# ALEXA BLINDS CONTROL

**Eshwar Pamula**
**Department of Mechanical Engineering, The Ohio State University**

## Purpose

The core purpose of my project is to help me wake up early in the morning by opening my blinds automatically when my alarm rings. While there were existing options, I wanted to build my own from scratch.

## Device Background

**Motivation**
Waking up early is a habit I have been struggling with ever since I moved to America and since college. Over the winter break, I want to go back to my old habit of waking up early and starting my day on a positive note. Using Alexa and having my blinds open automatically upon sunrise, this would help me get back to old and better habits.

**Current products on the market:**
The existing solutions on the market are usually full on new blind systems or an expensive home automation tool that is out of reach for students especially.

1**. Tilt My Smart Roller Shade:** Easy to install with a solar charging option, compatible with Alexa and SmartThings via the Tilt Bridge, priced at $399.

2**. Lutron Serena Shades:** Customizable with over 150 options, works with all major assistants, starting at $448 for a 3' x 5' window.

3. **Graywind Roller Shades:** Customizable blackout curtains, compatible with Alexa and SmartThings, no additional bridge needed, priced at $158.99.

**Problems with current products:**

1. Additional hubs or bridges required for full smart home integration.
2. Complex or multi-person installation processes.
3. Limited size, style, and fabric customization options.


**Figure 1: Examples of existing products in the market.**
Clockwise from the top: Tilt Smart Roller Shades and Lutron Serena Shades

## The Design Process

The design process for your smart blinds' opener began with identifying the primary requirements: integration with **Alexa** and **Arduino** control, a **housing** for the stepper motor, and a connection mechanism to the existing blinds. Considering space constraints and the need for a seamless look, initial sketches and 3D models were created.

The first iteration involved a simple box housing for the stepper motor, with subsequent versions adding dedicated entry and exit points for the wiring, ensuring a clean setup and preventing wire tangling.

The integration of a mini breadboard atop the housing box allowed for easy adjustments and maintenance of the circuitry. The **Arduino** and **ESP** module required a specialized housing, leading to a tailored 3D printed case that allowed for secure placement and easy access to the pins for wiring.

One of the critical design challenges was creating a **connector** between the stepper motor and the blind's opener to ensure a robust and reliable connection, finally settling on a **3D printed** tube that could attach firmly to both the stepper and the blind opener, providing the necessary torque without slippage.

| Functional Requirements | Non-functional Requirements |
|---|---|
| 1. The device must open and close blinds via a stepper motor controlled by Arduino.<br>2. It should receive commands through Alexa voice control.<br>3. The design must incorporate an IR receiver for remote signal input.<br>4. ESP Module must be connected to the local WIFI to receive a signal. | 1. Aesthetic integration with the apartment setting.<br>2. Maintain a compact form factor for the housing box and components.<br>3. Ensure user-friendly installation and maintenance. |

**Figure 2: Functional and non-functional requirements**

## Major Design Components

| | |
|---|---|
| Arduino and ESP Circuit | The function is to get the stepper motor on command. |
| Stepper Motor Housing | Houses the stepper motor ensuring all the wires are safe and stabilizes the motor using screws. |
| Arduino Housing | Protects the electric circuit from any unforeseen circumstances. |
| Blind to Stepper Connector | The main connector to the stepper motor to ensure the blind opens smoothly upon Alexa's command. |

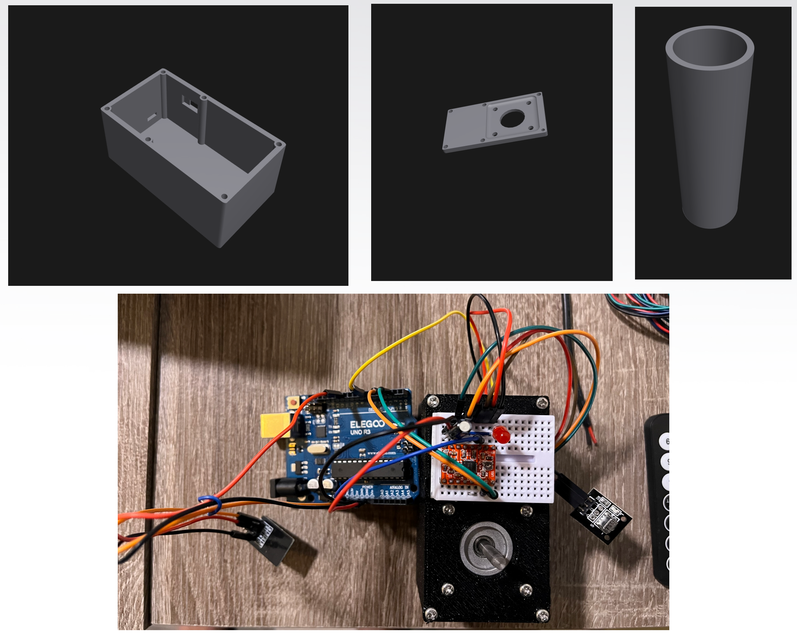

**Figure 3: Alexa Blind Control Mechanical Components**
Clockwise from the top: Stepper Motor Housing, Top Plate for Motor Housing, Blinds Connector, and Circuit.

## Testing and Analysis

To test the smart blinds opener, we conducted a series of streamlined procedures:

**Functionality Test:** We had individuals experienced with smart home technology, as well as novices, interact with the device using Alexa and the IR remote to ensure it correctly executed commands. The device was deemed acceptable if it successfully responded to 95% of commands.

**Durability Test:** Technicians subjected the device to repeated open-close cycles to mimic extended usage, with an acceptance benchmark set at 1,000 cycles without failure.

**Connectivity Test:** The development team tested Bluetooth and IR signal consistency over a standard range, expecting no more than a 5% signal drop-off to maintain acceptable performance.

The results were positive, and we noticed only a slight code snippet needed to be changes to use this commercially.

**References and Acknowledgements**

*Electronic Wings: Interfacing ESP8266 Module with Arduino Uno*

*Sinric  Pro and Amazon Alexa*

*Shahil Merchant (OSU Class of 2025)*