Oregon Institute of Technology

Automated Smart Blinds

Design, Fab, Test, Improve

*Authored by*:

Chad Revel

Drew Deriata

Michael Roberts

# Table of Contents

Contents

[Table of Contents 1](#_Toc41817098)

[Design - Stepper Motor 1](#_Toc41817099)

[Hardware 1](#_Toc41817100)

[Software 2](#_Toc41817101)

[Fabrication 2](#_Toc41817102)

[Testing 3](#_Toc41817103)

[Hardware 3](#_Toc41817104)

[Software 3](#_Toc41817105)

[Analysis/Interpretation 4](#_Toc41817106)

[Hardware 4](#_Toc41817107)

[Software 4](#_Toc41817108)

[Improvement 4](#_Toc41817109)

[Conclusion 5](#_Toc41817110)

**Design** -Stepper Motor

Hardware

The Stepper Motor’s initial hardware design was relatively straightforward to implement. However, this was before the motors were installed onto the blinds. Once we started to install the motors, we discovered a few problems. The first problem was how was the up/down motor supposed to be installed. To resolve this issue, we cut a hole in the back of the blinds and attached the motor using screws and bolts. The second problem that arose after we installed both motors. Namely, we didn’t have the proper hardware to operate the strings that controlled the blinds going up and down. This was resolved by creating a spool that would house the strings and then attached it to the motor.

Software

The software design of the motors started with implementing the rotation. Each complete rotation of the motor takes 64 repeated patterns of 8 signals to the coils. This was done through a state machine that would keep track of how many rotations it was on during the motor’s operation. Allowing us to know exactly what position the blinds would be in. The next part was figuring out how many rotations would be needed to raise, lower, open, and close the blinds. The final part was setting up a call system, allowing the microprocessor to start and stop the motors depending on what functions were triggered.

**Fabrication**

The motors had to be connected onto two separate mechanical parts of the blinds. The first was the arm that controlled the opening and closing of the blinds. The second was the strings that operated the raising and lowering of the blinds. Attaching the first motor to the arm was done by installing a connecting socket to both the motor and the arm. The second motor required a similar socket, in addition to a spool that would house the string.

**Testing**

Hardware

Hardware testing for the motors consisted of the following steps.

1. Complete a full rotation of the motor.
2. Rotate the motor both ways to specific positions.
3. Use the motor to rotate a load (move blinds to open and closed position)

These first two steps were done by using buttons to control the direction of the motors. Once this step was complete, we were ready to put a load onto the blinds to confirm they would have enough torque to operate at a desirable capacity.

Software

In conjunction with the hardware testing, we had to make sure that the motors would be able to make a full rotation in both directions. For this test we used a state machine that would go through each of the eight steps required for a full rotation. We initially had problems with the motors only going one direction, but this was the result of human error in the code. After fixing the issue, the motors rotated in both directions. Once the motors were installed along with the other parts, we moved onto the next phase of testing. For this phase, we tested that the alert functions would trigger properly and start the motor’s function.

**Analysis/Interpretation**

Hardware

The positions of the motors were a bit unusual since we originally wanted to have both motors inside the blinds. However, this proved to be unachievable due to the blinds not enough room to house them. The second reason was not having a way to properly mount the motors inside the blinds. While the motors had holes for screws to go through and allow us to install them, we still had to make sure that all the mechanical parts would not touch other parts and prevent them from functioning properly.

Software

Once the motors were attached and fully operational, we had to make the decision to have the up/down motor only operate for a certain amount of time. The primary reason for this was that this prototype and we wanted to keep the movements shorter for quicker demonstration purposes. This was only for the up/down motor since it had to operate longer compared to the open/close motor.

**Improvement**

One of the implemented improvements was increasing the amount of time it took to complete a full rotation. This was needed since the up/down motor would stall during its raising operation. By increasing the time it took to complete a full rotation, this allowed the magnetic fields inside of the motors to generate during each step; preventing it from stalling. However, this also proved that the motors would be too slow with the time we set it to work on. After fine tuning the speed, we found 0.62 seconds the minimum amount of time needed to complete a full rotation without stalling, while providing the least amount of waiting time in order for the blinds to go up and down.

One future improvement would be to possibly have a stronger motor that would allow for a heavier weight capacity. But this would require both hardware and software alterations, thus having to rework a good portion of the project.

**Conclusion**

The stepper motors were a vital part of our project, and the most challenging part of our physical design. We did not anticipate having mechanical issues with the motor speed based on the load, so we initially set the rotational speed very low. By experimenting with the speed based on blind size, we were able to come to a compromise between speed and reliability, improving the design from its original implementation.