# Universal Bike Sharing Lock

Group 16
Patrick ODonnell,
Armin Mohammadi, Jihoon Lee

# Problem

- Bike sharing services increasing in popularity in urban areas (Chicago, NY, etc.)

- No similar services for mopeds or scooters (faster than bikes, cleaner than cars)

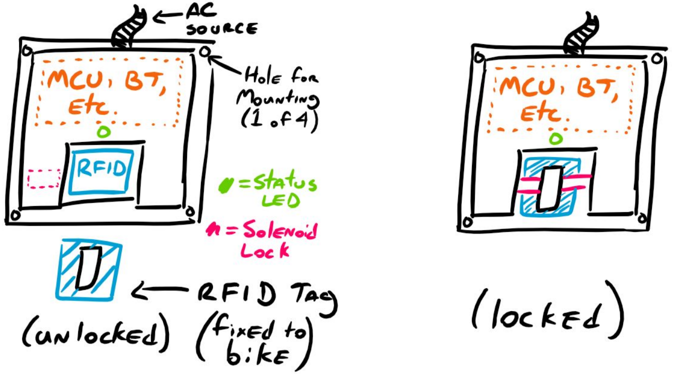
- How to widen scope of a bike-sharing service?

# Solution

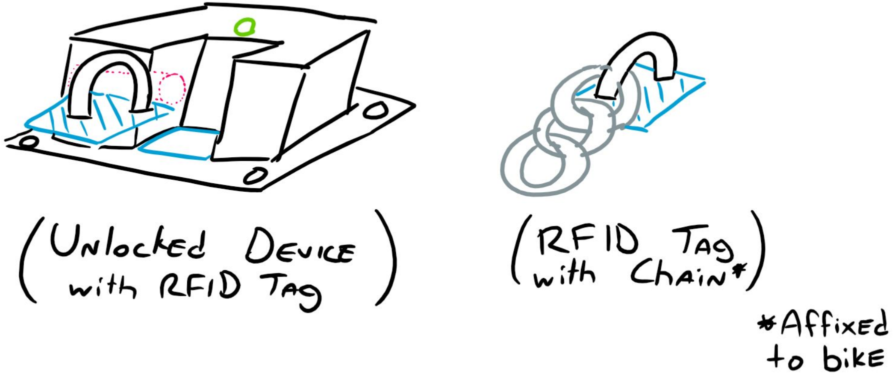

- Create a **universal bike sharing lock**
  - Allow bikes, scooters, and mopeds be shared in rental service
  - No custom bikes needed
  - Emulate similar services like Divvy and Zipcar

- Main Requirements
  - Quick, easy unlocking of bike for rental
  - Fast, secure locking of bike for return
  - Device must have constant source of power to preserve security of bike

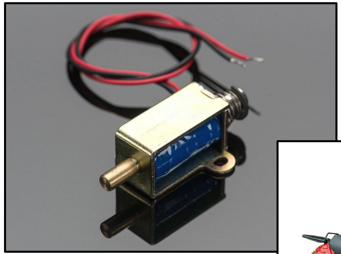
## Top View




## Angle View



# Design

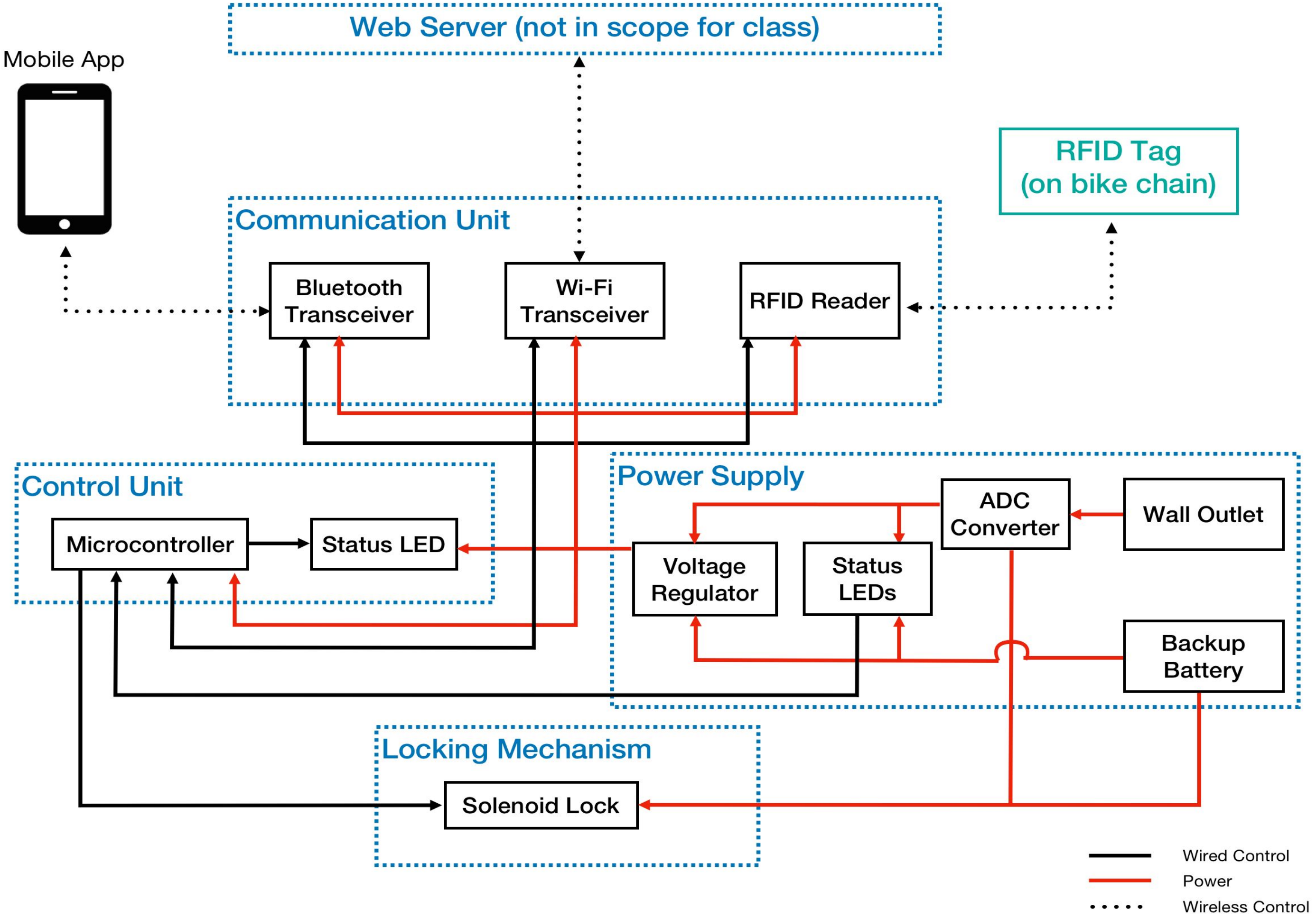


- Locking Mechanism
  - Designed with push-pull solenoid
    - Opens and closes based on current flowing through it
- Power Block
  - Convert AC power from outlet to DC
    - Drop down to appropriate levels for locking mechanism, control unit, etc.
- Control Block
  - Executes logic for renting/returning a bike (i.e. when to lock and unlock)
- Communication Block
  - Bluetooth for rental, RFID for return, Wi-Fi for sending data to server

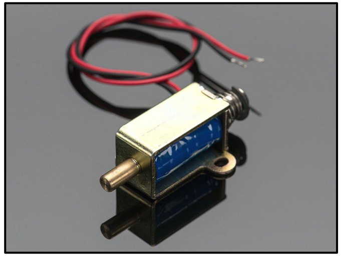
# Challenges

- Power
  - All components need to be powered at all times (for security and proper functionality)
  - Different components need different levels of power

- System integration
  - Need all components to work together properly
    - Communicate between components


Web Server (not in scope for class)
Mobile App
RFID Tag
(on bike chain)
Communication Unit
Bluetooth Transceiver
Wi-Fi Transceiver
RFID Reader
Control Unit
Microcontroller
Status LED
Power Supply
ADC Converter
Wall Outlet
Voltage Regulator
Status LEDs
Backup Battery
Locking Mechanism
Solenoid Lock
Wired Control
Power
Wireless Control

# Locking Mechanism



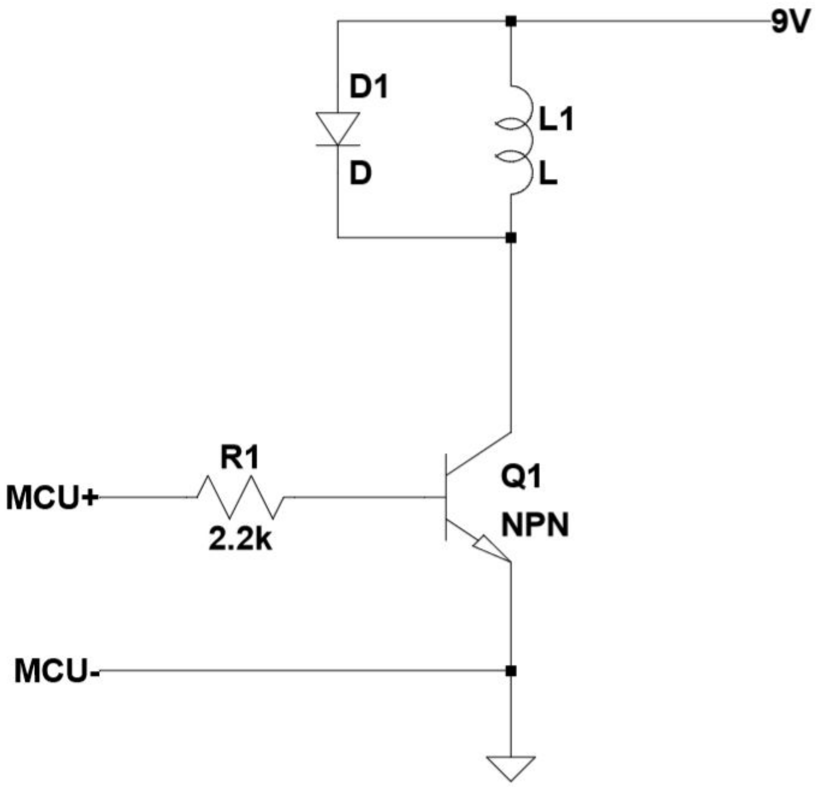
- Automatically controlled
  - Using **push-pull solenoid**
    - Pushes and pulls based on current flow
    - Chosen to guarantee proper locking/unlocking
      - Instant opening and closing
- Use BJT-Diode circuit to toggle solenoid on and off with constant voltage
  - Driven by MCU I/O pin

# Solenoid

- Needs 6.5-12V and 1A of current to operate swiftly and properly
    - Requires extra 250 mA to initially drive solenoid

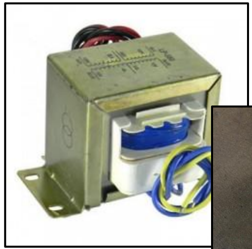
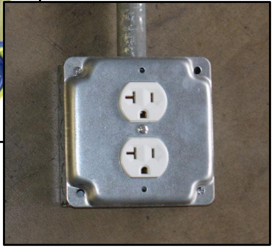
- Stays locked (pushed out) until voltage drops below 3.11V

- Designed to prevent bike chain/wire from being detached from device when fully installed in body




Solenoid Control Circuit

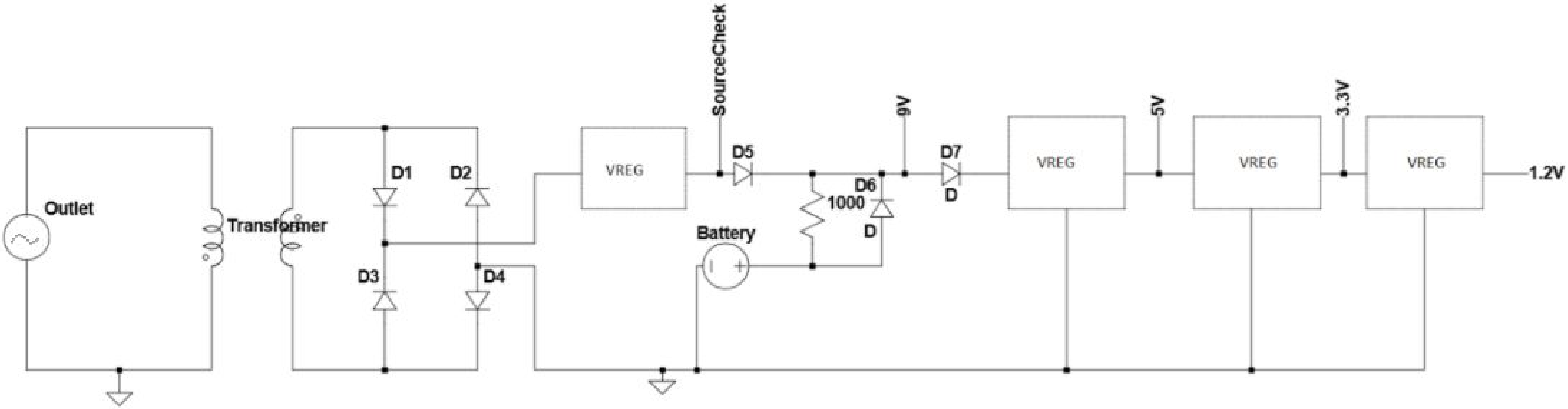
# Power

- Integral to security of bike

- Power needs for device:
    - Solenoid lock needs ~ 9V with 1A when driven
    - Various control devices (MCU, Wi-Fi, RFID, etc.) need 1.2 - 5V each

- Implementation
    - Constant power source from wall outlet
    - Use automatic backup battery for temporary use (for emergency power loss)
        - Used NiMH rechargeable 9V batteries
        - MCU detects power cut and sends alert over Wi-Fi to server
        - Put control unit in low-power mode to extend battery life




Power Conversion Circuit

# Power R&V + Results

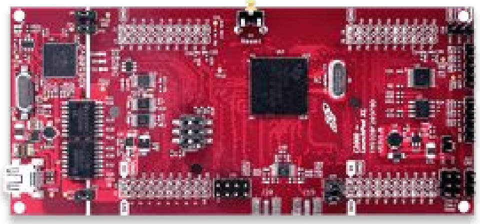
- Voltage Regulation
  - 

| Desired (V) | Measured (V) |
|---|---|
| 9.1 | 9.105 |
| 5 | 4.998 |
| 3.3 | 3.305 |
| 1.2 | 1.25 |

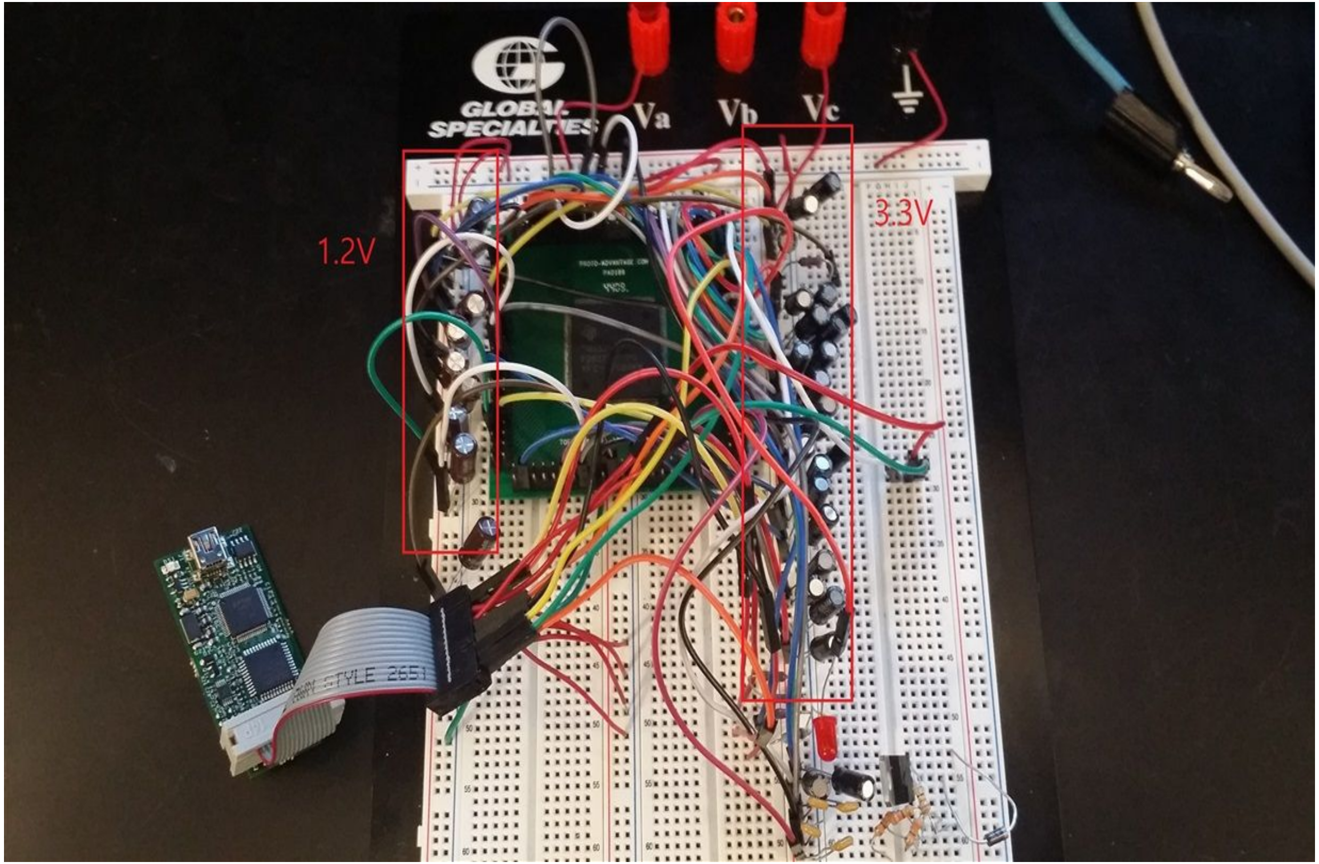

- Battery Recharged at a rate of 0.1mA

- Battery would last ~40 minutes
  - Discharge at a rate of 1mV/s
  - Locking mechanism uses more power when opening and closing
    - Drains current battery faster

# Control Unit



- Center of all functionality
  - Drives solenoid lock
  - Handles information to/from communication unit
    - RFID, Bluetooth, Wi-Fi
- Microcontroller: **TI C2000 Delfino Dual-Core MCU**
  - Maximum 200MHz operation frequency
  - Supply voltage: 1.2V (core), 3.3V (I/O)
  - Various connectivity options (UART, SPI, USB, CAN, I2C)
  - 1MB internal flash memory
  - 400 MIPS per cpu, 800 MIPS total




MCU on Breakout Board

# Control Unit R&V

- UART communication with Bluetooth module and Wi-Fi module with Baud rate of 9600
  - Verification: Check serial terminal data output
- Activation of low power mode when power supply fails ( less than 200mA core current consumption )
  - Verification: Measure current through the core power supply
- Server is notified of power failure within 20 seconds
  - Verification: Time duration from DC disconnect to server notification

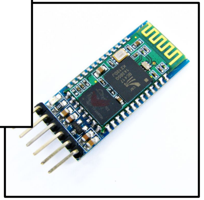
# Control Unit Results

- Successful printout of data from MCU (both over Wi-Fi and Bluetooth serial terminal)

- Core current consumption decreased from 187mA to 155mA
  - Static Power consumption (V * I): 0.224mW -> 0.186mW
  - Idle mode reduces 5mA of additional current draw

- Server instantly received power failure system message from MCU (w/ Wi-Fi) when DC source disconnected

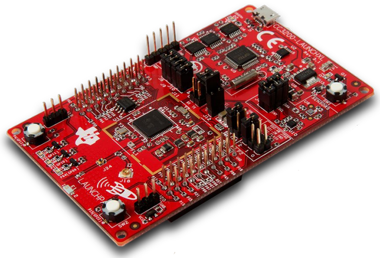
# Communication Unit

- Bluetooth
  - For remote unlocking when renting
  - Initially planned to use TI CC2640R2F Bluetooth chip
  - **HC-05 Module**
    - Wireless Bluetooth serial transceiver
    - 1 UART channel

- RFID
  - For automatically locking upon bike return
  - **MFRC522 Module**
    - Passive (with range of ~1 inch)
    - 1 SPI channel

# Communication Unit



- Wi-Fi
  - Used for sending real-time alert of DC power loss
  - Can be used for relaying rental data to vendor
  - **TI CC3200 SimpleLink Wi-Fi Unit**

- Implementation
  - All communication modules controlled by MCU
    - Bluetooth
      - Sends user request to MCU
      - Receives acknowledgement message from MCU
    - Wi-Fi
      - MCU sends system message to Wi-Fi module.
      - Wifi module relays the message to the server through TCP protocol
    - RFID
      - RFID module sends a single data bit depending on the presence of ID tag

# Communication Unit R&V

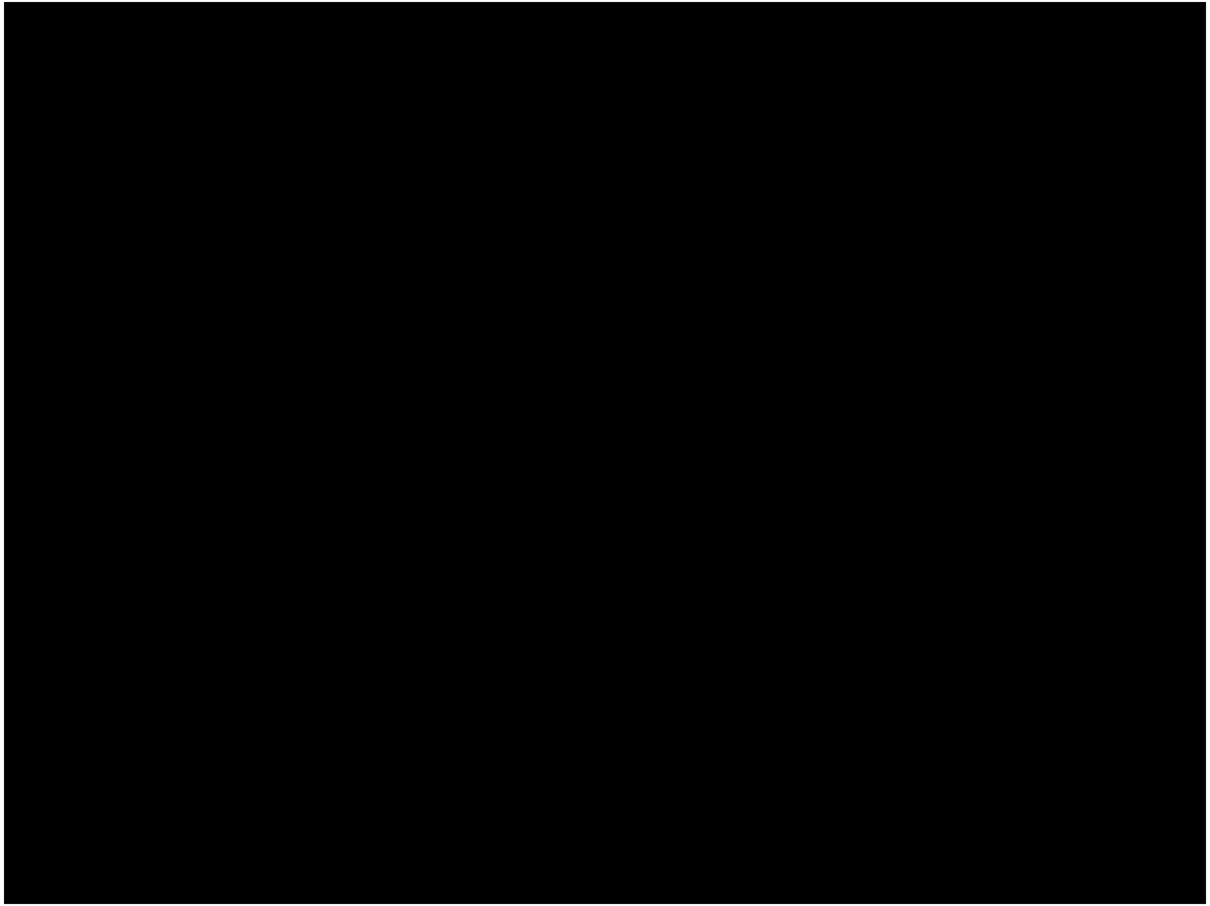
- A mobile device can communicate via Bluetooth 3 ft away
  - Verification: Lock/Unlock functionality through commands on Bluetooth serial terminal
- Must be able to communicate with an independent TCP server at 8Mbps
  - Verification: Confirmation of server message delivered from wifi module on the power failure sequence

# Communication Unit Results

- Solenoid lock was released when command was sent by the bluetooth terminal on the mobile device
    - Could communicate with Bluetooth from opposite side of lab
- Right after disconnecting the main power source, system message containing power failure warning was received through the TCP server running on a PC (Python)

# Overall Results

- Able to complete locking/unlocking logic with MCU (Bluetooth and RFID)
  - RFID handled with Arduino (which talks with MCU via UART) due to SPI complexity and time constraints
- Able to power all electronic components
  - Can successfully switch from DC to battery and keep components powered
  - Used 9V rechargeable batteries, larger battery would be longer lasting
- Wi-Fi capability implemented
  - Completed back-up alert
  - Potential to expand to send rental/return data to a server (for a bike sharing service)



Demo of Unlocking/Locking Functionality

Demo of Power Alert Over Wi-Fi

(See Local Video)

# Demo of Power Alert Over Wi-Fi 2

# Conclusions

- Many electronic parts with varying states use up variable amounts of power
- Can avoid security problems if safety less reliant on constant source of power (i.e. lower power requirements of system)
- Multiple functionalities important for making complete system (RFID/BT for rental, Wi-Fi for data relay, locking mechanism), but harder to integrate together in practice than in theory

# Potential Improvements

- Utilize locking mechanism with lower power sink
- Opt for active RFID (for easier scanning)
- Cheaper MCU
- Use larger, more powerful rechargeable backup battery
- Collect rental information and send to server
    - The “service” part of a bike sharing service

# Lessons Learned

- Repeatedly iterating to improve and better focus an idea/design
- When/how to modify design to achieve end goals by a deadline
- Integrating many components into one cohesive system
    - MCU controlling all components, powering all electronics sufficiently

# Thank You!

# Sources

[1] W. Hu, “More New Yorkers Opting for Life in the Bike Lane,” 30-Jul-2017. [Online]. Available: https://www.nytimes.com. [Accessed: 18-Sept-2017].

[2] S. Dave, “Life Cycle Assessment of Transportation Options for Commuters.” [Online]. Available: http://files.meetup.com/1468133/LCAwhitepaper.pdf. [Accessed: 19-Sept-2017].

[3] I. M. International, “How Divvy Works: Join, Unlock, Ride, Return,” Divvy Bikes. [Online]. Available: https://www.divvybikes.com/how-it-works. [Accessed: 04-Oct-2017].

[4] “How Does Zipcar Work?,” Car Sharing from Zipcar: How Does Car Sharing Work? [Online]. Available: http://www.zipcar.com/how. [Accessed: 04-Oct-2017].

# Sources (2)


[5] “South Side, Chicago, IL. to North Side, Chicago, IL. at 11:00AM CST by Car.,” Google Maps. [Online]. Available: https://www.google.com/maps. [Accessed: 04-Oct-2017].

[6] Texas Instruments, “TMS320F2837xD Dual-Core Delfino Microcontrollers” [Online]. Available: http://www.ti.com/lit/ds/symlink/tms320f28377d.pdf. [Accessed: 03-Oct-2017]

[7] Texas Instruments, “CC2640R2F SimpleLink Bluetooth low energy Wireless MCU” [Online]. Available: http://www.ti.com/lit/ds/symlink/cc2640r2f.pdf. [Accessed: 03-Oct-2017]

[8] NXP Semiconductors, “MFRC522” [Online]. Available: https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf. [Accessed: 03-Oct-2017]

# Sources (3)

[9] Texas Instruments, “CC3200 SimpleLink Wi-Fi and Internet-of-Things Solution, a Single-Chip Wireless MCU” [Online]. Available: http://www.ti.com/lit/ds/symlink/cc3200.pdf. [Accessed: 03-Oct-2017]

[10] IEEE.org, "IEEE Code of Ethics", 2017. [Online]. Available: http://www.ieee.org. [Accessed: 19-Sept- 2017]

[11] Arduino, “How to Drive Solenoids with Arduino” [Online]. Available: http://playground.arduino.cc/uploads/Learning/solenoid_driver.pdf. [Accessed: 03-Oct-2017]