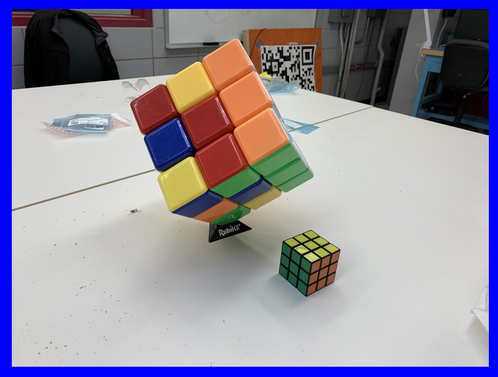
# Self-Solving Rubik's Cube

sdmay20-29
Taylor Burton, Jacob Campen, Casey Cierzan, Joe Crowley,
Annie (Yung-Hsueh) Lee, Keegan Levings-Curry, Luke Schoeberle

Client/Advisor: Dr. Zambreno

## Problem Statement and Requirements

- A self-contained, self-solving Rubik's cube
  - Can be scrambled by hand
  - Solves itself with no intervention
- Use for recruitment at ISU
  - Displays the possibilities of our degree
  - Hands-on recruitment tool



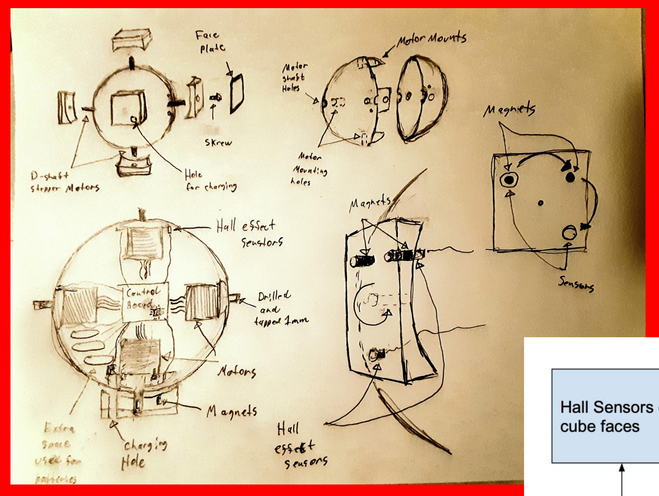
**Functional Requirements**

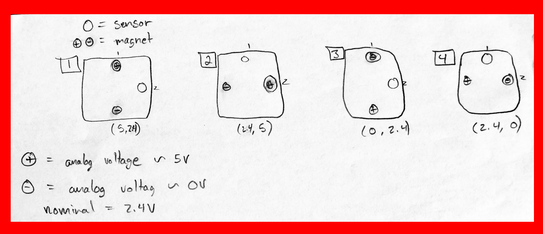
- Must be solved in 2 minutes or less
- The battery should last for at least 1 solve
- Should not rely on external devices (like cameras or robot arms)
- Starts in a solved state

**Nonfunctional Requirements**

- Should look like a standard Rubik's cube
- Easily turnable by the user
- Should last at least 3 years
- Side length should be 11 cm
- The budget should not exceed $750

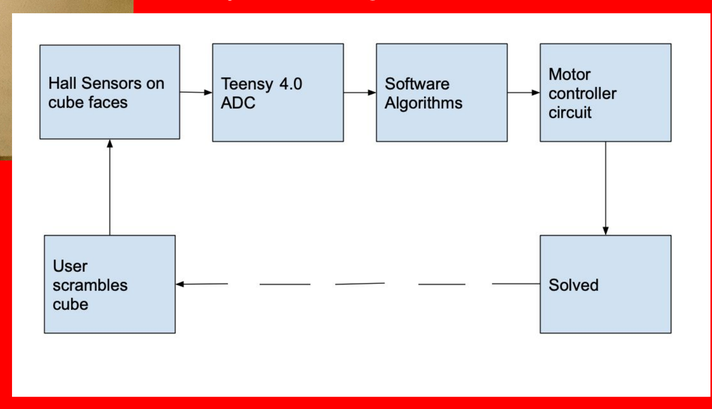
## Concepts and System Overview




**System Overview**

- Teensy 4.0 microcontroller
- Hall Effect sensors
- Solving algorithms
- Motor controller system
- System integration code




## Software

- Embedded code on the microcontroller
- A mix of pure C code and Arduino code
- The software contains 4 main parts:
  - Rotation detection software
  - Rotation simulation algorithms
  - Solving algorithms
  - Motor control software

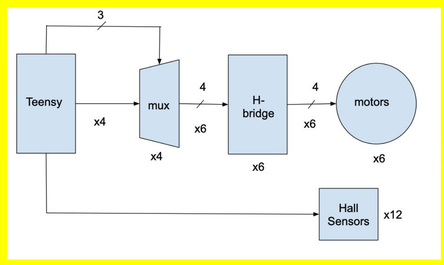

Cube Visualization

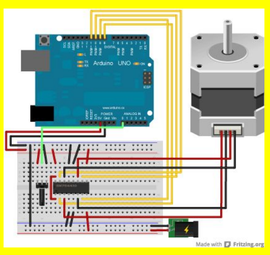
**Software Operation Overview**

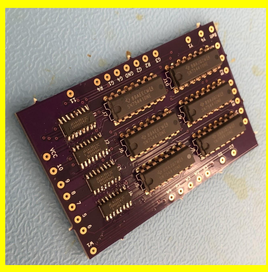
- Initially, physical cube movement triggers the rotation detection software
- Next, the rotation detection software encodes the rotations in a text file and executes the rotation simulation algorithms
- The rotation simulation algorithms use the text file to scramble the cube in software
- Next, the solving algorithms encode the rotations for solving the cube in a text file
- Finally, the motor control software uses the new text file to send rotations to the motors, which physically solves the cube

## Hardware

- Main Components
  - Hall Effect sensors
  - Teensy 4.0 microcontroller
  - Batteries
  - Stepper motors
    - Can be turned manually
  - PCB
    - Multiplexers
    - H-bridge motor driver
    - Bendable substrate in Rev. 2
- Mechanical Considerations
  - Size of cube
  - Internal space constraints



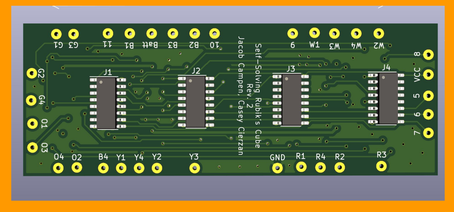



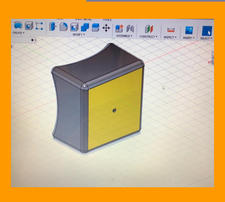


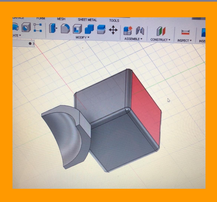
PCB Rev.1

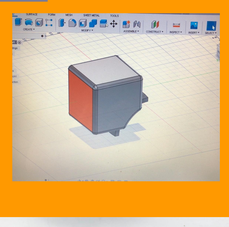
PCB Rev. 2

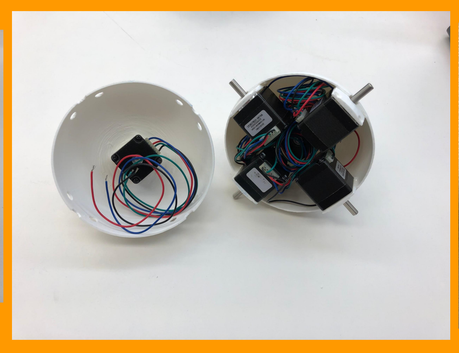
## Testing and Prototyping











**Testing Process**

- Unit Testing
  - Ex: Motor control circuit
- Integration Testing
  - Ex: Motor integration
- System Testing
  - Ex: Holistic motor verification

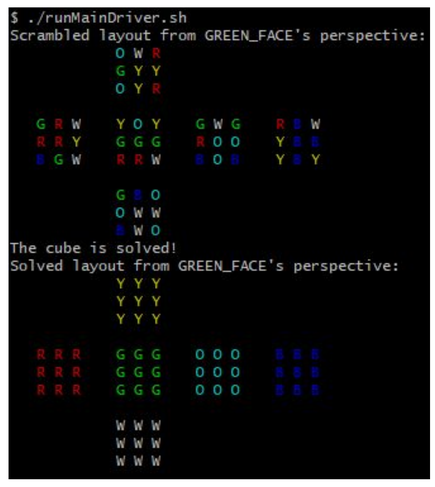
## Constraints and Outcome

**Final Deliverables due to COVID-19**

- Material for the physical prototype
- Completed solving algorithms
- Untested system code
- CAD models
- Documentation

**Unable to Test**

- PCB Rev. 2
- 3D-printed components
- Fully constructed cube
- System code




Completed solving algorithm example

**Engineering Standards and Design Practices**

- Follow IEEE standards
- Push early and often
- Document everything
- Follow a tight budget
- Ensure maintainability