Automatic Bike Locking System

Rohan Sreerama, Cameron Obrecht, & Praneeth Mekapati

ECE 445
TA: Hanyin Shao
University of Illinois at Urbana-Champaign
Fall 2022

Agenda

- What our project is
- What it does and how it works
- Fabrication process, shortcomings, and the future

Problems We Address

- About 2 million bicycles are stolen in the United States every year, with a total value of around $350,000,000[1]
- A good lock costs money, and replacing a bicycle costs many times more
- Individual bike locks can easily be cut with the use of bolt cutters

Existing Solutions

Bicycle - Bike U Lock,20mm Heavy Duty Combination Bicycle D Lock Shackle 4ft Leng…

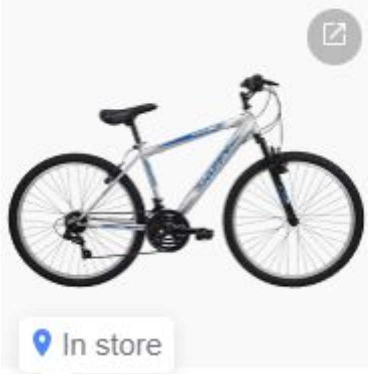
$29.95

Kryptonite - Bicycle - New-U KryptoLok Standard U-Lock with Cable 9 In …

$70.95

Kryptonite - Bicycle - Evolution 1090 Integrated Chain

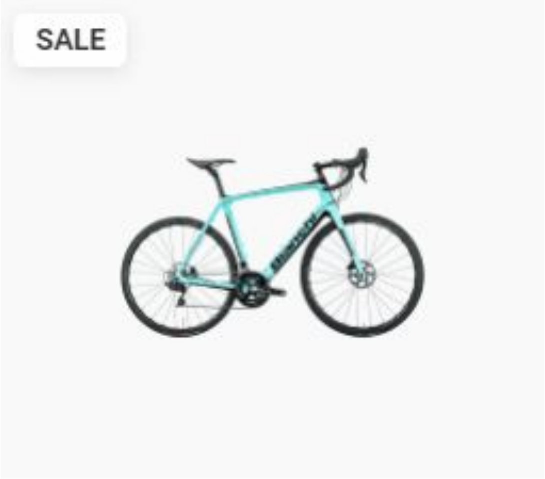
$110.00



Huffy Men's Highland 26" Mountain Bike - Silver/Blue

$149.99

Target



Bianchi - Infinito CV Disc 105 Road Bike - 2019,...

$3,413.99 ~~3,700~~

Used

Our Solution

- A pay-to-park bicycle lock, meant to be affixed to existing bicycle racks (not integrated, for lower cost)
- Purchasable by businesses and building owners, rather than bicycle owners
- Rugged and secure design, not penetrable by bolt cutters
- Fault tolerant and secure system built to withstand intrusion.

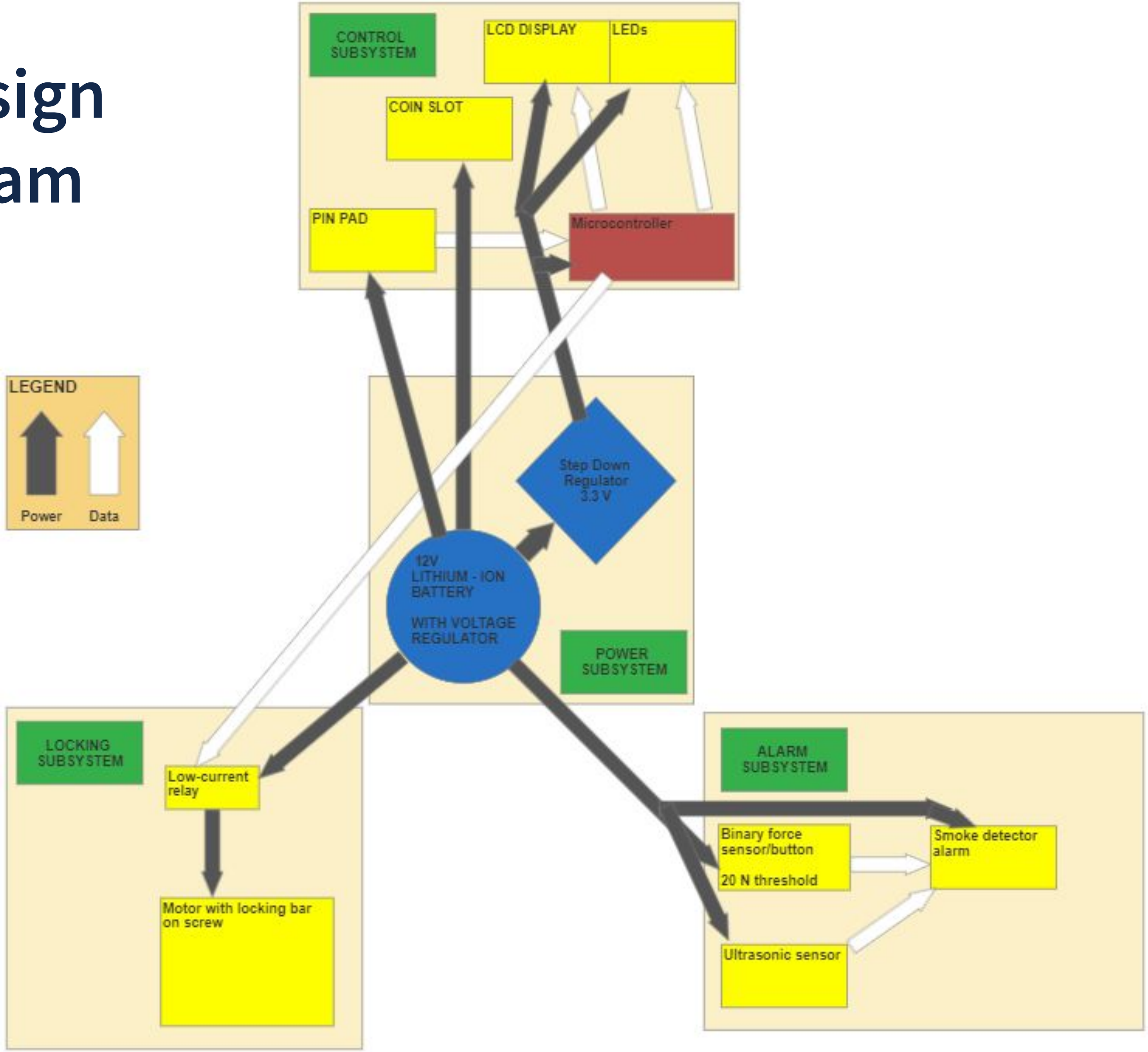

Mounts
Accelerometer inside
Keypad
PLEASE INSERT COIN
▼ COIN REJECT BAR
Coin slot
Display
Ultrasonic sensor
Locking bar
Guidance lights

Usage

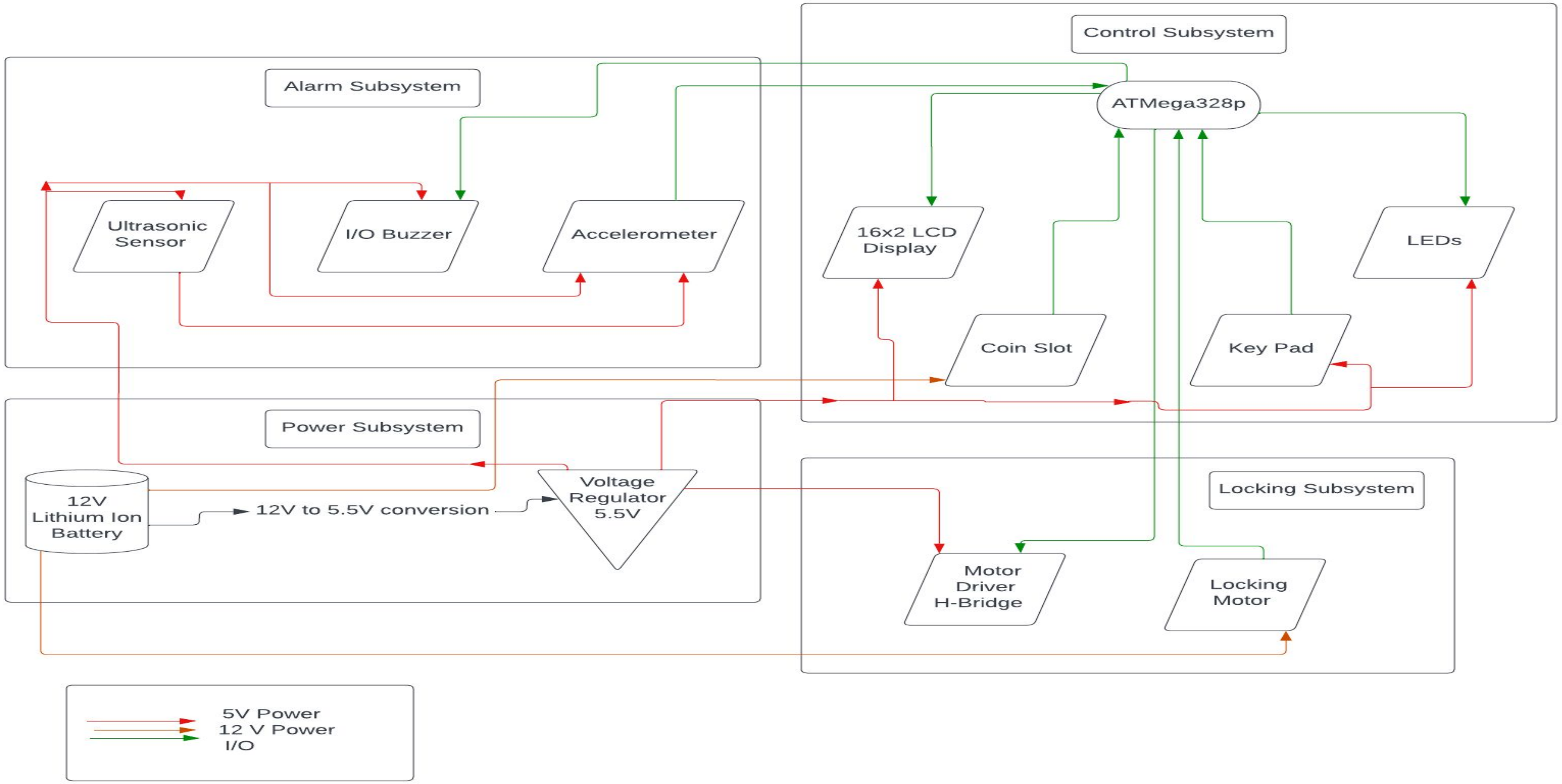
High-Level Requirements

- Be secure
  - Intrusions can be detected via accelerometer measurement deviations and the ultrasonic sensor.
  - Intrusions are appropriately reported, causing a buzzer to alarm.
- Accept payment honestly
  - System should fairly compute hourly cost and request the appropriate amount from the user according to the keypad input.
- Register input and display output correctly
  - Display the calculated cost on the LCD based on the user input for the # of hours

Original Design Block Diagram



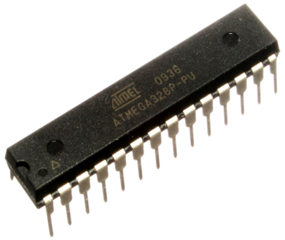
Final Block Diagram

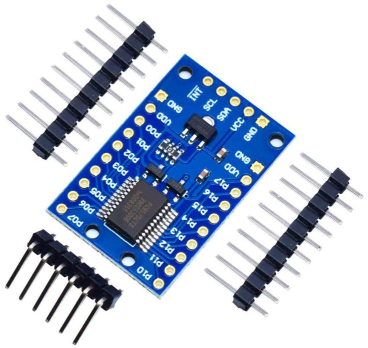


Power Subsystem

- 12-volt NiMH battery, commonly used in remote-control cars
- L7805 5-volt linear regulator for computing elements
- Utilizes decoupling capacitors to smooth out voltage delivery to core components and logic elements.

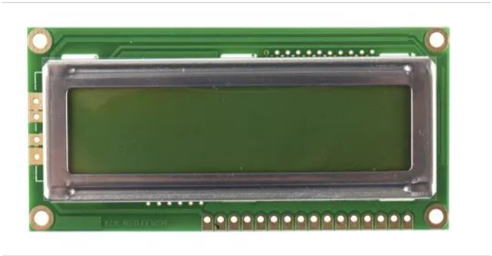
Control Subsystem





- ATmega328P with GPIO expanders
  - Sends/receives I/O signals to all the electronic components embedded in the system

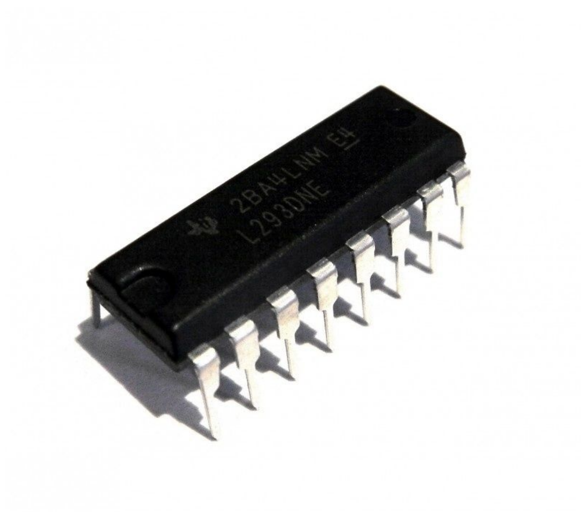
- Keypad, coin slot, LCD display, and guidance LEDs as I/O



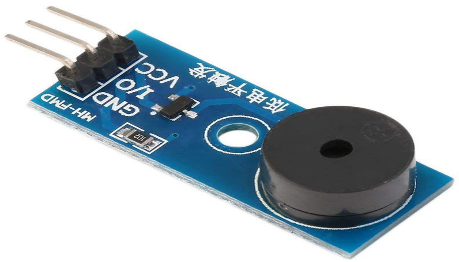
Locking Subsystem

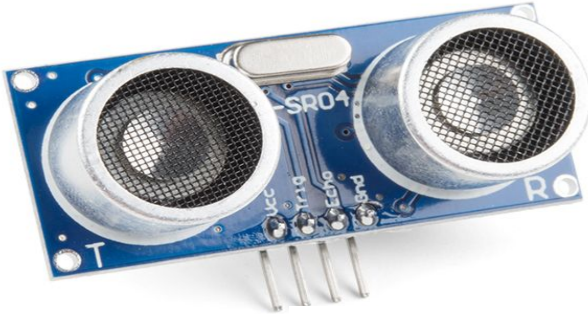
- Screw-driven locking bar
- 12-volt geared motor with encoder
- Motor driver (H-Bridge chip)

  - Controls the speed and the direction of the motor for the locking bar



Alarm Subsystem





- Ultrasonic sensor

  - Calculates the proximity distance of an entity from the machine

- Accelerometer

  - Detects unexpected movements of the machine from its initial position

  - Detects vibrations from tampering or grinding of the machine

- Buzzer

  - Outputs a high-pitched sound when the threshold has been crossed for both the ultrasonic sensor and accelerometer (Ultrasonic threshold - 150 cm, Accelerometer threshold - |10|)

Deliberate Changes

Power subsystem

- Originally 3.3 V, with two regulators for 3.3 V and 12 V

Control subsystem

- Reduction in number of guidance LEDs
- Change of display
- Addition of more decoupling capacitors
- Abandonment of on-board programming circuit

Alarm subsystem

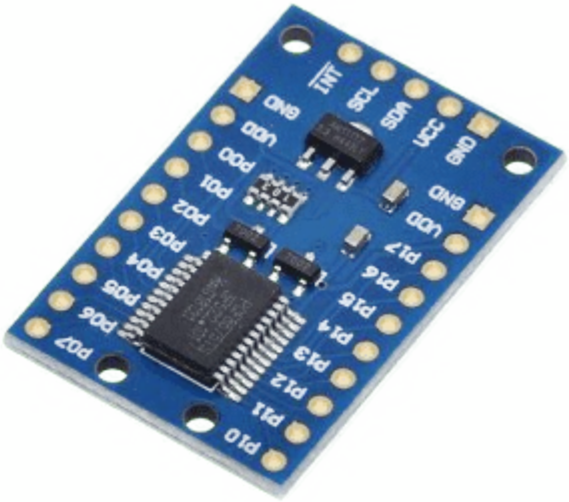
- Accelerometer instead of force sensors

Later Changes Due to Unexpected Obstacles

Control subsystem
- Wiring rigs on PCB to different pins on ATmega, due to misunderstandings
- Addition of a second ATmega
- Attempted use of a different GPIO expander model
- Abandonment of motor encoder functionality, relying instead on timing
- Abandonment of display connector in favor of I^2C "backpack" on display
- Different and better-proven keypad

Lock subsystem
- Use of different motor driver, due to a better-proven alternative

Alarm subsystem
- Different accelerometer, due to an overlooked voltage incompatibility



Building the Appliance

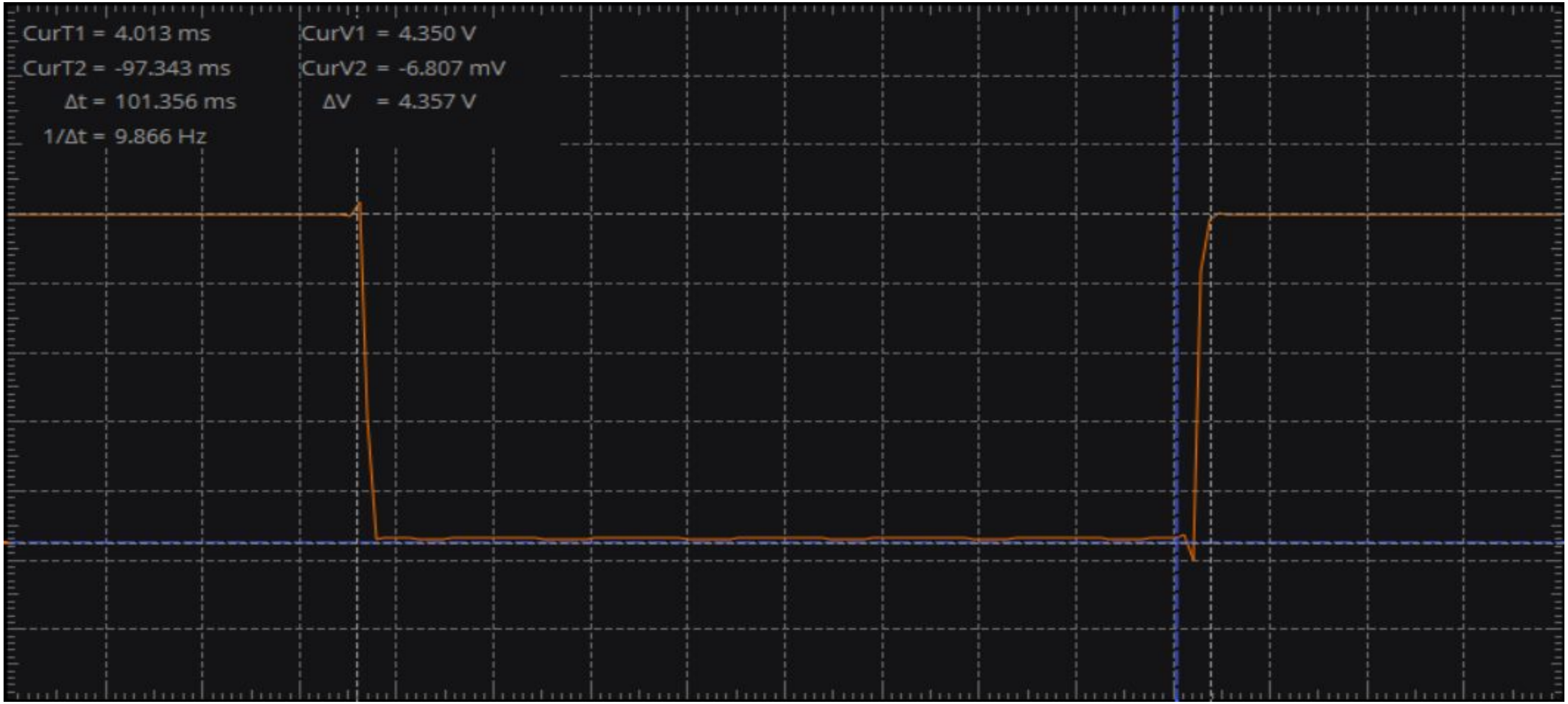
- Component selection
- Schematic creation
- PCB layout
- Correspondence with the machine shop
- Software development

Arduino Security Data

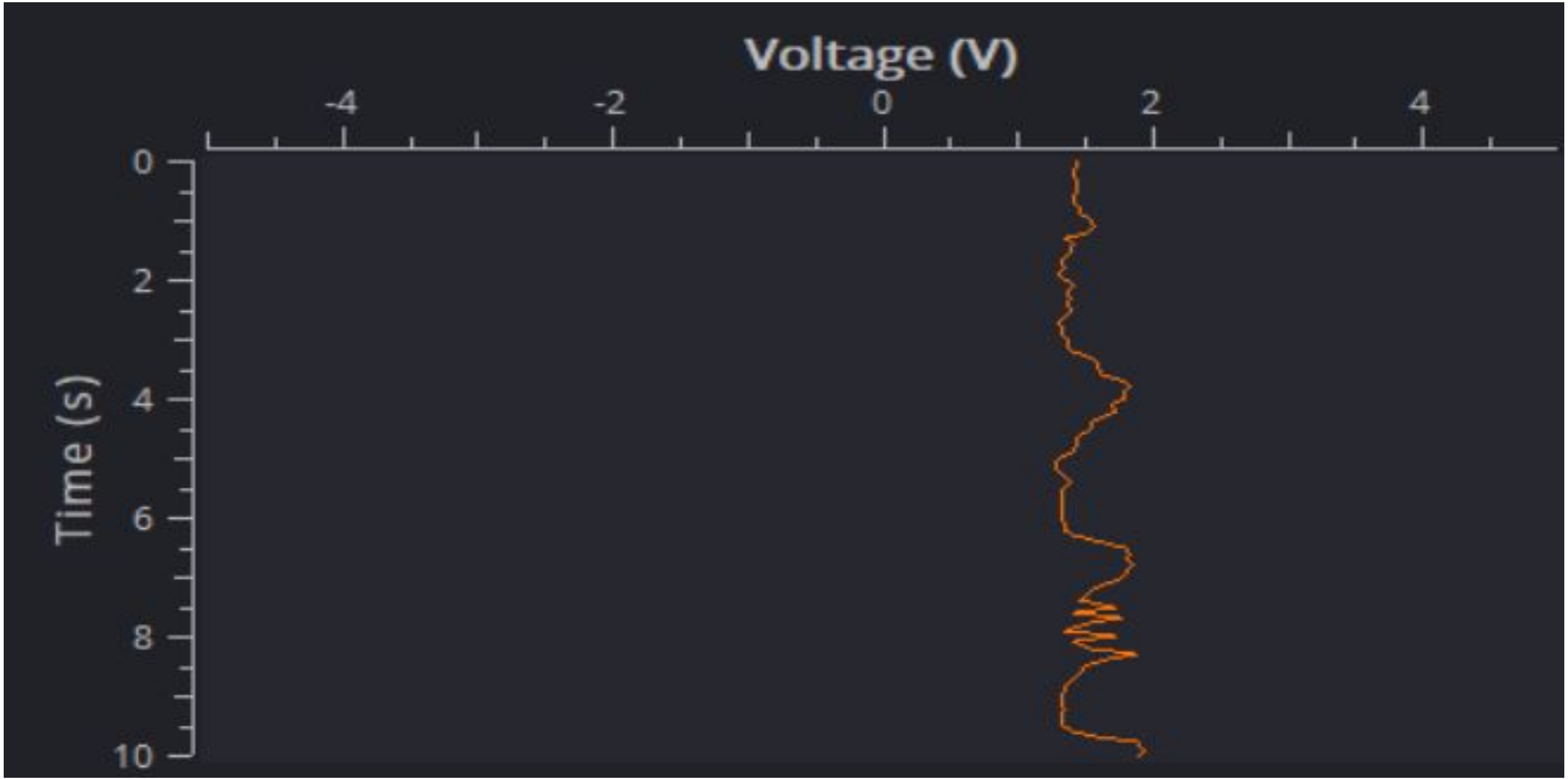
```
17:51:36.229 -> x=0
17:51:36.229 -> y=13
17:51:36.259 -> z=-6
17:51:36.259 ->  $
17:51:36.259 -> Distance: 9
17:51:38.501 -> x=10
17:51:38.501 -> y=13
17:51:38.501 -> z=2
17:51:38.534 ->  $
17:51:38.534 -> Distance: 9
17:51:43.212 -> x=1
17:51:43.212 -> y=-11
17:51:43.212 -> z=4
17:51:43.212 ->  $
17:51:43.212 -> Distance: 9
17:51:46.478 -> x=-2
17:51:46.478 -> y=-11
17:51:46.511 -> z=5
17:51:46.511 ->  $
17:51:46.511 -> Distance: 10
17:51:53.302 -> x=0
17:51:53.302 -> y=12
17:51:53.335 -> z=-6
17:51:53.335 ->  $
17:51:53.335 -> Distance: 10
17:51:56.533 -> x=-3
17:51:56.533 -> y=2
17:51:56.566 -> z=-11
```

```
17:52:27.451 -> y=-3
17:52:27.451 -> z=2
17:52:27.451 ->  $
17:52:27.451 -> Distance: 10
17:52:34.705 -> x=30
17:52:34.705 -> y=67
17:52:34.738 -> z=2
17:52:34.738 ->  $
17:52:34.738 -> Distance: 72
17:52:42.352 -> x=1
17:52:42.352 -> y=47
17:52:42.352 -> z=-12
17:52:42.352 ->  $
17:52:42.352 -> Distance: 11
17:52:44.365 -> x=12
17:52:44.365 -> y=35
17:52:44.365 -> z=-11
17:52:44.397 ->  $
17:52:44.397 -> Distance: 10
17:52:47.332 -> x=-3
17:52:47.364 -> y=-12
17:52:47.364 -> z=5
17:52:47.364 ->  $
17:52:47.364 -> Distance: 45
17:52:49.571 -> x=0
17:52:49.571 -> y=32
17:52:49.571 -> z=-9
17:52:49.604 ->  $
17:52:49.604 -> Distance: 83
```

Coin Slot Signal Test



Accelerometer Signal Test (x-axis only)



Shortcomings

- Original KXTJ3-1057 accelerometer gave bad output (voltage too high)
- Original wiring on the PCB was not going to work (unaware of SCL and SDA designations)
- Difficulty in using the alphanumeric display without I^2C accessory
- Lack of knowledge of software to support PCA9555 GPIO expanders
- Generally we did not allow enough time to correct for these failures, not evident before.
- Software development seemed to take more time than forecast
- Some lack of coordination and cross-functional knowledge (cases of the motor drivers and accelerometer outputs)

What Would We Do Differently in the Process

Starting knowledge was extremely important. We did not know what we did not know about the components, because we unwittingly made assumptions.

If we did it again, we would:

- Use common components with plenty of Arduino support and references
- Find or make breadboard DIP testing rigs for surface mount chips
- Prototype the entire circuit on a breadboard first
- We would experiment with more microcontrollers to determine their reliability and utility.

Possible Improvements to the Design

- Credit card reader in place of coin slot
- ESP32 microcontroller for wireless payment and communication
- Full color display, with physical buttons

This all would require much more intricate software than our current design.

Functional abilities

Ability	Success?
System welcomes new user and initiates start sequence	Yes
System requests number of hours to lock the bike and accepts input from keypad (also displays this on LCD)	Yes
System computes cost to lock bike for desired number of hours and reports to the user via LCD	Yes
System accepts correct number quarters via coin slot and reports amount inserted by user on LCD after each coin drop	Yes
System, upon receiving payment, tells user to move aside and engages motor to lock the bike in place.	Yes
Buzzer sounds when irregular accelerometer and ultrasonic sensors values are detected (occurs when device is shaken vigorously)	Yes

Conclusions: What We Have Learned

- We understand that we have to really know the functionality of the controller pins (e.g. SDA SCL)
- Prototyping the breadboard was very important prior to creating the final PCB
- Understanding the code execution at a low level is integral to debugging timing issues, etc.

References

1. "The Ultimate Guide to Bicycle Theft Statistics & Insurance." Simple Bike Insurance, Biker, 2 Oct. 2021, https://simplebikeinsurance.com/guide-bicycle-theft-insurance/.
2. "EcoTank Et-2803 Wireless Color All-in-One Cartridge-Free Supertank Printer with Scan and Copy." *C11CJ66203 | EcoTank ET-2803 Wireless Color All-in-One Cartridge-Free Supertank Printer with Scan and Copy | Inkjet | Printers | For Work | Epson US*, Epson America, Inc., 2022, https://epson.com/For-Work/Printers/Inkjet/EcoTank-ET-2803-Wireless-Color-All-in-One-Cartridge-Free-Supertank-Printer-with-Scan-and-Copy/p/C11CJ66203.
3. "ATmega328/p." Atmel Corporation, Jun. 2016, https://datasheet.octopart.com/ATMEGA328P-MU-Microchip-datasheet-65729177.pdf
4. "PCF8575 I2C 16-Bit Digital Input Output Expander." Create.arduino.cc, Project Hub, 7 Aug. 2019, https://create.arduino.cc/projecthub/xreef/pcf8575-i2c-16-bit-digital-input-output-expander-48a7c6