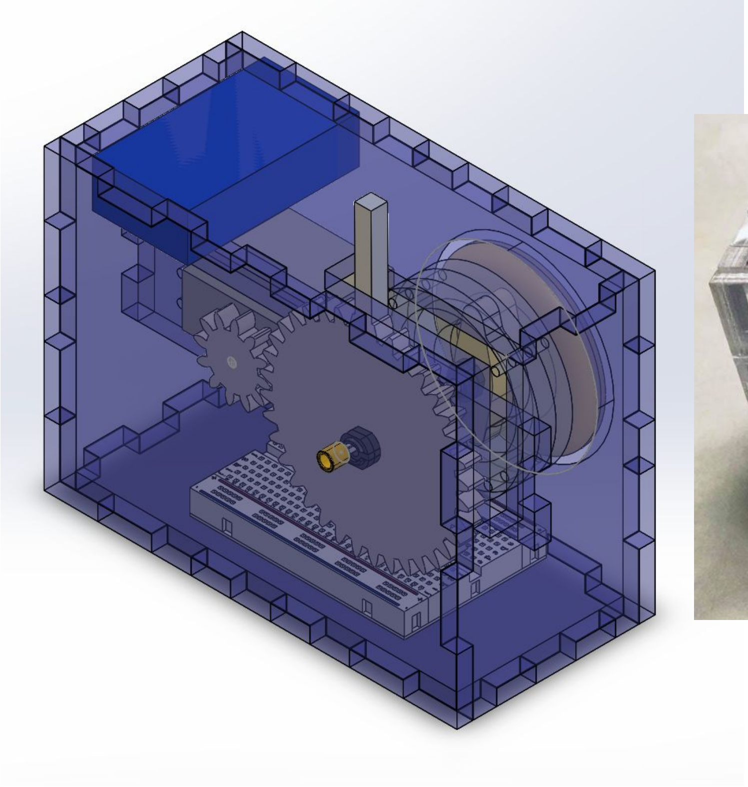
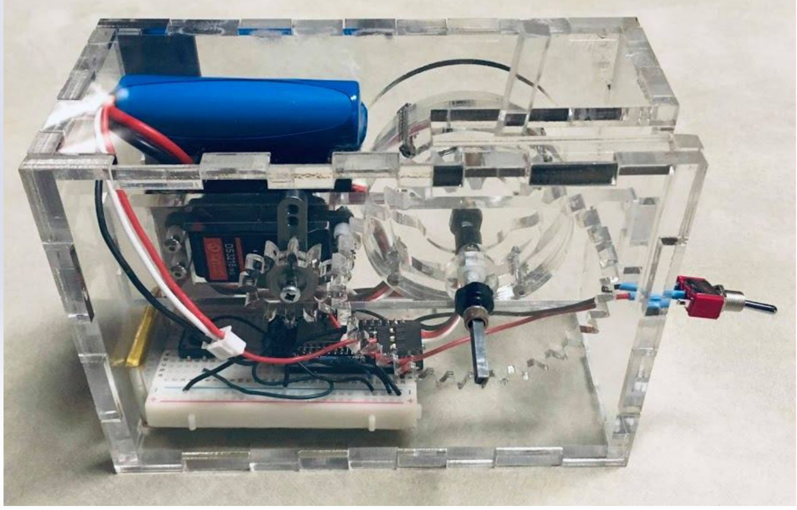
Wireless Deadbolt Unlocker





Mechanical Engineering 130: Design for Planar Machinery
December 6th, 2018
Group 10:
Asena Yildiz
Miyuki Weldon
Connie Yu
Lawrence Su
Brandon Huang

1. BACKGROUND

Central to the function of doors is the ability to lock them via a latch. Although there are many designs for these latches, one of the most pervasive type is the deadbolt. A deadbolt lock can be toggled between open and closed positions only by turning the key which makes deadbolt locks more secure against entry without a key [1]. As with any form of security, all locks including the deadbolts, come with the disadvantage of potentially locking out the owner if they forget or misplace their key. As most locks are purely mechanical in nature, for most people's doors, there is no other way to prove to the system the authenticity of the owner and so the owner must resort to other means: calling a locksmith, calling their landlord, etc.

Current solutions to this problem exist by replacing the locking system with a smart system. These include keycard locks, password lock systems, and other similar electronic systems that are "smart" and give the owner an alternative method to authenticate themselves. Extending a system's base of trust can increase the surface of attack for a malicious person, but these can be resolved with good design. An example would be numberpad passwords can be very easy to brute force especially if they are short with little entropy, such as the 4 digit numeric passwords are common in these products. For this example, solutions could be making the password alphanumeric or longer to increase password entropy or to simply add an increasingly long lockout period after incorrect attempts to deter attackers. The central issue with these systems, however, is that they are often expensive, full replacements to one's locking system, which may be improbable to implement if the owner a renter and cannot replace their door locking mechanisms on their demand.

Our goal then is to design a mechanism that automatically engages or disengages door locks to allow for remote locking or unlocking of a door with a deadbolt lock without the need of installing a custom lock. This device would be an IoT (Internet of Things) device and allow for users to lock or unlock their door with their phone. The device would also allow for the user to engage the lock via a button or manually if they are inside. As with all IoT devices, there are potential risks, but these can be mitigated by good implementation [2].

2. TIMELINE AND TASK ASSIGNMENT

Our team self-selected roles and assignments based off personal interests and skills. The following can be taken as a rough outline of task assignment, although we all helped in other tasks as well when necessary.

Asena Yildiz: Materials, Write Up/Report, Poster
Miyuki Weldon: Laser Cutting and Assembly
Connie Yu: Electronics and Assembly
Lawrence Su: Analysis, Write Up/Report
Brandon Huang: Write Up/Report, Editor

Figure 2.1 is a GANTT chart for the overall project timeline. We started the project brainstorming ideas in late August, narrowing down ideas and until we had our final design remaining by September 25th which was the day of the first design review. Analysis and prototyping of our design followed with documentation and write-up tasks arising near the end of the project on December 6th.

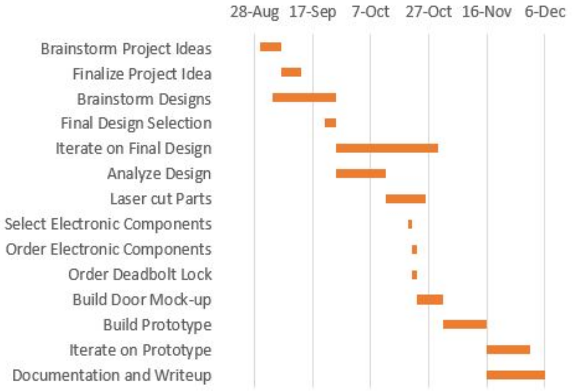


Fig 2.1 Timeline of Tasks.

3. SPECIFICATIONS AND REQUIREMENTS

The wireless deadbolt unlocker has many specifications and requirements to ensure its successful application. Our requirements are as follows:

Essential:

ǒ ***The device should provide enough torque to reliably rotate a lock into its locked and unlocked positions.***

ǒ ***The unlocker should be compact and able to stick on the door and be easily removable, without damaging the door.***

Desirable:

ǒ ***The unlocker has IoT support to remotely toggle the lock.***
ǒ ***The device would be controlled through an online app.***
ǒ ***Device has buttons to lock or unlock the door from the inside without using IoT.***
ǒ ***Warning of low-battery on device to warn user to charge it.****
ǒ ***Device is battery operated.***
ǒ ***Servo is powered only when the lock needs to be remotely controlled. Otherwise, the motor is off so that the lock can still be unlocked by key.****

**** For future modifications***

For our project, we managed to successfully reach most of requirements, including many of our desired requirements. Later in the written report, we will describe our future modifications.

For our engineering specifications, we wanted to impose limitations to our design to make the most compact and reliable design.

Firstly, our weight needs to be less than 3 lbs. Currently, our design is less than 3 lbs, allowing it to be easily attached to a door with simple 3M adhesive double-sided tape.

Secondly, our overall size must be no larger than 9in by 7in by 4in. This included the mounted electronics and physical mechanism. It needs to be small so it can fit on the door without interfering with the door handle. The size limitations are also because the design also needs to be compact and modular, so that it can be removed and replaced easily. Currently, our design in much less than our size limitations: 3in by 4.5in by 6.25in.

Thirdly, since our design is ideally battery operated, we wanted to make the design powered by rechargeable lithium polymer (LiPo) batteries. Ultimately, we wanted our design to be contained as 1 unit, and that includes our power source. Since home doors are not necessarily near an electrical power outlet, we wanted to make our design contain the internal power source. However, we are still working on ways to allow the user to know when to recharge their battery.

Fourthly, our less engineering specification is the mechanism motion. Our mechanism utilizes rotational motion, ultimately moving the deadbolt lock by rotating a rigid plate that fits onto the deadlock lock of the user's door.

4. CONCEPTUAL DESIGN

Through our brainstorming, we started with many ideas for our device's mechanism. The earliest designs, as seen in Figure 4.1, drew inspiration from the four bar mechanisms we were studying earlier in course of the project. These designs brought up interesting questions, such as how we would design enough clearance for the linkages while also making the mechanism as compact as possible.

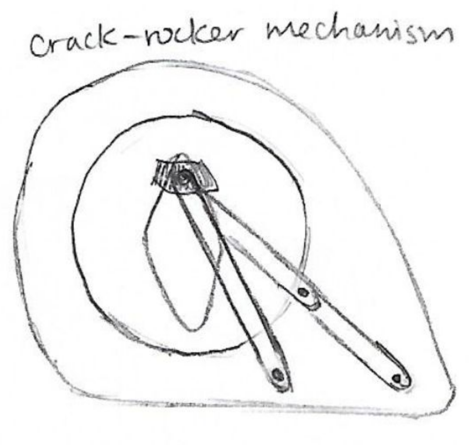


Fig 4.1 An early design using linkages

After working through several designs involving linkages, we realized that we were projecting too much complexity to the problem. Instead we started from the simplest idea, which was to attach the motor directly to the lock handle. From there we came up with designs with sliders that would turn as seen in Figure 4.2. These mechanisms that contain pure rotation and the fewest moving parts optimized for reliability and longevity.

Fig 4.2 An early design using rotational sliders

Our last major step forward before our final design was to use gears that would turn a toothed holder for the lock handle. Using gears would allow us to lower the gear ratio and increase torque applied to the handle to ensure our device can physically engage the lock. The design in Figure 4.2 uses such gears. The central piece interfaces with the actual lock handle while the two pins would hold the piece in while allowing it to turn in the slots.

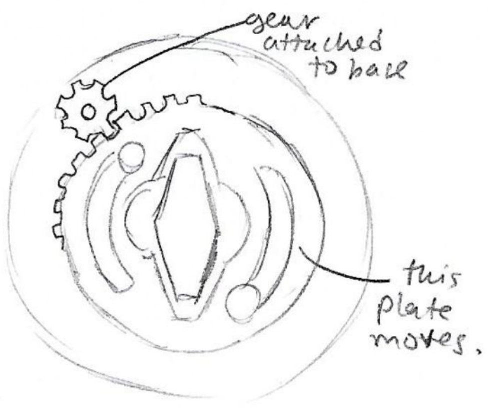


Fig 4.3 An early design using gears. The slots and pins on either side would hold the center piece

5. MECHANISM DESIGN

Bill of Materials

Tables 5.1 and 5.2 show the mechanical and electronics (respectively) materials and components used in our device. The screws and nuts were used to hold the device together. The bearings and spacers were used to maintain the right spacing on our central axle. The rest of the parts were cut from the acrylic sheet. For electronic components, the controller was an arduino uno with a NodeMCU wifi shield for the IoT connection. The buttons were included to allow users inside the door to manually toggle positions. We used LiPo batteries to allow the device to recharge. The 7.4V powers the servo motor while the 3.3V powers the controller.

Table 5.1 Mechanical components and their quantities

Parts	Quantity
4-40 Screws	4
4-40 Nuts	4
Bronze Bearings	2
8mm Spacer	1
¼” Spacer	1
3 mm Spacer	1
3” Machine Key (Axle)	1
Rubber Shaft Collar	2
¼” Acrylic Sheet	16” x 32” Sheet

Table 5.2 List Electronic components

Parts
Breadboard
7.4V LiPo Battery
3.3V LiPo Battery
9kΩ Resistor
Push Button
Toggle Button
NodeMCU ESP8266
Digital Servo Motor (20 kg-cm Torque)

Central Plate

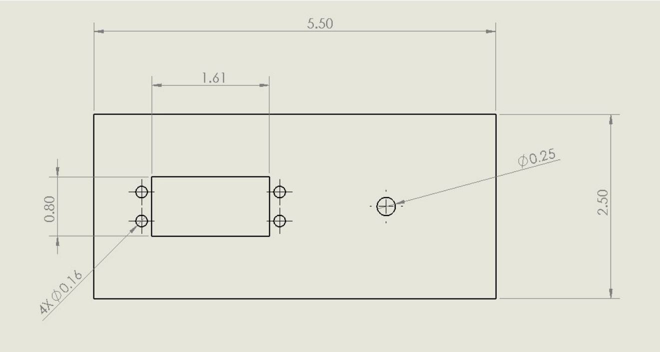


Fig 5.1 Drawing of the Central Plate

Figure 5.1 shows the central plate of our device which holds all our components together. The gears are mounted on top of this plate. The rectangular cutout is where the servo motor fits and is mounted where the 4 screw holes are. The circular hole on the right is used to connect the large gear to the offsets and lock piece, described below. This allows for the movement of the lock piece to unlock or lock the deadbolt lock.

Device-Lock Interface

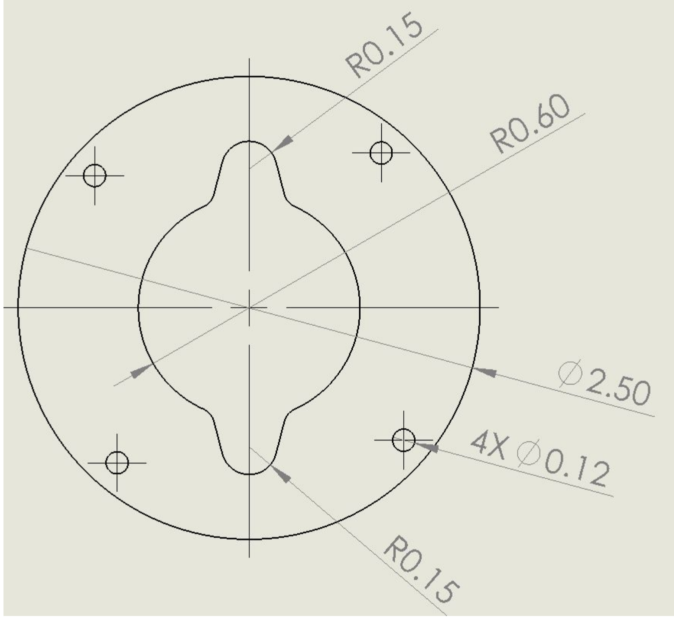


Fig 5.2 The Interface between our device and lock handle. This is attached to other pieces via four screws.

Figure 5.2 shows the interface between the device and lock handle. The lock fits snuggly into the lock piece's internal cutout shape, and the offsets, which are connected to the lockpiece, help turn the lock piece. If needed, additional padding can be added to the slot to allow for more flexibility in handle shapes and to reduce wear on the component.

There are two additional offset pieces which are shown in Appendix Figures A.1 and A.2. These two offset pieces are stacked on top of each other to help offset the lock piece, which needs to cover the deadbolt lock. Offset 1 (Figure A.1) is connected to the large gear via the square cutout in its center, and offset 2 (Figure A.2) and lock piece are connected to offset 1 with 4 screws near the circumference of the circle.

Servo Motor

From research and simple experiments we estimated the necessary torque required to turn a lock was approximately 10 kg-cm. Thus, we chose our servo motor with maximum nominal torque of 20 kg-cm to have a large margin of error and to allow our device to operate on locks that may be older, worn, and harder to turn.

Gears and Gear Ratios

In order to toggle between the open and closed positions, the lock handle must be turned a total of 90 degrees. Wanting to avoid uncertainties in the accuracy of the servo motor, we aimed to simply use the full 270 degree rotation of the servo motor. Thus, we aimed to have a 3:1 ratio. This also has the benefit of increasing the mechanical advantage of our system.

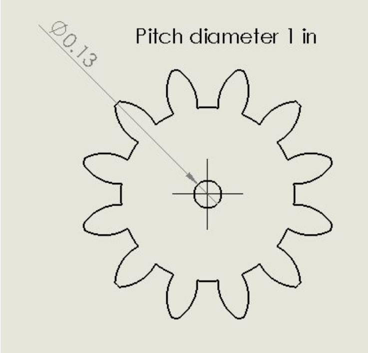


Fig 5.1 Small gear attached directly to the motor

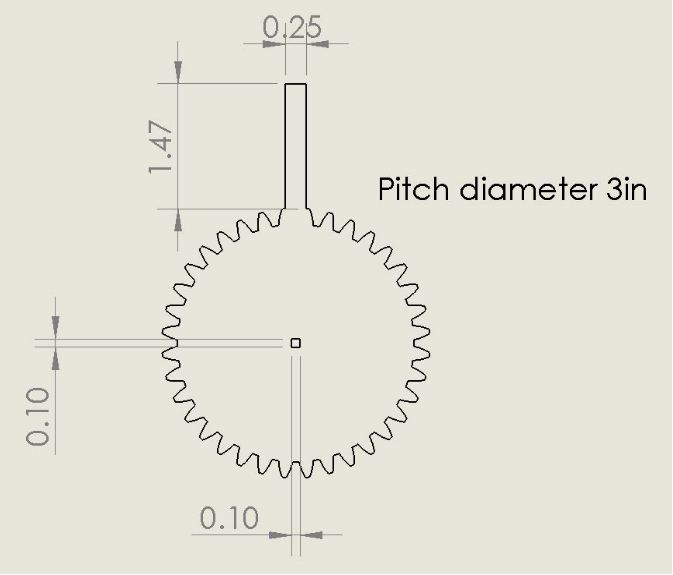


Fig 5.2 Larger gear attached to the central shaft that turns the lock

These gears allow movement. The servo turns the small gear, and because of the gear ratios mentioned above, the large gear turns to the correct angular displacement. The large gear's jaws interlock with the small gear, and as the large gear turns, the lock piece, pictured below, also turns. This is due to how the pieces are attached. An important design choice was the latch at the top of the large gear. This latch allows for the user to turn the large gear themselves when the servo motor is not powered, so they are able to unlock/lock their deadbolt lock manually when they are inside their home.

Electronics and IoT

To control the servo motor, we picked the NodeMCU ESP8266. This chip allows for easy internet connectivity, and the Blynk IoT platform was chosen to pair with the chip for hosting wireless capabilities. The Blynk phone app was designed to have a virtual button for unlocking and unlocking the door by having the ESP8266 control the servo motor. The app features email compatibility so that whenever the door is unlocked using the app, a message is sent to a designated email.

Batteries were chosen to power the electronics so that the device would not need to be plugged in. A 3.3V battery was chosen to power the NodeMCU ESP8266 through the Vin pin, and a 7.4V battery was chosen to power the servo motor, which requires a higher voltage.

Enclosing Box

The enclosing box simply has to contain all the components while being as small and lightweight as possible. It requires a slot near the top for a manual toggle between locked and unlocked positions and a hole on the back to allow the device to interface with the lock handle. It also needs an adequate surface on the pack in order to allow for an attachment point to the door via double sided wall mounting tape. The 2D drawings can be found in the Appendix figures A.3 to A.8

Animation

The device is designed to lock or unlock the door within 0.5 seconds. In order to see the whole process clearly, the animation of both locking and unlocking (counterclockwise and clockwise rotations) was created, and the whole animation lasts for 4 seconds. Here is the URL for the animation uploaded on youtube.

https://www.youtube.com/watch?v=tbYNsoH7JIA.

Here is the URL demonstrating what this virtual animation looks like translated into the physical prototype:

https://www.youtube.com/watch?v=oM8lAEfFCic.

Motion Analysis

In consideration of the toughness of measuring the velocity or the torque directly through the actual device, the motion analysis is mainly done with a background of solidworks simulation. The results are ideal ones regardless of the frictions or power loss. So the result curves are sine waves. Although the results might behave far from the real world, the results are still useful to help figure out some dangerous points.

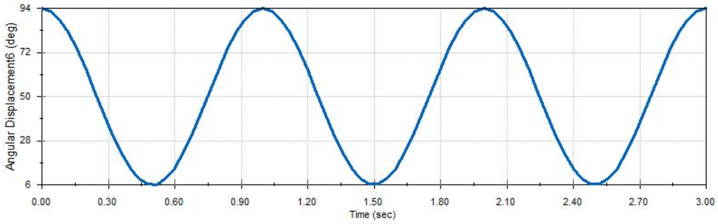


Fig 6.1 The angular displacement for the large gear.

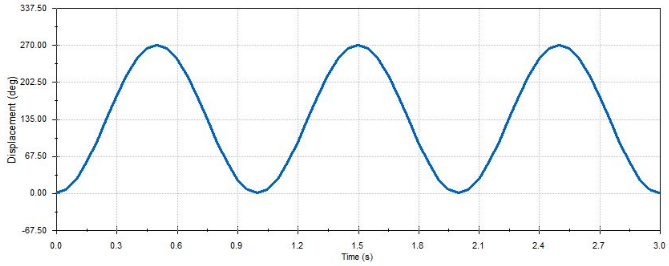


Fig 6.2 The angular displacement for the small gear.

Due to the its specifications, the servo motor only rotates 270 degrees, therefore the small gear does as well. The gear ratio is 1:3 resulting in the angular displacement of the large gear to be within 90 degrees, in order to turn the lock.

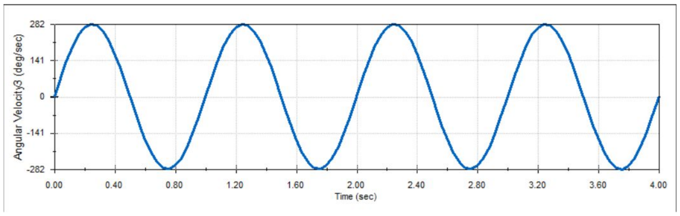


Fig 6.3 The angular velocity for the large gear

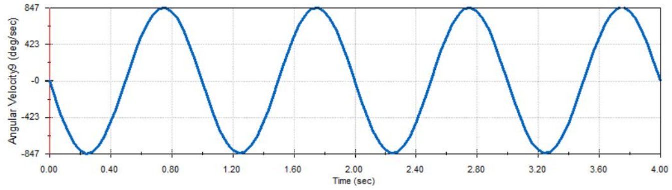


Fig 6.4 The angular velocity for the small gear.

Because we want our mechanism to complete a locking and unlocking cycle in one second, our maximum angular velocity of the small gear is 540 deg/sec. Computing from the ratio, the maximum angular velocity of the large gear is 180 deg/sec.

We estimated our torque required to turn the deadbolt to be around 3 kg-cm. This would require the large gear to turn with 3 kg-cm of torque and through the 1:3 gear ratio, the small gear would only need to output 1 kg-cm of torque. This quantity is far below the motor's stall torque of 18 kg-cm.

Because we did not input the mass of each assembly units, the actual amount of torque cannot be calculated by solidworks. However, the required input torque is crucial to know. Our design weakened the actual contact of the gears to the lock, and the mass of large gear is one of the most influential sections to the torque. Similarly, in a real process of locking or unlocking the door, we have to pay attention to the harm to the device generated by the torque from the gear's initial when the gear is about to stop.

Stiffness Analysis

The torque provided by the servo motor induces torsional stress in our device's components. The three parts that would be most affected are the the acrylic piece with the lock cutout, the big acrylic gear, and the steel shaft connecting the cutouts. To analyze the stress and displacement in these parts, SolidWorks FEA was conducted on worst-case scenarios.

For the steel shaft, only 1.16" was analyzed: this length is the distance between the servo motor providing torque and the lock cutout turning the lock. A torque of 3 kg-cm (roughly the expected torque needed to turn the lock) was applied on the end of the shaft, with the other end fixed (note that this is a worst-case analysis: in reality the shaft would rotate, not remain fixed). The shaft is made of 1018 stainless steel, which has a yield strength of 370 GPa. FEA revealed that the max stress is half of this yield strength, giving us a factor of safety of 2. Displacement was found to be negligible, with max displacement being 0.113mm.

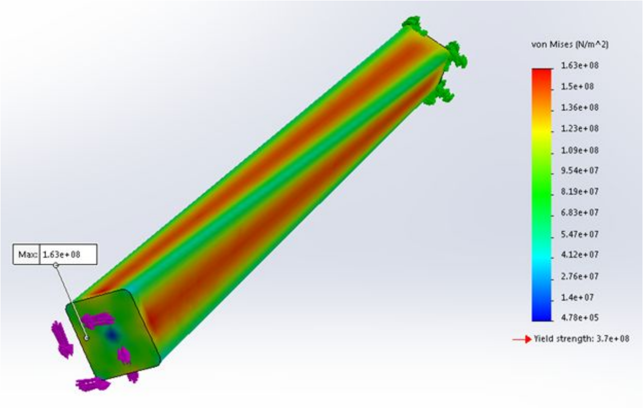


Fig 6.3 Stress distribution on steel shaft

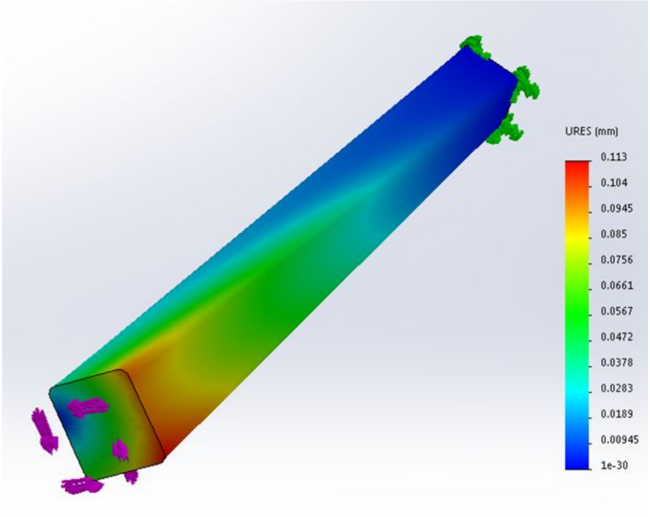


Fig 6.4 Displacement surface plot on steel shaft

To analyze stress in the acrylic parts, FEA was conducted on the big gear, with 3 kg-cm of torsion applied on the four faces of the square cutout where the shaft is inserted. Because this is the same cutout on the lock cutout piece, this analysis is applicable to both acrylic parts. Acrylic's yield strength is 45 GPa, but max stress was found to be 3.8GPa, giving us a factor of safety of almost 12. Displacement was also found to be negligible, with a max of only 0.002mm.

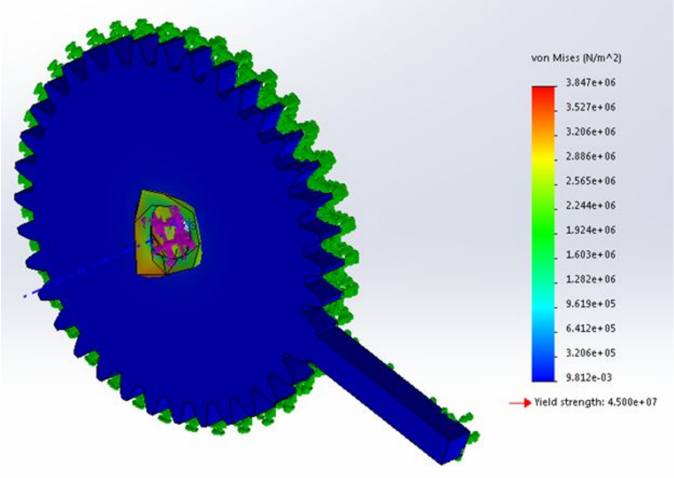


Fig 6.5 Stress distribution on acrylic cutout

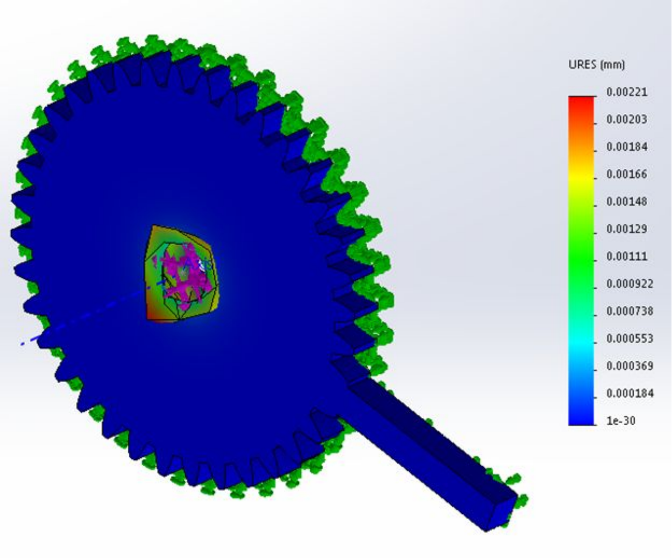


Fig 6.6 Displacement surface plot on acrylic cutout

From these analyses, we gained reassurance that our material choices of steel for the motor shaft and acrylic for the gears were feasible and unlikely to break or deform in our final design.

7. 3D MODEL AND EXPLODED VIEW

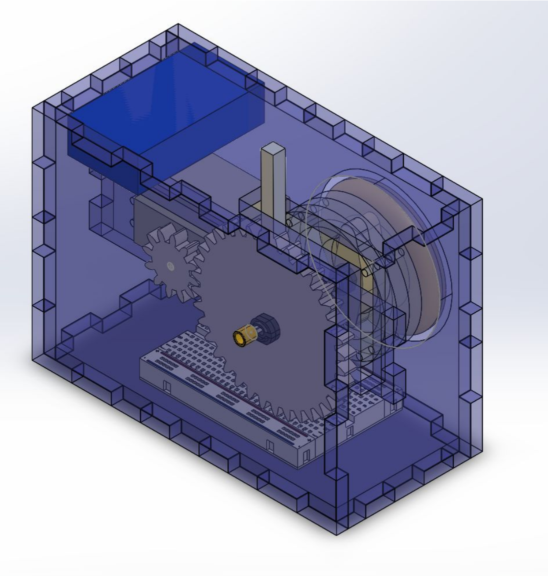


Fig 7.1 The 3D model of the device.

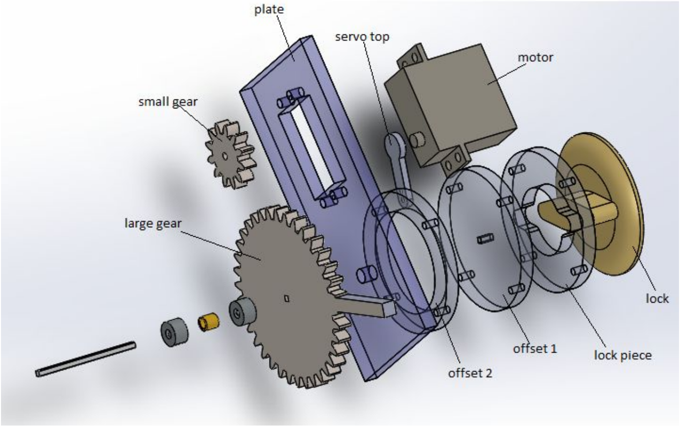


Fig 7.2 The exploded view of our device.

8. DISCUSSION, CHALLENGES AND FUTURE MODS

Some challenges we faced were creating a compact design that holds both the electronics and the mechanism that is placed directly over the deadbolt lock. Because of our constraints, we needed to make use of our limited space so that the mechanism can be used modularly. As mentioned earlier, we also had a complete redesign of our mechanism because the friction between slots and the screws in the previous design prevented the mechanism movement. When working on incorporating electronics, our team members with the most experience with IoT had difficulties troubleshooting our electronics and internet interface, as well as finding the right batteries to power our mechanism. We also had challenges with the batteries, incorrectly connecting them to their grounds, but through troubleshooting, we managed to discover our error.

For our future modifications, we hope to have an output to warn users about low-battery on their devices so users can remember to swap out their batteries. This was originally attempted by monitoring voltage of the microcontroller's GPIO pins: as the connected battery's voltage lowers over time, the output decreases from the original spec of 3.3V and the magnitude of this decrease can be translated to battery percentage. There are currently two batteries in the design, but ideally this would be reduced to a single battery for easy charging. This could be accomplished by using a switching regulator to down-regulate the 7.4V battery to 3.3V for the board to use, while also having the 7.4V for the motor. We also want to power the servo motor only when is it in use, by using transistors to accomplish this. This way, we save energy and the lock can still be unlocked by the user's key (when the servo is powered, it resists movement so a transistor is necessarily to programmatically disconnect the servo, thereby allowing the latch to be moved with a key). Lastly, we want to create a way to customize the wireless deadbolt unlocker for different lock sizes. We only designed it for the lock we acquired, but in the future, we can explore ways to calibrate our design for different deadbolt locks.

The wireless deadbolt unlocker, even though it went through a redesign, was successfully built, and we managed to incorporate all of our essential requirements and specifications while also almost incorporating many of our desirable requirements as well. As mentioned, because for this class we are focusing primarily on creating a working mechanism, we did not focus on efforts in finishing all of our desired requirements. These requirements, such as warning the user of low-battery or using a transistor to disconnect the servo motor when it is not in use, can be easily fulfilled in future modifications. Overall, we were successful in reaching our goals for this project, even though we had several challenged along the way.

9. ACKNOWLEDGEMENTS

We, the authors, would like to acknowledge the UC Berkeley Mechanical Engineering teaching staff and lab technicians that made our project possible. We would like to give specific thanks to Professor Ken Youssefi and Graduate Student Instructor Jackie First for organizing the project and giving guidance through the project.

10. REFERENCES

[1] "Dead Bolt." ***Wikipedia,*** Wikimedia Foundation, 12 Nov. 2018, en.wikipedia.org/wiki/Dead_bolt.

[2] Smith, S. and Marchesini, J., 2008, Craft of System Security, Addison-Wesley Professional, MA.

11. APPENDIX

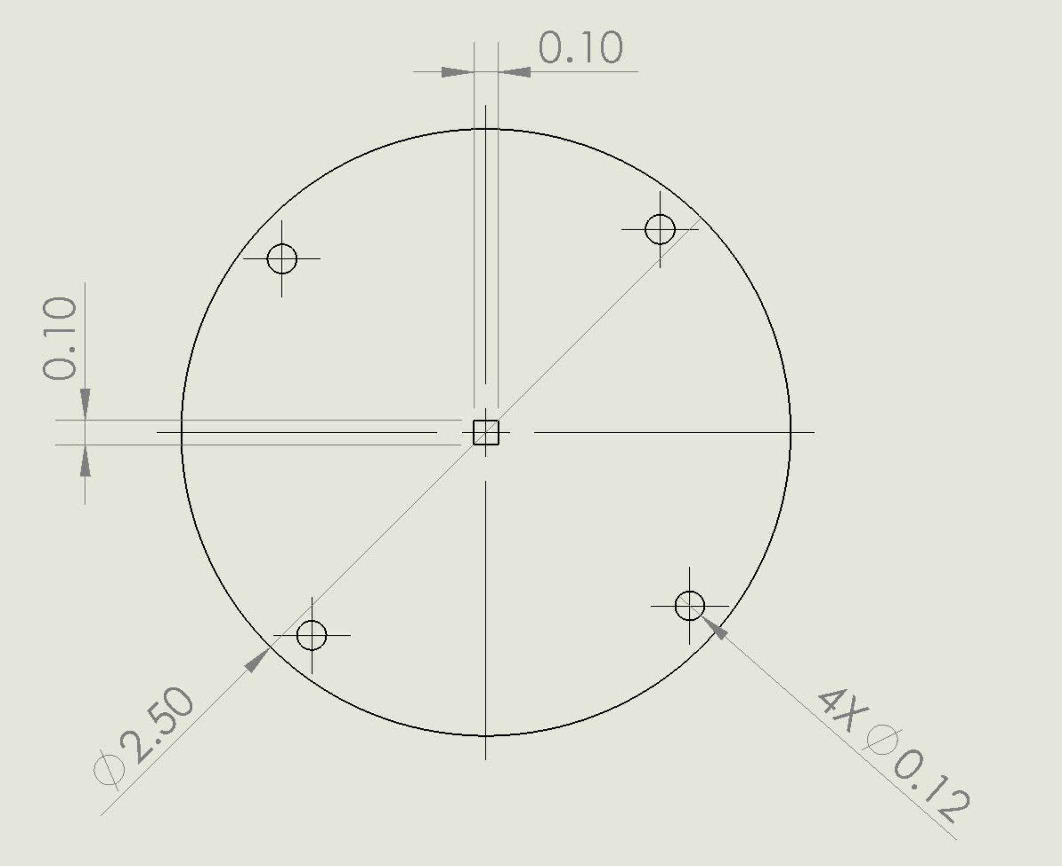


Fig A1. Offset 1 for the device-lock interface

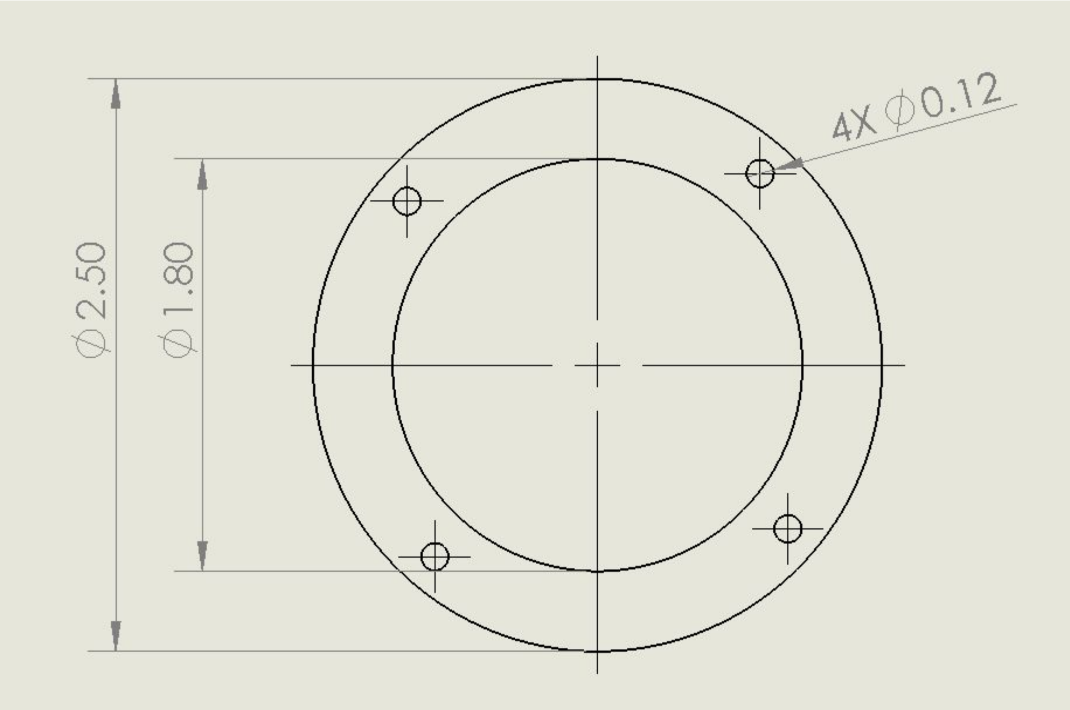


Fig A2. Offset 2 for the device-lock interface