Group 4: Auto Sun Visor

BLAIR HUANG, XIAOYANG TIAN, SIYING WANG

Contents

INTRODUCTION

AUTO SUN VISOR

- Our group designed an auto-adjustable sun visor that can change position based on the direction of the sun relative to the vehicle.

PROBLEM

Sun glare causes approximately 9,000 traffic accidents annually in the United States based on data of the National Highway Traffic Safety Administration

Sun glare impairs visibility

Adjusting the sun visor while driving can lead to distraction and reduced visibility, which increases the risk of accidents, including potential collisions with pedestrians.

SOLUTION	Advantage	Deficiency
Polarized Sunglasses	reduce sun glare in certain areas	• Inconvenient: Drivers who already need prescription glasses need to customize prescription sunglasses or add another polarized filter • Not effective: Reduced visibility/still cannot prevent glare from side
Electrochromic Glasses	effectively reduce sun glare	Expensive

Auto Sun Visor

- Safe choice for drivers to avoid manually modifying the visor position.
- Provide a compromise between the expense of photochromic glasses and the manual adjustment of sun visors or wearing sunglasses.
- The visor blocks the sunlight accordingly under the reaction time that won’t let people’s eyes be exposed under sunlight.

HIGH-LEVEL REQUIREMENTS

1. A 12v DC power should be able to supply the entire system, while a backup power supply connects to the remaining systems within 3 seconds if the vehicle's power source fails, ensuring the visor returns to its original position when DC power is cutting off.

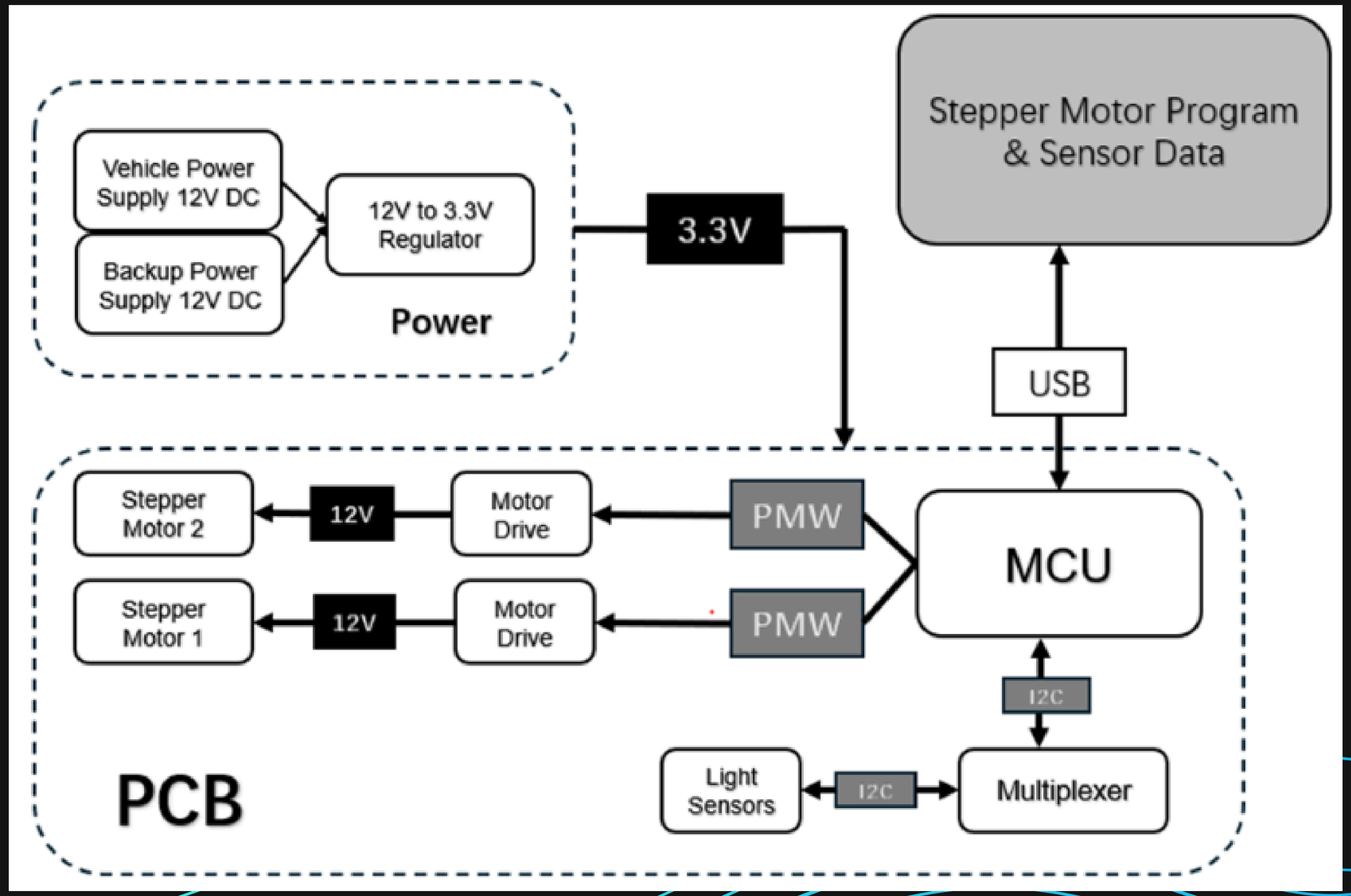
2. The microcontroller need to provide a proper visor angle to cover the sunshine based on the light sensors input.

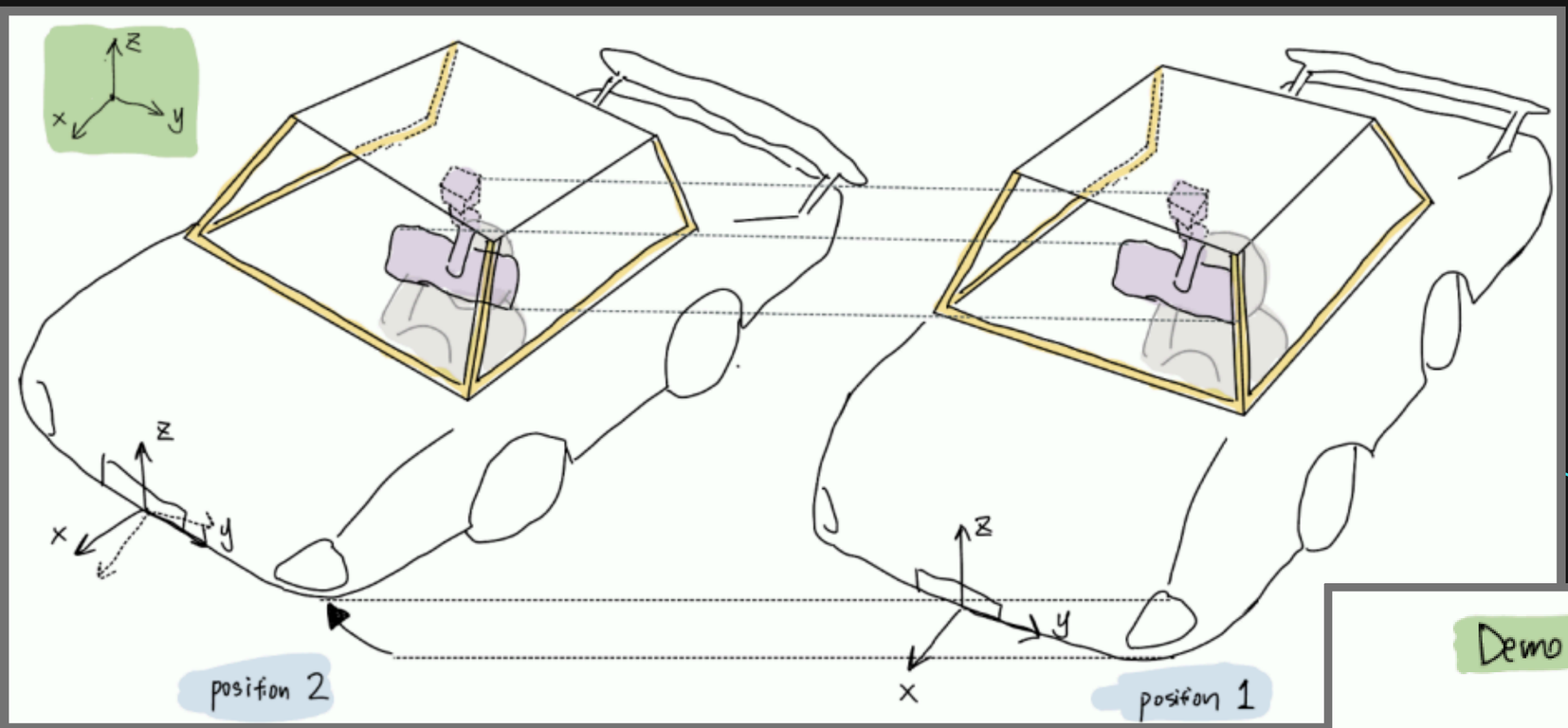
HIGH-LEVEL REQUIREMENTS

3. The light sensor must be able to indentify different level of light intensity (lux) as data input and correctly input the data into the MCU unit within 1 second.

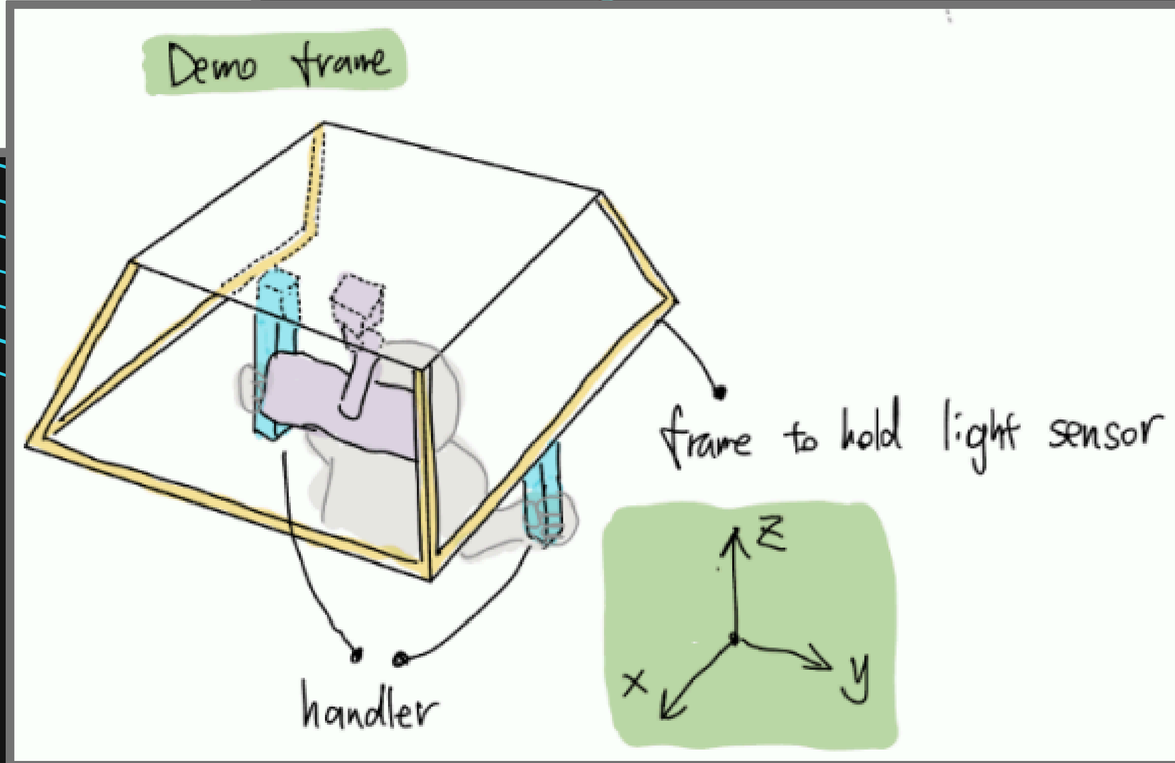
4. The motor should be capable of moving the shading board to the correct position within 1 second after receiving a move command.

BLOCK DIAGRAM



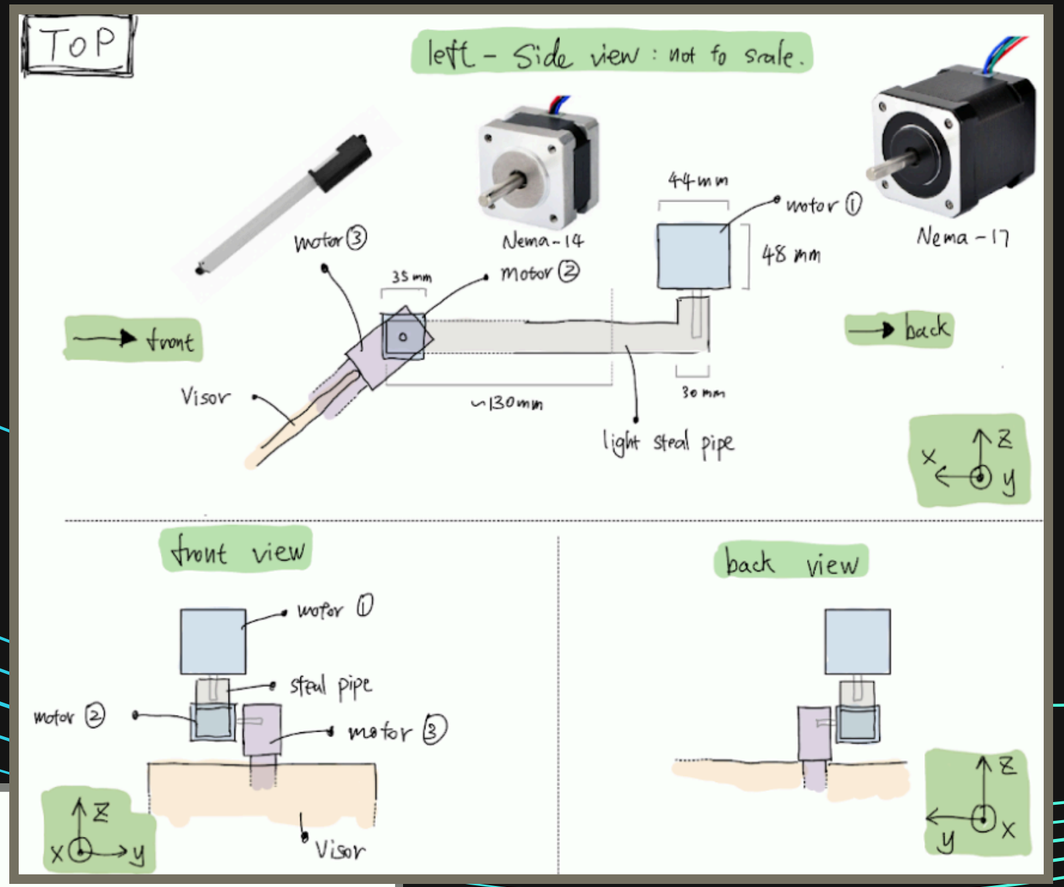


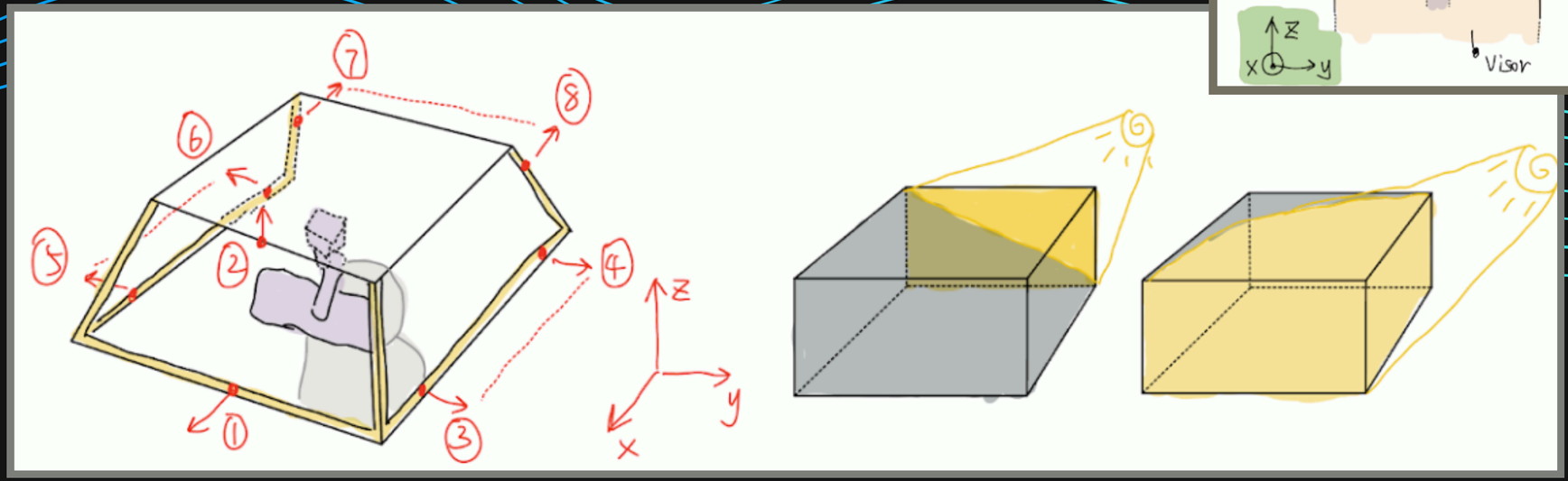
To accommodate demo, we made the car frame design, although we made some modification latter of the demo frame...



PHYSICAL DESIGN

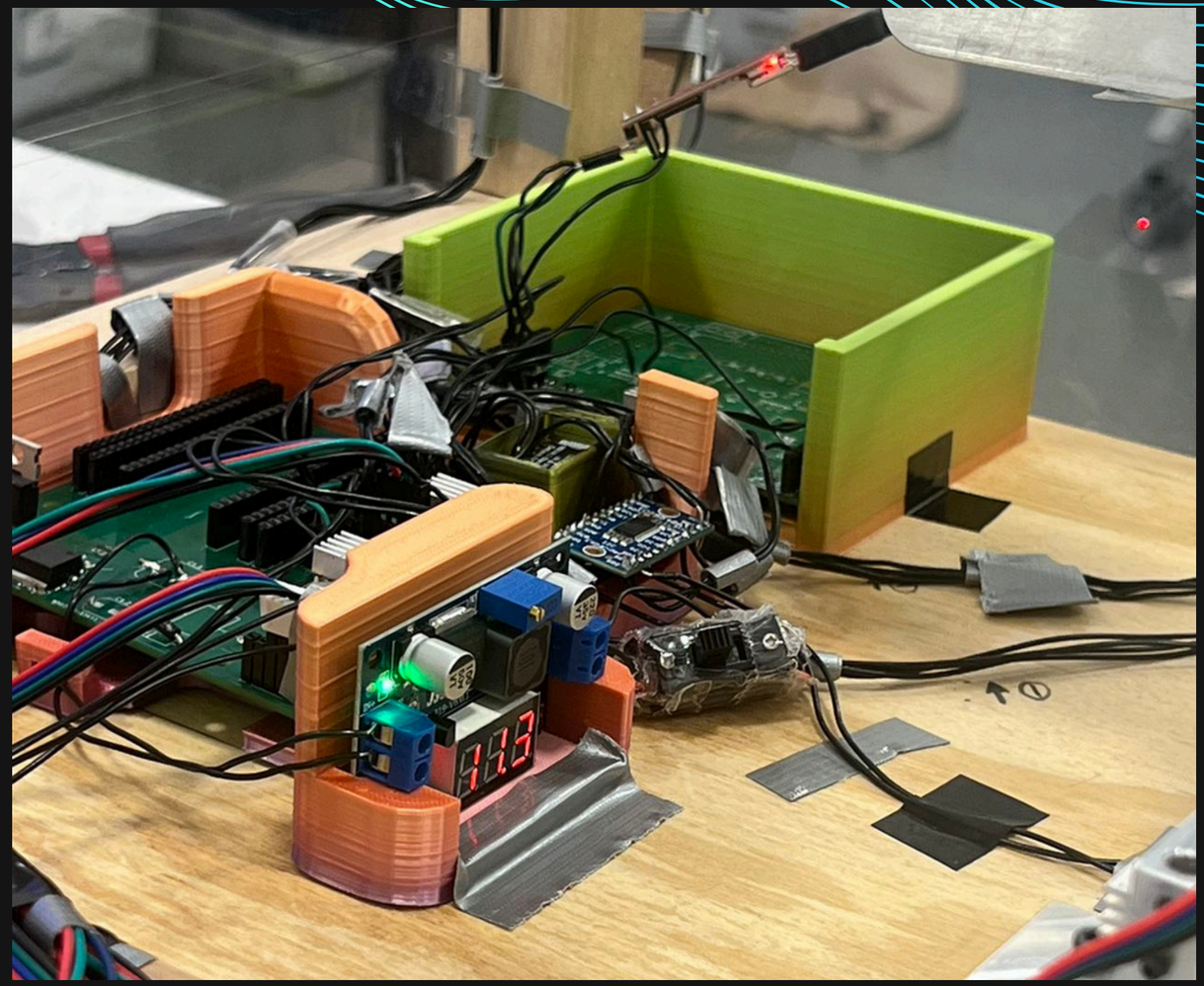
PHYSICAL DESIGN





Changes in Design

1. Delete the Linear Motor
2. Use regulator Modules.
3. Change the 14-NEMA to 17-NEMA
4. Change the demo frame from hand hold to desk demonstration.

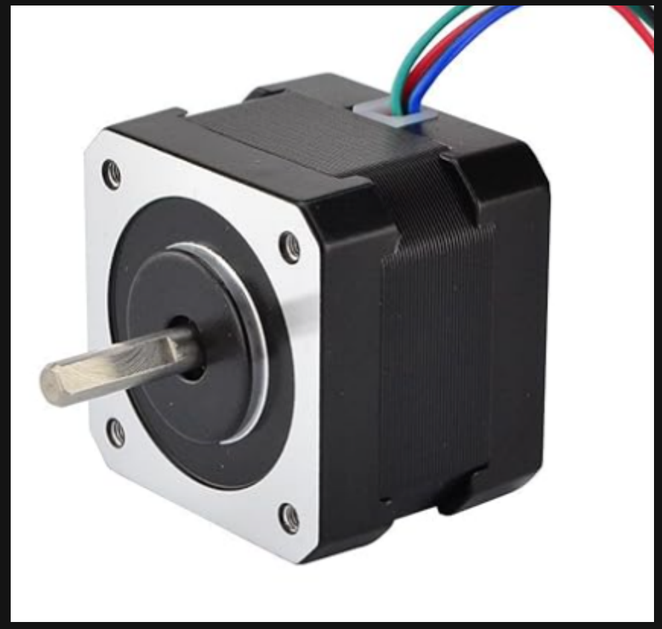


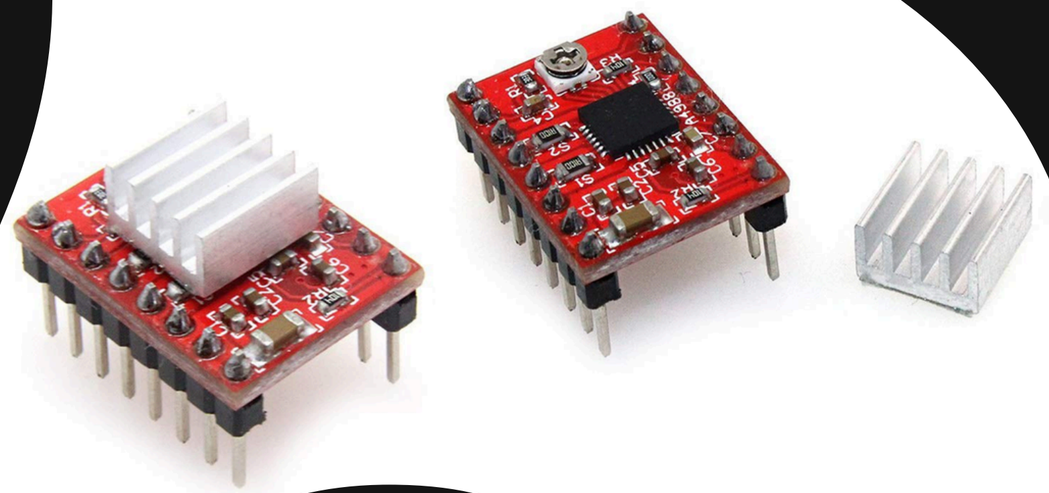
Subsystems: Motor Subsystem

2 Motor Dirvers and 2 Stepper Motors

Motor Driver A4988: operate motors

Stepper Motor N17: move the shading board

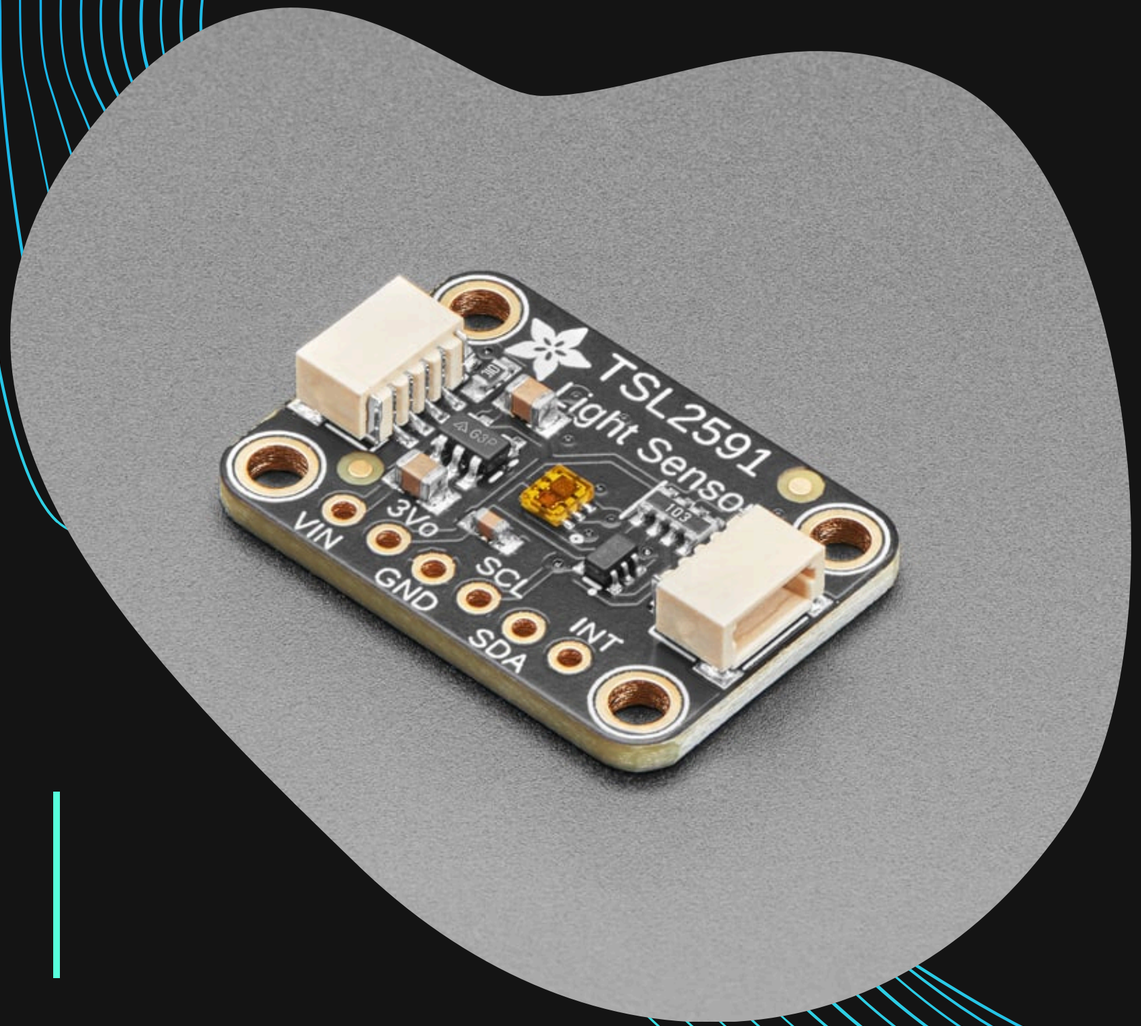




Requirements	Verifications
• Be able to react to the order from control unit within 2s. • Be able to change to different angle with given order from control system.	• Power up the motors and ensure each motor works well (just the basic check whether motors work) • Connect with the microcontroller PCB board. Use the backend to send orders. Make sure the motor can move. • Move within 2s

Subsystems: Sensor Subsystem

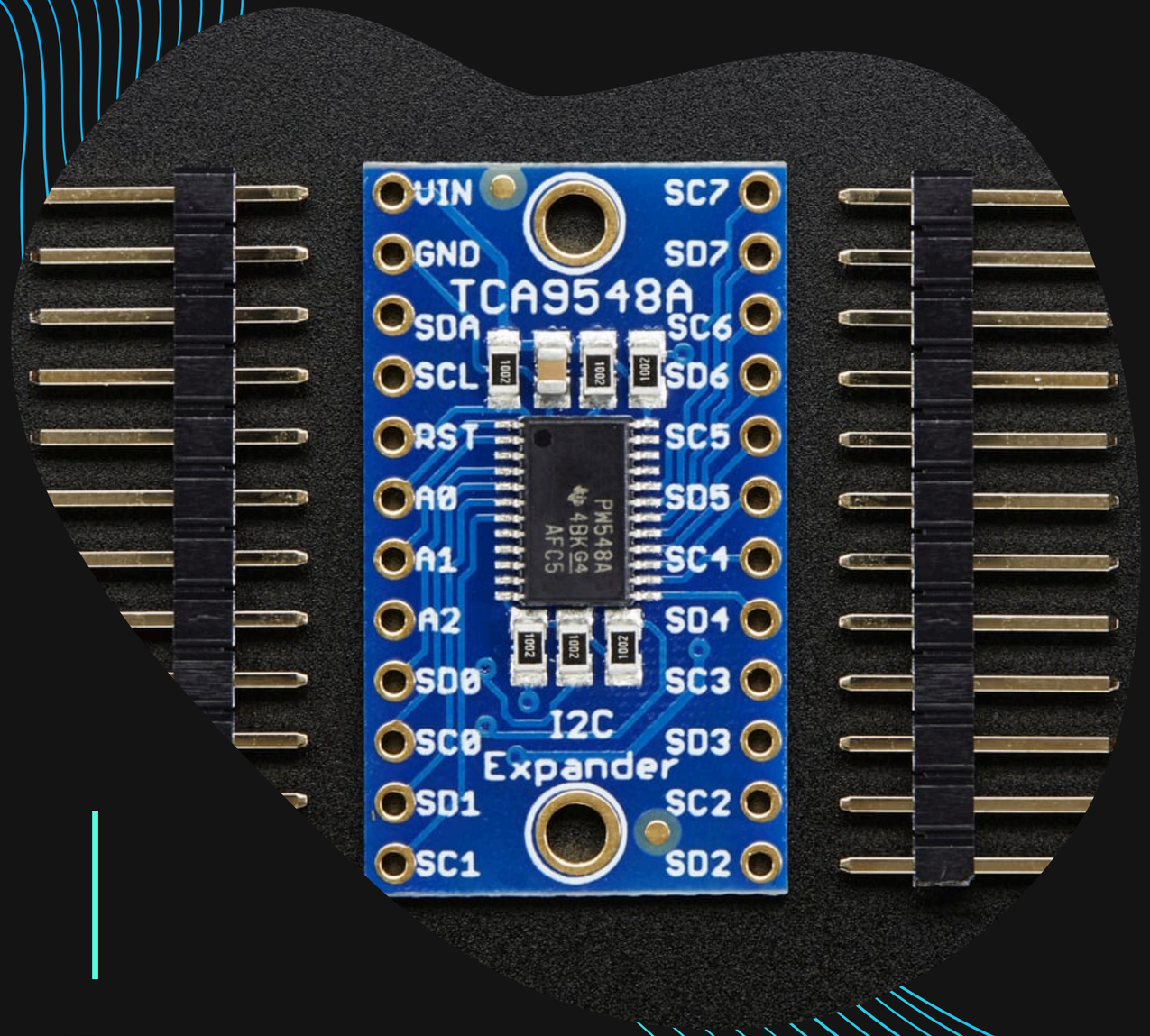
8 Sensors and 1 Multiplexer



- 7 light sensors will be placed at different positions at the frame.
- These light sensors will receive light intensity information, enable the control system to determine how the visor should rotate.
- 1 light sensor will help is decide whether the backup power is used based on the light intensity of LED.

Subsystems: Sensor Subsystem

8 Sensors and One Multiplexer

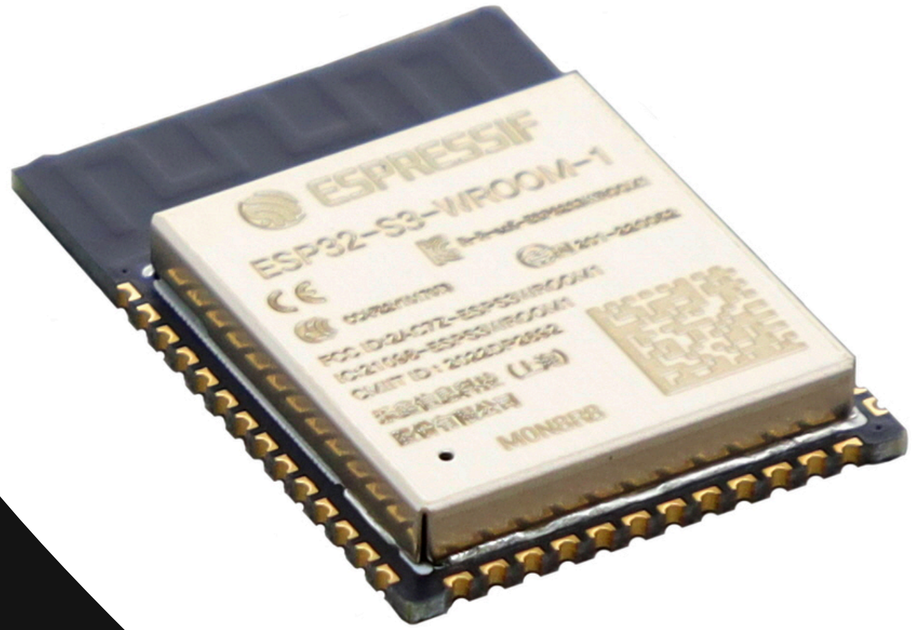


- Multiplexer: let 8 same-address light sensors hooked up to one microcontroller. It works like a gatekeeper, which shuttles the commands to the selected set of I2C pins with command.

Requirements	Verifications
Be able to detect light intensity changes within 1s	• Power up the light sensor and connect all of them to the backend. • Try covering by hand or shining a lamp onto the sensor to experiment with the light levels. • Check the data from the backend and make sure the lux output is valid. • Repeat step 2 and measure response time. Ensure the sensor takes less than 1s to give feedback.

Subsystems: Microcontroller

ESP32-S3-WROOM-1



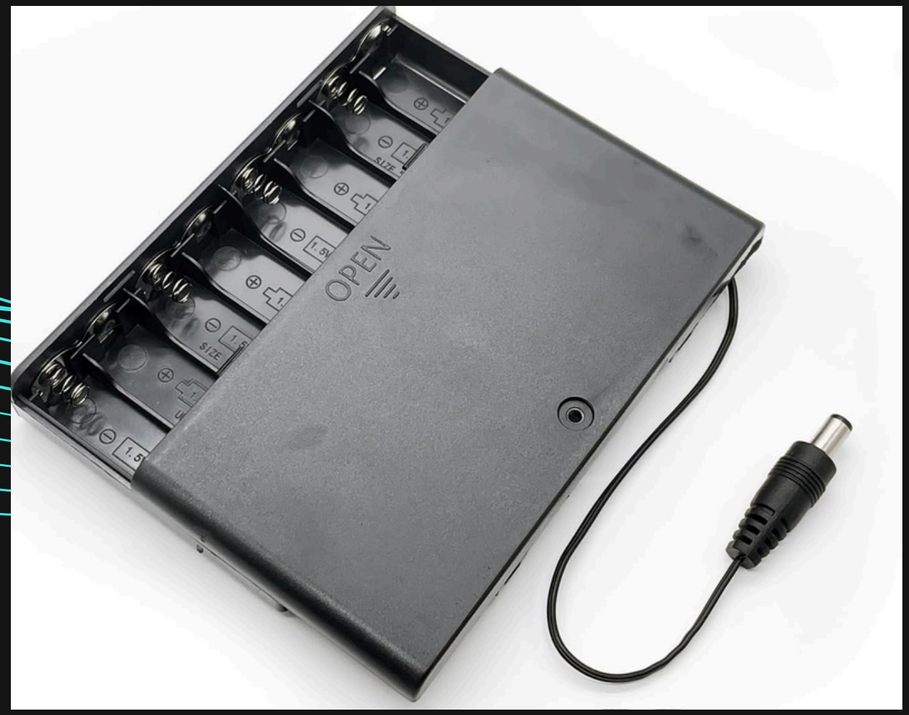
The microcontroller receives the output data from the 8 light sensors. The output of the sensor chip should be lux thus our receiving microcontroller should be able to read in lux, and our programing algorithm also in lux units to reflect. Then, it transmitted commend via cable to the motor drivers to control each motor.

Used ardurino to program it.

Requirements	Verifications
• Be able to receive digital signal from sensor unit continuously. • Be able to send order to motor with given light intensity from sensor unit. • Be able to process input lux and send order with 2s (motor should begin to move after light data change).	• Power up sensor and PCB board and connect sensor and PCB together. • Waving hand on the sensor and printing out the input of PCB at the backend. • Connect the motor with the PCB board and send a simple order to make the motor move. • To check whether the program is correct, use the model frame and specific light source. Then verify the board is in the correct position.

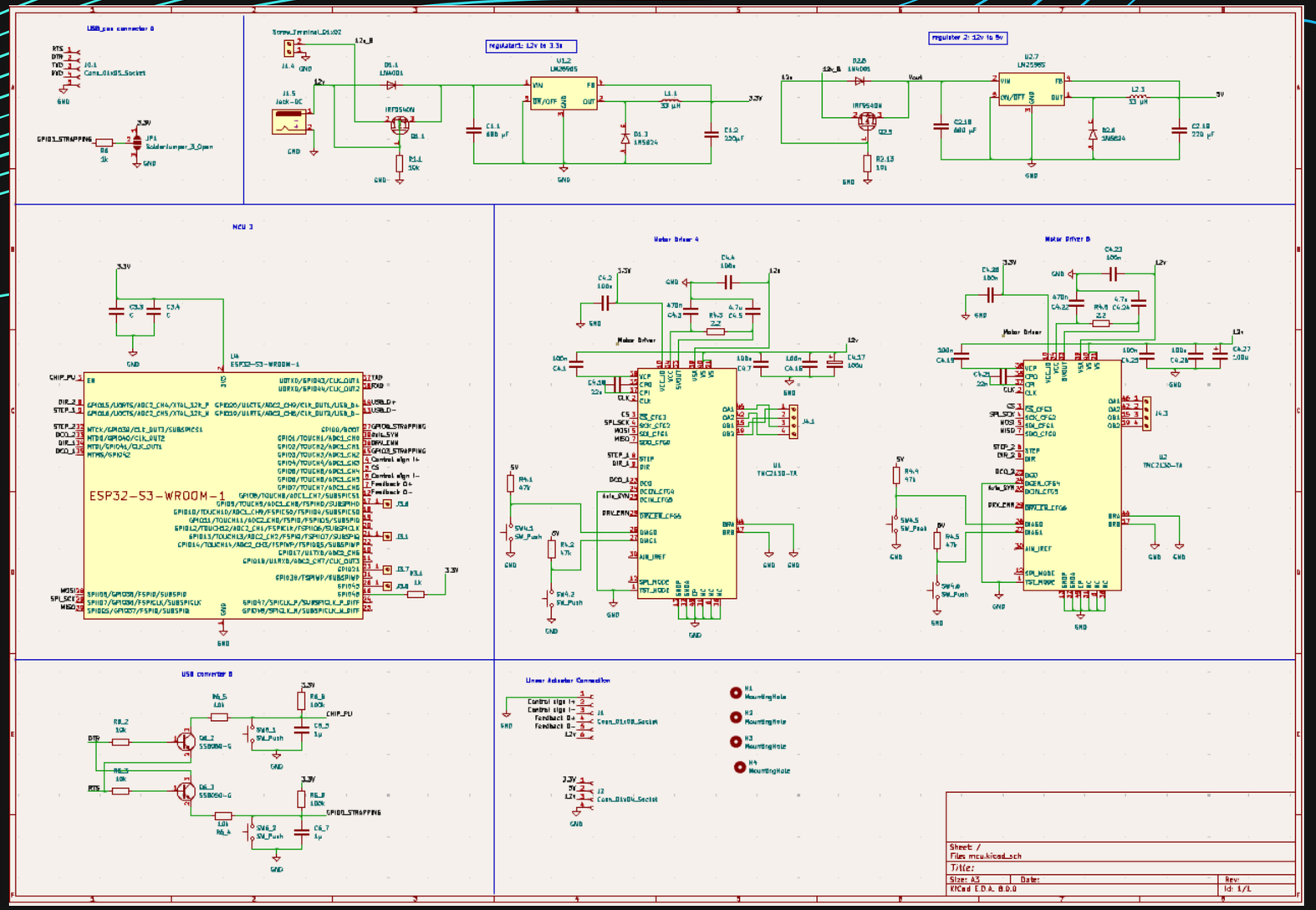
Subsystems: Power Subsystem

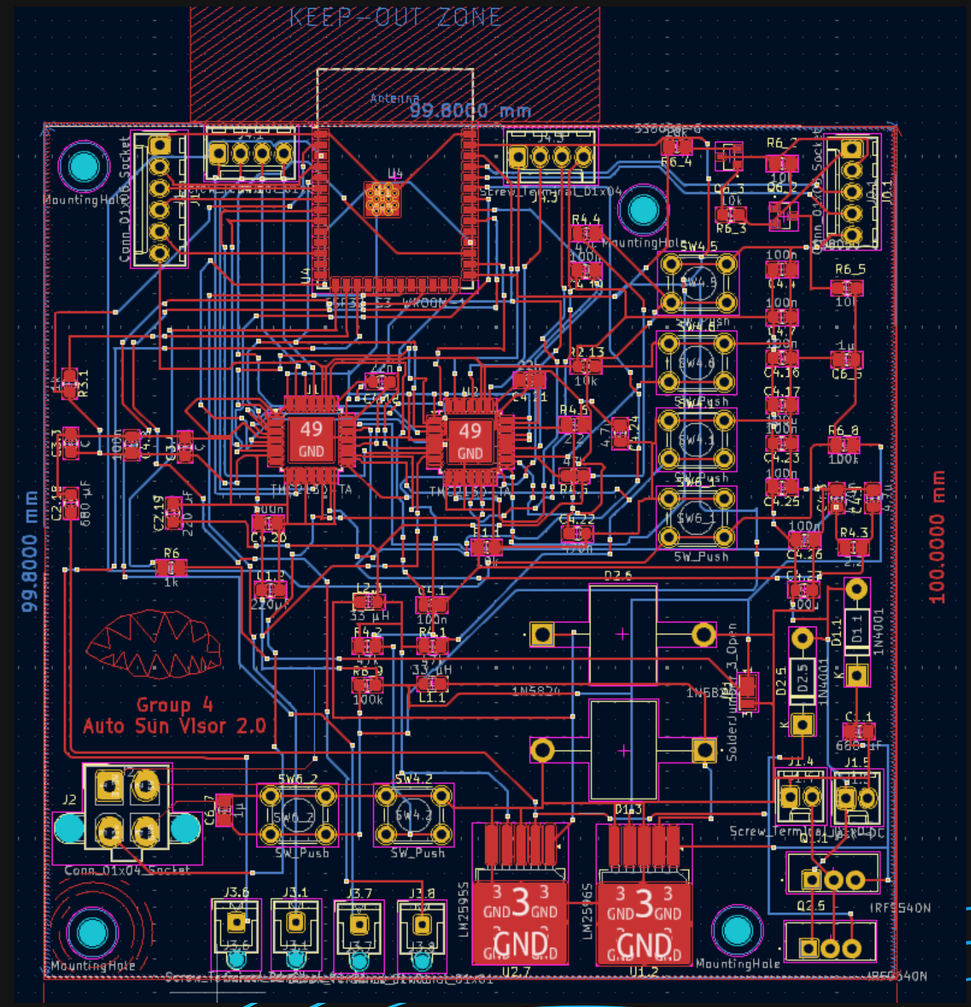
12V DC Power Supply: simulate the real vehicle electricity.
It can charge the whole visor system with steady power.



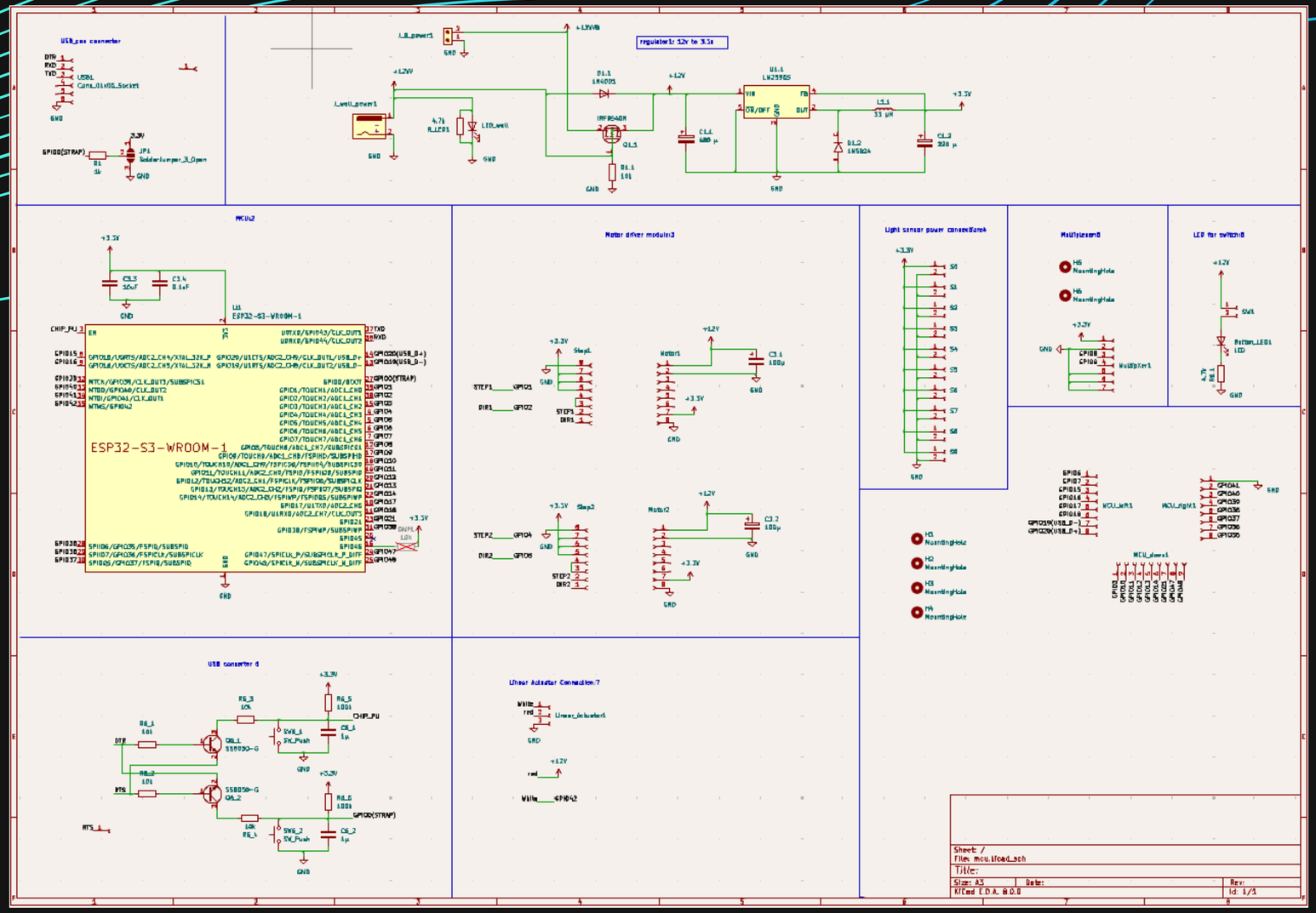
12V DC backup battery: work as the backup power supply. Should be able to connect to the remaining systems within 1 seconds in case the vehicle's power source fails to support the system when the car engine is cut off.

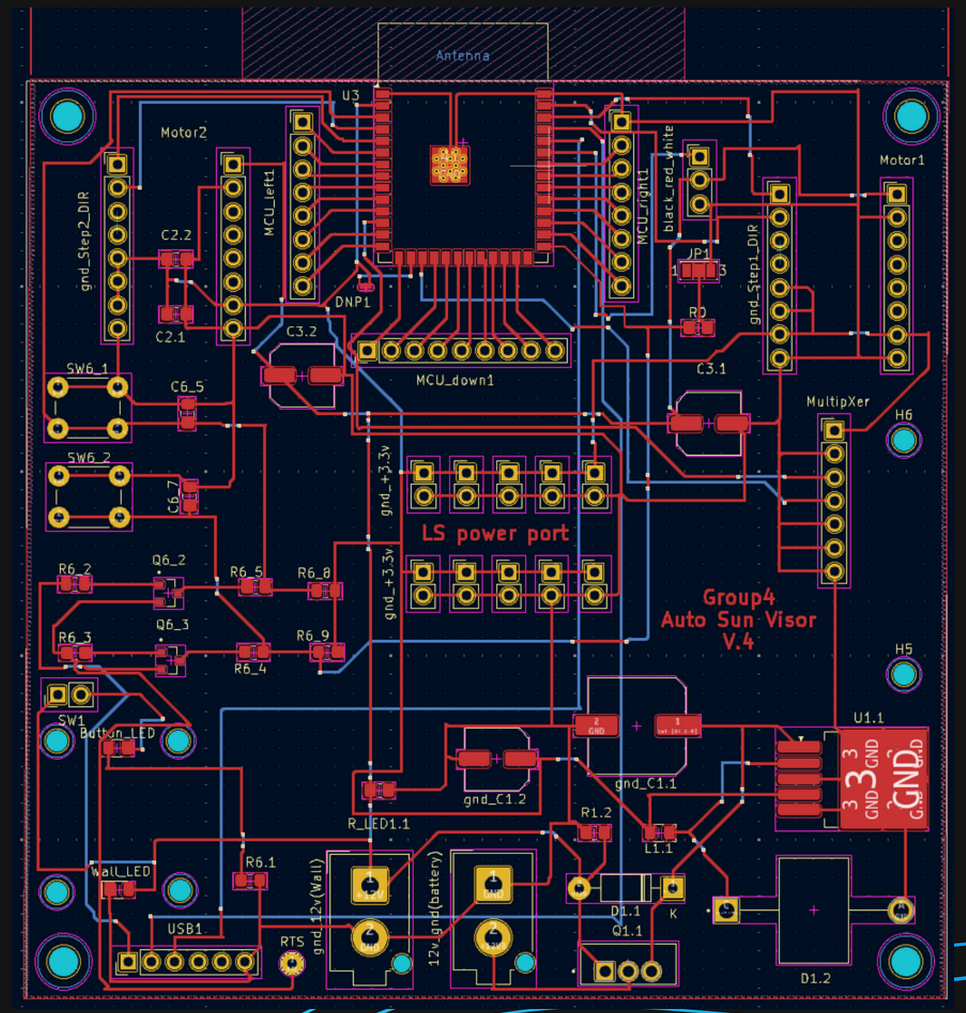
Requirements	Verifications
• Be able to provide 11.5V to 12.5V power supply to make all the subsystems work. • Be able to switch from car providing power to back up power when vehicle’s power fail	• Measure the voltage of every branch and item to check whether they match with the voltages. • Shut down the car providing power and measure the magnitude at every branch and item to check whether backup power works successfully.



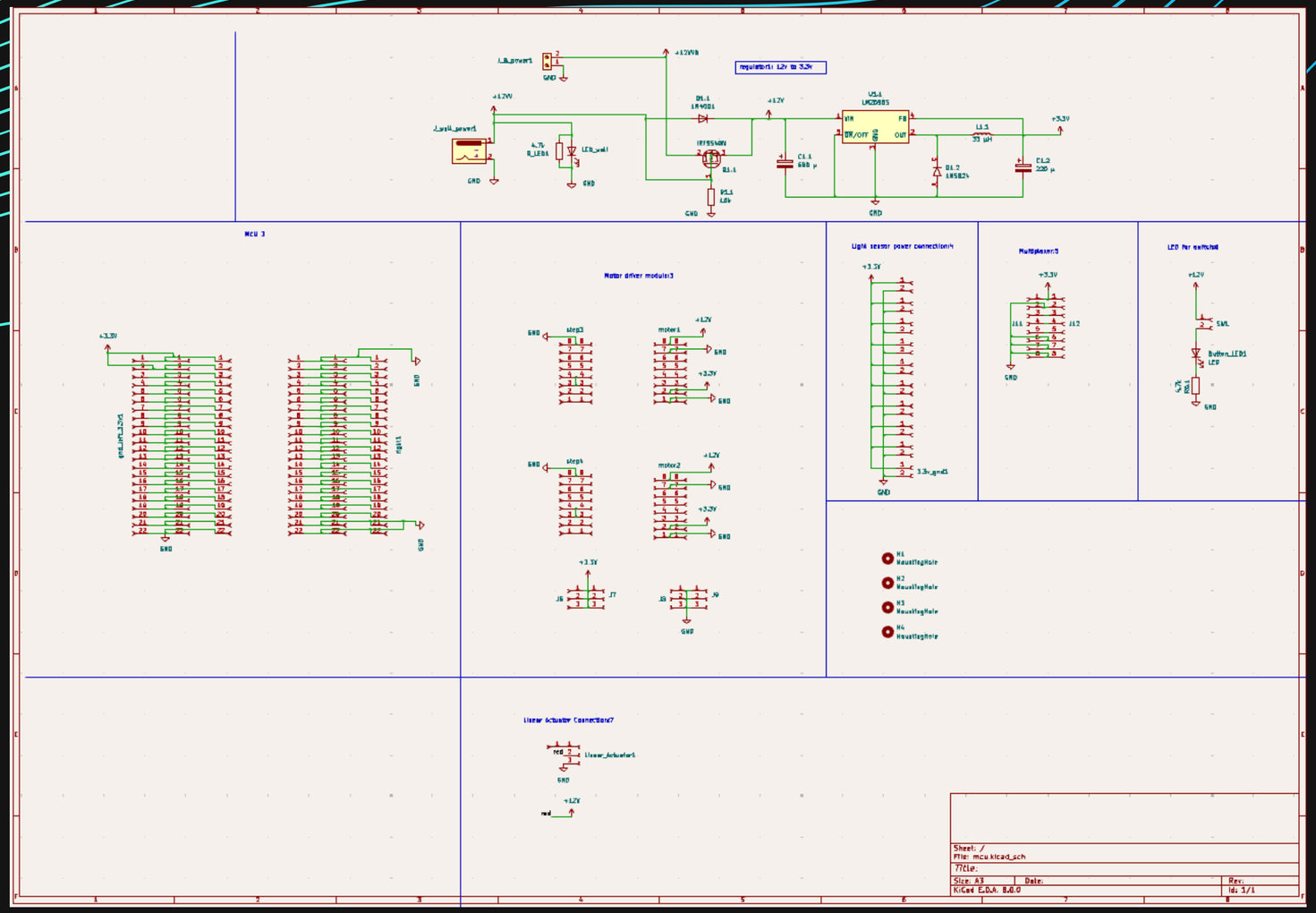


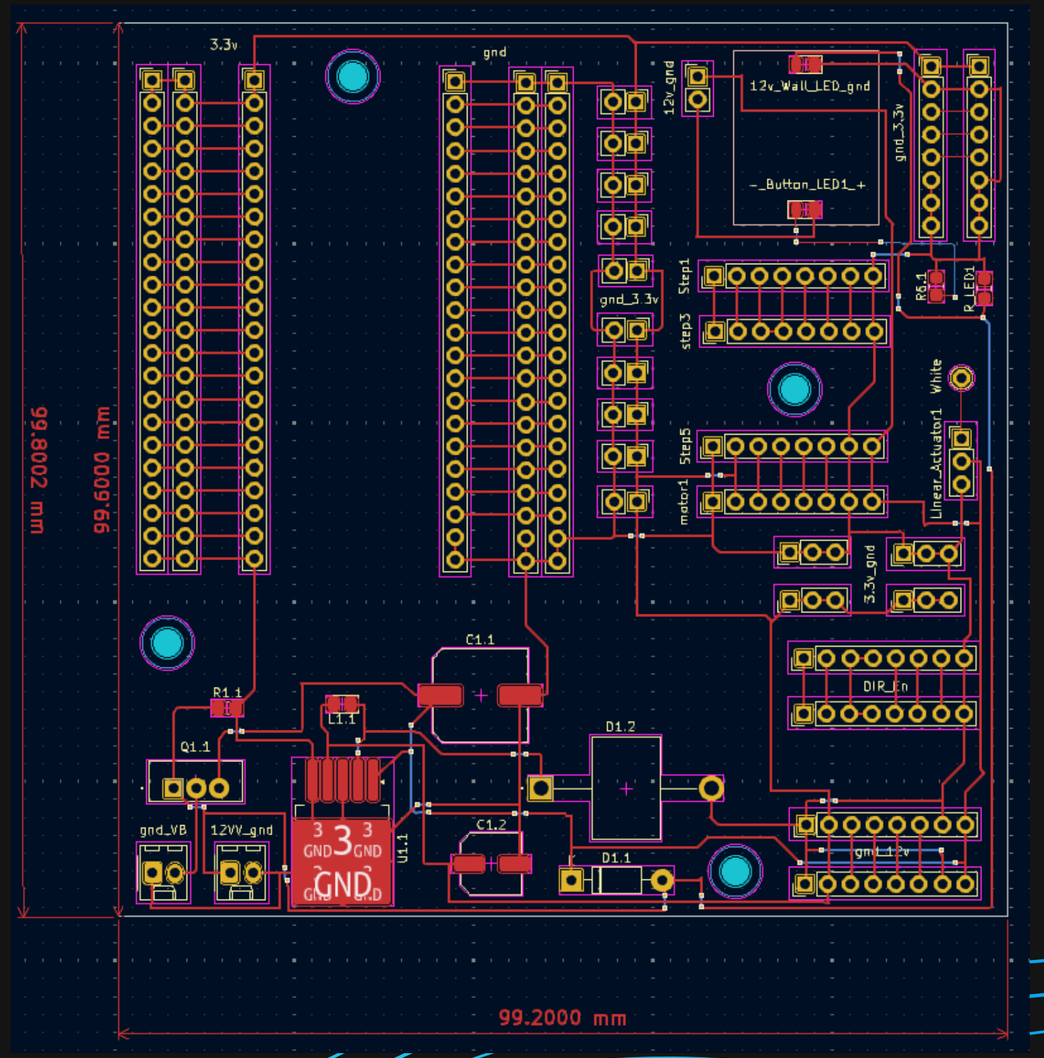
Schematic & PCB design





Schematic & PCB design

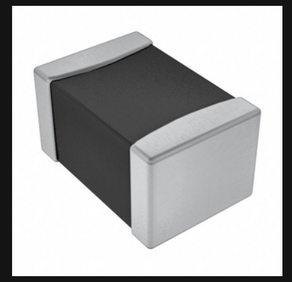




Schematic & PCB design

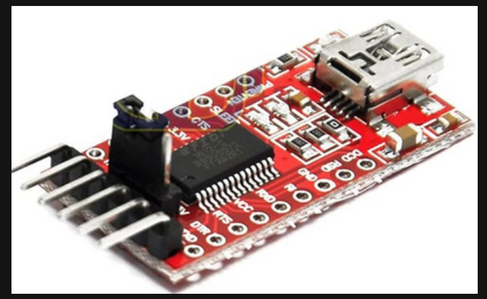
Test Result (OFFCIAL VIDEO: HTTPS://WWW.YOUTUBE.COM/WATCH?V=LTPQOFYQXXU)

Challenges



In the 12V to 3.3V regulator circuits, the 33uH inductor always burns. It may be caused by the large current and high temperature through the inductor.

We incorrectly estimated the torque needed to support the pole and shading board so that the motor connected to the top was unable to move the shading board.



ESP32-S3-WROOM-1 chip doesn’t work in some PCB boards because of the bad connection in the soldering and the bad connection of the USB-to-UTART adapter.

Conclusion & Future

THANK YOU