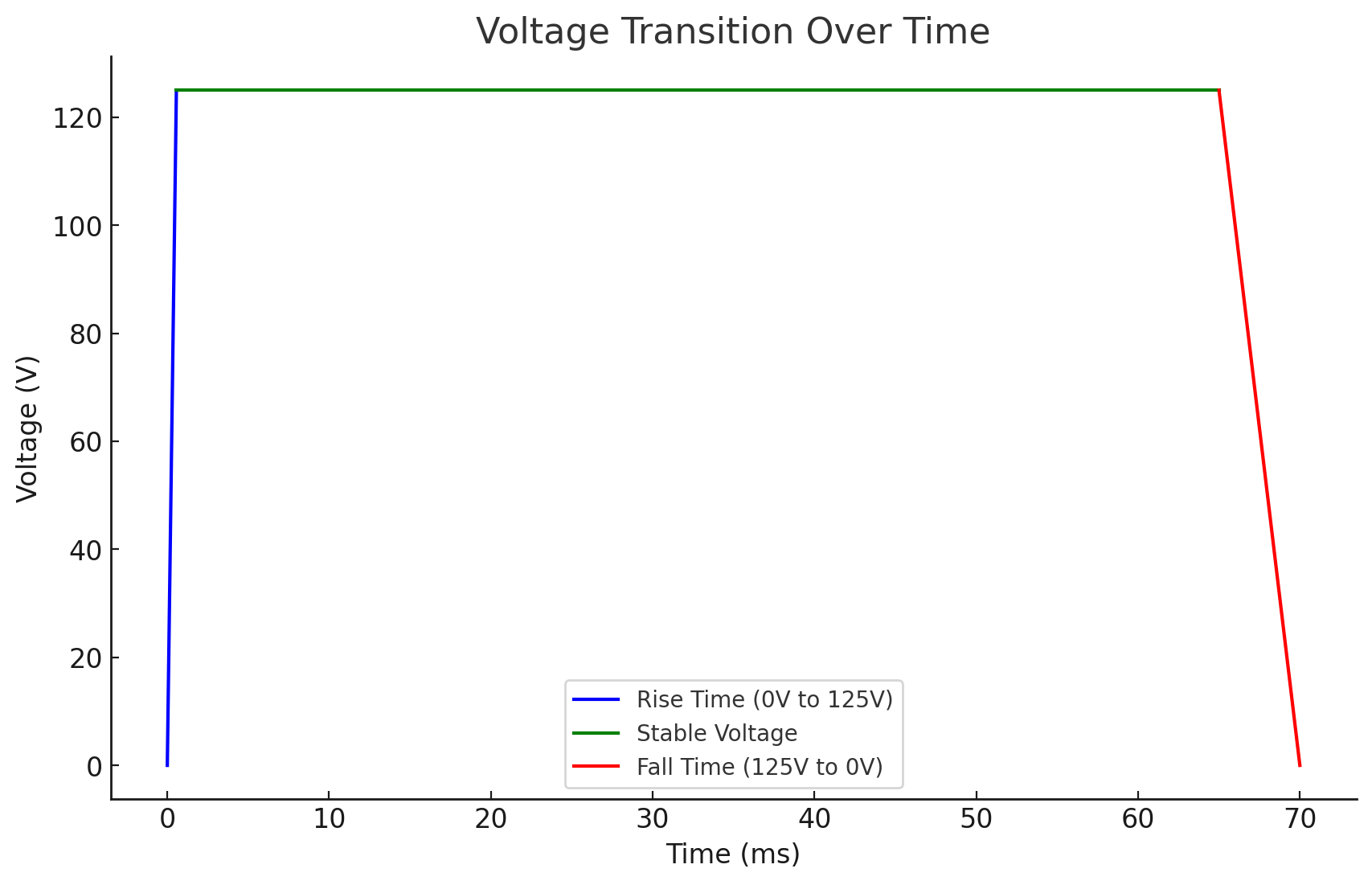
Appendix D: Portal Box API

| **HTTP Method** | **API Mode** | **Parameters** | **Response** |
| --- | --- | --- | --- |
| GET | check\_reg | mac\_adr: MAC address of the device | Integer (1 if registered, 0 if not) |
| PUT | register | mac\_adr: MAC address of the device | Boolean (success status) |
| GET | get\_profile | mac\_adr: MAC address of the device | JSON array with equipment profile: [{"id", "type\_id", "name"[0], "location\_id", "name"[1], "timeout", "allow\_proxy", "requires\_training", "charge\_policy"}] |
| POST | log\_started\_status | equipment\_id: ID of the equipment | Success status |
| POST | log\_shutdown\_status | equipment\_id: ID of the equipment  card\_id: ID of the card used (or 0 if not applicable) | Success status |
| POST | log\_access\_attempt | equipment\_id: ID of the equipment  `card\_id`: ID of the card used  successful: 1 if successful, 0 if not | Success status |
| POST | log\_access\_completion | equipment\_id: ID of the equipment  card\_id: ID of the card used | Success status |
| GET | get\_card\_details | card\_id: ID of the card  equipment\_id: ID of the equipment type | JSON array with card/user details: [{"user\_balance", "user\_auth", "user\_active", "card\_type", "user\_role"}] |
| GET | get\_user | card\_id: ID of the card | JSON array with user details: [{"name", "email"}] |
| GET | get\_equipment\_name | equipment\_id: ID of the equipment | JSON array with equipment name: [{"name"}] |
| POST | record\_ip | equipment\_id: ID of the equipment  ip\_address: IP address of the device | Success status |

Appendix E: FSM flow verification list

1. Show config.json values to set expectations for outputs
2. Connect ESP32 to serial port for console outputs
3. Shutdown/Reset Button Testing
   1. Remove power from PB for 5 seconds, then reconnect
      1. Box displays “IdleNoCard” after setup within 15 seconds Pass
   2. Press shutdown button
      1. Box displays “Shutting Down…” within 1 second Pass
      2. Use console to verify PB OS has shutdown Pass
   3. Press shutdown button
      1. Box displays “IdleNoCard” after setup within 15 seconds Pass
      2. Verify PB reads correct config.ini equipment info by reading system log ␨
4. Proxy Card Inserted While Idle
   1. Start in “IdleNoCard” state
   2. Insert proxy card
      1. Verify PB displays “IdleUnauthCard” state Pass
      2. Verify equipment power off, USB data blocked Pass
      3. Verify system log shows the unauthorized card Pass
      4. Verify web portal shows unauthorized access with card ID Pass
   3. Remove card
      1. Verify PB displays “IdleNoCard” state Pass
5. User Card Inserted When Idle, User Not Authorized for Equipment
   1. Start in “IdleNoCard” state
   2. Insert valid user card (not authorized for equipment)
      1. Verify PB displays “IdleUnauthCard” state Pass
   3. Remove card
      1. Verify PB in “IdleNoCard” Pass
6. User Card Inserted When Idle, Authorized but Inactive User
   1. Start in “IdleNoCard”
   2. Insert card
      1. Verify PB enters “IdleUnauthCard” with message Pass
   3. Remove card
      1. Verify PB in “IdleNoCard” Pass
7. Unknown Card Inserted While Idle
   1. Start in “IdleNoCard”
   2. Insert card not in MySQL database or local storage
      1. Verify PB in “IdleUnauthCard” Pass
   3. Remove card
      1. Verify PB in “IdleNoCard” Pass
8. Modify configuration on Makerportal to have a timeout.
9. Shutdown Card & Obtaining Config from MySQL Database
   1. Insert Shutdown card
      1. Verify PB shows “Shutting Down…” Pass
      2. Verify ESP32 is shutdown with console Pass
      3. Verify USB interlock is blocking data Pass
      4. Verify power to equipment is off Pass
   2. Remove card
   3. Press shutdown/reset button
      1. Verify PB returns to “IdleNoCard” Pass
      2. Verify equipment profile is read correctly with serial system log Pass
10. Authorized User Ends Access by Removing Card & Pressing Button
    1. Start in “IdleNoCard”
    2. Insert active, authorized card
       1. Verify PB in “RunningAuthUser” Pass
       2. Verify equipment power on Pass
       3. Verify USB data enabled Pass
       4. Verify web portal shows authorized access with card ID Pass
    3. Remove card
       1. Verify PB in “RunningNoCard” Pass
       2. Verify PB is beeping Pass
       3. Verify power to machine and USB data enabled Pass
       4. FUTURE: Show interactive message on screen
    4. Press button on keypad
       1. Verify PB in “IdleNoCard” Pass
       2. Verify power & USB data disabled Pass
11. Authorized User Ends Access With Equipment Timeout
    1. Start in “IdleNoCard”
    2. Insert active, authorized card
    3. Verify PB in “RunningAuthUser” Pass
    4. Wait one minute
    5. Remove card
       1. Verify PB in "RunningNoCard" Pass
    6. Reinsert card before grace period ends
       1. Verify PB in "RunningAuthUser" Pass
       2. Verify PB enters "RunningTimeout" only after a period of time equal to the equipment timeout (from last insertion of card) Pass
       3. Before grace period ends, verify equipment power and USB data are enabled Pass
       4. Before grace period ends, verify PB is beeping Pass
    7. Before grace period ends, press button
       1. Verify PB returns to "RunningAuthUser" Pass
       2. Verify PB moves to "RunningTimeout" after a period equal to configured equipment timeout Pass
       3. Verify PB moves to "IdleAuthCard" after a period equal to the configured grace period Pass
       4. Verify equipment power, USB data both disabled Pass
       5. Verify web portal shows end of authorized access with card ID Pass
       6. Verify that an email is sent to the user TBD
    8. Remove user card
       1. Verify PB moves to "IdleNoCard" Pass
12. Authorized Trainer to New User Card
    1. Start in "IdleNoCard"
    2. Insert authorized trainer card
       1. Verify PB in "RunningAuthUser" Pass
    3. Remove trainer card
       1. Verify PB in "RunningNoCard" Pass
    4. Insert unauthorized user card
       1. Verify that PB enters "RunningTrainingCard" Pass
       2. Verify web portal shows authorized access with card ID Pass
       3. Verify that equipment power and USB data are enabled Pass
    5. Wait 30 seconds
    6. Remove the unauthorized user card
       1. Verify that PB enters "RunningNoCard" Pass
    7. Reinsert the unauthorized user card before grace period ends
       1. Verify that PB enters "RunningTrainingCard" Pass
       2. After time period equivalent to equipment timeout, verify that PB displays "RunningTimeout" Pass
    8. Before grace period expires, press button
       1. Verify PB enters "RunningTrainingCard" Pass
       2. After time period equivalent to equipment timeout, verify PB enters "RunningTimeout" Pass
    9. Before grace period expires, remove user card
       1. Verify PB enters "IdleNoCard" Pass
       2. Verify equipment power + USB data are disabled Pass
       3. Verify web portal shows end of authorized card access with card ID Pass
13. Admin to New User Card, Insert Unknown Card During "RunningNoCard"
    1. Start in "IdleNoCard"
    2. Insert admin card
       1. Verify PB in "RunningAuthUser" Pass
    3. Remove admin card
       1. Verify PB in "RunningNoCard" Pass
    4. Insert unauthorized card
       1. Verify PB in "RunningTrainingCard" Pass
    5. Remove unauthorized card
       1. Verify PB in "RunningNoCard" Pass
    6. Before grace period expires, insert unknown RFID card
       1. Verify PB continues "RunningNoCard" Pass
    7. Press button before grace period ends
       1. Verify PB in "IdleNoCard" Pass
14. Authorized User Attempts to go to New User Card
    1. Start in "IdleNoCard"
    2. Insert active, authorized, non trainer/admin card
       1. Verify PB in "RunningAuthUser" Pass
    3. Remove authorized card
       1. Verify PB in "RunningNoCard" Pass
    4. Before grace period ends, insert unauthorized user card
       1. Verify PB remains in "RunningNoCard" Pass
    5. Before grace period ends, remove unauthorized card
       1. After grace period, verify PB in "IdleNoCard" Pass
15. Authorized User to Proxy Card
    1. Start in "IdleNoCard"
    2. Insert active, authorized user card
       1. Verify PB in "RunningAuthUser" Pass
    3. Wait 30 seconds
    4. Remove user card
       1. Verify PB in "RunningNoCard" Pass
    5. Before grace period ends, insert proxy card
       1. Verify PB in "RunningProxyCard" Pass
    6. Remove proxy card
       1. Verify PB in "RunningNoCard" Pass
    7. Insert proxy card
       1. After time period equivalent to equipment timeout, verify PB in "RunningTimeout" Pass
    8. Before grace period expires, press button
       1. Verify PB in "RunningProxyCard" Pass
       2. After period equal to equipment timeout, verify PB in "RunningTimeout" Pass
    9. Before grace period expires, remove proxy card
       1. Verify PB in "IdleNoCard" Pass
       2. Verify web portal shows end of authorized access with original card ID Pass
16. Verify Transitions from "RunningNoCard" if Entered From "RunningProxyCard"
    1. Start in "IdleNoCard"
    2. Insert active, authorized card
       1. Verify PB in "RunningAuthUser" Pass
    3. Remove user card
       1. Verify PB in "RunningNoCard" Pass
    4. Before grace period expires, insert a proxy card
       1. Verify PB in "RunningProxyCard" Pass
    5. Remove proxy card
       1. Verify PB in "RunningNoCard" Pass
    6. Before grace period expires, reinsert proxy card
       1. Verify PB in "RunningProxyCard" Pass
    7. Remove proxy card
       1. Verify PB in "RunningNoCard" Pass
    8. Before grace period ends, insert unauthorized user card
    9. Wait 3 seconds
       1. Verify PB in "RunningNoCard" Pass
    10. Press button (message)
17. Verify Transitions from "RunningNoCard" if Entered from "RunningTrainingCard"
    1. Start in "IdleNoCard"
    2. Insert active, authorized, trainer card
       1. Verify PB in "RunningAuthUser" Pass
    3. Remove user card
       1. Verify PB in "RunningNoCard" Pass
    4. Insert unauthorized user card before grace period ends
       1. Verify PB in "RunningTrainingCard" Pass
    5. Remove unauthorized user card
       1. Verify PB in "RunningNoCard" Pass
    6. Reinsert the same unauthorized user card before grace period ends
       1. Verify PB in "RunningTrainingCard" Pass
    7. Remove unauthorized user card
       1. Verify PB in "RunningNoCard"
    8. Reinsert a different unauthorized user card before grace period ends
    9. Wait 3 seconds
       1. Verify PB in "RunningNoCard" Pass
    10. Insert a proxy card
        1. Verify PB in "RunningNoCard" Pass
    11. Press button
18. Verify user with positive balance for equipment with a charge TBD
19. Verify user with zero balance for equipment with a charge TBD
20. Grace period is reset when entering "RunningTimeout"
    1. Start in "IdleNoCard"
    2. Insert active, authorized user card
       1. Verify PB in "RunningAuthUser" Pass
    3. Wait at least 30 seconds
    4. Remove user card
       1. Verify PB in "RunningNoCard" Pass
    5. Wait about half of grace period
    6. Insert the user card
       1. Verify PB in "RunningAuthUser" Pass
       2. Verify PB in "RunningTimeout" only after equipment timeout period Pass
       3. Verify PB remains in "RunningTimeout" for time equivalent to grace period Pass
       4. Verify PB returns to "IdleNoCard" Pass

Appendix F: Voltage Stabilization Graph



Appendix G: L2 Block Diagram

