S.T.A.R.
SPOTLIGHT TRACKING WITH AUTOMATED RECOGNITION

Group 7 Authors

Gage Anderson
Optical Engineering

Travis Nguyen
Optical Engineering

Lucio Villena
Computer Engineering

Jerison Lau
Electrical Engineering

Review Committee

Dr. Jaesung Lee	ECE	Professor
Dr. Wei Sun	ECE	Professor
Dr. Stephen Eikenberry	PSE	Professor
Dr. Andrew Klein	PSE	Post Doc.

Advisors

Dr. Chung Yong Chan	ECE	Senior Lecturer
Dr. Aravinda Kar	CREOL	Professor

Table Of Contents

List of Tables

List of Figures

Chapter 1 - Executive Summary

The Spotlight Tracking with Automated Recognition (S.T.A.R.) system is a senior design project developed at the University of Central Florida by a multidisciplinary team of students in optical, electrical, and computer engineering. The project addresses a long-standing challenge in live entertainment, presentations, and theatrical productions; maintaining consistent, accurate spotlight tracking on performers without requiring a dedicated operator or expensive lighting console.

Traditional stage lighting relies on human operators or complex, high-cost control systems, both of which introduce limitations in precision, repeatability, and accessibility. S.T.A.R. presents an innovative solution by combining computer vision, real-time image processing, and motorized optics to create an autonomous moving-light system capable of following a performer dynamically. Using a CMOS imaging sensor and trained machine-learning algorithms, the system continuously identifies and tracks subjects on stage, directing a high-intensity spotlight via a two-axis motorized gimbal. This automation enables smooth, low-latency motion, achieving response times under 500 milliseconds while preserving consistent illumination on the selected performer.

The optical subsystem is modeled after the ETC Source Four ellipsoidal reflector design, a professional industry standard known for its efficiency and image quality. The S.T.A.R. system integrates a biconvex BK7 focusing lens and replaceable light source, initially using an HPL 757 halogen lamp with provisions for an LED upgrade. Motion control is achieved through high-torque servo motors driven by an ESP32-S3 microcontroller, while the Raspberry Pi Compute Module 5 executes real-time image recognition using the YOLO framework within OpenCV. Together, these subsystems enable optical precision, responsive tracking, and wireless user control via a mobile or web interface.

The system is designed for a variety of use cases, including theater performances, concerts, lectures, and live events, where it can autonomously follow presenters or performers with minimal setup. Its small, efficient design allows quick set-up and integration into pre-existing lighting networks for professional venues while remaining affordable and user-friendly for smaller productions and educational environments.

This document outlines the complete design process of the S.T.A.R. system, including the theoretical foundation, hardware and software research, and rationale behind key component selections. It details the architecture of the optical, electrical, and computational subsystems, supported by schematics and simulation data. Standards and design constraints relevant to safety, communication, and performance are discussed, followed by documentation on prototyping, fabrication, and testing. Administrative content such as the bill of materials, cost analysis, and project timeline is also presented. The report concludes with a summary of system achievements, lessons learned, and proposed improvements for future development phases.

Chapter 2 - Project Description

2.1 Background and Motivation

Stage productions, concerts, and presentations rely on spotlighting to direct the audience's attention to the appropriate subject and enhance performance quality. Traditionally, this task can be achieved via two options; a trained spotlight operator manning a heavy, large spotlight, or a costly lighting console that needs a programmer to operate [1]. While effective, these methods present limitations: consoles often cost upwards of $20,000, spot operators add to labor expenses, and manual control can introduce inconsistencies in timing and accuracy.

Advances in computer vision and security applications open new possibilities for automated spotlighting systems [2]. By utilizing imaging sensors, real-time subject tracking, and mobile control platforms, an autonomous spotlight can be made both affordable and reliable. There are many applications and systems where precision human tracking is essential for operation; utilizing these systems will be crucial in developing an accurate system.

The S.T.A.R. project aims to address these challenges. The system uses a CMOS camera and real-time human recognition algorithms to track performers on stage. A motorized gimbal directs the spotlight toward the tracked subject, ensuring consistent illumination. In addition, S.T.A.R. integrates wireless control via a mobile app or remote interface [3], eliminating the need for expensive lighting boards. S.T.A.R. demonstrates the feasibility of creating an accessible, user-friendly spotlight system by combining off-the-shelf imaging hardware, embedded processing, and motor control.

2.2 Existing Products & Related Work

CHROMA: Color Harmonization from Skin Tone Recognition

CHROMA was a previous Senior Design project completed at the University of Central Florida that focused on improving lighting conditions for photography. The goal of the project was to create a smart lighting system that could automatically adjust brightness and color temperature based on a subject's skin tone. Using a small camera called the Image Capture Unit (ICU), the system analyzed the subject's image and adjusted nearby LED panels to produce accurate and consistent lighting. This process helped photographers capture more natural-looking images, especially in situations with varying skin tones or inconsistent light.

The CHROMA system used a Raspberry Pi 5 Model B as its main controller and featured several RGBCCT LED panels capable of producing both warm and cool color temperatures. A wireless connection allowed communication between the camera and the light panels, so adjustments could be made in real time. The project demonstrated how combining image processing, adaptive lighting, and color science could improve photography and make it more accessible to beginners. CHROMA's design approach is an example of how optical systems and embedded electronics can work together to automate lighting control and reduce the need for manual adjustments.

CHROMA could integrate into our S.T.A.R project in later iterations. Having a moving spotlight that could optimize colors depending on skin tone would be a groundbreaking innovation in stage lighting technology. Though CHROMA was designed for lighting panels in photography, it could still have a great impact in theatre, as optimizing the appearance of your performer is crucial. There would be great difficulty during implementation though, as the size of our light engine is significantly smaller than the one that was utilized in CHROMA. The imaging component though, is already built into S.T.A.R

Follow-Me 3D

Follow-me 3D is a tracking and control system used in large-scale productions. Unlike a standalone automated spotlight, Follow-Me is a software platform that integrates with existing moving head fixtures. It works by using cameras placed around the venue to track performers in three dimensions, then translates that data into a DMX signal which is sent to multiple moving lights. Operators can assign fixtures to specific performers, adjust beam parameters, and manage complex stage environments in real time. The system supports multi-fixture tracking, allowing one operator to control dozens of spotlights following several performers simultaneously.

This product displays the effectiveness of tracking systems with automated lighting instruments, but its complexity and price make it not suitable for small-scale usage. S.T.A.R. will fill this gap by utilizing the same camera tracking, while delivering a smaller scale system that can be used by anyone. S.T.A.R will have a single camera built into the unit, rather than needing multiple cameras around the perimeter of the stage. This makes the system much cheaper and easier to integrate.

MAC Viper Profile

The Viper Profile is a powerhouse in the moving light industry. Its mechanical system is a proven model for moving head lighting instruments. It utilizes a motorized yoke design, where stepper motors control both the pan and tilt directions. While it does not utilize any human tracking systems listed above, its intricate mechanics like the adjustable iris and motorized gimbal will be an inspiration of our design elements within S.T.A.R.

2.3 Goals and Objectives

At its foundation, the system aims to detect and track a single moving subject in lit environments, using a CMOS imaging system and real-time recognition algorithms to guide a two-axis motorized gimbal that delivers smooth spotlight tracking with less than 300 ms of latency. Advanced goals build upon this by refining multi-target detection, enabling user selection of subjects, optimizing gimbal motion for silent operation, and incorporating a wireless control interface for calibration and override. Stretch goals further extend the system's functionality by supporting DMX/Ethernet integration for compatibility with professional lighting consoles, networking multiple units for coordinated coverage, and adapting to environmental challenges such as smoke, fog, or high-contrast lighting. Additional objectives include implementing built-in lighting effects, reducing optical distortion through post-processing, and exploring higher-wavelength detection capabilities such as thermal imaging in the far-infrared range. Together, these goals and objectives provide a clear roadmap for developing a robust, real-time, and versatile autonomous spotlighting system.

2.3.1 Basic Goals

1. Detect and track a single moving person 7.5 meters away.
2. Control a motorized gimbal to pan/tilt the spotlight toward the tracked subject's center of mass.
3. Provide smooth, low-latency spotlight movement (<500 ms total system delay).

2.3.2 Advanced Goals

1. Enable multi-target detection from 7.5 meters away with single-target spotlight focus.
2. Optimize gimbal motion for continuous and smooth operation.
3. Add wireless control interface (web app or mobile) for calibration and override.

2.3.3 Stretch Goals

1. Network multiple S.T.A.R. units for coordinated spotlight coverage.
2. Incorporate environmental adaptation, such as handling smoke, fog, or high-contrast lighting.

2.3.4 Basic Objectives

1. Design a multi-lens system using commercial off the shelf components capable of providing a clear image of performers at the specified depth of field.
2. Use human recognition algorithms (i.e. computer vision or ML) to identify and track a single subject and communicate to a gimbal system for pan/tilt control.
3. Validate the system for latency through stage test scenarios with one performer.

2.3.5 Advanced Objectives

1. Refine the vision algorithm to detect multiple subjects and allow selection of which performer to follow by the user.
2. Calibrate the motor control to minimize jitter, noise, and overshoot in gimbal movement.
3. Design and test a mobile/remote app interface for manual override and basic spotlight control.

2.3.6 Stretch Objectives

1. Implement DMX/Ethernet support to allow for integration with existing lighting systems and console control.
2. Incorporate detection of higher wavelengths including that of the Far-Infrared (9µm - 12µm) range in order to provide thermal imaging.

2.4 Description of Features / Functionalities

There are many features and components of the S.T.A.R project that must integrate to create a fully functional, operating system. This section will do two things; show complete schematics of both the optical system and prototype, then explain the unit in its three main components; the rotating gimbal, imaging system, and illumination system. It is important to note that the illumination system is mounted on a rotating gimbal, and the imaging system is placed separate from the design. Thus, when the gimbal rotates, the spotlight moves in correspondence to its motion, yet the imaging system remains still as it is capturing the entire scene, without changing its viewing angle. This is to maintain the observation that the subject that is being tracked is actually in motion rather than continuously centered on the imaging plane. By separating the motion of the light from the camera, the system keeps a fixed view of the scene while allowing the spotlight to move freely. This design makes tracking easier because the camera does not constantly re-center the subject, which would hide real movement. Instead, the gimbal is used only to point the illumination, while the imaging system stays stationary and observes the full field of view.

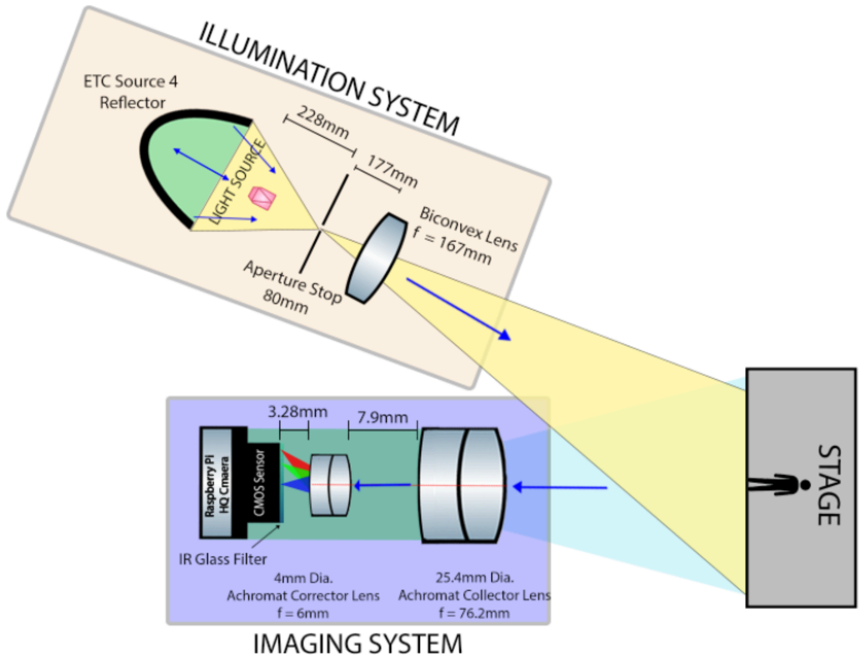


Figure 1: Optical Schematic

The figure above demonstrates all of the optical elements in the S.T.A.R system. The light yellow box shows the illumination system, from the illumination source to the focusing lens. The components in the blue box are the imaging system, from the sensor to the collector lens. The blue lines show that the light from the illumination system goes onto stage, and the light that reflects off of the subject (red box) goes back into the imaging system.

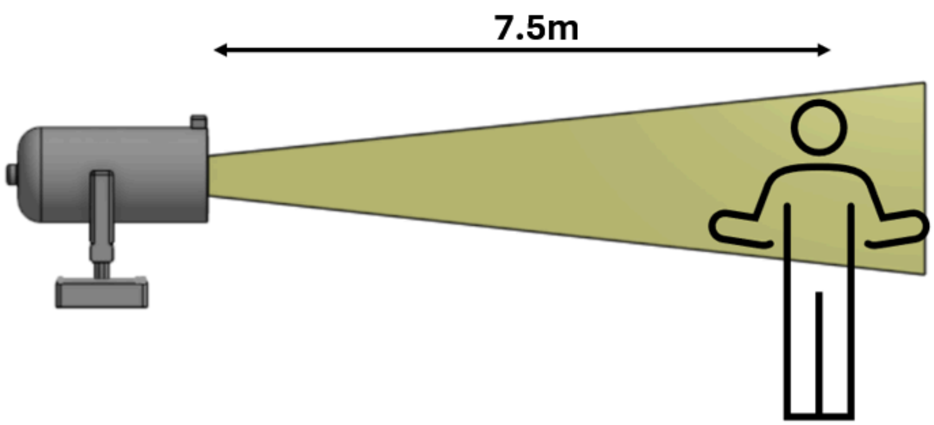


Figure 2: Prototype model of S.T.A.R.

The prototype shown above displays the S.T.A.R. motorized spotlight system capable of autonomous human tracking. The goal is to illuminate the entire upper half of the subject from 7.5 meters away to allow for hand motions/actions to be seen by the audience. The cylindrical housing contains the light source, reflector, and lens system. Inside the gimbal are servo motors for two-axis movement (pan and tilt), The base houses the control electronics, including the microcontroller, motor drivers, PCBs, and power sources. Our CMOS sensor system is mounted at the top of the output of the system. The system operates as a standalone unit, requiring only power and a simple user interface (remote or mobile app) rather than a traditional lighting console.

Rotating Gimbal

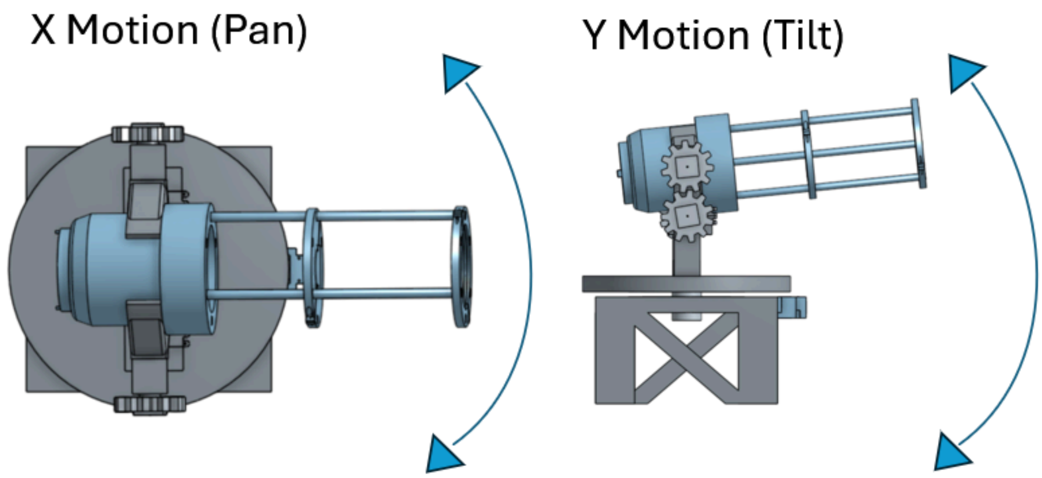


Figure 3: Gimbal motion axes of the S.T.A.R. system. The left view shows rotation about the X-axis (pan), and the right view shows rotation about the Y-axis (tilt).

A core feature of the S.T.A.R. system is the motorized gimbal that allows the spotlight to rotate in two separate axes: the horizontal and vertical direction. In professional stage lighting, motorized gimbals are most commonly found in moving head fixtures, such as the MAC Martin Viper. These instruments are some of the most versatile in the entertainment industry, and require the most skill to manipulate. These fixtures use multi-axis gimbal systems not only to provide pan and tilt movement, but also to manipulate other parameters such as zoom, focus, and beam shaping. By contrast, S.T.A.R. aims to simplify the gimbal system by providing only the most essential element: pan and tilt. The gimbal serves as the method for subject tracking, translating the output of the vision system into physical movement of the illumination system. The design incorporates two degrees of freedom (pan and tilt):

- Pan (X-axis rotation): Allows the spotlight to sweep left and right across the stage.
- Tilt (Y-axis rotation): Adjusts the vertical angle to follow performers as they move closer or farther from the camera's elevation.

Human Detection

A key feature of the S.T.A.R. system is the ability to detect and track humans on stage in real time. This function eliminates the need for a dedicated spotlight operator and ensures consistent, automated coverage of performers. Human tracking will be achieved using a CMOS imaging system combined with computer vision algorithms that identify and follow targets as they move [4]. The camera will be mounted onto the rotating gimbal, so the subject will always be in view of the camera system [5].

To be able to track humans, the system will utilize a machine learning model from OpenCV that's trained for object detection. After taking input from the CMOS imaging system, it will communicate it through the MCU, where an application will be used alongside the system to help users select which human in view of the camera will be targeted by the system.The output of the application should then be able to track the selected human, adjusting the camera movement (pan and tilt) as the human moves [5].

Illumination System

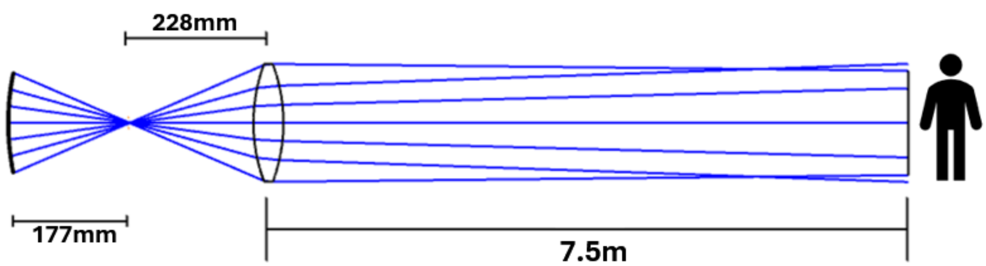


Figure 4: (from left to right) Illumination reflector, gate, and lens; pointing towards subject

The most critical component of the S.T.A.R. system is its illumination system, which is based on the ellipsoidal reflector spotlight (ERS) design (see Fig. 5). Ellipsoidal reflector systems have been a pillar of stage lighting for decades, with the release of the Source Four ERS by Electronic Theatre Controls (ETC) in 1992 setting the modern industry standard for efficiency, beam quality, and optical control. The ERS illumination system is unique in that it employs a combination of reflective and refractive optics to efficiently collect and shape light from a compact source. Specifically, an ellipsoidal reflector images the light source located at its first focal point onto a second focal point at the gate, where shutters, gobos, or apertures define the beam. A lens assembly positioned downstream of the gate then collimates or projects the shaped light toward the stage, producing a controllable and well-defined beam profile.

Traditionally, ERS fixtures utilize a tungsten-halogen four-filament lamp as the light source. These lamps are capable of very high luminous output and provide a continuous spectral distribution desirable for theatrical applications; however, they also generate significant heat, consume substantial power, and require frequent replacement due to limited filament lifetime. Thermal management is therefore a major design consideration in conventional ERS fixtures, influencing material selection, airflow, and optical spacing. In contrast, the S.T.A.R. system adopts a high-power LED emitter as its illumination source. Unlike filament lamps, the LED emits light over a relatively large area with high electrical-to-optical efficiency, significantly reduced infrared radiation, and a greatly extended operational lifespan. The use of an LED source reduces thermal load on optical components and enables longer continuous operation, while also introducing new optical design constraints related to source size, angular emission, and étendue. As a result, the S.T.A.R. illumination system preserves the core optical principles of the ERS design while adapting them to accommodate modern solid-state lighting technology.

2.5 Specifications

Table 1.0: Overall System Specifications

Parameter	Description	Target Value & Unit(s)
Tracking Latency	Time delay between performer movement and spotlight response	≤ 500 ms (basic), ≤ 300 ms (advanced)
Tracking Accuracy	Angular error between spotlight center and the selected performer's center of mass	≤ 5° RMS at 10m
Detection Accuracy	The percentage at which a human subject is correctly identified upon first entering the field of view.	70%
Horizontal Field of View (FOV)	The viewable area that the sensor can image	7.5m (± 1.5m)
Depth of Focus	The acceptable focus depth that a performer can move on a stage	> 1m
Tracking Range	Distance at which subject can be reliably detected and tracked	7.5m (± 1m)
Frames Per Second (fps)	How many frames are captured by the camera per second	30 fps @ 1920x1080p

Lighting Conditions	Minimum illumination where tracking is possible	≥ 1000 Lumens
Spotlight Beam Size	Effective beam diameter at 10m	0.5m - 1m adjustable (>5° half cone)
Pan/Tilt Range	Gimbal rotation coverage	Pan: 180° max, Tilt: 40° min
Pan/Tilt Speed	Spotlight movement speed	≥ 180°/s pan, ≥ 120°/s tilt
Operating Power Consumption	Wattage consumed by system during operation (NOT including illumination source)	~ 611Watts
System Weight	Total weight of system (NOT including illumination source)	≤ 20 lbs
System Cost	Total cost of prototype build (not including redundancies)	≤ $800 prototype

Figure 5: House of Quality

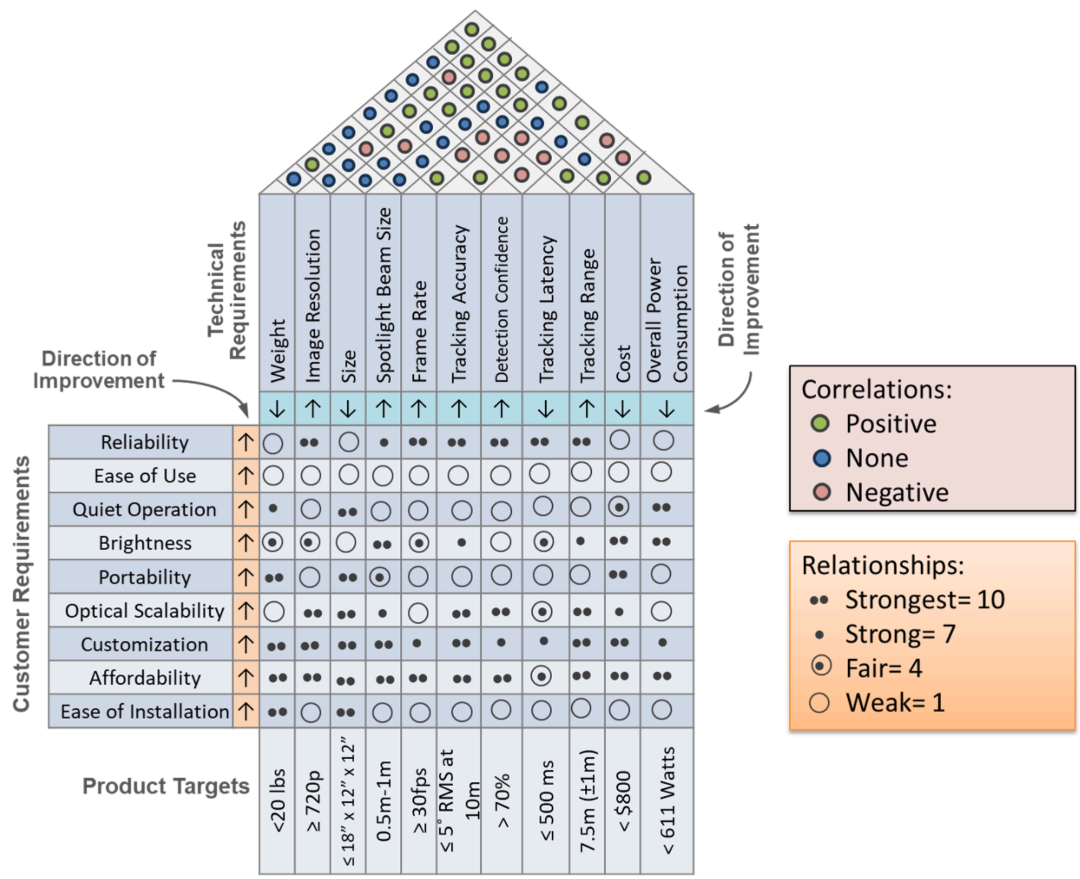


Table 1.1: Overall System Specifications

OPTICAL COMPONENTS			
Component	**Parameter**	**Specification**	**Unit**
Spotlight System			
Ellipsoid Reflector	1. Efficiency 2. Cone Angle 3. Focal Length	1. 90 2. 19 3. 237	1. % 2. Degrees 3. mm

LED Light Source	1. Intensity 2. Color Quality 3. Power	1. 5000 2. 85 3. 155	1. Lumens 2. CRI 3. Watts
Focusing Lens	1. Focal Length 2. Spot Size 3. Material	1. 177 2. 0.5-3 3. BK7	1. mm 2. m 3. NA
Imaging Camera Lens System			
Achromatic Doublet Collector Lens (MgF2 coated)	1. Diameter 2. Focal Length	1. 25.4 2. 76.2	1. mm 2. mm
Achromatic Doublet Imaging Lens (VIS 0° Coated)	1. Diameter 2. Focal Length	1. 4 2. 6	1. mm 2. mm
IR Cut Filter	1. Diameter 2. Cut Wavelength	1. 26mm 2. 650nm	1. mm 2. nm
Sensor (Sony IMX477 sensor)	1. Pixel Size 2. Megapixels 3. Wavelength Range	1. 1.55 x 1.55 2. 12.3 3. 400 - 700	1. µm 2. Mp 3. nm
ELECTRICAL COMPONENTS			
Component	**Parameter**	**Description**	**Target value & unit(s)**
MCU (ESP32-S3)	Operating Voltage	Runs motor control	3.3V
SoM (Raspberry Pi Compute Module 5)	Operating Voltage	Runs Computer Vision and MCU programming	5V

Motor (Pan/Tilt)	Max Speed, Max Standing Torque, Operating Voltage, Stall Current	2-axis servo system	0.12sec/60°, 41kg-cm, 5-8.4V, 3.9A
PWM Motor Controller (PCA9685)	Max PWM Outputs, Operating Frequencies	PWM controller for gimbal motors	16, Up to 1.6KHz
Power Supply (Raspberry Pi)	Power Output	DC power supply, USB-C connector	5.1V, 15W
Power Supply (PCBs and Motors)	Power Output	DC power supply	24V, 25A
WAGO Connector (2606-1102)	Rated Voltage, Rated Current	For connecting 24V power supply to PCB	300V, 31A

2.6 Hardware Block Diagram

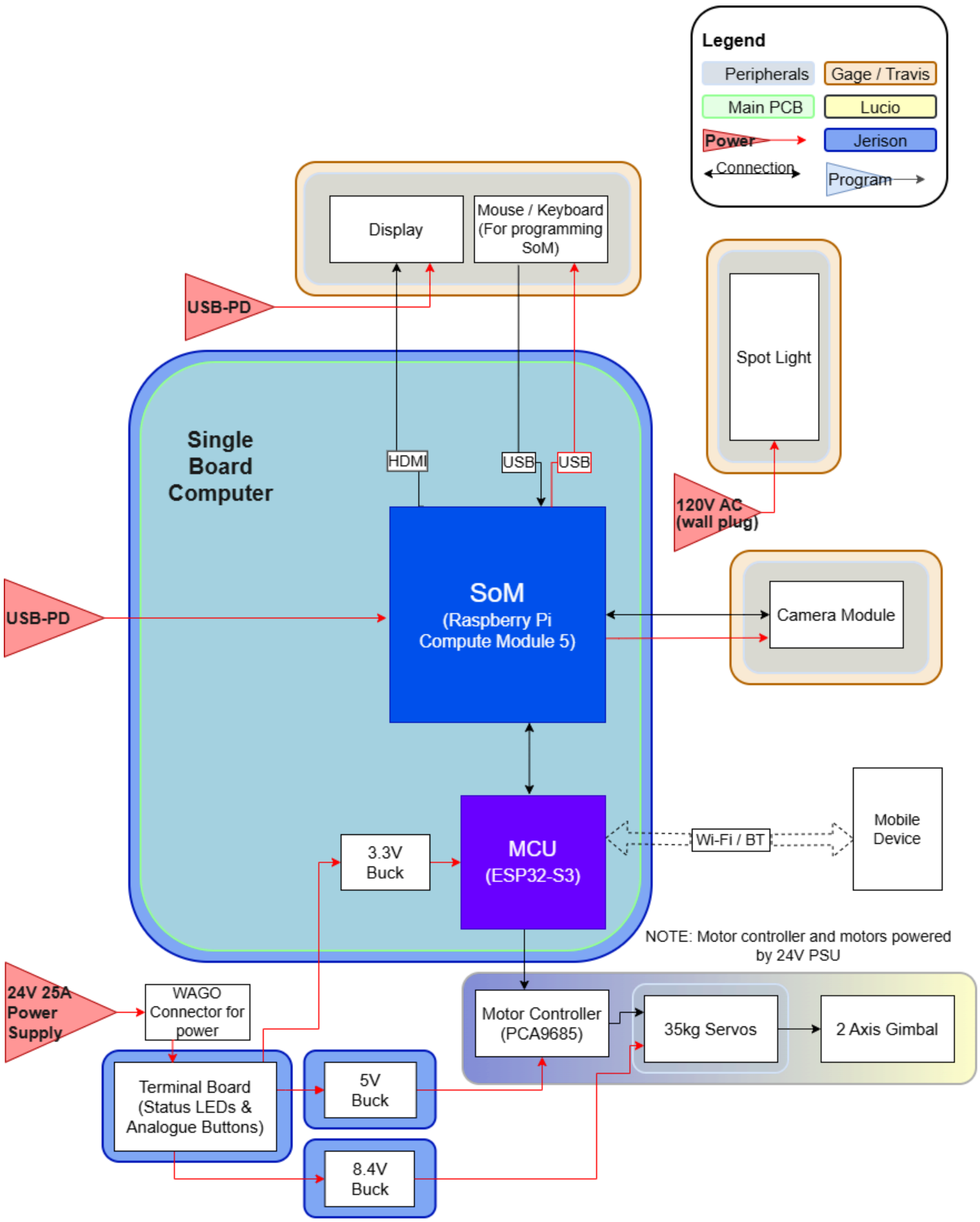


Figure 6.0: Hardware Block Diagram

Does NOT include lens system. Please refer to section 2.9 for diagrams of optical components.

2.7 Software Diagram

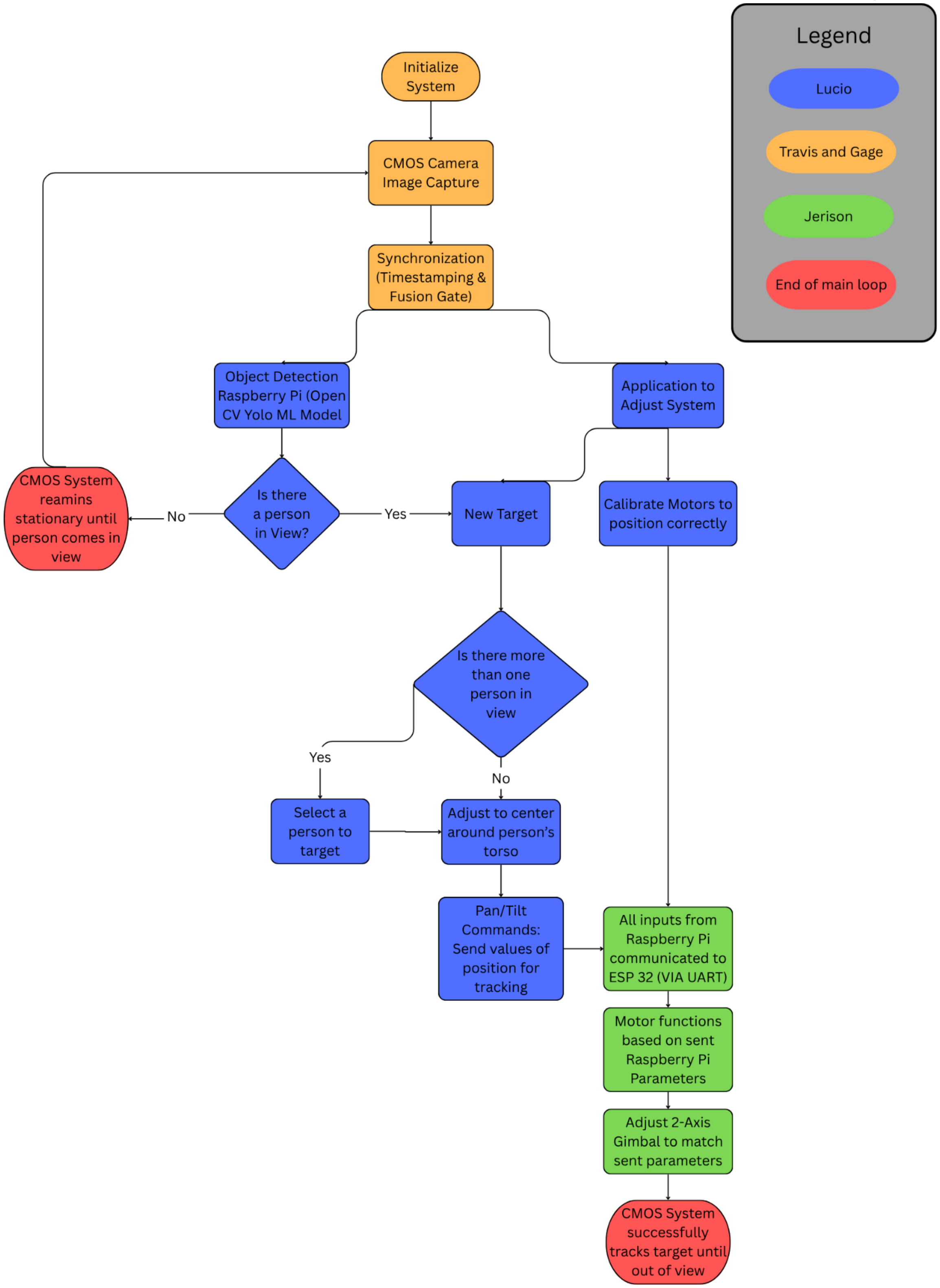


Figure 6.1: Software Block Diagram

2.8 Optical Components

Illumination system

The illumination system is composed mainly of two components: the source and reflector, and the focusing lens. The reflector is modeled after the ETC Source Four design, which is an elliptical housing coated with a reflective material to maximize light collection and projection efficiency. The lighting source can vary, as long as it is small enough to fit in the housing. For S.T.A.R., the Source Four lamp will be used. For the S.T.A.R. prototype, a Source Four lamp has been selected due to its high lumen output and strong overall efficiency. The primary drawback of this lamp is its heat generation, so a further investigation and testing of LED sources will be appropriate.

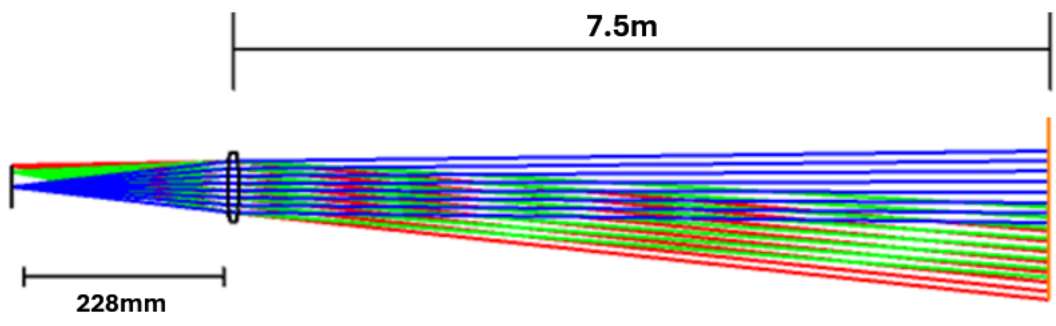


Figure 7.0 Cross section for output lens of S.T.A.R.

To be as cost efficient as possible, a biconvex lens is used to achieve a roughly 8.5 degree output half cone angle. Considering the variety of scenarios that S.T.A.R. will be used in, 8.5 degrees seems to be an optimal angle for varying distances. The lens will be placed at the end of the assembly. If we find that this angle is too wide for large distances, an iris will be placed at the gate of the optical assembly to reduce spot size.

Imaging System

The imaging system consists of a lens system and a CMOS sensor. This is a crucial optical element for the system as it determines the efficiency of the spotlight tracking. Glass choice is vital for building an optical setup to meet the parameters scoped for the resultant image. For this scope we are aiming to reduce as many aberrations as possible, leading to the design choice of using two types of glass: a crown lens and a flint lens. To maintain cost efficiency we explore the option of Commercial Off the Shelf Lenses (COTS) which provide a low cost alternative in comparison to a custom built lens.

The objective is to build a lens system that focuses light to a focal plane equivalent to the sensor size used. In this case we are using the **Raspberry Pi High Quality Camera**

12.3 megapixel Sony IMX477 sensor.

Table 2.0 Specifications of tracking camera

Raspberry Pi High Quality Camera 12.3 megapixel Sony IMX477 sensor	
Sensor Size	Horizontal: 6.32mm Vertical: 4.74mm
Pixel Size	1.55µm × 1.55µm
Megapixels	12.3
Wavelength Range	~ 400nm - 700nm
Lens sensor format	1/2.3” (7.9mm) or larger
IR Cut Filter	Integrated

For the crown lens, a common glass of choice to consider is an N-BK7 which when uncoated provides a transmission wavelength in the visible light spectrum to the near infrared wavelengths.

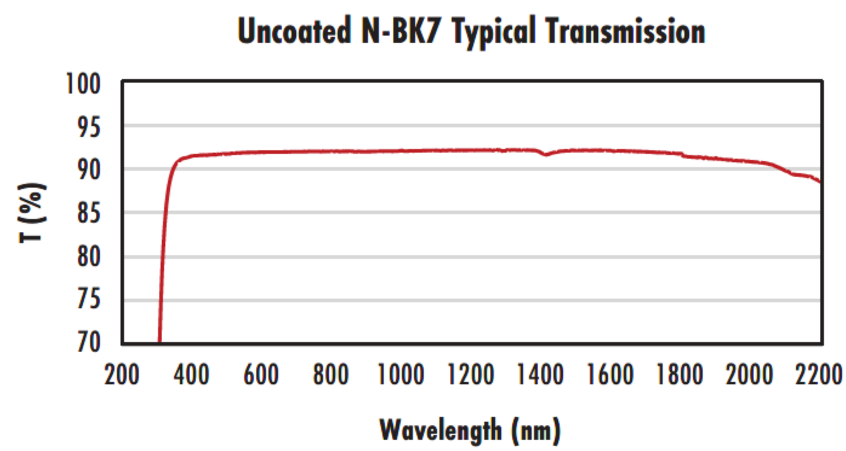


Figure 8.0: Typical Wavelength Transmissions of an Uncoated N-BK7 per Edmund Optics

As mentioned prior the lenses used are commercially off the shelf lenses. From Edmund Optics, the first frontal lens is the collector lens, gathering light to the optical system. This achromatic doublet features both a crown and a flint lens using substrates N-BK7 and N-SF5, respectively to minimize aberrations. The lens has a large diameter compared to other lenses within the system as it enables for more light to enter. The second lens is another achromatic doublet from Edmund Optics with an N-FK5 and F2HT substrate that serves as the corrector lens to mitigate the aberrations formed by the other optics, and produce a clearer image. The third and final lens of this system is also from Edmund Optics, with substrates N-BAF10 and N-SF10 and a smaller diameter and focal length to minimize the effective focal length of the overall system, and maintain its compactness.

Another practical component to consider is an iris. In lens design, the iris helps control the amount of light entering the system, which directly affects image brightness and exposure. By changing its diameter, it sets the f-number and influences depth of field as well as how forgiving the system is to focus errors. A smaller iris naturally reduces many optical aberrations by limiting the outermost rays, improving overall image quality. It also establishes the entrance pupil, which affects how the system accepts off-axis rays and how evenly light reaches the sensor across the field. Overall, the iris is essential for balancing light throughput, image sharpness, and depth-of-field performance in an optical system.

Chapter 3 - Research and Investigation

3.1.1 Reflecting vs. Refracting Optics

Generally, optical components can be divided into two main categories based on how they manipulate light: reflectors, which redirect light using mirrors or reflective surfaces, and refracting systems, which bend light using glass or other transmissive optical material. Both categories aim to achieve the same goal; image formation, beam shaping, or focussing. Their underlying physics, performance trade-offs, and material constraints differ significantly. Understanding these distinctions is crucial in selecting the appropriate configuration for a given optical system.

Fundamental Principles

Refracting optical systems operate on the fundamental physical principle known as Snell's Law of Refraction, which describes how light changes direction when passing between materials with different refractive indices. The law is expressed as:

$$n_1 \bullet sin(\theta_1) = n_2 \bullet sin(\theta_2) \tag{1}$$

where $n1$ and n2 are the refractive indices of the initial and secondary media, respectively, and θ1 and θ2 are the corresponding angles of incidence and refraction, measured relative to the normal at the surface interface.

A lens is formed from two or more refracting surfaces, typically spherical or aspheric, designed to redirect incident light rays to a specific focal region. The optical power of a lens is described by the Lensmaker's equation:

$$\frac{1}{f} = (n - 1)(\frac{1}{R1} - \frac{1}{R2}) \tag{2}$$

Where f is the focal length, n is the refractive index of the lens material, and R1 and R2 are the radii of curvature of the two surfaces. Because refraction depends on both material and surface geometry, lens designers can very precisely control how rays converge to form an image or spot.

Unlike refracting optics, which bend light as it passes through transparent materials, reflective optics redirect light entirely by bouncing the incident rays off of a surface. Some reflective optical components are completely flat, to be applied in redirecting incoming beams. Other reflective surfaces are designed with a curvature, to control the direction and convergence of light rays with its shaped surface. The curvature of a reflective optical surface can be aspherical or spherical as well, with the focal length of a spherical mirror simply being

$$f = \frac{R}{2} \tag{3}$$

This means that the focal point of a spherical mirror lies halfway between the mirror's surface and its center of curvature [15].

There are many other reasons why you would want to choose reflective optics rather than lenses. Reflectors provide a way to focus light while limiting aberrations - unwanted effects during the focusing process - much more than a lens would. Reflective optics are much better at handling higher temperatures, which is important when working with high powered sources.

3.1.2 Illumination System Design

Cold Mirror Reflector

A key component of the illumination system is the cold mirror reflector, which collects light from the source and focuses it through the aperture while simultaneously managing the system's thermal load. In high-intensity lighting systems, such as those using powerful LEDs or discharge lamps, a significant portion of the emitted energy lies outside the visible spectrum as infrared radiation. If not properly managed, this infrared energy can heat the surrounding optical components, degrading performance or shortening the lifespan of the fixture [16]. The cold mirror reflector solves this problem by reflecting visible light forward while transmitting infrared radiation through the back surface, which can then be dispersed via a fan.

The cold mirror is typically constructed from a borosilicate glass substrate, such as BK7, which is coated with multiple thin dielectric layers. These alternating high and low refractive index materials such as titanium dioxide (TiO_2) and silicon dioxide (SiO_2) are deposited in carefully calculated thicknesses to create wavelength-selective interference effects [17]. The result is a mirror that achieves high reflectance in the visible spectrum (approximately 400–700 nm) and high transmittance in the infrared region (above 750 nm). This design allows the reflector to maintain a bright, efficient optical output while significantly reducing the thermal energy directed toward the front lens and aperture.

The geometric design of the cold mirror reflector is ellipsoidal, meaning it has two focal points that govern how light is collected and directed. When the LED or lamp source is placed at the first focal point, the reflected rays converge at the second focal point, forming a concentrated beam and imaging plane as shown in figure 9:

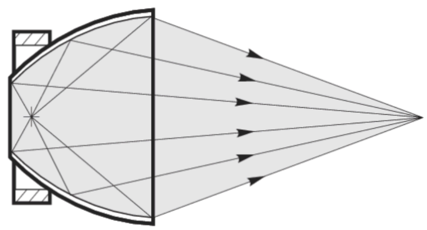


Figure 9. Ellipsoidal reflector diagram showing how light from the first focal point reflects to the second focal point. Adapted from [18].

This geometry maximizes optical efficiency by minimizing light spill and ensuring that nearly all usable light is redirected toward the focusing lens. Because the mirror reflects only visible light, the beam that reaches the second focal point remains cooler, protecting downstream components such as lenses and filters from thermal stress.

Because we aim to make our optical system as compact as possible, we want the f/# of the reflector to be as small as possible, giving us an optically "fast" system that will deliver a wide, bright cone of light at the gate. The relationship between f/# and NA is given by the equation:

$$f/\# = \frac{1}{2NA} \tag{4}$$

They are inversely proportional, so a larger numerical aperture will contribute to a smaller f/#, making the system much faster.

Using a cold mirror reflector also helps maintain color stability within the projected beam. By preventing the forward projection of infrared radiation, the system preserves the true color temperature of the light source. This ensures consistent illumination quality and prevents warm color shifts that could otherwise occur as optical components heat up. The selective spectral behavior of the coating therefore not only enhances thermal performance but also contributes to the optical accuracy and visual consistency of the overall system.

In practical terms, the cold mirror reflector forms the foundation of the illumination assembly, performing the dual roles of light collection and thermal filtration. Its material and coating design enable high reflectivity for visible wavelengths while passively managing heat dissipation, allowing for a more compact and reliable optical housing. By combining high optical efficiency, durability, and selective spectral reflection, the cold mirror reflector provides a critical balance between brightness and thermal safety, making it an ideal choice for high-performance lighting systems where clarity and stability are essential.

Focusing Lens

The focusing lens is the final optical element in the S.T.A.R. illumination assembly and plays a critical role in shaping the beam projected onto the stage. After light is collected and redirected by the ellipsoidal cold mirror reflector, it converges toward the gate where the intermediate image is formed. The focusing lens then refines this light into a well-defined, uniform beam with the desired angular spread. Because the optical performance of the entire fixture depends on the accuracy and clarity of this beam, the selection of the lens geometry, focal length, and material was conducted with particular care.

The lens assembly must meet several objectives simultaneously. It must produce a controlled beam with an approximate full output angle of 19°, maintain the high luminous output of the LED source, and minimize optical aberrations to ensure consistent illumination across the entire field. The lens also needs to remain stable under moderate thermal conditions from residual heat within the system while providing a means for precise mechanical adjustment to fine-tune focus. These combined optical and mechanical requirements defined the foundation for the focusing lens design.

The focal length of the lens primarily determines the relationship between the gate and the projection distance, dictating how strongly the beam converges or diverges [20]. In general, a shorter focal length produces a wider beam with greater divergence, while a longer focal length results in a narrower, more concentrated output. Similar to the reflector unit, the focusing lens should operate with a relatively small focal length relative to the system's numerical aperture (NA) to achieve a bright and efficient beam.

To begin, the numerical aperture of the lens can be expressed as:

$$NA = n(sin(\frac{\theta}{2})) \qquad (5)$$

where n is the refractive index of the medium (equal to 1 for air), and θ is the half-angle of the converging cone of light.

Next, the focal length is determined by the geometric relationship between the projection distance and the beam angle, given by:

$$f = \frac{r}{tan(\theta/2)} \tag{6}$$

Where r is the radius of the aperture (25mm), and θ is the output throw angle (19 degrees). Solving this equation with our parameters gives us a focal length of 167mm.

The final and most important mechanical parameter of the focusing lens is its entrance pupil diameter (EPD), which defines the clear aperture of the lens and ultimately determines its physical size, mass, and cost. It can be calculated using the relationship:

$$EPD = \frac{f}{f/\#} \tag{7}$$

This equation gives us an EPD of 103mm. This result means that the focusing lens requires a minimum diameter of 103 mm to capture the entire light cone without vignetting. A lens of this size offers a balance between optical performance and manufacturability, large enough to efficiently pass the collected light, yet compact enough for integration within the fixture housing.

With all of these parameters defined, we have enough information to model the lens system that will be used in the S.T.A.R system, shown in figure 10:

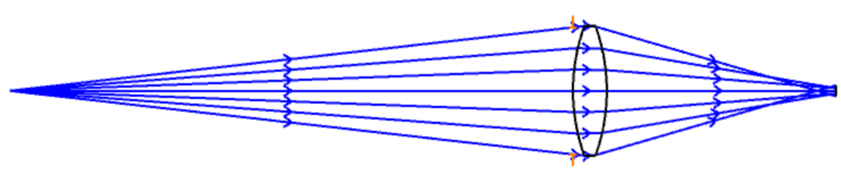


Figure 10: Focusing Lens Simulated in Zemax Opticstudio

Gate

The gate represents the image plane of the illumination system, located between the ellipsoidal reflector and the focusing lens. At this position, light collected by the reflector converges to form a sharp, controlled focal point before being refocused by the front lens into the output beam. Because this region defines where the system produces its most accurate and concentrated image, the gate serves as the optimal location for inserting image-forming and beam-shaping components. In professional lighting systems, this is where gobos, irises, and shutters are placed to sculpt or pattern the light with precision.

A gobo, short for "goes before optics," [21] is a thin template or stencil placed at the gate to shape the light beam into a specific pattern or image. When light passes through the gobo, the design is projected and magnified by the focusing lens onto the stage. Gobos are commonly made of stainless steel for simple, high-contrast patterns or dichroic glass for detailed, multicolor imagery [22]. Their interchangeability allows for easy customization, enabling projection of holiday themes, logos, scenic textures, or any design tailored to a particular event.

To maintain compatibility with existing professional standards, the gate in the S.T.A.R. system is designed to accept B-size gobos, matching the format used in ETC Source Four fixtures. This ensures compatibility with a wide range of commercially available gobos, allowing designers to use both stock and custom templates without modification. Although most infrared radiation is filtered by the cold mirror reflector, the gate region still experiences moderate heat, so ventilation is incorporated to protect the gobo and surrounding components. The result is a stable, modular system that maintains sharp, consistent projection while offering full creative flexibility.

Internal Baffles

One unique aspect of the optical design of the S.T.A.R. illumination system is the inclusion of internal baffles, also known as light traps, within the lens tube assembly. These are a series of annular, matte-black structures positioned along the interior walls of the focusing lens housing [23]. Their primary purpose is to absorb stray light, suppress internal reflections, and enhance overall image contrast. In high-intensity optical systems such as this, even a small amount of unwanted reflection can lead to visible artifacts, reduced beam sharpness, and loss of contrast. The baffles serve as passive optical filters, ensuring that only light within the designed optical cone contributes to the final projected image.

When working with imaging and illumination systems, one of the most common challenges is the presence of ghost reflections, which are faint, secondary images caused by light reflecting multiple times between lens surfaces or the inner housing walls. These reflections can produce blurred halos, double edges, or unwanted light spill that degrade image quality. By placing baffles along the optical path as shown in figure 11, these stray rays are intercepted and absorbed before they can reach the projection plane. This results in a cleaner, higher-contrast beam with sharply defined edges, which is especially important when projecting gobos or patterns.

Incorporating internal baffles into the design ensures that the S.T.A.R. system maintains the same high level of image fidelity and beam quality found in professional-grade fixtures such as the ETC Source Four. By controlling stray light within the housing, the system achieves greater optical efficiency, improved visual contrast, and a distinctly professional output that is free of unwanted glare or ghosting artifacts.

Figure 11: internal baffles in a stage lighting system

3.1.3 Materials

Focusing Lens

The choice of optical material plays a critical role in determining the performance, durability, and efficiency of the illumination system. Each lens material has distinct optical and mechanical properties; such as refractive index, dispersion, transmission range, and thermal stability. All of which influence how light is refracted and transmitted through the system. For the S.T.A.R. focusing lens, it was important to select a material capable of maintaining high optical clarity and withstanding the heat generated by the LED source and reflector assembly. To evaluate potential options, several candidate materials were analyzed, including optical plastics, BK7 crown glass, and SF2 dense flint glass. The following table summarizes the optical characteristics and suitability of each material for use in the S.T.A.R. system.

Table 3: Comparison of Optical Materials for Focusing Lens

Material	Refractive Index (n)	Abbe Number	Thermal Stability	Application / Notes
Optical Plastic (Acrylic)	~1.49–1.59	High (low dispersion)	Very low	Lightweight, inexpensive alternative for low-power systems; suitable for prototypes or environments with minimal heat exposure.
N-BK7 (Borosilicate Crown Glass)	1.5168 @ 587.6 nm	64 (moderate dispersion)	High	Industry-standard optical glass; excellent clarity, durability, and transmission; used in most general-purpose imaging and illumination systems.

SF2 (Dense Flint Glass)	1.6477 @ 587.6 nm	33.8 (High dispersion)	High	High-index material; useful for compact optics or achromatic doublets when paired with crown glass; more expensive and heavier than BK7.

Among the compared materials, N-BK7 was selected for the S.T.A.R. system's focusing lens. It offers an ideal balance of optical performance, mechanical stability, and cost-effectiveness. BK7 provides high visible and near-infrared transmission, a moderate refractive index, and low dispersion, making it suitable for imaging without introducing significant chromatic aberration. While SF2 provides a higher refractive index, its increased dispersion and weight make it less favorable for compact optical systems. Plastic lenses, though lightweight and inexpensive, were excluded due to their limited thermal tolerance under high-intensity illumination.

Reflector

The choice of reflector material has a significant impact on both the optical performance and thermal efficiency of the illumination system. The reflector's purpose is not only to redirect light from the source toward the gate but also to control how much infrared (IR) radiation and heat are transmitted through the optical path. Materials differ in how they reflect visible light, manage heat, and withstand prolonged exposure to high temperatures. Factors such as reflectivity, spectral selectivity, durability, and manufacturability must all be balanced to ensure long-term performance and safety. To determine the most suitable material for the S.T.A.R. illumination system, several common reflector types were compared, including polished aluminum, coated aluminum, dichroic (cold-mirror) glass, and reflective polymer films. The following table summarizes their relative optical and thermal characteristics.

Table 3.1: Comparison of Reflector Material Types

Material	Reflectivity (Visible)	Thermal Stability	Spectral Behavior	Notes Applications
Bare Aluminum	~85–90%	Moderate; may oxidize	Reflects visible + IR	Common, low cost, but projects heat forward.
Aluminum with Enhanced Coating (Al + dielectric)	~93–95%	Slightly better than bare Aluminum	Reflects visible; partial IR removal	Higher performance than bare metal reflector;

				improved durability.
Dichroic Glass (Cold Mirror)	~95–98% visible; IR transmitted	High; excellent stability	Reflects visible, transmits IR	Ideal for heat management in thermal systems.
Silicone / Polymer Reflective Film	~97% (depending on coating)	Good (varies by film type)	Reflectivity varies by film	Used in cost-sensitive applications; limited thermal endurance.

After evaluating these materials, the dichroic cold mirror reflector was chosen for the S.T.A.R. system. This material provides exceptional visible-light reflectivity (approximately 95–98%) [24] while allowing unwanted infrared radiation to pass through the back of the reflector, preventing heat buildup near the gate and lens assembly. The glass substrate offers excellent dimensional and thermal stability, ensuring stability under continuous operation. Unlike bare or coated aluminum, which can oxidize or discolor over time, the dielectric coating on the cold mirror is stable and maintains performance for thousands of hours.

Although polymer reflective films and metallic coatings are cost-effective and lightweight, they cannot withstand the high temperatures produced by a concentrated LED source. The cold mirror reflector, by contrast, combines high optical efficiency with robust heat management, resulting in a brighter, cooler, and more stable beam. For these reasons, the dichroic cold mirror was determined to be the optimal material for the reflector in the S.T.A.R. illumination system.

Instrument Housing Material

The outer body of the S.T.A.R. illumination system has to fill many roles; it provides mechanical support for the optical components, protects the internal electronics, and ensures safe operation under high temperature conditions. The housing must therefore be structurally rigid, lightweight, and thermally resilient, while also minimizing the total weight to allow smooth operation on its motorized gimbal assembly.

Because the enclosure is 3D printed, the choice of filament determines how well the system withstands heat generated by the LED and reflector, as well as how easily complex features such as mounts, ventilation channels, and cable paths can be fabricated. Several thermoplastics were evaluated for this purpose, including standard PLA, ABS, PETG, HT-PLA, and polycarbonate (PC). Each material offers distinct trade-offs in printability, strength, and thermal behavior. The following table summarizes their key characteristics and suitability for use in the S.T.A.R. illumination system.

Table 3.2: Comparison of 3D Printing Materials for Fixture Housing

Material	Heat Resistance	Density / Weight	Printability / Cost	Notes / Suitability
Standard PLA	~60–65 °C glass transition	~1.24–1.30 g/cm³	Very easy, low cost	Excellent for prototypes, but poor heat resistance for lighting housings.
ABS	~100–110 °C deflection temperature	~1.04–1.08 g/cm³	Moderate (warps; enclosure needed)	Better heat resistance than PLA but still marginal near high radiant heat sources.
PETG	~70–80 °C heat resistance	~1.27 g/cm³	Good printability, moderate cost	Balanced properties, but not ideal for prolonged exposure to high heat.
HT-PLA (High-Temperature PLA)	Up to ~150 °C when annealed	~1.24–1.30 g/cm³	Similar to PLA; slightly higher cost	Engineered for heat applications; lightweight and thermally resistive
PC (Polycarbonate)	Glass transition ~145–150 °C	~1.20–1.22 g/cm³	Difficult print (high temp, enclosure needed)	Excellent heat and strength properties but higher cost and complexity.

After comparing several common 3D-printing materials, HT-PLA was chosen for the S.T.A.R. housing due to its unique combination of thermal resistance, low weight, and printability. Once annealed, HT-PLA achieves a glass transition temperature of up to 150C, allowing it to maintain structural integrity even in the thermally intense environment surrounding the LED module and reflector [25]. This represents a substantial improvement over standard PLA and PETG, which begin to soften around

60–80 °C, and provides sufficient safety margin for extended operation. The improved heat tolerance ensures that the housing will not warp, deform, or lose alignment when exposed to the fixture's radiant heat.

Beyond thermal performance, HTPLA provides excellent dimensional stability and ease of fabrication. Its compatibility with standard FDM 3D printing allows for easy prototyping and precise geometric constraints without the need for costly machining or molds. The ability to print custom component mounts, internal airflow channels, and complex mechanical interfaces directly into the design simplifies assembly and reduces part count.

Overall, the choice of HTPLA offers an optimal balance between thermal resistance, strength, and manufacturability. It supports reiteration during the development phase while maintaining the heat tolerance and rigidity needed for reliable performance in a lighting environment. This makes it a practical and efficient material choice for the S.T.A.R. system's housing, combining the benefits of lightweight design with sufficient durability for continuous optical operation.

3.1.4 Illumination Source Comparison

Selecting the appropriate illumination source is one of the most critical aspects of designing a system like the S.T.A.R. project. The light engine defines the optical, thermal, and mechanical behavior of the entire fixture. The choice of light engine directly affects brightness, color rendering, efficiency, and even physical parameters such as system weight, length, and power distribution. There are many different illumination technologies on the market, such as traditional halogen-tungsten lamps, high-power LED COB modules, and modern RGBW light engines. Each unique engine presents advantages and limitations that must be carefully evaluated. Understanding the characteristics and constraints of each source type allows for a more efficient and optimized optical design, ensuring proper integration with other subsystems. The following subchapters discuss these illumination technologies in detail, outlining their respective principles, strengths, and trade-offs, followed by a direct comparison to determine the most suitable source for the S.T.A.R. illumination system.

LED COB Modules

Chip-on-Board (COB) LED modules are compact, high-efficiency light engines that integrate multiple LED dies onto a single substrate, creating a bright, uniform emission surface. This design minimizes optical losses and improves thermal conductivity compared to traditional discrete LED arrays, allowing for higher luminous density and simpler optical alignment. With efficiencies exceeding 90 lm/W and lifetimes over 25,000 hours [26], COB modules provide an excellent balance of brightness, energy efficiency, and reliability. Their continuous, even output eliminates visible hotspots, producing a clean beam ideal for projection and imaging applications. Below is a table that demonstrates the differences between a COB module and a traditional LED array;

Table 4: Comparison of LED Light Source Technologies

Source Type	Efficiency (lm/W)	Lifetime (hours)	CRI Range	Thermal Load
Discrete LED Array	60–80	10,000–15,000	80–90	Moderate
LED COB Module	90–110	25,000+	80–95	Low–Moderate

COB modules also offer exceptional design flexibility, supporting a wide range of color temperatures and CRI values while remaining compact enough for lightweight optical housings. Although they require a constant-current driver and heat sink, these integration needs are minimal compared to the overall performance benefits. For the S.T.A.R. illumination system, the COB LED module was selected for its high lumen output, low thermal load, and long service life, making it the most practical and efficient choice for a gimbal-mounted optical assembly.

Halogen-Tungsten Lamp

The halogen-tungsten lamp has been a staple in stage lighting for decades, most notably popularized by the ETC Source Four lighting fixture. These lamps are praised for their extremely bright output and their warm, natural quality of light they produce. These characteristics are highly valued by lighting designers for creating visually appealing and inviting lighting scenes. Even with the rise of LED technology, many manufacturers continue to emulate the distinctive look and warmth of halogen-tungsten illumination to satisfy designers seeking the original look. Optically, these sources are ideal; the spectrum they produce matches almost perfectly to the blackbody spectrum, as shown in figure 12:

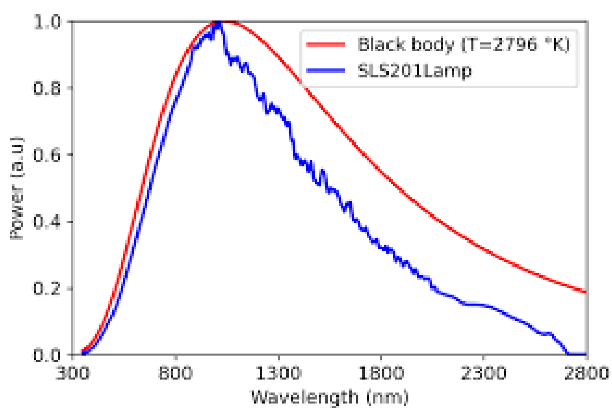


Figure 12: Spectral power distribution comparison between a blackbody source at 2796K and a halogen lamp. Adapted from [27]

However, despite their appealing light quality, halogen-tungsten lamps come with significant drawbacks. The most notable is their extreme heat generation, which far exceeds that of modern solid-state light sources. A large portion of their electrical power is converted into infrared radiation rather than visible light, leading to surface temperatures that can reach 518 °F (270 °C). This high thermal output not only reduces efficiency but also imposes strict design requirements for ventilation, heat shielding, and material selection—making halogen lamps less practical for compact or automated systems like the S.T.A.R. unit. They are also extremely sensitive to handling; touching the surface even once will transfer oils onto the glass housing, causing irregular heating and ultimately lead to an explosion of the lamp.

RGBW Engine

RGBW light engines combine red, green, blue, and white LEDs into a single integrated system, enabling full-spectrum color mixing and adjustable white-light output. By varying the intensity of each color channel, the engine can produce a nearly limitless range of hues and color temperatures—from warm tungsten-like ambers to bright, daylight-balanced whites. This makes RGBW technology highly attractive in entertainment and architectural lighting, where flexibility and dynamic control are prioritized. The inclusion of a dedicated white emitter also improves color rendering and brightness compared to traditional RGB systems, which often struggle to produce clean, high-intensity whites. This inclusion still does not come close to the robustness of a traditional incandescent source, as shown by the spectrum of an RGBW engine in figure 13

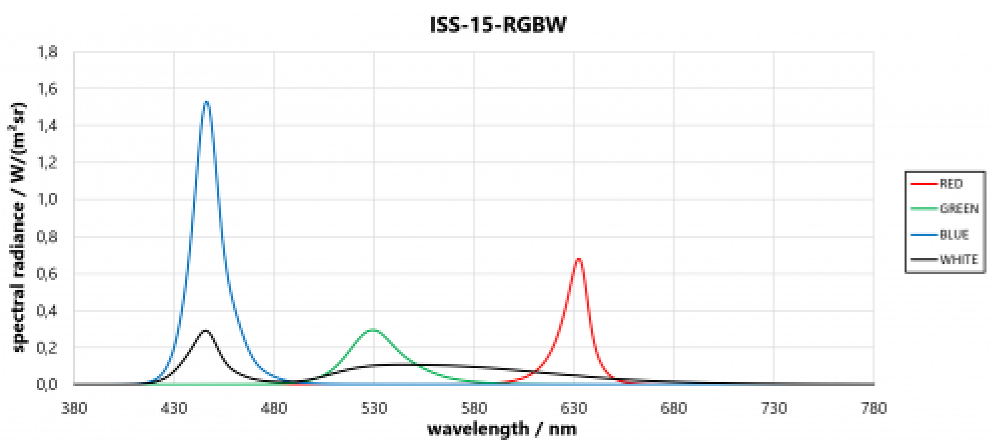


Figure 13: Spectral radiance of an RGBW LED engine, showing individual red, green, blue, and white emission peaks. Adapted from [28]

Despite their versatility, RGBW engines introduce increased system complexity. Each color channel requires precise current control and calibration to achieve consistent color balance and intensity. This demands a specialized driver and often digital signal control such as DMX-512, increasing both the cost and electronic bearing of the system. Additionally, because the total electrical power is divided among multiple emitters, the overall luminous output of an RGBW engine is typically lower than that of a comparable single-source COB module. While RGBW fixtures are excellent for applications that benefit from color effects or tunable white output, their added cost, control requirements, and reduced simplicity make them less ideal for compact, fixed white-light systems like the S.T.A.R. unit.

Comparison

After evaluating multiple illumination technologies, it became clear that each source offers unique strengths and limitations depending on the intended application. Halogen-tungsten lamps excel in color rendering and produce the warm, natural tones favored by many lighting designers but suffer from extreme heat output and short lifespans. RGBW light engines provide unmatched color versatility and dynamic control but introduce additional complexity, cost, and lower overall brightness. In contrast, LED COB modules offer the best balance of efficiency, thermal stability, and luminous performance, making them ideal for compact and automated systems. The following table summarizes the performance characteristics of these illumination sources, highlighting the criteria that guided the selection of the most suitable light engine for the S.T.A.R. illumination system.

Table 5: Comparison of Halogen, LED COB, and RGBW Light Sources for Stage Illumination

Criteria	HPL 757 Halogen Lamp	High-Power LED COB Module	Integrated RGBW Engine	Best Option
Luminous Output (lm)	≈ 5000	4500–6000	3500–5000	LED COB Module
Efficiency (lm/W)	≈ 18	≈ 90	≈ 70	LED COB Module
CRI	> 85	80–90	> 90	Integrated RGBW Engine
Thermal Load	High	Moderate	Low	Integrated RGBW Engine
Lifetime (hours)	300–500	25,000+	20,000+	LED COB Module
Cost (USD)	~$20	~$80	~$150	HPL 757 Halogen Lamp

Integration Ease	Direct fit to housing	Requires adapter + driver	Custom driver needed	HPL 757 Halogen Lamp

Furthermore, the COB module's broad, uniform emission pattern pairs exceptionally well with the ellipsoidal cold mirror reflector used in the S.T.A.R. optical assembly. The reflector's geometry is designed to collect light from an extended source and redirect it toward the system's focal plane with high efficiency. Because the COB emits over a relatively wide solid angle, its light distribution closely matches the acceptance profile of the ellipsoidal reflector, minimizing spill light and maximizing usable flux at the gate.

3.2 Imaging System

3.2.1 Technical and Theoretical Considerations

Object Magnification

Before selecting hardware components for lenses, it is important to dive into the math and theory driving the decisions to be made. A preliminary factor to consider is the target object magnification needed to properly image the performing stage onto the sensor. This target magnification can be directly calculated using the height of the output image, and the height of the object using the equation below.

$$M = \frac{h_i}{h_o} \tag{8}$$

When using this equation it is important to make note that h_o and h_i are the width of the stage and the width of the sensor, respectively. With a target stage size of 7620mm, and a horizontal sensor size of 6.32mm, the magnification can be calculated as a 1250x demagnification. Such magnification could be easier to achieve with a three lens system, however, opted to be achieved with a two lens system instead.

With the calculated magnification, another variable that can be equated is the back focal length of the lens system to the sensor. There is another ratio that relates magnification. The distance from the object to the lens (u), as well as the distance of the lens to the image/sensor (v) can also be used as shown below.

$$M = \frac{u}{v} \tag{9}$$

For the values of magnification and the fixed value of v, the distance u can be calculated as 6.1mm. This is verifiable through the thin lens equation where

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \tag{10}$$

As our distance from the object to the lens is significantly greater than our distance from the image to the lens ($v >> u$), it is a safe approximation to make that our focal length f = u. Therefore when viewing our lens system overall, we are seeking for an effective focal length (EFL) of about 6.1mm.

3.2.2 Sensor Technology Comparison

CCDs vs CMOS

In order to properly display an image, a choice in sensor must be made. There are two prominent types of sensors when considering an imaging system: a charge-coupled device (CCD) and a complimentary metal-oxide semiconductor (CMOS). These devices both produce an electrical signal from incident light received. Since we are aiming to use our image for human-tracking, it is important to consider the advantages and disadvantages of both sensors.

CCDs provide benefit in the pixel to pixel variation allowing for more precise optical measurements and less noise in the image produced, which is important as the applications for the project would most likely be used in a low light setting. This is achieved as the charge is transferred and read on a single amplifier. Although there are benefits, there are some drawbacks. CCDs tend to process at slower frame rates due to the readout architecture, and require a higher consumption of power. They also tend to draw more heat to the system, which is important to consider as the design needs to heavily monitor heat transfer from the illumination source.

CMOS cameras provide a parallel readout that allows for an increased frame rate, ideal for real-time imaging. For the pixel array in a CMOS sensor, each pixel contains its own amplifier with an analog-to-digital converter, which supports the faster frame rate. Since we are focusing on tracking motion on a stage, frame-rate is a key parameter to account for, making it a great choice to consider. These sensors also tend to exert less heat, removing the necessity of a method to cool the product. Overall, the choice between a CMOS and a CCD was determined by the factor of framerate speed and temperature, conditions that are necessary to consider for the scope of the project so communication between the real-time motion and the human tracking can be more efficient. To save costs, the CMOS sensor technology is the route we considered for this project, and thus we explored cameras using them.

Table 6: Comparison of CCD and CMOS

Specification	CDD	CMOS
Frame rate	Slower	Faster
Temperature control	Requires cooling	Does not require cooling
Power consumption	High power usage	Low power usage
Output noise level	Low	High
Cost	More expensive	Less expensive

Global shutter vs rolling shutter

When reviewing various sensor technologies, there are two primary capturing techniques used by sensors that define how light is captured over time. Global shutters capture an entire scene at a precise instant in time as each pixel in an array begins and ends their exposure in a simultaneous fashion. In doing so, image distortions are put to a minimum as real time movement in the object does not affect how each pixel sees it over time. Rolling shutters on the contrary, reads and exposes an image line-by-line sequentially. With this process, each row of pixels has a slight offset in time in the microseconds as there is real-time motion in the object. This can cause slightly more distortion effects when imaging. It is more common to see rolling shutters in most consumer CMOS applications. The simpler pixel readout technique tends to output lower costs, and show to be more compact. Since S.T.A.R. is aiming to rely on a Raspberry Pi system camera, the compactness and cost of these sensors are the factors we prioritized when looking for a camera.

ArduCam Pi Hawk-eye for Raspberry Pi

During the initial search for an imaging sensor, the attention of a high megapixel camera seemed to be beneficial for our design. With a higher sensor size, and thus a higher megapixel count, more incident light can be captured by the sensor, allowing for discrete sampling points. This means the optical image that is formed on the sensor would contain a finer image. With a high megapixel count, and a pixel size of 0.8 µm × 0.8 µm, the camera is able to perform 2x2 binning. This sensor-level technique takes a 2x2 array of pixels and merges them to form one effective pixel, reducing the output image by a factor of 4. Through this, the total still resolution is effectively 16mp, which is a deduction, but each pixel grants an increase in signal as the total light is from the previously combined array. As more light is captured by each pixel of the sensor, it is important to compare the ratio of signal-to-noise (SNR). WIth a 4x greater signal input, the noise of the overall image is decreased, and thus this 2x2 binning technique is very useful in low-light imaging, an environment that S.T.A.R. would likely be used in.

Raspberry Pi HQ Camera

Another point of interest for an imaging sensor was the Raspberry Pi High Quality (HQ) Camera, a popular imaging module built using the Sony IMX477 sensor. With a 12.3 megapixel CMOS sensor, this camera has the capabilities of communicating with a Raspberry Pi Compute Module 5, the latest models of those MCUs. The sensor uses a backside-illuminated CMOS architecture, where light sensitivity is improved and noise is reduced, making it stronger for low-light performance, and high dynamic range. This camera supports C- and CS-mount interchangeable lenses, increasing the amount of optical control onto the system, an important feat, as S.T.A.R. requires a great magnification to image a large stage in its practical application. Similar to the Hawk-eye, the HQ camera also uses a rolling shutter as opposed to global shutter, which may cause some distortions for more fast-paced scenes.

Imaging Sensor Selection

In the final selection of the sensor for S.T.A.R., there were many benefits of each camera we considered. The ArduCam Pi Hawk-eye provides a larger sensor size accompanied by a greater megapixel count, allowing for 2x2 barreling for less noise and greater low-light performance. However, the choice we made was the Raspberry Pi HQ Camera, primarily due to its ability to incorporate C and CS-mount lenses. With the flexibility of changing lenses, more optical control is gained, and thus the effective Field of View, depth of field, and magnification can precisely be customized to fit the needs of S.T.A.R. With this, more degrees of freedom are open to the optical design of the overall system to meet the target parameters, and goals of the design.

Table 7: Specification Comparison of ArduCam Pi Hawk-eye vs. Raspberry Pi HQ Camera

Specification	ArduCam Pi Hawk-eye	Raspberry Pi HQ Camera
Net Price	$65	$50
Shutter Type	Rolling Shutter	Rolling Shutter
Sensor	Sony IMX686	Sony IMX477
Sensor Resolution	9152 x 6944 pixels	4056 x 3040 pixels
Still Resolution	64 megapixels	12.3 megapixels
Horizontal Image Area	Not Specified	6.287mm
Vertical Image Area	Not Specified	4.712mm
Pixel Size	0.8 µm × 0.8 µm	1.55 µm × 1.55 µm
Horizontal Field of View	74 degrees	Lens Dependent
Vertical Field of View	55 degrees	Lens Dependent
Lens Mount	Non-interchangable	C/CS or M12 Mount

3.2.3 Stage Imaging System Design

Glass Materials and Geometry

Before considering lenses used to build the imaging system, it is critical to consider the variations a lens can be fabricated in from geometry to material. Glass material is an important aspect of lens design, as it has a major role in apparent aberrations in the output image. There are two primary lens categories to consider when building a lens system, a crown glass and a flint glass. Crown glass offers low-dispersion material characterized with a relatively low refractive index ($n < 1.50$) and a high Abbe number causing incident light to bend moderately and colors to separate less, minimizing the chromatic aberration in a lens system. On the contrary, flint glass offers a high refractive index with high dispersion, bending light more strongly with a greater degree of separation between colors, causing greater chromatic aberration. With a high refractive index, these glasses are beneficial for compact lens design systems. Both glass types have its essentials in a lens design, and it is important to consider this when making our selection of lenses for the tracking camera.

With the two primary types of glass to choose from, the geometry of a lens is also an important factor to consider. A singlet lens is made from a single piece of glass, and is typically a convex or concave element that focuses or diverges light based on its radii of curvature. As we are primarily relying on commercial off the shelf lenses over custom fabricated lenses, cost is an important factor when choosing the lenses we will use for the system. From commercial lens retailers, singlets are typically easier to manufacture and tend to be a more cost-effective choice for lens design. On the downside, they suffer from chromatic aberration, as the refractive index of these lenses tend to change with varying wavelengths, causing a degree of separation between the focus distances of different colors of light. This would cause more color fringing in our output image, which is not ideal for a program to accurately identify humans. In contrast to this, a doublet combines two lenses either cemented together or air-spaced. Doublets are typically composed of both a crown and a flint glass lens to cancel out the chromatic dispersion caused by the individual singlet of the lenses. This causes doublets to significantly reduce chromatic aberration allowing for a sharper and more accurate image that can further be improved for other aberrations. Although at a higher cost margin than its singlet counterpart, achromatic doublets are the primary choice we seek when choosing lenses to build our system.

Wavelength and Coatings

Before looking into a lens, it is important to be aware of the target wavelength that we are imaging. Since we are relying on the visible wavelength spectrum, the aim is to “block” out incident light at wavelengths beyond or below 300nm-750nm by reflecting them. As we are using commercial off the shelf lenses, there is much variety to the types of coatings provided. The main coatings we considered were those with a transmission range from the 400-750nm region. These wavelengths provide minimal surface reflection at the targeted wavelengths and intentional reflection at wavelengths outside of those ranges to to help mitigate sensitive wavelengths to our IMX477 sensor.

Collector Lens

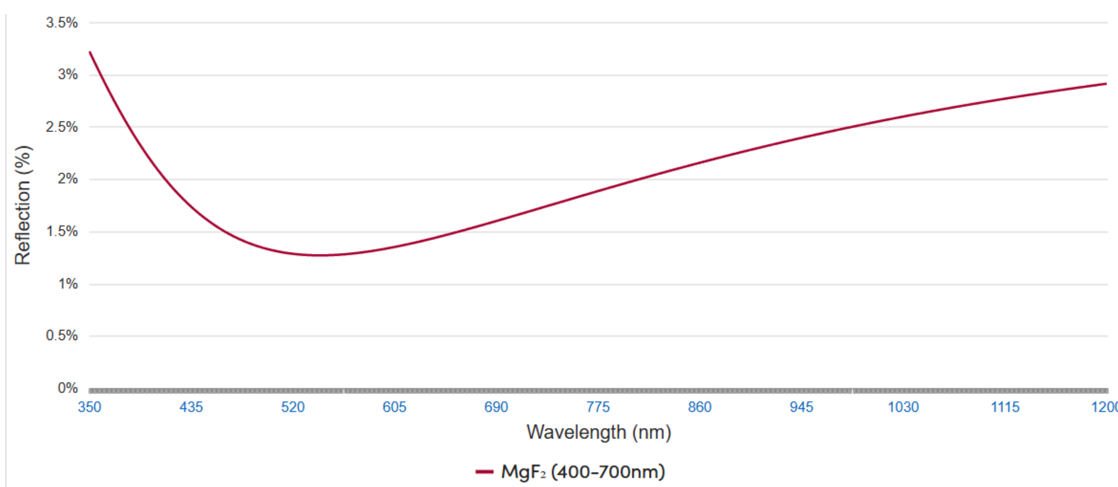
Since we are imaging from a target distance of >7.5 meters, the collector lens ideally should have the maximum light collection of the system. A collector lens with a large diameter is sought for as it improves the amount of photons to reach the sensor, and to reduce vignetting of the overall image used for the tracking software. This lens ultimately determines the optical efficiency of our imaging system, as it defines how much light the overall system can accept, also known as the numerical aperture (NA). Providing a lens that promotes a high NA allows for a greater field cone angle that can transmit the collected light into the rest of the system. The formula relating the NA and the maximum cone angle is as follows.

$$NA = n \times sin(\theta) \quad (11)$$

It is important to note that n refers to the refractive index of the medium that the light is propagating in. In our case, the medium is air with a refractive index of 1, making NA a direct relation to the angle.

Edmund Optic's #49-768 lens provides a large diameter relative to the system we are using. With a 25.4mm diameter, and a 76.2mm focal length, enough light can be collected to ensure a clear image is formed. The lens contains an MgF2 coating which has a low reflectivity rate at wavelengths from 300nm-400nm as shown in the figure below.

Figure 14: Reflectance (%) of an MgF2 Coated Lens



The collector lens was chosen as the system's front element because it establishes the initial numerical aperture, field coverage, and ray geometry that the rest of the optics must refine. Its primary role is to gather sufficient light from the distant stage while maintaining a manageable entrance pupil size and minimizing off-axis aberrations that could propagate downstream. A commercial achromatic or well-corrected singlet was selected to provide reliable performance across the visible spectrum, ensuring that chromatic errors remain controlled before the rays encounter the corrector and imaging lenses. The chosen lens offered an appropriate focal length and diameter to capture the full field required for stage imaging without introducing excessive distortion or demanding unrealistic spacing within the compact design. Overall, this collector lens provided the stable, well-defined optical starting point needed for the multi-element system to achieve high-quality demagnified imaging.

Imaging Lens

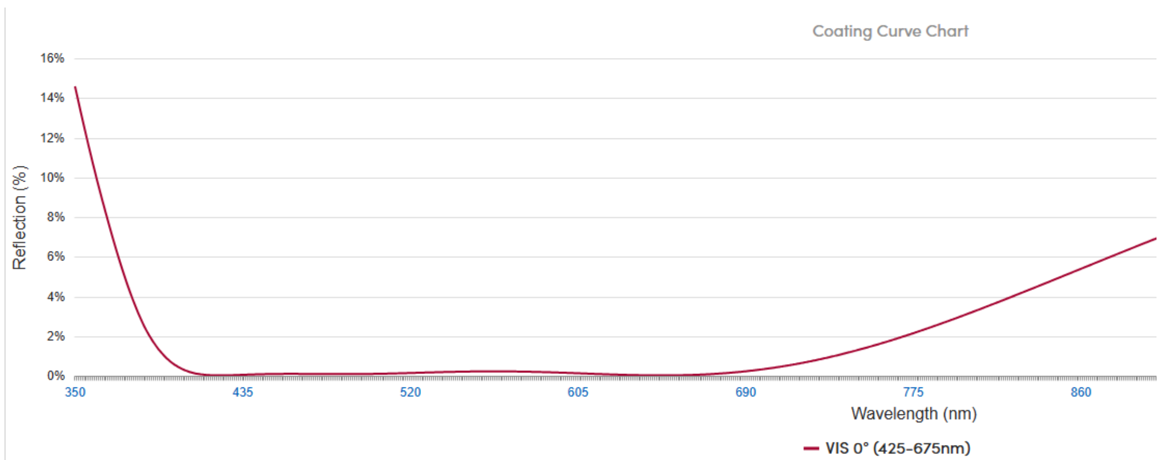
The second lens of the design is the imaging lens, primarily used to eliminate aberrations by other optical components. This lens is pivotal for the system design as it improves the image quality by compensating for distortions such as spherical aberration, coma, and field curvature. Through this, corrector lenses extend the usable field of view of an imaging system and allow for other optical components to be used, without mitigating the quality of the output image. In an "ideal" lens, the shape is aspheric where the surfaces are contoured to counteract spherical aberration. Spherical aberration is a distortion caused by light focusing at different points rather than at the center as you leave off-axis. Compared to traditional spherical lenses, asphere lenses have curvatures that gradually change across the surface to maintain a uniform focus across the field and correct for off-axis distortions.

Edmund Optic's TECHSPEC #63-702 is the lens we considered for this design. It is an achromatic doublet that contains a VIS 0° coating with a 4mm diameter, and a 6mm effective focal length. The substrates of this lens contain an N-BAF10 as the first crown element of the doublet and an N-SF10 as the flint element. This is the lens we chose as our imaging lens.

The lens was selected to refine and balance the optical performance of the system by addressing aberrations introduced by the front field lens and shaping the ray bundle before it reaches the imaging lens. Its role required a lens with moderate power and good chromatic control, ensuring that off-axis rays and higher-order aberrations were suppressed early in the optical path. The commercially available element from Edmund Optics with a suitable curvature profile and broadband coating was chosen to maintain high transmission and predictable behavior under varying field angles. The lens diameter and spacing flexibility also made it well suited to the compact mechanical layout of the design, allowing it to condition the light without introducing vignetting or complicating alignment. In the final configuration, this corrector lens provided the necessary balance of aberration control and system compatibility to support a clean, high-fidelity image at the sensor.

As the final lens for this commercial off the shelf lens system, the aim of this lens is to control the effective focal length of the overall system to achieve the desired magnification. With the previously calculated 6.1mm design, it is imperative that our effective focal length reaches that target. Choosing a proper imaging lens is important as it serves as the determinant of the focus and resolution of our final output image onto the sensor plane. Based on its position, this lens also helps us achieve the magnification needed for S.T.A.R. to image the stage. Since this element is closest to the detector, it is highly subject to internal reflections, so we aimed to find a lens that can fit these needs.

Figure 15: Transmission of a VIS 0° Coated Lens



Optical Irises

When deciding where to place the iris in a lens design, the first major consideration is aberration control. Positioning the iris near the front of the system tends to suppress aberrations that originate in early lens elements, such as coma and spherical aberration, by limiting the highest-angle rays before they propagate deeper into the optical train. Placing the iris near the middle of the system often creates a more balanced distribution of ray angles through each lens surface, which can help reduce field-dependent aberrations and keep the design more symmetric. If the iris is placed closer to the rear, it typically emphasizes the optical behavior of the later elements, enabling finer control over image-side numerical aperture and helping match the lens to a specific sensor format or entrance pupil requirement. Each of these locations changes the way rays interact with the optics, so designers must weigh which aberrations or performance metrics matter most.

Another important consideration is how the iris placement affects vignetting, field coverage, and mechanical packaging. If the iris sits too far from the optical center, the system may clip off-axis rays more aggressively, leading to uneven illumination across the sensor. For wide-field imaging, such as stage or theater capture, placing the iris near the location of the natural stop (where the optical system would naturally limit rays even without a physical aperture) helps maintain full field while controlling the cone angle of rays that reach the sensor. The space available between elements must also be considered, ensuring the iris can physically fit without interfering with lens mounts or introducing stray reflections. The optimal iris placement is the point where optical performance, illumination uniformity, and mechanical feasibility intersect, producing a system that is both high-quality and practical to build.

For our system, we chose to place the iris at the point where it naturally becomes the system's aperture stop, allowing us to control ray angles efficiently while preserving the full field needed to image a stage from distance. At this location, the iris suppresses marginal rays before they can contribute significant aberrations, which is especially important given the compact lens spacing and the high demagnification we're targeting. It also provides a clean, well-defined entrance pupil that Zemax can optimize around, ensuring consistent illumination across the sensor without introducing avoidable vignetting. Mechanically, this position offers enough clearance between elements to mount a physical stop without interfering with lens holders or risking reflections from nearby surfaces. Overall, placing the iris here gave us the strongest balance between optical performance, aberration control, and practical assembly.

Optical Housing

The housing for the lens system acts as the structure for this system. It must provide a rigid and stable region that preserves optical alignment under real conditions, including vibrations, temperature changes, and handling during setup. Because the design uses multiple commercial lenses spaced at precise and narrow intervals, the housing needs internal features such as fixed spacers, threaded retainers, or machined shelves to maintain those distances with tight tolerances. Typically a cylindrical metal body, composed of aluminum or stainless steel, is preferred for its mechanical stability, low weight, and ease of machining. However, to maximize cost efficiency, 3D printing the housing is the most practical route, especially for S.T.A.R.'s custom multi-element optical system. Additive manufacturing allows precise control over internal features such as lens seats, baffles, and alignment channels without the high machining costs associated with metal fabrication. It also enables rapid iteration, providing the flexibility to adjust spacing or geometry and reprint parts within hours rather than waiting for outsourced components. Using high-strength polymers or resin materials provides enough rigidity for stable optical alignment while keeping production costs low, which is ideal for prototyping or small-batch builds. By taking advantage of 3D printing, the housing becomes both affordable and highly customizable, allowing the system to be optimized without exceeding budget constraints.

3.2.4 Existing Camera Imaging Technologies

CCTV Surveillance Cameras

CCTV lens systems are a compact, multi-element optical design that serves as the imaging core of many surveillance and machine-vision cameras. Often in the C-mount or CS-mount fixture types, the optical design is heavily complex. Including typically between four and eight glass elements, these systems incorporate achromatic doublets, or aspheric surfaces in order to keep a compact fixture capable of reducing aberrations. These cameras typically operate at fast apertures such as f/1.4 to f/2.8 in order to accommodate low-light surveillance. For S.T.A.R., this would be a great system to incorporate, as the compactness and low-light capabilities would be suitable for this design. However, with the complexity of the design, and the amount of optical components involved, cost plays a factor into choosing this as the basis for our design, as it is hard to achieve the quality and compactness without the use of an aspheric lens, an expensive component to purchase off the shelf.

Pan-Tilt-Zoom Cameras

Existing tracking camera systems are designed to automatically follow a subject through a combination of optics, sensors, and intelligent tracking algorithms. Many commercial solutions use integrated pan-tilt-zoom (PTZ) cameras equipped with fast autofocus and wide-to-telephoto zoom lenses that maintain sharpness as the subject moves across the stage or performance area. These systems typically rely on computer vision techniques, infrared beacons, or motion-prediction algorithms to determine where the subject is going and adjust the camera orientation accordingly. Their optical assemblies are optimized for consistent image quality across a broad zoom range, meaning their lens groups are engineered to maintain focal plane stability and minimize breathing as the camera adjusts focus in real time.

A widely used example is the Sony SRG-X120 PTZ, which features a 12x optical zoom lens and real-time subject tracking powered by Sony's AI analytics, allowing the camera to maintain sharp focus as performers move across a stage. Another example is the Lumens VC-A61P, a 4K PTZ camera with a 30x optical zoom system designed to deliver consistent image quality while auto-tracking presenters or performers using onboard motion detection.

3.3 Electrical Hardware

3.3.1 SoM vs SBC

When selecting a computational platform, one of the key design decisions involves choosing between a System-on-Module (SoM) and a Single-Board Computer (SBC). Both serve the same fundamental purpose — providing the processing, memory, and interface capabilities needed to run advanced algorithms — but they differ significantly in flexibility, scalability, and integration complexity.

A System-on-Module, such as the Raspberry Pi Compute Module 5, integrates the processor, GPU, memory, and storage into a compact, self-contained board that interfaces with a custom-designed carrier board. This modular approach offers a high degree of design flexibility, allowing engineers to tailor the carrier PCB to the project's specific requirements. The carrier board can include only the necessary interfaces reducing system complexity and improving reliability. It also allows designers to implement custom power distribution and noise filtering circuits to enhance stability. While the initial engineering and PCB design costs are higher, the result is a more compact, efficient, and professional-grade embedded system that can be easily scaled or upgraded by replacing the SoM without redesigning the entire system.

In contrast, a Single-Board Computer, such as the Raspberry Pi 5, provides a complete computing platform with all necessary I/O ports, power management, and connectors on a single board. SBCs are highly advantageous for rapid prototyping and early-stage development, as they eliminate the need for custom hardware design. Developers can immediately begin testing software, algorithms, and peripherals such as the camera and motor controller without concern for hardware integration. SBCs also benefit from mature software support, well-documented drivers, and extensive community resources, which can accelerate debugging and reduce setup time. However, this convenience comes at the cost of reduced flexibility and a larger physical footprint. Because the I/O and power architecture are fixed, it becomes challenging to optimize the system for compact enclosures or specialized power domains. SBCs are also more susceptible to electrical noise and thermal inefficiencies when used in close proximity to high-power components.

The Compute Module 5 was originally chosen because of its excellent processing power for computer vision tasks and the ability to create a completely custom carrier board for I/O. However, highspeed boards were deemed too complex. HDMI needs very specific PCB requirements to function correctly. In the interest of cost-savings and time constraints, we have decided to not design a custom carrier board. Nevertheless, we will still be using a CM5 and CM5IO board. The PCB requirement will be satisfied by a custom MCU board, custom motor controller, and custom LED and switch terminal board.

During our final demonstration, we had to purchase a SBC Raspberry Pi 5 to replace our Compute Module 5. This will be explained in detail in a later chapter.

Table 8 – SoM and SBC Comparison

Attribute	System-on-Module (SoM)	Single Board Computer (SBC)
Integration Level	Core computing components only (CPU, RAM, storage, GPU). Requires carrier board.	Fully integrated with I/O ports, power management, and interfaces on a single PCB.
Design Flexibility	Highly customizable—carrier board can be tailored for specific project needs.	Limited customization—fixed I/O layout and hardware configuration.

Scalability	Easier to upgrade by replacing the compute module without redesigning the carrier.	Typically requires replacing the entire board to upgrade hardware.
Manufacturing Cost	Higher initial design cost but scalable for production runs.	Lower cost for prototypes and small projects.
Thermal Management	Can be optimized with custom heat dissipation solutions.	Fixed thermal design based on vendor layout.

3.3.2 MCU vs FPGA

One of the most critical hardware decisions involves selecting between a microcontroller unit (MCU) and a field-programmable gate array (FPGA). Both technologies are capable of handling real-time control and signal processing tasks, but they differ substantially in terms of programmability, complexity, cost, and overall suitability for specific applications. The decision ultimately rests on balancing computational performance, design flexibility, and system efficiency.

FPGA development typically requires advanced expertise in hardware description languages (such as Verilog or VHDL), longer design cycles, and specialized toolchains. FPGAs also tend to consume more power and are substantially more expensive than MCUs, making them less practical. While it is technically feasible to implement motor control or communication logic on an FPGA, these functions are more efficiently executed by an MCU, which already provides built-in peripherals and optimized hardware support for such operations.

Table 9 – MCU and FPGA Comparison

Attribute	MCU (e.g., ESP32-S3)	FPGA
Processing Type	Sequential software execution on processor cores.	Parallel hardware logic execution with reconfigurable gates.
Programming Complexity	Easy—uses C/C++ or MicroPython.	High—requires HDL (VHDL/Verilog) and synthesis tools.
Development Speed	Rapid—large ecosystem and open-source libraries.	Slower—requires detailed logic design and timing closure.

Power Consumption	Low; ideal for embedded and battery-powered systems.	High; often requires cooling and significant power overhead.
Cost	Very low (<$10 for ESP32-S3).	High ($50–$200+ depending on FPGA).
Best For	General-purpose control, networking, real-time tasks.	High-speed signal processing or custom hardware pipelines.
Reason Chosen for S.T.A.R.	The ESP32-S3 provides real-time motor control, Wi-Fi, and I^{2}C communication in a compact, low-cost package.	FPGA unnecessary for S.T.A.R.'s moderate control and vision needs.

For the S.T.A.R. system, the ESP32-S3 microcontroller was chosen over an FPGA because it offers a superior balance of computational capability, cost, and ease of integration.

The ESP32-S3 was chosen over other popular microcontrollers such as the STM32 family, Arduino Portenta, or Teensy 4.x series because it offers the best balance between performance, cost, wireless connectivity, and ecosystem maturity for the specific requirements of our system. While each alternative MCU family has its own strengths, the ESP32-S3's combination of computational capability, built-in wireless interfaces, and developer accessibility makes it uniquely well-suited for a distributed embedded system that demands both real-time control and networked communication.

Table 10 – ESP32 and STM32 Comparison

Attribute	**ESP32-S3**	**STM32 (e.g., STM32F4/F7)**
CPU Architecture	Dual-core Xtensa LX7 @ 240 MHz	ARM Cortex-M4/M7 @ up to 480 MHz
Wireless Connectivity	Integrated Wi-Fi + Bluetooth 5.0	Requires external modules
AI/ML Support	Built-in vector instructions for AI acceleration	Limited DSP capability

I/O Interfaces	I^{2}C, SPI, UART, PWM, USB, ADC, DAC	Extensive peripheral support
Ease of Development	Excellent—supported by Arduino, ESP-IDF, and MicroPython	Good—requires STM32CubeIDE
Cost Efficiency	Very low (<\$10)	Moderate (\$10–\$30)
Chosen for S.T.A.R.	Offers wireless connectivity, powerful dual-core processing, and efficient servo control via I^{2}C (PCA9685).	STM32 better for high-precision control but costlier and lacks wireless integration.

3.3.3 Motors

Servo motors were chosen for the S.T.A.R. system due to their precision control, integrated feedback mechanisms, and smooth, predictable motion characteristics, all of which are essential for maintaining accurate spotlight positioning and stable target tracking. Unlike stepper motors, which move in discrete increments and can exhibit vibration or "jitter" during operation, servos provide continuous positional feedback via an internal potentiometer or encoder, allowing for closed-loop control. This ensures that the spotlight's pan and tilt axes remain steady, even under dynamic tracking conditions where frequent adjustments are required. The result is a system capable of fluid, responsive movement that mirrors professional-grade gimbal performance—ideal for live performance environments where smooth motion is crucial to visual quality.

Another advantage of using servo motors is their simplified control architecture. Each servo integrates a motor, feedback sensor, and driver circuit into one compact package, allowing for direct control via PWM (Pulse Width Modulation) signals. This drastically reduces the external circuitry required compared to stepper or DC motors, which often need complex H-bridge drivers and external encoders for feedback. By interfacing the servos through an Adafruit PCA9685 PWM driver, the system can control multiple channels simultaneously with high precision and minimal signal noise. Furthermore, servos inherently hold their position when idle, maintaining the spotlight's orientation without consuming additional processing power or continuous correction signals from the ESP32-S3 microcontroller.

The torque ratings for the selected servo motors were determined through a "more-is-better" mentality. We chose 35kg-cm servos for the gimbal which ended up being enough to handle the pan and tilt loads. However, if they were found to be too weak in testing, we would have swapped out the servos for ones of higher torque rating. Since the stage will always be in front of the spotlight, 360 degree rotation is not necessary for our motors.

Table 11 – Servo and Stepper Comparison

Attribute	Servo Motors	Stepper Motors
Control Method	Closed-loop control with integrated feedback (potentiometer or encoder)	Open-loop control (closed-loop requires external encoder)
Position Accuracy	High accuracy maintained through continuous feedback and correction	Accuracy depends on step count; prone to missed steps under load
Motion Smoothness	Smooth, continuous motion ideal for fluid pan/tilt tracking	Discrete stepwise motion can cause vibration or “jitter”
Stability During Dynamic Motion	Maintains stable positioning during frequent and rapid adjustments	Can oscillate or lose position during rapid or variable loads
Feedback Mechanism	Built-in position feedback enables real-time correction	No inherent feedback; external sensors required for verification
Control Complexity	Simple PWM-based control with integrated driver electronics	Requires external motor drivers and often additional feedback hardware
External Circuitry Required	Minimal (PWM controller such as PCA9685)	Moderate to high (stepper driver, current regulation, optional encoders)
Position Holding	Actively holds position when idle without additional control logic	Must be continuously energized to hold position, increasing heat
Noise and Vibration	Quiet operation suitable for live performance environments	Can produce audible noise and mechanical vibration
Torque Characteristics	High torque at low speeds; torque chosen via rated servo strength	Torque drops off rapidly at higher speeds
Rotation Range	Typically limited (e.g., 180°–270°), suitable for fixed-orientation systems	Capable of continuous 360° rotation
Suitability for S.T.A.R.	Excellent—supports smooth tracking, stability, and simplified integration	Less ideal due to vibration, complexity, and tracking instability

3.3.4 Motor Controllers

There are several available motor controller options for driving servo motors, each varying in complexity, channel capacity, communication interface, and control precision. Common servo control solutions include direct microcontroller PWM generation, dedicated servo driver ICs (such as the PCA9685 or TLC5940), and advanced motion control boards (such as Pololu Maestro or Adafruit 16-Channel PWM/Servo Driver). Each has its advantages, but for a project like S.T.A.R. —which requires reliable, synchronized, multi-axis servo control integrated with embedded processing—the Adafruit PCA9685 stands out as the most practical and efficient choice.

Direct PWM generation through a microcontroller, such as the ESP32-S3, is a simple and inexpensive method for controlling a few servos. The ESP32 can generate PWM signals using internal timers, allowing positional control via software. However, this approach becomes less practical as the number of servos increases, since PWM signal generation consumes significant processor resources and can introduce timing jitter if the CPU is simultaneously handling computationally intensive tasks—such as vision processing or wireless communication. For this reason, the S.T.A.R. system required a dedicated servo driver that offloads PWM signal generation from the main controller, freeing the ESP32-S3 to focus on system logic, camera interfacing, and communication tasks.

The Adafruit PCA9685 was selected due to its high channel count (16 channels), I2C communication interface, and precision 12-bit PWM resolution. These features provide smooth, accurate positional control and eliminate jitter in servo movement. Because it communicates over I2C, the PCA9685 can be easily interfaced with both the ESP32-S3 and the Raspberry Pi Compute Module 5, enabling a flexible architecture where the Raspberry Pi manages vision and tracking algorithms while the ESP32 handles real-time motion control. The PCA9685's ability to generate PWM signals independently of the microcontroller ensures consistent timing and reliable performance, even under heavy processing loads.

Another key advantage is that the PCA9685 includes a built-in oscillator, meaning it does not rely on the timing accuracy of the controlling microcontroller. This guarantees consistent servo response regardless of CPU load or software timing fluctuations. Additionally, the driver supports an external power source for the servo rail, allowing high-current motors to be powered independently of the logic circuitry—essential for protecting sensitive electronics like the Raspberry Pi CM5 and ESP32-S3 from voltage dips or noise caused by servo operation.

Alternatives, such as the Pololu Maestro or TLC5940, offer advanced features like scripting or current feedback, but they come at significantly higher cost and complexity. The Maestro, for example, is better suited for robotics applications that require trajectory planning or multi-servo choreography, while the PCA9685 provides a simpler and more cost-effective solution optimized for smooth PWM control and scalability.

In summary, the PCA9685 was chosen because it offers high-precision, multi-channel PWM generation over I2C, offloads timing control from the main processor, and supports independent servo power input—all in a compact, affordable package. These attributes make it ideally suited for S.T.A.R.'s two-axis servo gimbal system, where reliable, synchronized motion and system stability are paramount.

Table 12 – Motor Controller Comparison

Attribute	PCA9685 (Adafruit)	TLC5940	Pololu Maestro
Control Type	12-bit PWM over I2C	12-bit PWM over SPI	USB/UART control with scripting
Channels	16 independent servo channels	16 channels	6–24 channels (depending on model)
Communication Interface	I2C (addressable up to 62 devices)	SPI (faster, but fewer devices)	USB/UART (standalone controller)
Ease of Integration	Very easy—widely supported by Arduino, ESP32, and Raspberry Pi	Moderate—requires more configuration	Easy, but adds software complexity
Power Handling	Separate servo power rail; supports up to 6V	External current sink required	Integrated power management per servo
Cost	~$10	~$7	$25–$50
Chosen for S.T.A.R.	Offers high channel count, reliable timing, and I2C compatibility with ESP32-S3 for smooth servo control.	TLC5940 lacks flexibility for multi-servo synchronization.	Maestro provides advanced features but is too costly and over-specified.

3.3.5 WAGOs

Wago connectors were chosen for the S.T.A.R. system primarily due to their reliability, safety, and ease of maintenance when dealing with multiple power and signal connections within a complex electromechanical setup. Unlike traditional screw terminals or soldered joints, Wago lever or push-in connectors provide a tool-free, vibration-resistant connection that ensures long-term electrical integrity — a crucial factor in systems subject to movement, such as a motorized pan-tilt spotlight.

From a technical standpoint, Wago connectors employ a spring-clamp mechanism that maintains consistent contact pressure on the wire conductor, even under thermal expansion or vibration. This design minimizes the risk of loose connections, arcing, or intermittent faults that could otherwise occur in high-current or mobile applications. In the S.T.A.R. project, this is particularly important since the 24V, 25A power rail supplies multiple subsystems (such as the servo motors, buck converters, and control electronics) that must remain electrically stable at all times.

Another major advantage is modularity and serviceability. The S.T.A.R. system may undergo iterative testing and component swaps during development. Wago connectors allow rapid prototyping and safe reconfiguration of the wiring harness without soldering or crimping tools, reducing downtime and minimizing the potential for wiring errors. The connectors also support wide wire gauge ranges and are compatible with both solid and stranded conductors, making them ideal for mixed-signal and power distribution setups found in this project.

In addition, the compact form factor and clean aesthetic of Wago connectors contribute to a neat and professional internal layout of the 3D-printed enclosure. This enhances not only safety and organization but also compliance with good electrical design practices, such as maintaining proper creepage and clearance distances between high- and low-voltage lines.

3.3.6 PSU

The power supply selection was a critical design consideration, as it directly influences system stability, electrical safety, and performance reliability. After evaluating the total current draw and voltage requirements of the major subsystems—including the servo motors, motor controllers, and embedded electronics—a 24V 25A power supply unit (PSU) was selected. This configuration provides a total available power of 600W, which ensures adequate headroom for peak loads while maintaining efficiency and thermal stability during continuous operation.

The choice of 24V as the main supply voltage was driven by the power requirements of the high-torque servo motors used to drive the two-axis gimbal. These servos demand relatively high current, particularly during rapid acceleration and deceleration cycles when both torque and current draw peak. Operating them at 24V allows for more efficient torque delivery and reduced current flow compared to lower-voltage systems (such as 12V), which in turn minimizes resistive losses along the power lines.

The 25A current capacity was selected to provide sufficient margin above the nominal load to prevent voltage sag or instability during high-power transients. Servos, in particular, can draw several amps each under full load.

A 600W PSU provides flexibility for powering auxiliary electronics such as cooling fans, control boards, and sensor modules without the risk of overloading the main power rail. To enhance system reliability, individual subsystems are isolated using voltage regulators or buck converters that step down the 24V rail to the required logic levels (e.g., 5V for the PCA9685, 3.3V for the ESP32-S3).

In summary, the 24V 25A PSU was chosen to deliver a balance of power, efficiency, and headroom suitable for a system combining high-torque actuation with sensitive embedded control electronics. It ensures that the S.T.A.R. system can maintain stable operation during rapid servo movement while providing ample reserve power for future scalability and operational safety.

Table 13 – PSU Power Budget

Component	Voltage Requirement	Current Requirement
ESP32	3.3V	1A
PCA9685	5V	400mA
35kg Servo Motors (3x)	8.4V	3.9A / Each (Stall)
TOTAL	**~16.7V**	**~13A**

3.3.7 Power Converter

A buck (step-down) switching converter is the preferred topology for converting our system's high-voltage rail (e.g., 24 V) down to the logic and servo rails (e.g., 5 V, 3.3 V) primarily because of efficiency and thermal advantages. A buck converter achieves high efficiency by switching a transistor rapidly and using an inductor to store and release energy; only the difference between input and output power (plus converter losses) is dissipated as heat. This makes buck converters especially appropriate when the input-to-output voltage difference is large and when the load current is moderate to high — exactly the case when stepping 24 V down to 5 V or 3.3 V to feed ICs, drivers, and MCUs. The high efficiency reduces heat generation, relaxes thermal management requirements (smaller heatsinks and fewer fans), and extends component life.

A linear regulator (LDO), by contrast, drops voltage by dissipating the voltage difference across a pass element as heat. The power loss is simply the voltage drop multiplied by current, so when Vin is much higher than Vout and currents are significant, the heat becomes excessive — sometimes impractically so. LDOs are attractive only in scenarios with small voltage drops, very low currents, or when very low output noise is required for sensitive analog circuits. In practice for S.T.A.R., using LDOs alone to step 24 V to logic rails would require large heatsinking and greatly increase thermal stress and inefficiency.

Table 14 – Buck and LDO Comparison

Attribute	Buck Converter (Switching)	LDO (Linear Regulator)
Efficiency	High (typically 80–95% depending on load and design).	Low when Vin ≫ Vout (efficiency = Vout/Vin).
Heat dissipation	Low (most energy converted, small losses).	High (dissipates (Vin−Vout)×I as heat).
Output noise & ripple	Switching noise and ripple present; requires filtering (good layout, LC filters, low-ESR caps).	Very low noise; excellent for sensitive analog/ADC/clock rails.
Transient response	Good, but requires loop compensation; faster switching designs respond well.	Very fast and clean transient response for small load steps.
Complexity and BOM	Higher: inductor, diode/synchronous FETs, switching IC, output/input capacitors, layout care.	Lower: single component, few external caps.
Size	Can be compact but requires inductor (board area).	Very compact, minimal components.
EMI considerations	Requires EMI mitigation (layout, filtering, shielding).	Minimal EMI.
Quiescent current (standby)	Can be low; some designs have µA standby, others higher.	Often very low quiescent versions exist (suitable for battery standby).
Cost	Moderate (higher component count).	Low (cheap and simple).

3.3 Communication Methods

We compared HTTP/REST, WebSocket, and MQTT for app↔Pi command/telemetry. For tight closed-loop control, persistent bidirectional messaging is preferred to meet the ≤ 300 ms end-to-end budget.

HTTP/REST

Stateless request/response calls are ideal for configuration and infrequent actions. Each command incurs HTTP method/headers and, unless kept-alive, connection/TLS handshakes; even with keep-alive, request framing and server routing add non-trivial overhead. Polling for feedback (vs push) either increases latency (long intervals) or CPU/network (short intervals). This is suitable for setup, logs, and firmware/version checks, not for continuous joystick/pose streaming.

WebSocket

A single HTTP upgrade creates a long-lived, full-duplex TCP channel with very small per-message headers. Both directions can push asynchronously with minimal context switching, enabling high-rate control (e.g., 20–100 Hz deltas), immediate acknowledgments, and event-driven backpressure. Each connected client holds state on the server; with one controller per spotlight this is trivial. Mature libraries provide heartbeats and robust reconnects.

MQTT

MQTT provides lightweight pub/sub via a broker (e.g., Mosquitto) with QoS levels and retained messages, excellent for multi-client dashboards or intermittent connectivity. Headers are tiny and latency is low; the broker adds a minor LAN hop. QoS 1/2 add ack rounds for delivery guarantees; tune per topic (e.g., control at QoS 0, telemetry at QoS 1). Overhead is broker management—often unnecessary for a single app↔spotlight pair today, but the clean path if multiple operators or a cloud mirror are added later.

Table 15 – Control Channel Comparison

Criterion	HTTP/REST	WebSocket	MQTT
Setup cost	Request per action; keep-alive helps	One upgrade; then steady state	Broker + client connects
Push capability	No (client-pull only)	Yes (bi-directional)	Yes (publish/subscribe)
Per-msg overhead	High (headers)	Low (frame header)	Low (tiny header)
Backpressure	Ad-hoc	Event-loop awaits/queues	Broker buffer + QoS acks

Reconnect	Per request	Heartbeats & fast resume	Retained messages + QoS
Best use here	Config, logs	Primary real-time control	Future multi-client/IoT

Choice: WebSocket for low-latency control; REST for configuration; MQTT reserved for multi-client expansion.

3.5 Embedded Communication

UART

A framed, line-oriented protocol (start byte, command ID, payload, checksum, end byte) with sequence numbers and ACK/NACK provides reliable delivery at 115.2–921.6 kbps. This comfortably supports control deltas (pan/tilt targets, mode changes) at tens of Hz with negligible link latency. Timeouts and limited retries on the Pi side align with the app's heartbeat so faults propagate cleanly.

I2C

Best used on the ESP32's local bus for devices such as a PCA9685 PWM/servo driver or auxiliary sensors. It reduces wiring inside the actuator enclosure and leverages addressing on a short, well-terminated bus. Not preferred for the Pi↔ESP32 interconnect due to open-drain signaling, pull-up tuning, and shared-bus jitter.

SPI

Offers the highest throughput and most deterministic timing via a dedicated clock. It becomes attractive only if future requirements include bulk binary transfers (e.g., dense motion profiles or high-rate sensor data) across the Pi↔ESP32 boundary. The additional wiring and chip-select handling are unnecessary for the present command/ack workload.

Table 16 – Embedded Communication Comparison

Method	Wires	Clocking	Practical Rate	Determinism & Jitter	Pros	Cons
UART	TX, RX, GND	Asynchronous	115.2 kbps–1 Mbps	Good with framed packets and acks	Very simple wiring; easy	No shared addressing; half-duplex

					debugging; resilient	per line pair
I2C	SCL, SDA (+pull-ups), GND	Synchronous (open-drain)	100 kHz/400 kHz/1 MHz	Bus arbitration; moderate jitter under load	Two-wire multi-drop; addressing	Pull-ups + bus integrity; not ideal for long runs
SPI	SCLK, MOSI, MISO, CS, GND	Synchronous (push-pull)	1–20+ MHz	Very good, clocked	Highest throughput; deterministic	More wires; chip select mgmt

Choice: UART for the Pi↔ESP32 primary control channel; I²C reserved for ESP32-local peripherals; SPI deferred for possible high-bandwidth future extensions.

3.6 Software

The software subsystem of the S.T.A.R. system is responsible for real-time human detection, target tracking, motor control communication, and user interaction through a mobile application. To ensure low latency, modularity, and scalability, the software design is evaluated by separating it into two major domains: Computer Vision and Mobile Application. Each domain is further decomposed into layered components, where multiple candidate tools are analyzed and compared before selecting a final implementation.

3.6.1 Computer Vision Software

The computer vision subsystem executes on an embedded computing platform and performs image capture, human detection, motion tracking, and coordinate generation for motor control. To justify design decisions, the vision stack is decomposed into platforms, frameworks, and object detection algorithms.

3.6.2 Computer Vision Platforms

Raspberry Pi Compute Module 5

The Raspberry Pi Compute Module 5 provides a Linux-based embedded computing environment capable of running full computer vision pipelines. It supports native camera interfaces, GPU-assisted image processing, and broad compatibility with OpenCV and machine learning frameworks. Its balance of computational capability, power efficiency, cost, and ecosystem maturity makes it well suited for real-time embedded vision tasks in the S.T.A.R. system.

NVIDIA Jetson Nano

The NVIDIA Jetson Nano was evaluated due to its integrated CUDA-capable GPU, which allows for accelerated neural network inference. While this platform excels in deep learning workloads, its higher cost, increased power consumption, and limited availability make it excessive for single-target tracking, where lightweight detection models already meet performance requirements.

Table 17 - Platform Comparison

Criterion	Raspberry Pi CM5	NVIDIA Jetson Nano
Primary Role	Embedded vision processing	GPU-accelerated edge AI
CPU	Quad-core ARM Cortex-A76	Quad-Core ARM Cortex-A57
Compute Capability	Moderate–High (CPU + GPU)	High (CUDA GPU)
Power Consumption	4-8 W	5 - 10 W
Embedded Suitability	High	Medium
Cost Efficiency	High	Medium

Choice: The Raspberry Pi Compute Module 5 is selected as the primary computer vision platform due to its strong embedded performance, Linux support, and compatibility with real-time vision frameworks while also being a more power efficient and affordable selection.

3.7 Computer Vision Frameworks

OpenCV

OpenCV is a comprehensive computer vision library that supports camera input, image preprocessing, geometric transformations, object tracking, and deep neural network inference. Its ability to combine classical vision techniques with modern deep learning models within a single framework simplifies system integration and ensures predictable execution timing. OpenCV's maturity and optimization for ARM-based systems make it ideal for real-time embedded vision.

TensorFlow Lite

TensorFlow Lite is designed specifically for efficient inference of trained neural networks on edge devices. It supports quantization and hardware acceleration, allowing models to run with reduced memory and computational overhead. However, TensorFlow Lite focuses solely on inference and requires additional libraries such as OpenCV for camera handling and tracking.

PyTorch with TorchScript

PyTorch provides a flexible environment for developing and training machine learning models. When exported using TorchScript, models can be deployed for inference. Despite this flexibility, PyTorch introduces higher runtime overhead and longer startup times, making it less suitable for lightweight embedded systems.

MediaPipe

MediaPipe offers highly optimized pipelines for predefined vision tasks such as pose estimation and hand tracking. While it delivers excellent performance for supported tasks, its reliance on predefined processing graphs and limited customization options restrict its suitability for specialized tracking applications.

Table 18 - Framework Comparison

Criterion	**OpenCV**	**TensorFlow Lite**	**PyTorch / TorchScript**	**MediaPipe**

Primary Role	End-to-end CV toolkit	Edge inference for TF models	Research DL; scripted deployment	Prebuilt perception graphs
Edge Readiness	High (CPU/NEON; broad backends)	High (quantization, delegates)	Medium (heavier runtime)	High (mobile/edge focused)
Pipeline Breadth	Wide (capture → inference → tracking)	Narrow (inference only)	Narrow → Medium	Medium (task-focused)
Integration Effort	Low	Medium	Medium–High	Medium

Choice: OpenCV is selected as the primary computer vision framework due to its unified support for camera input, tracking, deep learning inference within a single embedded-optimized library and its compatibility with the Raspberry Pi

3.8 Object Detection Algorithms

YOLO (You Only Look Once)

YOLO-based object detection models perform object localization and classification in a single forward pass through a convolutional neural network, enabling high detection accuracy with low inference latency. Lightweight variants such as YOLO-Tiny and YOLO-Nano are well suited for embedded platforms, providing reliable human detection while maintaining real-time performance. These models demonstrate strong robustness to partial occlusion, scale variation, and lighting changes, making them appropriate for live tracking applications.

MobileNet-SSD

MobileNet-SSD is designed for mobile and embedded systems, combining a lightweight backbone network with a single-shot detection framework. While it achieves fast inference with reduced computational requirements, it generally exhibits lower detection accuracy and reduced robustness to occlusion and dynamic backgrounds compared to

YOLO-based models. This limitation can result in less stable tracking under real-world conditions.

Histogram of Oriented Gradients (HOG) with SVM

The HOG with Support Vector Machine approach is a classical object detection method that extracts gradient-based features and applies a trained classifier for detection. Although computationally efficient and suitable for controlled environments, it struggles with rapid motion, partial occlusion, and complex or cluttered backgrounds. These limitations reduce its reliability for real-time human tracking in dynamic scenes.

Table 19 - Algorithm Comparison

Criterion	**YOLO (Tiny / Nano)**	**MobileNet-SSD**	**HOG + SVM**
Detection Accuracy	High	Medium	Medium
Inference Latency	Low (~33–66 ms / frame)	Very Low (~25–50 ms / frame)	Medium (~100–200 ms / frame)
Throughput	15-30 FPS	20-40 FPS	5-10 FPS
Robustness	High	Medium	Low
Resource Usage	Moderate (~300-500 MB)	Low (~200-350 MB)	Moderate (~150-250 MB)
Modern Relevance	State-of-the-art	Widely used	Legacy

Choice: A YOLO-based lightweight detector (YOLO-Tiny / YOLO-Nano) is selected due to its superior balance of accuracy, robustness, and real-time performance on embedded hardware.

3.9 Mobile Application Software

The mobile application provides the primary user interface for system control, allowing users to start and stop tracking, manually override motor movement, calibrate motors, and select new targets. The mobile software is to be a minimal stack as it only is used for control and does not need to be a fully deployed application.

3.9.1 Frameworks

React Native

React Native enables cross-platform mobile application development using JavaScript and native UI components. It allows developers to maintain a single codebase for both Android and iOS while rendering platform-native interfaces. Its large ecosystem, strong community support, and seamless integration with REST APIs and WebSocket-based backends make it well suited for real-time control applications.

Flutter

Flutter uses a compiled rendering engine written in Dart to deliver consistent UI behavior across platforms. While it offers strong performance, it introduces an additional language and framework that increases onboarding and integration complexity for JavaScript-based teams.

Native Android and iOS Development

Native development offers the highest level of platform integration and performance but requires maintaining separate codebases, increasing development time and complexity.

Table 20 - Frontend Comparison

Criterion	**React Native**	**Flutter**	**Native Android / iOS**
Performance	High	High	Very High
Codebase Count	Single	Single	Multiple

Development Speed	High	Medium	Low
Ecosystem Support	Very Large	Medium	Platform-specific

Choice: React Native is selected as the mobile frontend framework due to its single-codebase development model and strong JavaScript ecosystem.

3.10 Database and State Storage

In-Memory Storage

In-memory storage provides the lowest possible access latency and is used to maintain real-time system state, including target coordinates, motor commands, and current operating mode. Because this data is transient and frequently updated, in-memory storage ensures deterministic access times and avoids I/O blocking that could degrade control loop performance.

SQLite

SQLite is a lightweight, file-based relational database that provides persistent storage without requiring a dedicated database server. It is well suited for infrequently accessed data such as system configuration parameters, calibration constants, and user preferences. Its minimal runtime overhead and local file storage make it appropriate for embedded Linux environments.

Cloud-Based Databases

Cloud-based databases offer scalability and remote accessibility but introduce additional network latency, external connectivity requirements, and operational complexity. For a real-time, locally controlled system such as S.T.A.R., these drawbacks outweigh the benefits, particularly for time-critical control and vision data

Table 21 - Storage Comparison

Criterion	**In-Memory Storage**	**SQLite**	**Cloud Database**

Latency	Very Low	Low	High
Persistence	No	Yes	Yes
System Complexity	Very Low	Low	High
Embedded Suitability	High	High	Low

3.11 ESP32 Firmware

Language & Runtime (C/C++)

C/C++ with Arduino core or ESP-IDF provides deterministic access to UART, PWM (LEDC), GPIO, timers, and FreeRTOS. Lower abstraction yields tighter loops and lower jitter than interpreted environments—ideal for servo timing. For PWM/LEDC, select frequency/resolution to preserve duty precision and avoid servo chatter (e.g., 50–300 Hz typical for hobby servos; higher update rates demand careful filtering). For the UART link, 115 200 bps is a simple baseline; compact, coalesced frames keep well below saturation. If dense telemetry is later required, 460 800–921 600 bps is available with ESP-IDF (validate cable integrity/level shifting). With FreeRTOS, separate UART RX/TX, motion control, and safety/limit checks into distinct tasks with priorities; use queues/semaphores to bound latency and protect timing.

IDE/Tooling

Arduino IDE accelerates iteration and serial debugging through simple builds and a rich library ecosystem. PlatformIO + ESP-IDF is available for lower-level drivers, custom build flags, and partition tuning when tighter control or optimization is needed.

Table 22 – Platform Language Trade-offs

Criterion	C/C++ (ESP32)	Python (Pi)
Execution	Native (no VM)	Interpreted

Jitter	Very low (timers/ISRs)	Low–moderate (GC/GIL; mitigated with async)
Hardware access	Direct (ESP-IDF/Arduino)	Via bindings (serial, GPIO)
Role	Motor control, PWM, UART	Vision, orchestration, server

3.12 End-to-End Interaction & Timing Budget (How it Works)

1. **App → ESP32 (Control & Mode Selection):**
   The mobile application establishes a low-latency wireless connection with the ESP32-S3 and transmits user commands such as mode selection (Idle, Manual, Auto), Directional inputs, and start/stop requests. Control messages are rate-limited (e.g., 30–60 Hz) and coalesced per UI frame to maintain responsiveness while minimizing network overhead. The ESP32 immediately acknowledges received commands, enabling the application to measure round-trip latency and present accurate system status to the user.
2. **ESP32 (Coordination & Arbitration):**
   Upon receiving app commands, the ESP32 acts as the system's mode authority. It determines whether motor outputs are driven by manual user input or by autonomous tracking telemetry from the Raspberry Pi. Control messages destined for the vision subsystem are forwarded to the Pi over a dedicated serial link using compact binary frames (e.g., header, command ID, payload, checksum). UART reception is handled through a ring buffer to prevent data loss under burst traffic.
3. **Pi (Vision Processing & Server Services):**
   The Raspberry Pi CM5 runs the OpenCV-based YOLO tracking pipeline and computes target error metrics, including torso centroid offset and tracking confidence. A lightweight tracker interpolates between detection frames to maintain smooth telemetry. A FastAPI service operates on the Pi to manage vision-related configuration, monitoring, and non-time-critical system functions, without participating directly in real-time motor arbitration.
4. **Pi → ESP32 (Tracking Telemetry):**
   Tracking updates are transmitted from the Pi to the ESP32 at a bounded update rate suitable for stable control. The ESP32 validates telemetry freshness and applies rate limiting or damping as needed before incorporating it into motor control logic. If telemetry becomes stale or invalid, the ESP32 safely reverts to Idle or Manual mode.
5. **ESP32 (Actuation):**
   The ESP32 firmware converts the selected control input into PWM setpoints using LEDC timers. Motion smoothing, rate limiting, and mechanical constraint enforcement are applied before commanding the pan and tilt servos. Servo

updates execute on a fixed periodic timer to ensure deterministic behavior independent of communication jitter.

6. **ESP32 → App (Feedback):**
System state, tracking status, and summarized telemetry are relayed back to the mobile application by the ESP32 at a UI-friendly rate. Feedback is throttled to match rendering capabilities, ensuring smooth visualization without saturating the wireless link.

Table 23 – Control-Loop Budget (Design Targets)

Segment	Typical Work	Budget Notes
App UI frame	Gesture → delta coalesce	Keep < 16.67 ms; avoid heavy JSON on main thread
App→ESP 32 transport	WebSocket send	Small binary frames; heartbeat for loss detection
Pi handling	Async route → UART	Avoid blocking; bounded queues for bursts
UART	Bytes on wire	115 200 bps baseline; compact frames minimize latency
ESP32 actuation	Parse → PWM setpoint	Timer-driven; prioritize motion over logs
Feedback return	Telemetry/centroid	Throttle to UI frame rate

Selected Stack (recap with performance rationale)

- **Mobile:** React Native (native feel; efficient UI updates; lightweight command batching for responsive control).
- **Computer Vision / Pi Services:** Python + OpenCV on Raspberry Pi CM5; YOLO-based object detection for real-time human tracking; compact telemetry generation (target offset, confidence) suitable for low-latency control.
- **Communication:** App → ESP32 over low-latency Wi-Fi (LAN primary, ESP32 AP fallback) for commands and status; ESP32 ↔ Pi via UART for deterministic telemetry exchange; REST endpoints on Pi for configuration and diagnostics.

- **ESP32 Firmware:** C/C++ (ESP-IDF/FreeRTOS); mode arbitration and real-time motor control; UART-framed messaging with validation; LEDC PWM for pan/tilt servos; I^2C for local peripherals.

Chapter 4 - Standards and Design Constraints

4.1 Industrial Standards

Designing the S.T.A.R. system means balancing real-world engineering limits with a system that is safe, reliable, and practical. Even though this is a student project, the final device still needs to follow the general ideas behind lighting industry standards and safety guidelines. Since this is a lighting fixture with electrical components, moving parts, and bright illumination, the design process has to consider electrical safety, heat management, optical performance, and user safety. This chapter explains the standards and constraints that shaped our decisions and how they influenced the design.

Even though student prototypes don't go through official certification processes, it's still important to understand the standards used in professional lighting equipment. These standards help guide design decisions so that the final system operates safely and reliably.

4.1.1 Optical Standards

Lighting Fixture Safety — IEC 60598

The IEC 60598 luminaire safety standard is directly relevant to the S.T.A.R. project because it defines the fundamental electrical, thermal, and mechanical safety requirements expected of professional lighting fixtures. Since S.T.A.R. operates similarly to a stage luminaire—with a high-power LED source, reflector optics, control electronics, and a motorized gimbal—it must be designed with the same safety considerations used in commercial products.

Electrical safety requirements in IEC 60598 guided how the S.T.A.R. system is constructed and wired. The design assumes proper insulation between energized components and any user-accessible surfaces, and all exposed conductive parts of the housing are intended to be bonded to earth ground to reduce shock risk in the event of a fault. Internal wiring is routed to avoid sharp edges and moving parts, with strain relief provided where cables pass through the gimbal to prevent wear or accidental disconnection during motion.

Thermal management is another major focus of the standard and was a key design consideration for S.T.A.R. The high-power LED produces significant heat, so passive heat sinking and conservative drive currents are used to limit temperature rise within the enclosure. The fixture housing is designed to withstand continuous operation without reaching unsafe external surface temperatures, reducing both burn and fire hazards. These measures align with IEC 60598 requirements for temperature limits and thermal endurance.

Mechanical safety requirements are especially important due to S.T.A.R.'s motorized gimbal and potential overhead mounting. The gimbal structure and mounting interfaces are designed with appropriate safety factors to support static and dynamic loads during operation. Fasteners and structural components are selected to resist loosening under vibration, and cable routing through rotating axes is implemented to avoid pinching or excessive bending.

IEC 60598 also addresses photobiological and environmental considerations. LED output levels are selected with photobiological safety in mind to reduce hazardous light exposure during normal operation. While the current prototype is not fully rated for outdoor use, the enclosure layout allows for future improvements such as sealing, gaskets, or rated connectors if operation in dusty or damp environments is required.

Optical Safety — IEC 62471

IEC 62471, titled "Photobiological Safety of Lamps and Lamp Systems," is the international standard that defines how to evaluate the optical hazards produced by light sources. It focuses on the potential biological effects of radiation in the visible, ultraviolet (UV), and infrared (IR) ranges on both the eyes and skin. The standard applies to LEDs, lasers, and all other types of lamps, describing how to measure their spectral output and determine safe exposure limits.

This standard directly applies to S.T.A.R. because it uses a high-intensity LED source and optical system that focuses light into a controlled beam. Since the fixture is designed to track and follow performers, its light output will be directed towards people on stage, making eye and skin exposure an important consideration. Testing under IEC 62471 would determine the risk group classification for S.T.A.R.'s LED engine and help verify that it can be safely operated in close proximity to performers.

Design features such as lens selection, beam shaping, and diffuser coatings can all influence optical safety. By referencing IEC 62471, the S.T.A.R. team can make informed decisions about limiting hazardous brightness levels, minimizing blue-light exposure, and including appropriate warning labels if needed. In short, this standard ensures that S.T.A.R. delivers professional lighting performance without posing photobiological risks, aligning it with the same safety expectations as commercial LED stage fixtures.

4.1.2 PCB Design Standards

PCB Quality and Acceptability — IPC-A-600K

IPC-A-600 defines the acceptability requirements for printed circuit boards before and after assembly. It establishes visual and dimensional criteria used to evaluate bare PCBs, including conductor spacing, annular ring integrity, hole quality, solder mask coverage, and laminate conditions. The standard categorizes boards into different classes based on end-use reliability requirements, ranging from general consumer electronics to high-reliability applications. By following IPC-A-600, this project ensures that fabricated

PCBs meet consistent quality benchmarks and are free from defects that could compromise electrical performance, manufacturability, or long-term reliability.

Qualification and Performance Specification for Rigid Printed Boards — IPC-6012

IPC-6012 specifies the qualification and performance requirements for rigid printed circuit boards. Unlike acceptability standards, IPC-6012 focuses on defining how a PCB must be designed, fabricated, and tested to meet electrical, mechanical, and environmental performance expectations. It covers aspects such as material selection, copper thickness, hole reliability, dielectric integrity, and thermal stress tolerance. Compliance with IPC-6012 helps ensure that the PCB design for this system can withstand operational stresses such as current loading, thermal cycling, and mechanical vibration, making it well suited for an electromechanical system involving motors, power electronics, and embedded processors.

Acceptability of Electronic Assemblies — IPC-A-610

IPC-A-610 establishes the acceptability criteria for electronic assemblies, providing guidance on workmanship quality for soldered components, connectors, and wiring on populated PCBs. This standard defines what constitutes acceptable, marginal, and defective conditions for solder joints, component alignment, lead trimming, and mechanical mounting. IPC-A-610 is particularly important for ensuring consistent assembly quality when integrating components such as microcontrollers, motor controllers, connectors, and power circuitry. Adhering to this standard reduces the risk of intermittent faults, mechanical failures, and premature system degradation, thereby improving overall system reliability and serviceability.

4.1.3 Electrical Standards

Electrical Safety — NFPA 70 (NEC)

NFPA 70 is an important standard for S.T.A.R. project because it governs how electrical systems are wired, grounded, and protected to prevent shock, fire, and equipment damage. Since S.T.A.R. integrates a high-power LED source, motor drivers, control electronics, and an external power connection, the system must be designed with the same electrical safety practices used in professional lighting equipment and installations.

For S.T.A.R., NEC guidelines directly influence decisions related to power distribution and wiring. The current draw of the LED and motors requires appropriate wire gauge selection to prevent overheating, while fuses or current-limiting protection are necessary to guard against short circuits or fault conditions. Proper grounding of all exposed metal components is critical to reduce shock risk, particularly because the fixture may be handled during setup, adjustment, or maintenance. The NEC also addresses insulation ratings, strain relief, and connector selection, all of which are relevant where power and control cables interface with the moving gimbal assembly.

The NEC is especially relevant because S.T.A.R. is intended to operate in environments similar to theaters, performance venues, or event spaces. NFPA 70 includes provisions specific to stage and entertainment lighting installations, which emphasize safe practices for portable equipment, temporary power connections, and systems that may be rigged overhead. These requirements align closely with how S.T.A.R. would realistically be deployed, handled, and powered in a real-world setting.

Applying NFPA 70 as a design reference helps ensure that S.T.A.R. is not only electrically functional, but safe to integrate into existing venue infrastructure. Following NEC principles reduces the risk of electrical failure, fire hazards, or unsafe operating conditions, and supports the project's goal of transitioning from a prototype system into a fixture that reflects professional lighting industry expectations.

Safety of Machinery: Electrical Equipment of Machines — IEC 60204-1

IEC 60204-1 establishes the fundamental safety requirements for the electrical, electronic, and programmable electronic equipment used in industrial machinery. The standard addresses the entire electrical system of a machine, including power distribution, grounding and bonding, overcurrent protection, wiring practices, emergency stop circuits, and protective interlocking. Its primary objective is to reduce the risk of electrical shock, fire, and hazardous machine behavior during normal operation, maintenance, and fault conditions. For motor-driven systems such as servo-controlled gimbals and automated lighting equipment, IEC 60204-1 provides guidance on safe motor control circuits, power isolation methods, and fail-safe operation. Compliance with this standard ensures that electrical systems are designed and constructed in a manner that promotes operator safety, system reliability, and maintainability.

Environmental Testing — IEC 60068

IEC 60068 is a comprehensive series of standards that defines methods for evaluating the ability of electronic and electromechanical components to withstand environmental stresses encountered during their lifecycle. These stresses include temperature extremes, humidity, vibration, mechanical shock, and exposure to dust or moisture. Rather than specifying performance requirements, IEC 60068 provides standardized test procedures that allow designers and manufacturers to assess durability, reliability, and long-term performance under controlled conditions. For systems incorporating servo motors, sensors, and embedded electronics, such as automated tracking or motion-control platforms, IEC 60068 testing helps validate that components will continue to operate correctly when subjected to motion-induced vibration, heat from power electronics, or varying ambient conditions. Adhering to IEC 60068 testing methodologies supports robust system design and reduces the likelihood of environmental-related failures in real-world operation.

4.1.4 Industry Standards

Universal Serial Bus (USB)

Universal Serial Bus (USB) is an industry-standard interface designed to provide a unified solution for data communication and power delivery between electronic devices. USB defines both the physical connectors and the electrical, signaling, and protocol layers required for reliable device interconnection. Over successive revisions, USB has evolved to support increasing data rates, higher power delivery, and enhanced device management, making it suitable for applications ranging from low-speed peripherals to high-performance embedded computing platforms. In embedded and control systems, USB is commonly used for device configuration, firmware updates, peripheral expansion, and powering low- to moderate-power electronics. Its standardized connectors, hot-plug capability, and widespread support across operating systems make USB a practical and robust interface for development, debugging, and user interaction in complex systems. We will be using USB for several connectors in the S.T.A.R. system

High-Definition Multimedia Interface (HDMI)

High-Definition Multimedia Interface (HDMI) is a widely adopted digital interface standard for transmitting uncompressed video and multi-channel audio over a single cable. HDMI specifies the electrical signaling, connector form factors, communication protocols, and content protection mechanisms required to deliver high-quality audiovisual data with minimal latency. It supports a range of resolutions, color depths, and refresh rates, enabling real-time display of high-definition and ultra-high-definition content. In embedded and vision-based systems, HDMI is frequently used to output live video feeds, graphical user interfaces, and diagnostic visualizations to external monitors. Its ability to carry synchronized audio and video, along with its strong industry adoption and plug-and-play reliability, makes HDMI a valuable interface for system monitoring, debugging, and user-facing visualization. HDMI will be used to display a live feed of our tracking software to an external monitor.

4.1.5 Software and Control Standards

Functional Safety Considerations — IEC 61508

IEC 61508 is important to the S.T.A.R. project because it provides an approach to managing risk in systems that rely on programmable electronics and software to control motion. Unlike purely mechanical fixtures, S.T.A.R. depends on software driven decision making to command the gimbal, interpret tracking data, and respond to user input from a control application. This makes functional safety an important consideration, even in a prototype or research setting.

For S.T.A.R., the most relevant risks are not component failure alone, but behavioral failures, such as unexpected motion, loss of brightness control, loss of communication between the app and the fixture, or software crashes during operation. IEC 61508 emphasizes identifying these failure modes early and defining predictable, safe responses when they occur. Applying this ensures that the system fails in a controlled manner rather than entering an unsafe operation state.

Design principles that follow IEC 61508 S.T.A.R. ensure clear safe default behaviors. For example, if communication with the control app is lost, the system should halt gimbal motion or transition to a known neutral position rather than continuing to move autonomously. Similarly, software limits on motion speed, angular range, and acceleration help prevent hazardous behavior even if erroneous commands are received.

IEC 61508 also encourages separation between critical control functions and higher-level features such as user interface logic or tracking algorithms. This reduces the likelihood that a fault in non-essential software propagates into unsafe mechanical behavior. In addition, the standard promotes systematic testing and validation of safety-related functions, supporting confidence that the system behaves predictably under fault conditions. In industrial servo systems, this often includes features such as Safe Torque Off (STO), fault detection, and controlled stopping. While hobby servos typically do not implement certified safety features, these standards still inform best practices for system-level design, such as fail-safe defaults and emergency power removal.

While S.T.A.R. is not intended to be formally certified under IEC 61508, using the standard as a design reference demonstrates that functional safety was deliberately considered alongside performance and functionality. Incorporating these principles strengthens the overall reliability of the system and supports the project’s goal of transitioning from a proof-of-concept prototype toward a realistic, deployable lighting instrument.

4.2 Design Constraints

4.2.1 Illumination System Constraints

Size

The size of the illumination system places important constraints on the optical design of S.T.A.R., particularly in the selection of the reflector and lens assembly. When choosing the reflector, it was important to look at ones which are physically capable of housing the light source while also providing enough clearance for beam formation. The reflector must be large enough to capture a meaningful portion of the emitted light and redirect it toward the gate without clipping or excessive spill.

In addition, the diameter of the reflector must be compatible with the diameter of the lens system. The lens must be large enough to accept the converging beam from the reflector, which requires careful matching between reflector size, lens aperture, and focal geometry.

If the reflector is undersized or mismatched with the lens diameter, the beam will be truncated, reducing efficiency and introducing unwanted artifacts.

Proper spacing between the source, reflector, and lens is required to allow the light to properly converge toward a focal point before entering the lens system. This spacing directly affects the overall length of the illumination module and limits how compact the fixture can be made. As a result, the minimum viable size of the illumination system is driven by geometry rather than packaging alone. Our best effort was made to minimize the size and length of the illumination system, but there is only so much that can be done while allowing the light to shape and bend the way it needs to in order to achieve a proper image.

Material

Material selection for the illumination system is determined by both mechanical and thermal requirements. To minimize mass and reduce the load on the motorized gimbal, the illumination housing is fabricated using 3D-printed components rather than a traditional metal enclosure. While a metal housing would provide superior structural rigidity and thermal conductivity, its added weight would significantly increase the required torque of the gimbal system, negatively impacting tracking performance and responsiveness, and increasing power consumption.

Weight reduction is especially important because the illumination module is mounted at the end of the gimbal, where additional mass has the greatest effect on dynamic behavior. Using a lightweight polymer housing allows the system to maintain acceptable balance and motion control while still supporting the optical and electrical components.

Thermal performance is also a critical factor in material selection. The high-power LED generates substantial heat during operation, requiring the housing material to tolerate elevated temperatures without deformation or structural degradation. For this reason, HTPLA was selected over standard PLA due to its higher heat resistance and improved thermal stability. While HTPLA does not offer the same heat dissipation as metal, it provides a practical compromise for a prototype system when combined with dedicated heat sinking and controlled operating conditions.

Overall, the use of HTPLA enables rapid iteration, reduced mass, and sufficient thermal tolerance for the S.T.A.R. illumination system, while leaving open the possibility of transitioning to a metal housing in future revisions once gimbal load capacity and thermal requirements are further optimized.

Optical Étendue

Étendue is an important consideration in the design of the S.T.A.R. illumination system because it fundamentally limits how compact and efficient the optical system can be. Even if the correct optical components could be selected to reduce the physical size of the system, étendue would still impose a constraint on how tightly the emitted light can be collected and focused. The finite size and angular emission of the LED source set a minimum optical throughput that must be preserved throughout the system. As a result, attempting to force the light through a smaller reflector or lens aperture would lead to clipping, reduced efficiency, and degraded beam quality rather than true size reduction.

Because étendue cannot physically be reduced based on the optical prescription, the reflector and lens diameters must be large enough to accept the full area-angle product of the source. This the size of the illumination optics and prevents miniaturization of the fixture. In the context of S.T.A.R., this means that achieving sufficient brightness and beam uniformity requires an illumination system of a certain minimum scale, regardless of mechanical packaging constraints. Étendue therefore serves as a fundamental optical limitation that directly influences the overall size, efficiency, and performance of the lighting instrument.

Optical Alignment Tolerances

Optical alignment tolerances are a significant constraint in the S.T.A.R. illumination system due to the precision required between the light source, reflector, and lens assembly. Small positional or angular misalignments can introduce beam asymmetry, reduced efficiency, and degraded focus quality. Because the system relies on multiple optical elements working in sequence, alignment errors can accumulate, making tolerance control an important consideration during both design and assembly.

The use of 3D printing introduces challenges related to dimensional accuracy and repeatability. While additive manufacturing enables lightweight structures, printed parts are subject to layer resolution limits, thermal shrinkage, and print-to-print variability. These effects can cause deviations in bore diameters, mounting surfaces, and reference features, making it difficult to achieve tight concentricity and axial alignment without post-processing. Circular prints are also subject to warping or distorting along an axis, which results in loose fitting enclosures. As a result, the optical design must tolerate small misalignments rather than relying on ideal placement.

4.2.2 Imaging System Constraints

Mechanical Considerations

When designing optical components for detection of a wide field of view, there are several key constraints that must be considered for the camera system. One primary consideration is the necessity of a wide effective focal length, as the subject and stage will be at a distance ~7.5 meters and thus the system length will be increased. With incident light captured on a small sensor at a wide angle, the magnitude of demagnification is greater, requiring the need of precise alignment and spacing. The design of stable lens housing mounts is also required to prevent any tilt and turning of the lens that would deviate the incident light from hitting the sensor, especially since the overall system will be on a moving axis. Mechanically, the design must be as precise as possible, proving another constraint.

Discrepancies in the theoretical lens design in terms of spacing may cause an assortment of aberrations such as coma, chromatic aberration, and spherical aberration causing a decrease in image quality, an essential aspect needed in order to accurately track subjects moving along the plane of view. To prevent these aberrations, the use of aspheres and achromats are primarily considered as opposed to singlets to account for such wide angles and the potential of stray light.

Coatings and Materials

The coating and material of each lens acts as another optical constraint placed upon the lens design. Since the system will be operating in the visible light regime, the lenses must be coated to spectrally transmit the proper wavelengths. The usage of a crown fused with a flint lens is typically seen in achromatic doublets in order to correct light using a lower dispersion refractive index to a higher dispersion refractive index, respectively. The sensor also follows this spectral transmission property, where it can only observe a set wavelength range, requiring each optical component in the imaging system to follow the same spectral transmission so it can effectively permit wanted light through the system from the front end to the detector. These constraints provide the basis of what optics and materials are needed to successfully build an imaging system for the purpose of displaying an image capable of tracking subjects at wide angles in a far range.

4.2.3 Electrical Hardware Constraints

Mechanical Load and Torque Limitations

The pan–tilt gimbal must support the mass and moment arm of a Source Four–style stage light while maintaining smooth and accurate motion. This places a constraint on servo motor torque, gearbox strength, and mechanical rigidity of the gimbal structure. Excessive weight or poor mass distribution can lead to servo saturation, increased positional error, or mechanical wear. As a result, the mechanical design must carefully balance structural robustness with weight minimization, ensuring that the selected servos can provide sufficient torque margin under dynamic conditions such as rapid acceleration, deceleration, and continuous tracking motion.

Power Availability and Distribution

The system is constrained by the available power budget and the need to safely distribute high current from a 24 V, 25 A power supply to multiple subsystems, including servo motors, embedded processors, cameras, and auxiliary electronics. High-current loads such as servos introduce transient spikes and electrical noise, which can negatively impact sensitive digital components if not properly isolated. This necessitates the use of buck converters, dedicated power rails, protective circuitry, and appropriate conductor sizing. Thermal dissipation and connector ratings further constrain how power can be routed and managed within the enclosure.

Thermal Management

Heat generation from the illumination source, power electronics, and embedded computing hardware presents a significant constraint on enclosure design and component placement. High-power LEDs or lamps, voltage regulators, and processors can raise internal temperatures to levels that degrade performance or reduce component lifespan. The system must therefore incorporate adequate thermal mitigation strategies, such as heat sinks, forced-air cooling, ventilation paths, and temperature-rated materials. Thermal limits constrain continuous operating time and influence both mechanical layout and enclosure geometry.

Electrical Noise and Signal Integrity

High-current servo motors and switching power regulators introduce electrical noise that can interfere with low-voltage logic, camera signals, and communication interfaces. This constrains PCB layout, grounding strategy, cable routing, and component placement. Improper noise mitigation can result in erratic motor behavior, communication errors, or unreliable sensor data. As a result, the design must include decoupling capacitors, proper grounding schemes, shielded cables where appropriate, and physical separation between noisy and sensitive subsystems.

Physical Size and Enclosure Constraints

All system components, including PCBs, power supplies, motor controllers, and wiring, must fit within a 3D-printed enclosure while maintaining accessibility, airflow, and structural integrity. The enclosure size is constrained by stage-mounting considerations, aesthetic requirements, and portability. These limitations affect PCB dimensions, connector placement, and cable management. The enclosure must also provide mechanical protection and strain relief without impeding serviceability or thermal performance.

Interface and I/O Limitations

The number of available I/O pins, communication buses, and expansion interfaces on the ESP32-S3 and Compute Module constrains how many peripherals can be supported simultaneously. Servo channels, sensors, status LEDs, user controls, and communication interfaces must all share limited hardware resources. This necessitates the use of I^2C expanders, PWM drivers such as the PCA9685, and careful bus management. Bandwidth and addressing limitations further restrict how subsystems are interconnected and scaled.

Reliability and Safety Constraints

Because the system operates in proximity to people and uses high-power electrical and mechanical components, safety and reliability impose critical hardware constraints. Components must be rated for the expected voltages, currents, and environmental conditions, and mechanical assemblies must be designed to prevent unintended motion or structural failure. Fail-safe behavior, such as controlled shutdown during faults or power loss, constrains motor selection, power architecture, and control circuitry to ensure safe operation during live performance environments.

4.2.4 Software Constraints

Embedded Computing Resource Constraints (Raspberry Pi Compute Module 5)

The computer vision subsystem executes on the Raspberry Pi Compute Module 5, which imposes constraints on available processing power, memory, and hardware acceleration. While the platform supports Linux-based vision frameworks, its computational resources are limited compared to desktop or server-class systems. As a result, the software design must avoid large or computationally intensive deep learning models and instead prioritize lightweight detection algorithms and optimized pipelines. These constraints directly influence model selection, frame resolution, inference frequency, and the overall structure of the vision software to ensure real-time operation.

Real-Time Processing and Latency Constraints

The S.T.A.R. system requires near real-time response to target motion in order to maintain smooth and accurate spotlight tracking. Excessive processing latency in the computer vision pipeline or delays in command transmission can result in visible lag, overshoot, or loss of target alignment. This constraint limits the use of blocking operations, high-latency inference techniques, and unnecessary data processing during live operation. Consequently, the software architecture must emphasize asynchronous execution, predictable timing behavior, and efficient data flow between system components.

Microcontroller Firmware Timing Constraints (ESP32-S3)

The ESP32-S3 firmware is responsible for real-time motor control while concurrently managing communication with both the Raspberry Pi and the mobile application. Servo control requires precise and consistent PWM signal generation, which places strict timing constraints on firmware execution. Complex logic, blocking communication routines, or excessive computational overhead may interfere with deterministic motor behavior. Therefore, firmware tasks must be carefully prioritized, and real-time scheduling mechanisms must be used to ensure that time-critical motor control functions are not disrupted.

Communication Bandwidth and Latency Constraints

Inter-device communication within the system is constrained by available bandwidth, transmission latency, and reliability of wireless links. The ESP32-S3, in particular, has limited throughput and processing capability, making it unsuitable for handling high-bandwidth data such as raw or compressed video streams. This constrains the software architecture to transmit only low-bandwidth control commands and tracking telemetry through the microcontroller, while high-data-rate operations are handled directly by the Raspberry Pi. Communication protocols must therefore be efficient, lightweight, and resilient to intermittent connectivity.

Cross-Platform Mobile Application Constraints

The mobile application is required to operate across multiple platforms, including Android and iOS. Differences in operating system behavior, background execution policies, and networking restrictions limit the extent to which platform-specific optimizations can be applied. This constraint influences the selection of cross-platform development frameworks and restricts the use of certain native APIs. As a result, the application design must rely on portable libraries and standardized communication mechanisms to maintain consistent functionality across devices.

Software Modularity and Interface Stability Constraints

The S.T.A.R. system consists of multiple independently developed software subsystems, including computer vision processing, embedded firmware, and a mobile user interface. Integration of these components requires stable and well-defined interfaces to prevent cascading failures during development and testing. This constraint limits frequent changes to communication protocols or data formats once integration begins and necessitates early definition of application programming interfaces (APIs) and message structures.

Development Timeline and Debugging Constraints

The Senior Design project timeline imposes limitations on the complexity of software tools and frameworks that can be realistically adopted. Debugging distributed systems spanning embedded firmware, Linux-based processing, and mobile platforms is time-intensive and resource-demanding. This constraint discourages the use of experimental or poorly documented technologies and favors mature, widely supported software libraries that reduce integration risk and development overhead.

Reliability and Fault-Handling Constraints

The software must safely handle fault conditions such as communication dropouts, stalled vision processing, or loss of tracking data. However, implementing comprehensive fault-tolerant mechanisms across all software layers is constrained by system complexity and development time. As a result, the design is limited to essential safeguards, including timeout detection, controlled state transitions, and safe motor behavior during error conditions, rather than full redundancy or autonomous recovery at every level.

4.3 Practical Constraints

Time

The biggest time constraint for this project is the time to hypothesize, theoretically plan, test, and physically execute the demands of S.T.A.R. This is important as there are many considerations and priority orders that must come first before being able to fully implement the system such as meeting lecture deadlines. Testing and debugging the system requires a significant amount of time as well, as each system has issues that are inevitable. This process of optimization and further testing requires significant amounts of time to ensure that S.T.A.R. is meeting the metrics and specifications that were decided upon.

Another consideration is time delays from external factors, a major one being the shipping time, as the components ordered come from a variety of vendors which can take a while to order and ship. Coordinating the timing of the retrieval of products requires a lot of time allocation, and custom designed components such as lenses or printed circuit-boards tend to take a lot of time to acquire depending on the vendor itself. To mitigate the delay, the theoretical design and ordering of the parts must be done early to allocate more time to testing and optimizing the system.

Economic

Perhaps the most considered constraint of this project is the economics within the entire system. Cost is a major factor that influences the scope of the project, and the capabilities between the goals listed. Since this project is not sponsored, the cost allocations for S.T.A.R. is funded entirely out of pocket. With the efforts of four members, component selection and project capabilities became a limitation when trying to keep the cost of the design under a reasonable budget set amongst the group.

Many considerations were influenced by the constraint of cost. One of the primary subjects of consideration was the sensor used to create the imaging system. Since S.T.A.R. aims to track humans upon an incident stage, it made theoretical sense for our spectral transmission to operate in the mid-wave infrared (MWIR) to long-wave infrared

(LWIR) as it is expected for the product to be applied in settings with lower visible light levels. However, after initial research, sensors such as uncooled microbolometers capable of transmitting such long wavelengths proved to significantly exceed our cost allocations ranging between several hundred to several thousands of dollars potentially spent. Thus, the scope of the project was changed to the visible light spectrum, and expanding to the MWIR - LWIR regime to account for environmental obstacles became a stretch goal.

Another limitation with cost results in choice of lenses. This project was highly constricted by using commercial off the shelf (COTS) components. Custom lenses are ideal for S.T.A.R. to account for aberrations that can be caused through the lens design, however they prove to be significantly more costly compared to its commercial counterparts. The imaging system's assembly of lenses was then hypothesized with this in mind and a decision was made to have a slightly worse image quality for cheaper COTS components.

Chapter 5 - Comparison of ChatGPT with other Similar Platforms

5.1 Introduction

Artificial intelligence (AI) has become a common tool in engineering education and professional work. It is used to help with tasks such as research, data organization, documentation, and design development. In UCF's Senior Design course, AI tools like ChatGPT are used by students to generate project ideas, summarize technical topics, and improve report structure. These tools are also helpful for formatting documents, creating tables and figures, and simplifying large amounts of information. The main advantage of using AI in this course is efficiency; it allows students to focus more on engineering and design while spending less time on repetitive writing tasks. However, students must still verify results and apply their own judgment, as AI systems can make mistakes or present information that lacks context. When used correctly, AI supports learning by helping students communicate complex ideas clearly and stay organized throughout their project.

5.2 Overview of AI Language Platforms

Table 24 – Comparative Analysis of AI Assistant Platforms

Platform	Strengths	Limitations	Best Use
ChatGPT	Excellent contextual reasoning and report writing	Occasional factual inaccuracies	Technical documentation, brainstorming

Claude	Handles long documents and maintains tone	Limited technical depth	Summarizing long reports
Gemini	Real-time web access and multimedia use	Inconsistent with advanced topics, making confident decisions	Fact-checking and research
Copilot	integration for coding platforms like MATLAB	Not conversational or analytical	Software and coding projects

5.3 Use in Illumination System Design

AI was a helpful tool during the design of the illumination system, mainly as a research and organizational assistant rather than a decision-maker. While it couldn't directly calculate optical parameters like focal length, F-number, or beam divergence given my specific components, it was useful for gathering background information and helping guide design choices. For example, AI was used to locate reliable data sheets for the light source, find specific optical materials, and summarize manufacturing standards related to LEDs and lenses. It was able to explain specific parameters of my design choices, like how a "cold mirror" was different from a dichroic mirror. Overall, it's good at information, but bad at optical calculations.

AI is absolutely useless when asking for help with Zemax Opticstudio. It cannot help with solving issues, how to complete tasks within the program, or explaining anything within the software. When I prompt ChatGPT to help me with something simple like where to find the fan plot, it tells me to click on menus and buttons that do not (nor ever have) existed. This is okay though, since Zemax is an old software that hasn't developed much over the past few decades and there is an abundance of information online about how to troubleshoot every possible problem that you can run into. In general, AI is pretty bad at helping with optics at all. It cannot plot any useful information (transmission spectrums, ray aberration, spot diagram…), its predictions on how light will behave and its ability to ray trace fall flat almost every time.

5.4 Use In Imaging System Design

Please refer to Appendix E [5a][5b] and [6a][6b] for the given prompt and outcomes of ChatGPT in research for the design of the imaging system.

ChatGPT [5a][5b] did an amazing job in dictating initial research on parameters set for S.T.A.R.. Aiding in explaining the concept of magnification, and verifying whether the target stage size could realistically be imaged, the LLM provided formulas which appear to be accurate when figuring out a target magnification. Indicating the need of an image and an object size, the formula the LLM provided served as the main subset of the magnification parameter we sought for. ChatGPT [6a][6b] also provided a great explanation to further the concept of demagnification. Indicating that a "short effective focal length" must be used to achieve demagnification on a 7.5m stage, the goal of the commercial off the shelf lenses was made. The overall aim of the lenses we selected was to reduce the effective focal length and properly image the stage. Along with this, the LLM mentions that a small magnification occurs when the object to be imaged is significantly further than the image distance from the lens, which is an accurate explanation to match our design as we are capturing a stage 7.5m away.

5.5 Use In Hardware Design

During early design phases, LLM was used to generate and refine descriptions for hardware subsystems such as the gimbal, optical housing, and motor control interfaces. It assisted in translating complex design details into formal technical language appropriate for an engineering report. Additionally, it helped standardize formatting, improve clarity, and ensure coherence between sections such as system objectives, specifications, and comparative component analyses. This allowed us to focus more on implementation and testing, while maintaining professional-quality documentation.

From a technical perspective, ChatGPT has been instrumental in assisting with system analysis and integration design. By leveraging the model's ability to synthesize large amounts of technical information quickly, the design process became more efficient and better informed. Furthermore, ChatGPT provided valuable insight into design standards, hardware interfacing protocols, and control methodologies, reducing the time typically required for manual literature review.

For some examples, please refer to Appendix E [7a][7b], [8a][8b], and [9a][9b] for the given prompt and outcomes of ChatGPT in research for our hardware design.

5.6 Use In Software Design

LLM has been used as a productivity aid across the S.T.A.R. software stack—mobile app, Raspberry Pi backend, and ESP32 interface—without replacing engineering judgment. It helped us quickly draft architecture options (Flutter or React Native UI plus FastAPI on the Pi plus UART to ESP32), compare transports (REST vs WebSocket vs MQTT) against our latency goals, and turn those choices into consistent, report-ready prose that aligns with Chapter 3.

On the implementation side, we used the LLM to scaffold boilerplate code and configs (for example, FastAPI project layout, WebSocket endpoints, simple health and metrics routes, and UART read and write loops) and to generate lightweight test stubs (mock frames, timing logs, and packet parsers) that we then completed and validated. For the mobile app, it accelerated early UI wireframes and happy-path flows (start, stop, manual override, target select), which we refined during integration.

For computer vision, we used the LLM to outline OpenCV pipeline options (capture to preprocess to detect or track to pan or tilt command) and to enumerate trade-offs (frame rate vs model size vs latency). Final algorithm choices, thresholds, and model selection were validated through our own experiments; the LLM served to organize options and surface issues such as bounding-box smoothing to avoid servo jitter.

It also assisted with documentation assets: turning interface definitions into clear API tables (command schema for Pi to and from ESP32 over UART), drafting sequence and state-transition descriptions for figures in Chapter 7, and standardizing terminology across chapters so the app, backend, and firmware sections stay consistent.

Limitations: we did not rely on the LLM for final algorithm design, performance claims, or optics and vision truths without testing. All code and specifications generated were treated as drafts and verified through experiments, profiling, and hardware tests.

Chapter 6 - Optical Design

6.1 Illumination System Optical Design

In the S.T.A.R. illumination system, the optical performance depends on several key parameters: the numerical aperture (NA), the focal ratio (f/#), the focal length (f), and the entrance pupil diameter (EPD). These parameters define how light from the LED COB module is collected, focused, and projected onto the stage. The following calculations outline how each value is determined and how they relate to the design of the optical system.

Numerical Aperture

The numerical aperture defines the range of angles over which the optical system can accept or emit light. It directly determines the cone of light that passes through the optical path. The equation for numerical aperture is:

$$NA = n \bullet sin(\theta/2) \tag{12}$$

where n is the refractive index of the medium (air = 1.0) and θ is the full cone angle of the beam. The reflector used in this system has a full cone angle of 36 degrees, so the half-angle is 18 degrees:

$$NA = n \bullet sin(18/2) = 0.31 \tag{13}$$

This value represents a relatively wide cone of light, suitable for a high-output stage fixture where coverage and brightness are prioritized over tight imaging resolution. The chosen NA ensures that the beam can effectively fill the aperture of the focusing lens while maintaining uniform illumination at the gate.

F-Number (f/#)

The f-number, also known as the focal ratio, describes how "fast" or "slow" an optical system is. It represents the relationship between the focal length and the entrance pupil diameter. For optical design, the f/# is linked to numerical aperture through:

$$NA = \frac{1}{2f/\#} \tag{14}$$

Rearranging for f/#:

$$f/\# = \frac{1}{2NA} \tag{15}$$

Since we have just calculated our numerical aperture value, we can plug it back into the equation to get

$$f/\# = \frac{1}{2(.31)} = 1.618 \tag{16}$$

An f/# of 1.6 indicates a relatively "fast" optical system, meaning it can collect and project a large amount of light. For illumination purposes, this is ideal because it increases light throughput, which is directly related to the brightness of the beam projected on the stage.

Focal Length

The focal length is the distance from the optical element to the point where light rays converge into focus. For the S.T.A.R. system, the focusing lens creates a 19-degree output beam, matching the design of commercial stage spotlights. The focal length can be estimated using the formula:

$$f = \frac{r}{tan(\theta/2)}$$

(17)

where *r* is the radius of the aperture (25 mm) and θ is the output beam angle (19°). Substituting the known values:

$$f = \frac{25mm}{tan(19/2)} = 167mm$$
(18)

If you refer to figure 4, this distance (167mm) is very close to the distance from the aperture to the lens. This is on purpose; it's just that the light is *coming from* a focal point, and not *going to* a focal point. Light can go either way through a lens, it just depends on how your system is set up. The way we have it, putting the lens exactly 167mm away from the aperture would give us a perfectly collimated output. Since we don't want that, and instead want an output cone of 19°, we move the lens 10mm further away. The 10mm was determined via Zemax simulation, since there is no single equation (or set of equations) that could determine that exact distance.

Entrance Pupil Diameter (EPD)

The entrance pupil diameter defines the physical size of the optical aperture required to achieve a given focal ratio and focal length. It is important to us because it determines our lens diameter. Usually, the lens diameter is not equal to the EPD, but the simplicity of this particular subsystem allows us to make this decision. It can be determined from the equation:

$$EPD = \frac{f}{f/\#} \quad (19)$$

Substituting the calculated values:

$$EPD = \frac{167mm}{1.618} = 103mm \quad (20)$$

This means the lens assembly must have a minimum clear aperture of 103 mm (in our case, the diameter is equal to the clear aperture). This value also provides the mechanical constraint for the lens mount and determines the outer diameter of the optical tube. Choosing a lens of this size ensures the beam is not vignetted or cut off at the edges, preserving brightness uniformity.

Illumination Source

The ETC Source 4WRD is used as the LED illumination module for this project. Since it replaces a traditional tungsten Source Four lamp, understanding its electrical behavior helps ensure safe operation and proper power planning. The 4WRD unit operates from a standard AC power source and is designed to run on a switched, non-dimmable circuit.

The unit consumes roughly 150 watts at full output. For a 120-volt supply, this results in a steady current of approximately 1.25 amps, using the basic electrical equation:

$$I = \frac{P}{V} \quad (21)$$

Substituting values:

$$I = \frac{150W}{120V} = .63A \quad (22)$$

This is a fairly light load compared to traditional stage lighting fixtures, making the Source 4WRD more power-efficient and easier to power from standard outlets.

For heat calculations, LED fixtures still generate heat even though they are more efficient than tungsten lamps. The Source 4WRD produces roughly 450–500 BTU per hour [29]. This helps estimate ventilation needs and ensures that adjacent components, especially 3D-printed parts or sensitive electronics, remain below their temperature limits. Although this heat level is lower than an incandescent Source Four, it still needs active or passive airflow around the light engine to avoid heat buildup.

Table 25: Power and Thermal Specifications for the ETC Source 4WRD Light Engine

Parameter	Value
Input Voltage	100–240 VAC
Power	~150 W
Current @ 120V	~1.25 A
Inrush Current	~32 A
Heat Output	~450–500 BTU/hr
Circuit Type	Switched / Non-Dim

6.2 Imaging System Optical Design

Ansys Zemax Optic Studio Simulation

With the selected lenses to build the imaging system, there are several variables to account for to ensure that the target parameters are met. The main criteria to be met is the magnification of the overall system, aiming to hit a 1250x demagnification for a 7.5 meters wide stage placed 7.5 meters away from the optical system. Much of the target values we are seeking for such as effective focal length and f/# can all be simulated using Ansys' Zemax Optic Studio, a program for lens design. Referring back to the lenses and their parameters in Table 1.0, there are a few variables that can be controlled in order to optimize the output image produced by the system. Since we are using commercially off the shelf lenses as opposed to custom lenses, the only degree of freedom to vary is the distances between each lens, and the distance from the sensor. The schematic and simulation design and results are shown below in Figure 17.1, Figure 17.2, and Figure 18.

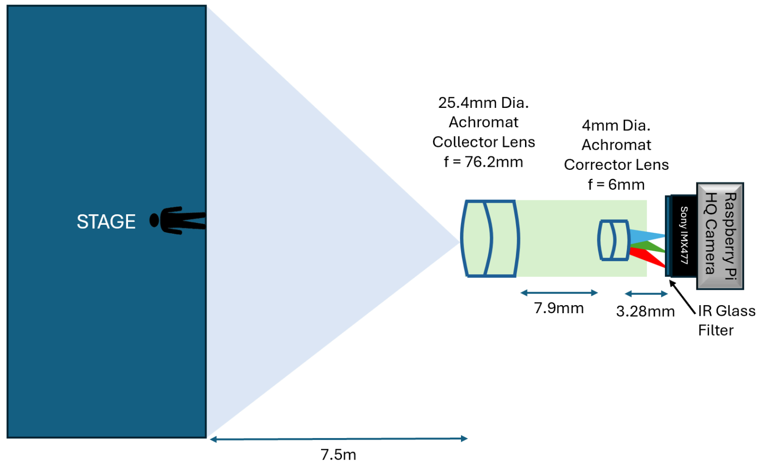


Figure 17.1 Optical Schematic of the Imaging System

Please note: The detailed simulation of the lens system is shown in Figure 17.2.

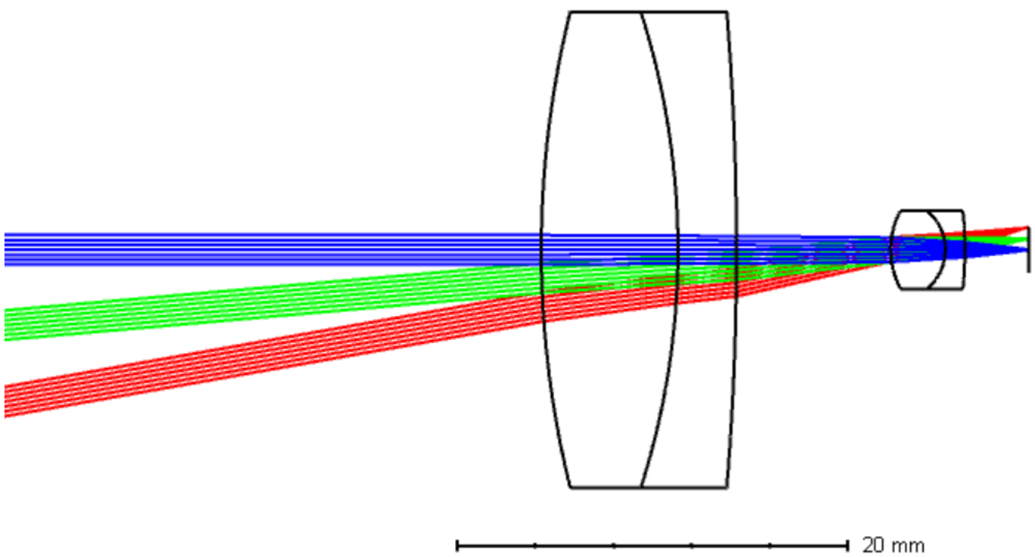


Figure 17.2 Zemax Simulation of Selected COTS Lenses

Figure 18. Zemax Simulated Extended Scene Analysis

Please note that the aperture stop is placed after the first achromatic doublet, with a diameter of 1.5mm, which is what causes the rays in the figure to not reach the full diameter of the first lens.

Through this simulation, and optimizing the system, it was found that the distance between the last surface of the system and the sensor that produces the smallest RMS spot size was at a distance of 3.281mm. It is pivotal to acknowledge that this distance does not equate to the effective focal length of the system, but rather the back focal length. For the effective focal length, that can be calculated with the knowledge of both image and object planes. For a multi-lens system, the image of the first lens becomes the object of the second lens, and the same concept applies for each lens. Using Zemax, this focal length is already calculated, with an optimized distance between the collector lens and the imaging lens being 7.9mm

With these distances, and the focal lengths of each lens, the effective focal length of the lens system is 5.904 as calculated by Zemax.

Table 26: Reported Calculated Values of the Imaging System from Zemax

Parameter	Reported Value
Entrance Pupil Diameter (EPD)	1.31mm
Effective Focal Length (EFL)	6.63mm

With these values, various numerical parameters can be calculated in order to verify that the target parameters have been met.

Magnification Calculations

From the object height of 25 feet or 7620mm, and a sensor size of 6.29mm horizontally, the theoretical target magnification can be calculated using

$$M = \frac{h_i}{h_o} = \frac{6.29}{7620} = \frac{1}{1250} \tag{23}$$

This result means a 1250x magnification system is needed to properly image the width of the stage. With the results of the Zemax report, we can calculate the true magnification the imaging system achieves, found by dividing the effective focal length over the distance of the stage to the system.

$$M = \frac{h_i}{h_o} = \frac{5.9}{7620} = \frac{1}{1290} \tag{24}$$

Although not precisely a 1250x demagnification, the system achieves a 1290x demagnification, which is about 3% higher with a slightly smaller image than the theoretical image, suitable for our design.

F/#

The working f/# of the system can also be determined from the Zemax report, using both the given effective focal length and the entrance pupil diameter. The calculated f/# can be found by dividing the effective focal length over the entrance pupil diameter. As aforementioned, the entrance pupil diameter was previously defined in the system from the aperture stop placed between the corrector and the imaging lens, pre-defined as 1.5mm. Thus, the working f/# of the system is:

$$f/\# = \frac{EFL}{EPD} = \frac{5.9}{1.5} = f/3.9 \tag{25}$$

The calculated value of f/3.9 is a perfectly reasonable speed that is acceptable for a compact imaging lens assembly, providing expected depth of field for an imaging system.

Chapter 7 - Hardware Design

7.1 Single Board Computer Design

The CM5 has 28 general purpose I/O (GPIO) pins which correspond to the GPIO pins on the 40way header. Since the CM5 can support up to five UART connections, we will be using one of those connections to transmit data to the ESP32.

The system on module (CM5) and microcontroller (ESP32) are what make up our single board computer. The Compute Module 5 handles all of the computer-vision tasks then transmits that information to the ESP32 where it handles all of the motor control for gimbal movement and other motor functions. The ESP32 will also be tasked with WiFi/Bluetooth transmission for the use of a mobile application.

7.1.1 CM5IO Carrier Board

A system on module is a compact module containing the CPU, RAM, and storage. However, It needs to be connected to a "carrier board" to function properly. Raspberry Pi does offer a readily available carrier board. We used this for testing and our final product.

The CM5IO carrier board exposes the Raspberry Pi Compute Module 5's GPIO via a standard 40-way header, giving access to 28 general-purpose I/O (GPIO) pins. Among its capabilities, the CM5 supports up to five independent UART interfaces, each mappable to specific GPIO pins through the device tree and config.txt configuration files.

For this project, UART2 was selected as the dedicated serial link between the Raspberry Pi CM5 and the ESP32-S3 microcontroller. The CM5 transmits data to the ESP32-S3 using GPIO4 (UART2_TXD) and receives data from the ESP32-S3 via GPIO5 (UART2_RXD). On the ESP32-S3 side, UART0 is used — it receives data on GPIO44 (RXD_0) and transmits back on GPIO43 (TXD_0). This cross-wired TX→RX arrangement is the standard UART crossover topology.

On the Pi side, UART2 is enabled and assigned to GPIO4/GPIO5 via the device tree overlay system. In /boot/config.txt (or /boot/firmware/config.txt on newer Raspberry Pi OS releases), the overlay, "dtoverlay=uart2", is added.

This maps the UART2 peripheral to GPIO4 (TXD) and GPIO5 (RXD) and exposes the interface as /dev/ttyAMA2 (or /dev/serial2). The serial port can then be opened in Python using a library such as pyserial.

This UART link is used to send real-time positional commands from the CM5 (which handles computer-vision processing) to the ESP32-S3, which in turn drives the 2-axis servo gimbal for spotlight orientation. The low-latency, deterministic nature of UART makes it well-suited for this type of motion-control communication.

7.1.2 ESP32

The ESP32 microcontroller serves as the primary motor control unit within the S.T.A.R. system, responsible for driving the 2-axis servo gimbal that orients the spotlight and WiFi/Bluetooth transmission for the use of a mobile application. Real-time positional commands are received from the Raspberry Pi Compute Module through a UART serial interface, ensuring low-latency and deterministic response for spotlight adjustments. Figure 23 shows the entire MCU schematic sheet.

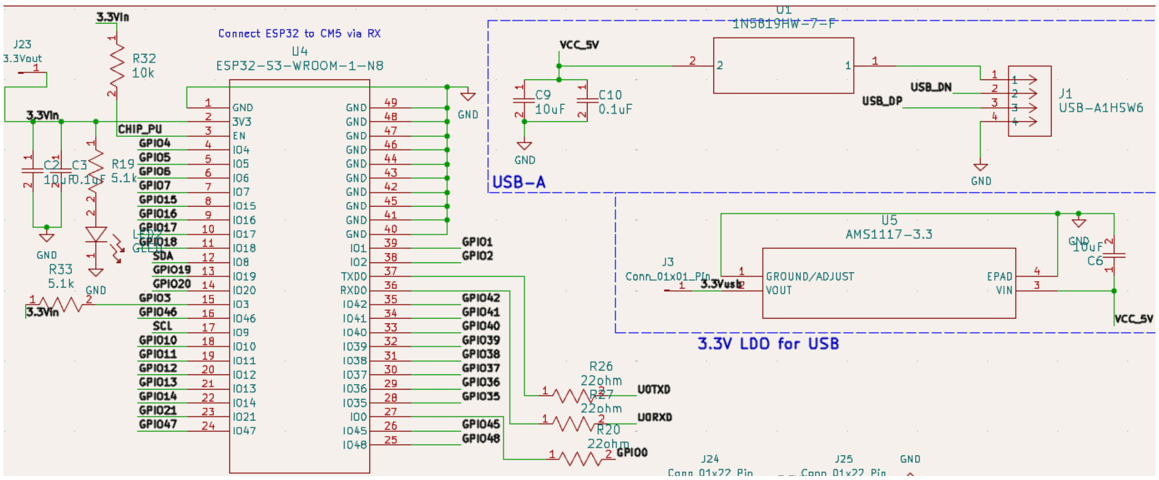


Figure 23.1: ESP32 Schematic

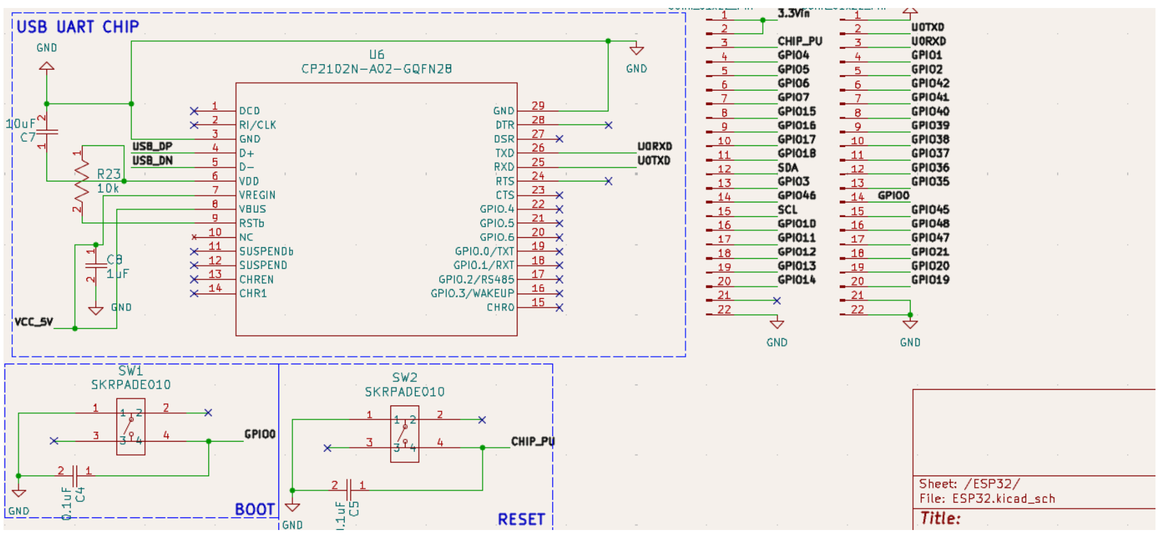


Figure 23.2: ESP32 Schematic

The ESP32 and its components will be on its own custom PCB. 3.3V Power will be supplied by the 600W PSU through a buck converter discussed later in this chapter. While the ESP32 will receive real-time commands from the Raspberry Pi, a USB-A USB port has been added for programming redundancy and debugging. While USB-C is the industry standard nowadays, Dr. Weeks recommends USB-A because there were some cases where students' USB-C receptacles fell off.

7.2 Motor Control

The Adafruit PCA9685 PWM driver is utilized as our dedicated servo control interface, providing precise and hardware-timed pulse generation for the servo motors. Operating via an I2C communication bus, the PCA9685 offloads the real-time PWM computation from the ESP32, enabling smoother and more stable servo motion without burdening the microcontroller’s internal timers. The use of the PCA9685 ensures deterministic timing, reduced signal noise, and scalable control, all critical for achieving the fluid and responsive spotlight tracking performance required by the system. For senior design 1, a standalone PCA9685 board was used. For senior design 2, our custom PCA9685 PWM controller was manufactured and assembled for our system. Figure 24 shows the entire PCA9685 schematic sheet.

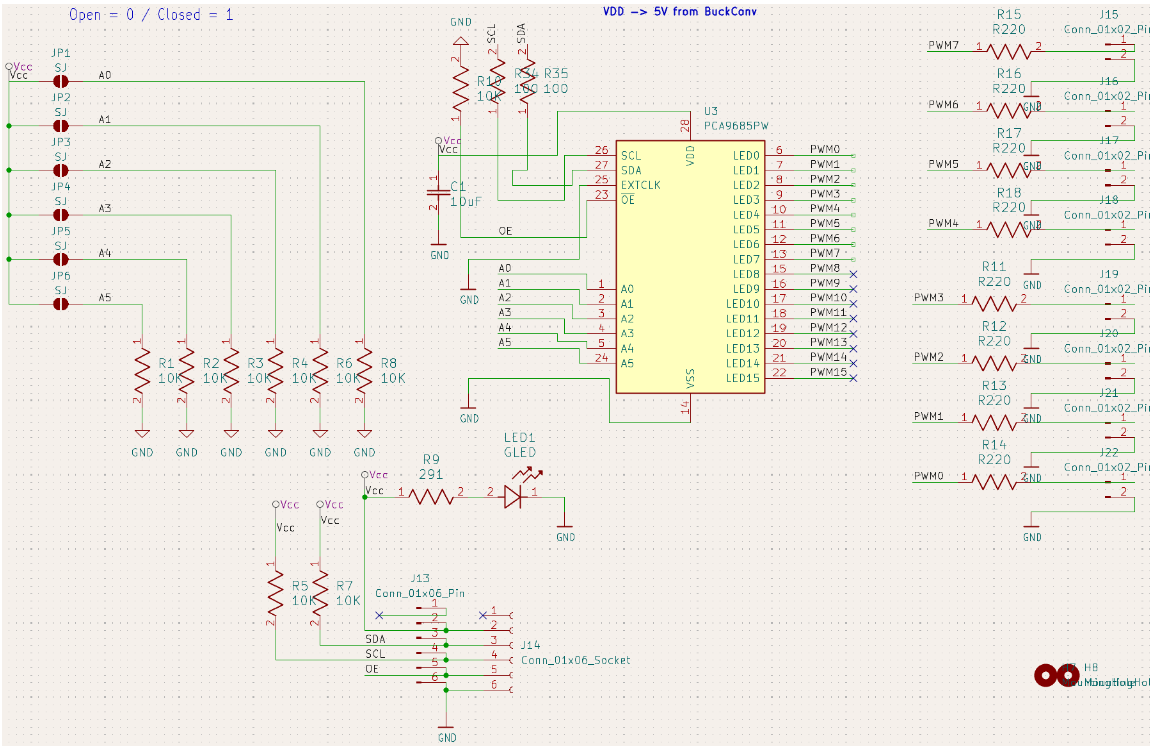


Figure 24: PCA9685 Schematic

The PWM controller needs a 5V supply. This will be provided by the 600W PSU through a buck converter. Since the PCA9685 has an output voltage limit of 6V, we are limited to the types of motors we can use. In order to use the high torque motors selected for our gimbal system, we must power the motors with an external source. This will be covered later in the chapter where we discuss the buck converters used for voltage regulation. A standard Adafruit PCA9685 PWM servo board has a jumper for external voltage supply to the motors, but even that is not rated for enough current.

7.3 Power Delivery / Electrical Power System

7.3.1 24V 25A 600W PSU

To provide ample power to our system, we needed a power supply unit with a high current output. Since there are multiple high torque servo motors being used, we needed enough amperage in case all of the motors reach their stall current at the same time. The

PSU plugs into a typical 110V/220V AC power outlet. Output voltage is adjustable from 0-24V DC. Appropriately rated wire will be attached to the screw terminals, then connected to the WAGO connector on the PCB. The PSU will be bought off amazon and not custom built in the interest of time and cost savings.

7.3.2 WAGO Connector

We are using the WAGO 2604-1102 PCB terminal block. It provides us with more than enough rated current and voltage for our application. It also comes with the added benefit of convenience with its push in cage clip lever system. Figure 25 shows the schematic for the WAGO connector.

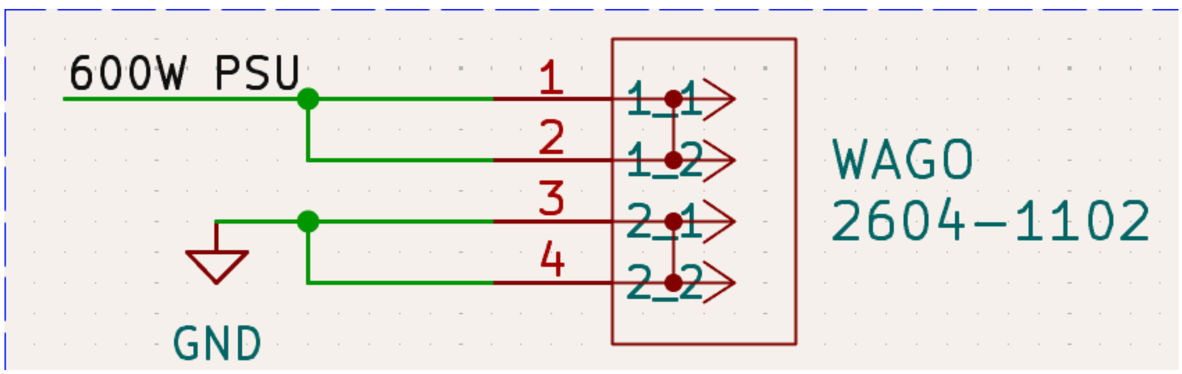


Figure 25: WAGO Schematic

7.3.3 Buck Converters

A buck converter is a type of switching voltage regulator designed to efficiently reduce a higher input voltage to a lower output voltage. Buck converters are commonly used to power low-voltage components such as microcontrollers, sensors, and communication modules from higher-voltage sources (e.g., 12V or 9V supplies) and provide relatively high levels of efficiency (85% - 95%). For our hardware design, we have used Texas Instrument components for all of our buck converters. It was extremely helpful being able to use the TI WEBENCH power designer to come up with the buck converter circuitry. TI ICs and circuitry were chosen based on a balance between cost, complexity, and efficiency.

All of our buck converters receive voltage from our 600W PSU and step it down to the desired voltage and current output. We will go into detail for each of them in the figures below.

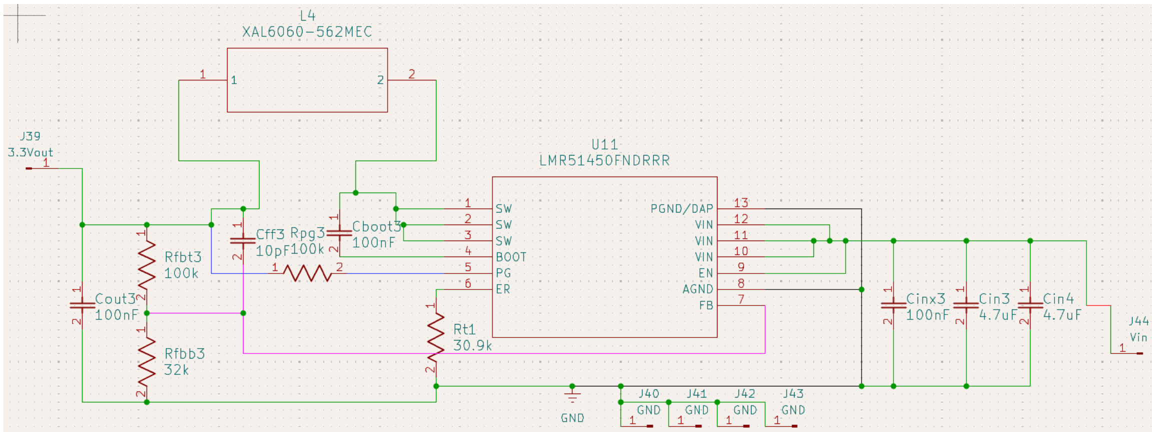


Figure 26: LMR51450 3.3V Buck Converter

In figure 26, we use a TI LMR51450 to take our PSU's 24V 25A down to 3.3Vout at 2A Iout. This buck converter will be used to regulate voltage to our ESP32.

In figure 27, we use a TI LM2576SX to step down the PSU to 5Vout at 1A Iout. This regulator is used to regulate power to the PCA9685. The LM2576 was chosen by the recommendation of Dr. Weeks. Previously, we had chosen a TPSxx model TI IC; however, the required layout for proper functionality of the IC is too complicated to stay within the PCB requirements stated by Dr. Weeks.

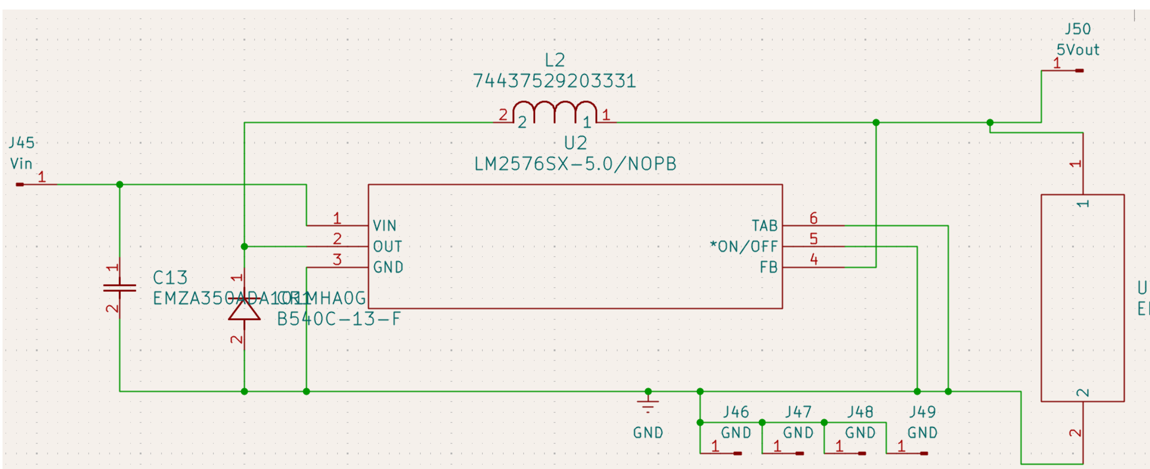


Figure 27: LM2576 Buck Converter

For our motors, we used similar TI components. Below in figure 28, we use another TI LMR51450 to take our PSU's 24V 25A down to 8.4Vout at 5A Iout. This buck converter will be used to regulate voltage to our 35kg motors. The addition of an AOD417 transistor and 2000uF capacitor at the output of the regulator is for reverse voltage protection and back-EMF protection. Originally we tried using a LM25141 IC. However, issues arose and we decided to design a new regulator around the LMR51450 for simplicity.

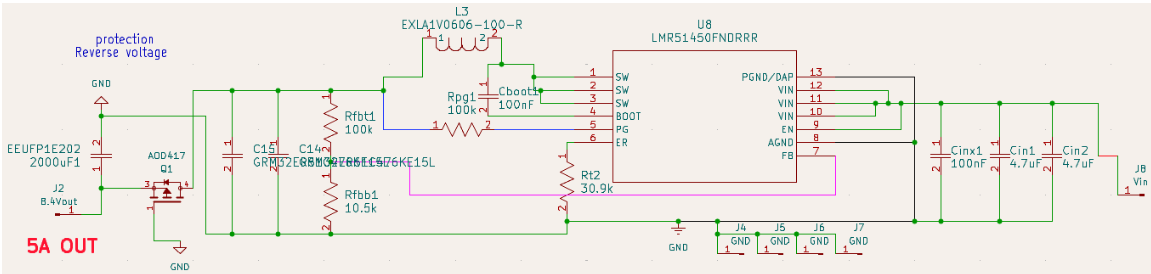


Figure 28: LMR51450 8.4V Buck Converter

7.3.4 CM5 Power

The Raspberry Pi Compute Module 5 (CM5) should be powered by a dedicated power source. It has been reported that the CM5's sensitive electronics can become unstable when the module is configured to both supply and receive power simultaneously. To prevent potential backfeeding and electrical noise, it is considered best practice to isolate the CM5's power supply from other system components. In this design, the CM5 is not powered by the 600W PSU used for other subsystems; instead, it is powered by a 27W USB Type-C power adapter provided by Raspberry Pi. The schematic shown below illustrates the power protection circuitry implemented for the CM5 to ensure stable and reliable operation.

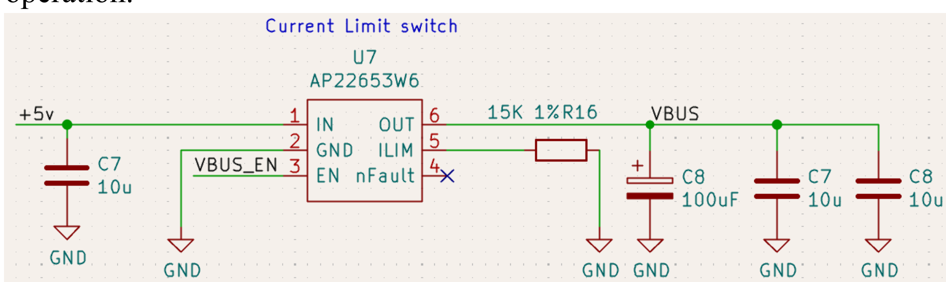


Figure 29: CM5 USB-C Current Limit Switch

7.4 Terminal Board

To reach the PCB requirements, we added a terminal board to hold additional peripherals such as status LEDs for each buck converter and power / kill switches. The status LEDs and rocker switches have been very helpful and convenient for us when testing the S.T.A.R. system. A terminal board has the advantage of being able to add additional peripherals in future revisions of the S.T.A.R. system. Voltage and current sensing components at the output of each buck converter that displays on the terminal board can potentially be added for QOL improvements. The schematic shown below is the basic relay outline for our terminal board.

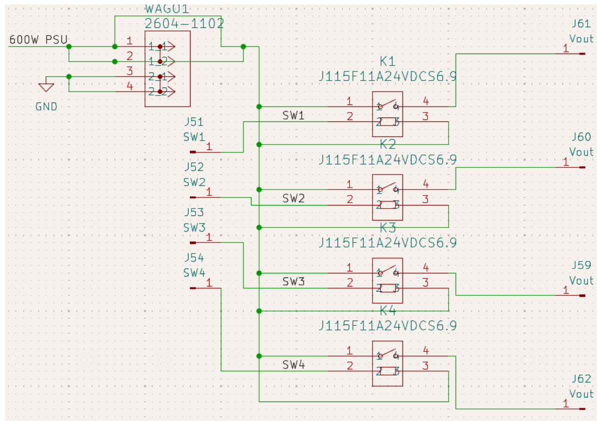


Figure 30a: Terminal Board Schematic

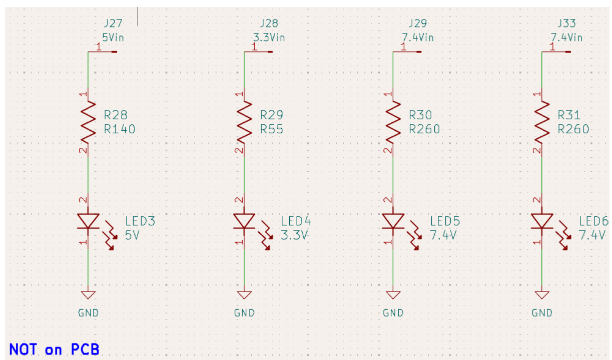


Figure 30b: Terminal Board Schematic

7.5 Gimbal Design

The proposed design for a two-axis pan–tilt gimbal is intended to provide precise, stable, and responsive motion control for a Source Four 19° stage light. Three 35kg-cm servo motors will be used for the gimbal. Two servos will be used for the tilt axis, one for each side of the stage light. The remaining servo will be used for the pan axis.

Mechanically, the gimbal consists of two main rotational stages: a pan axis located at the base and a tilt axis integrated within the upper stage. The pan assembly uses many steel ball bearings set in a grove between the upper and lower stages to relieve some of the frictional forces. The tilt axis uses multiple ball bearings in conjunction with the tilt gear shafts to handle the moment loads imposed by the weight of the light fixture. Both axes were 3D printed because we had access to a printer, it is easy to manufacture, and cheap to produce.

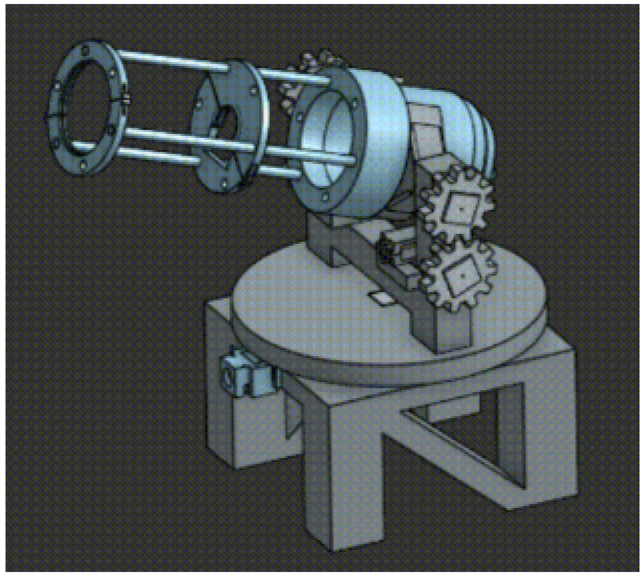


Figure 53: Gimbal Model

The gimbal control architecture consists of three major subsystems: motor control, vision processing, and power distribution. The motor controller interfaces directly with the servos, providing pulse width modulation signals to control the angular position of the motors. The controller receives angular setpoints from an ESP32 microcontroller, which executes low-latency position control and communicates with the Raspberry Pi Compute Module 5. The Compute Module handles the higher-level vision processing and target tracking, transmitting position commands to the ESP32 via a UART.

In testing and verification, the system was evaluated for motor load capability, smoothness of motion, and positional accuracy. Dynamic performance was measured in terms of latency, overshoot, and response to rapid direction changes, ensuring that the spotlight tracks targets seamlessly during live operation. Acoustic testing will confirm that motor and gearbox noise remain below acceptable thresholds for stage environments, while safety testing will verify reliable operation under power interruptions and limit conditions. The final design aims to combine robust mechanical construction, precise servo control, and responsive real-time tracking to create a professional-grade, automated spotlight system suitable for theater and live event applications.

7.6 System Housing Design

The electronic control system—including the Raspberry Pi Compute Module 5, ESP32, PCA9685 PWM driver, power regulators, and PSU is housed within a custom 3D-printed enclosure designed. The enclosure serves as both a protective shell and a thermal management solution for the sensitive electronics that power and control the S.T.A.R. system.

The enclosure is constructed from polyethylene terephthalate glycol (PETG) and polylactic acid (PLA). The housing is printed with 4 mm wall thickness and 10% infill density, providing sufficient rigidity while maintaining a manageable weight for gimbal-mounting or portable configuration.

The design incorporates magnets on the top and bottom panels for easy access to the PCB components. This has been very helpful while testing and trouble shooting the system. Cable management holes are integrated into the design to ensure secure routing of cables and wires. Many holes have been added to the panels of the housing to help with cooling, mounting points for components, and convenient passages for wires. A two-tier design was implemented to give us more room to place components. The housing is designed to allow for open arrangement of our PCBs. Each PCB is mounted using a screw and standoff. Holes on the front panel have been sized to allow a friction fit for our rocker switches and regulator status LEDs.

Thermal management is achieved through strategically placed ventilation slots and an auxiliary cooling fan mounted to the rear panel of the housing. Passive heat dissipation is enhanced by vent holes located along the enclosure's walls, promoting natural convection.

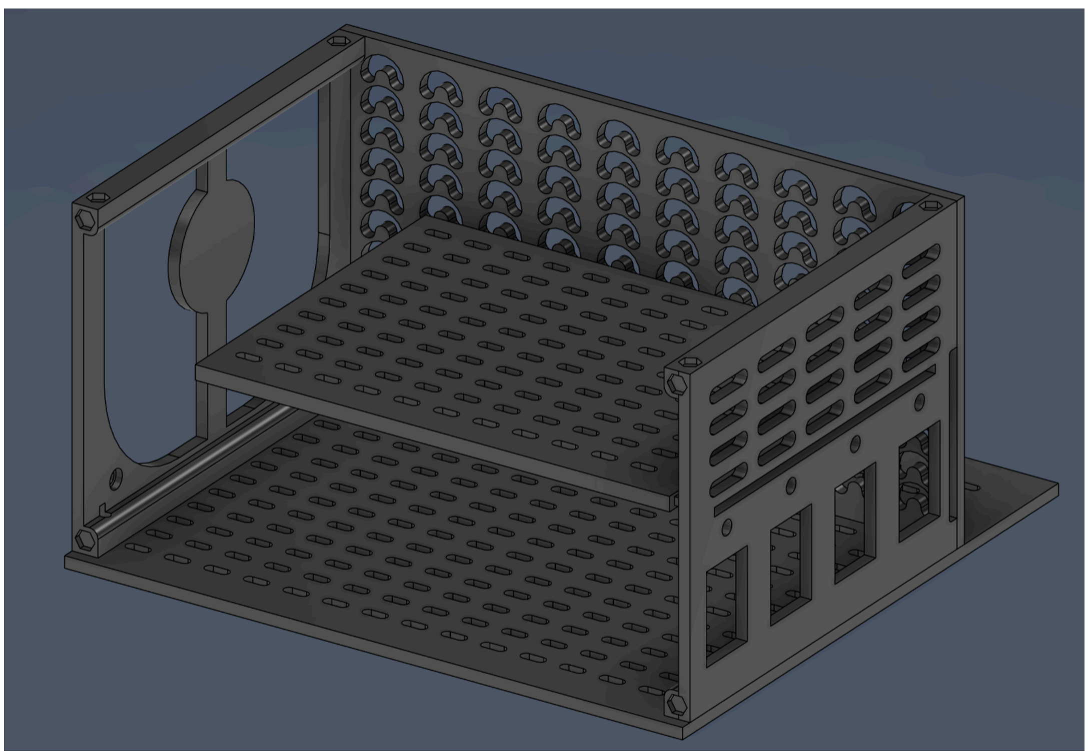


Figure 54: PCB Housing

Chapter 8 – Software Design

8.1 Software Architecture

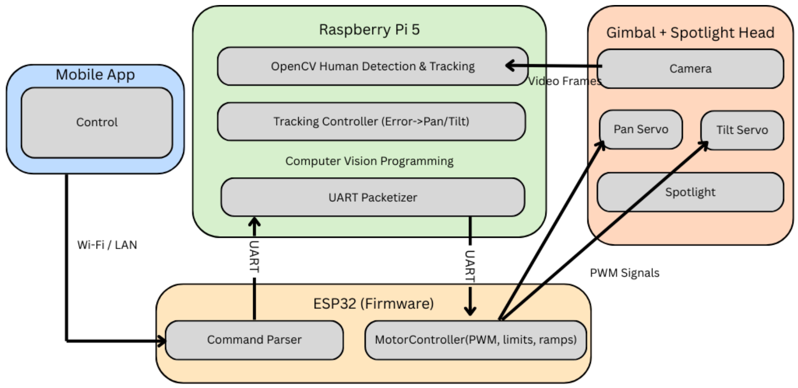


Figure 31: S.T.A.R. Software Architecture

The S.T.A.R. software is organized around three cooperating components that split responsibilities by computational load and real-time requirements: a Raspberry Pi Compute Module 5 (CM5) for vision processing, an ESP32-S3 for deterministic control and system coordination, and a mobile application that provides the user interface. In the revised architecture, the ESP32-S3 is the primary communication gateway and the system's mode authority. This means the mobile application connects directly to the ESP32 rather than using the Raspberry Pi as the user-facing endpoint. The app transmits user actions such as power, mode selection, manual movement commands, target selection requests, calibration routines, and configuration settings to the ESP32 over a wireless link. The ESP32 then forwards only the requests that require vision-side action to the Raspberry Pi and receives back tracking telemetry that describes the target's current position and tracking status. Using that telemetry, the ESP32 runs the motor-control loop that drives the pan/tilt servos and keeps the spotlight aligned with the performer.

The Raspberry Pi CM5 is dedicated to compute-intensive work and functions as the vision processing node. It receives live feed from the camera mounted on the gimbal and executes the human detection and tracking algorithm. The Pi continuously publishes compact tracking results to the ESP32 through a low-latency embedded link such as UART. This division improves robustness during live operation because the time-critical actuation path and user-command handling remain on the microcontroller, while the Linux-based Pi focuses on inference and tracking logic. The architecture is also designed with latency as a core constraint, targeting a total system response under roughly 500 ms. To support this target, the Pi-to-ESP32 communication is intentionally lightweight and frequent, and the ESP32 can enforce timeout logic so that stale telemetry triggers a safe behavior and a user-visible status update

8.2 Vision Algorithm and Target Tracking

The vision algorithm is the foundation of S.T.A.R.'s autonomous capability because it enables detection and continuous tracking of a performer inside the camera's field of view. Each frame captured by the gimbal-mounted camera is processed on the Raspberry Pi CM5 using a pretrained human detection approach (for example, an OpenCV-based detector or a similar model). When the model detects one or more people, the software produces a bounding box and confidence score for the person that is being currently tracked. From the selected detection, the vision software computes the target centroid and then determines the tracking error relative to the center of the image. This error is the key output used to steer the gimbal because it represents how far the performer is from the current aim point of the camera and spotlight.

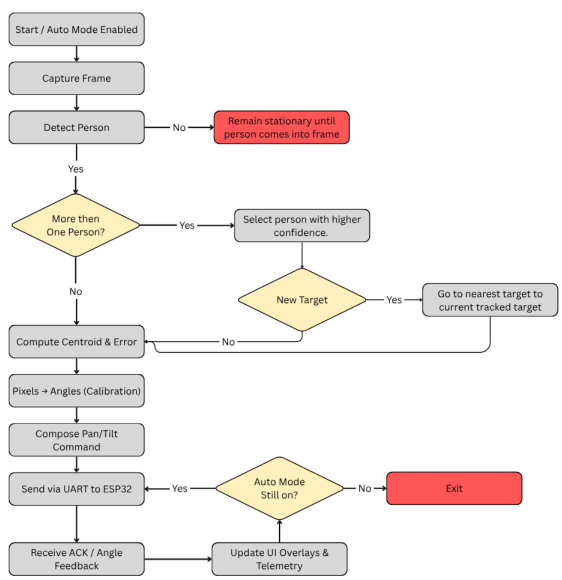


Figure 32: Continuous Tracking Loop Flowchart

In the design, the Pi does not issue direct motor commands. Instead, it outputs tracking telemetry that the ESP32 interprets and converts into actuation. Telemetry is designed to be compact and low overhead, typically describing horizontal and vertical offsets, confidence, and the current tracking state (for example, acquired, lost, or reacquiring). If multiple people are detected, the Pi can report multiple candidates or choose a best candidate using a simple heuristic such as largest bounding box, highest confidence, most central position, or which person came into view first while still allowing the user to override the target through the app. When the user selects a new target, that selection is sent to the ESP32 first and forwarded to the Pi so the vision pipeline can lock onto the requested performer and begin producing telemetry for that specific subject. This structure keeps the ESP32 as the single authoritative entry point for commands while still allowing vision-side target management.

To translate image-space error into physical motion, the system relies on geometric calibration that relates pixel displacement to angular displacement using the camera's field-of-view parameters. Rather than having the Pi compute final servo angles, the Pi can output normalized error values so the ESP32 can apply proportional or PID-style control, smoothing, and safety constraints in a consistent real-time loop. This is beneficial because the microcontroller can enforce motion limits and maintain stable motor behavior even if the Pi's frame rate fluctuates due to inference load. In challenging stage conditions such as uneven illumination or partial occlusion, the Pi can also report a "target lost" state and optionally apply prediction methods (such as Kalman-style estimation) to improve reacquisition behavior.

8.3 Microcontroller Firmware and Motor Control

The ESP32-S3 firmware is responsible for real-time gimbal motor control, system state management, and communication routing between the mobile app and the Raspberry Pi CM5. Its design is structured around continuously servicing two primary inputs: user commands arriving from the mobile app and tracking telemetry arriving from the Pi. The key architectural decision is that the ESP32 is the mode authority, meaning it decides which input source is allowed to drive the motors at any moment. This prevents conflicting actuation commands and ensures deterministic motor behavior during live operation.

In Auto mode, the ESP32 consumes the telemetry stream from the Pi and converts target error into pan/tilt updates. The microcontroller generates PWM signals for the servos and applies smoothing logic such as ramping, rate limiting, and PID-style damping to reduce overshoot and jitter. In addition to motion smoothing, the firmware enforces mechanical constraints so the gimbal cannot rotate beyond safe limits. The ESP32 also monitors telemetry freshness; if updates become stale beyond a timeout threshold, the firmware transitions into a safe behavior such as holding the last valid position, performing a slow scan, or returning to an idle-like posture while notifying the user through the app.

In Manual mode, the ESP32 applies user movement commands directly to the servos, supporting nudges or joystick-style velocity control. During this mode, the ESP32 either ignores incoming vision telemetry or instructs the Pi to pause tracking to reduce compute load and eliminate control contention. Because the microcontroller owns the mode state, the user does not experience the system "fighting" manual inputs. This is particularly important during performances where an operator may need to reposition the spotlight for creative intent or to recover from a difficult tracking moment.

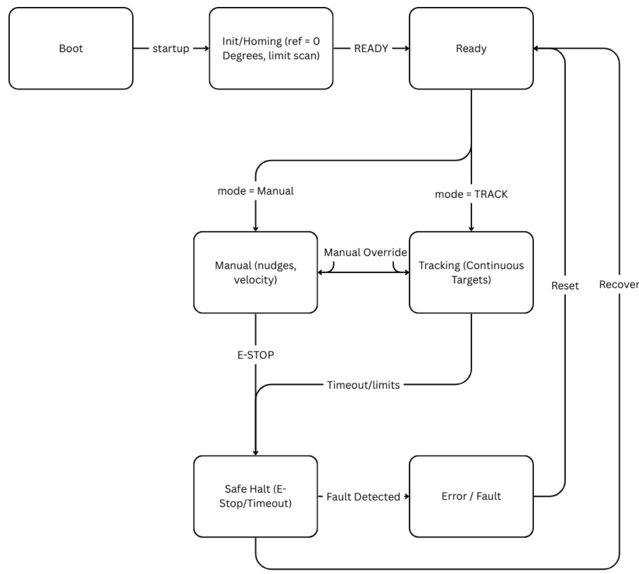


Figure 33: State Machine Flowchart

The embedded communication protocol between the Pi and ESP32 is implemented as a simple custom UART protocol that prioritizes reliability, low overhead, and ease of debugging. On the telemetry side, the Pi transmits compact messages describing offsets, confidence, and tracking state. On the control side, the ESP32 can issue commands such as starting or stopping tracking or selecting new target. This bidirectional exchange supports coordination without requiring the Pi to be exposed as the external networking endpoint.

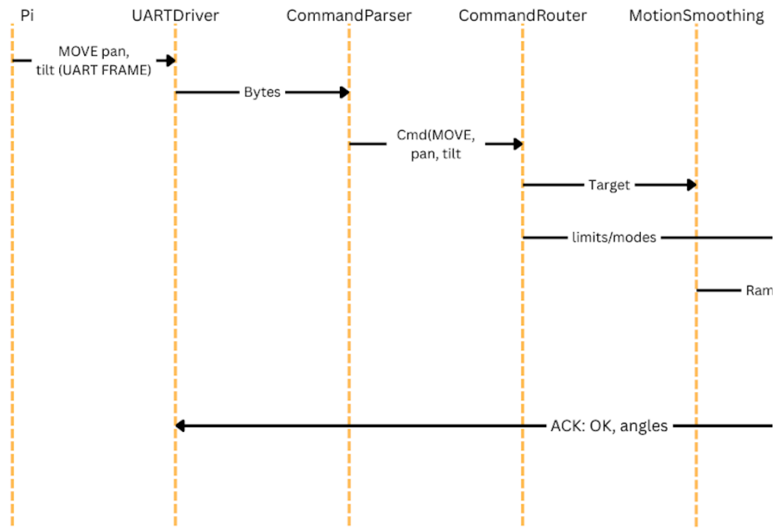


Figure 34.1: Move Sequence Flowchart

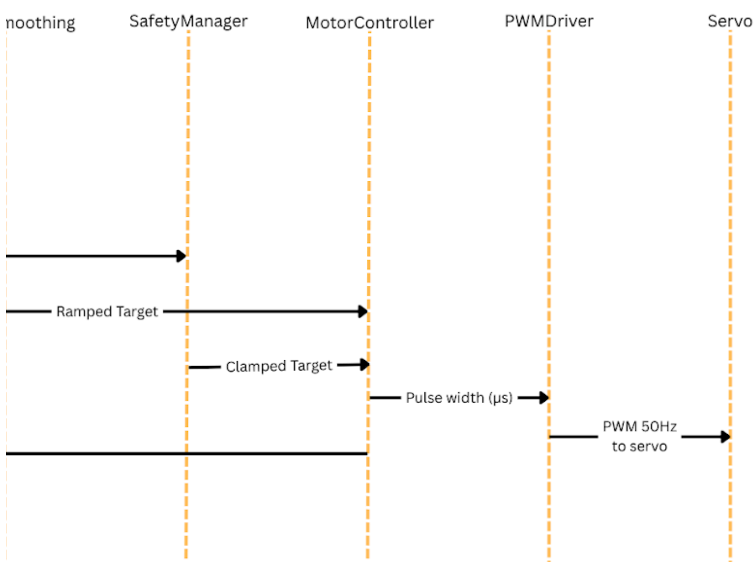


Figure 34.2: Move Sequence Flowchart

Internally, the ESP32 firmware is modular to improve maintainability and reduce failure coupling. The design includes distinct parsing modules for each input stream, such as an app command handler for decoding user actions and a telemetry handler that stores the latest vision tracking state. A motor controller module then consumes a unified "desired motion" signal derived from whichever input source is permitted by the active mode. During startup, the ESP32 initializes servo outputs to a safe home position, validates the embedded link to the Pi, and then reports readiness to the app. It can also provide periodic status updates such as current angles, active mode, tracking state, and error conditions.

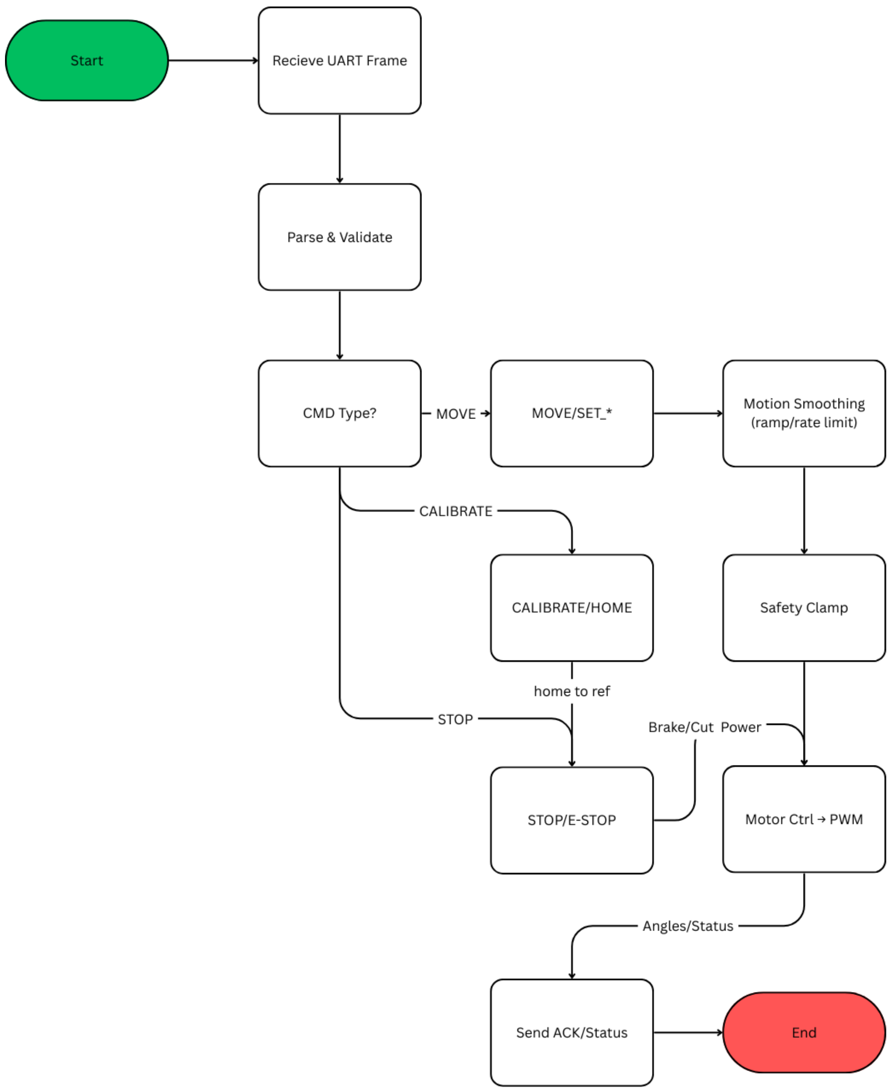


Figure 35: Activity – Handle Incoming Command Diagram

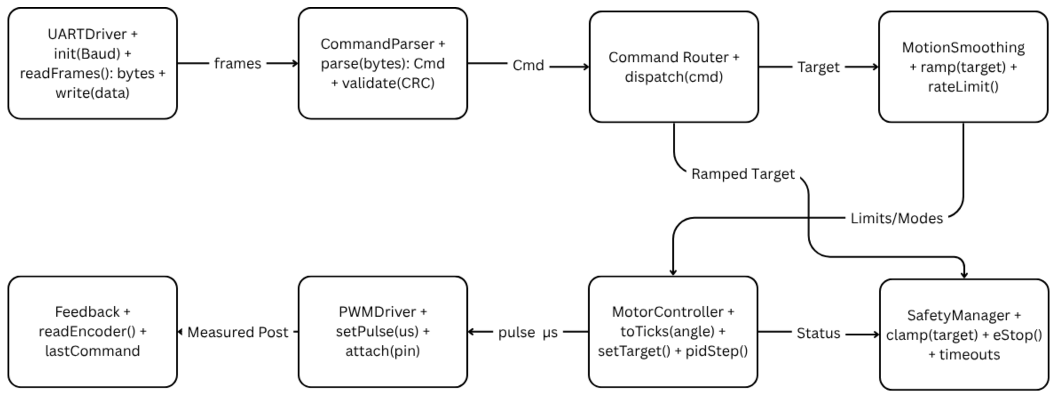


Figure 36: ESP Class Diagram

8.3 Mobile Application Interface and Wireless Communication

The mobile application is the user's primary interface for monitoring and controlling S.T.A.R. in real time, so its design emphasizes clarity, minimal clutter, and immediate responsiveness for live performance operation. In the revised architecture, the app connects directly to the ESP32-S3 over a wireless link such through Wi-Fi AP, making the microcontroller the single primary endpoint for commands and telemetry. This improves reliability by reducing dependence on Linux networking behavior during an event and ensures that command handling remains consistent regardless of the Pi's current compute load.

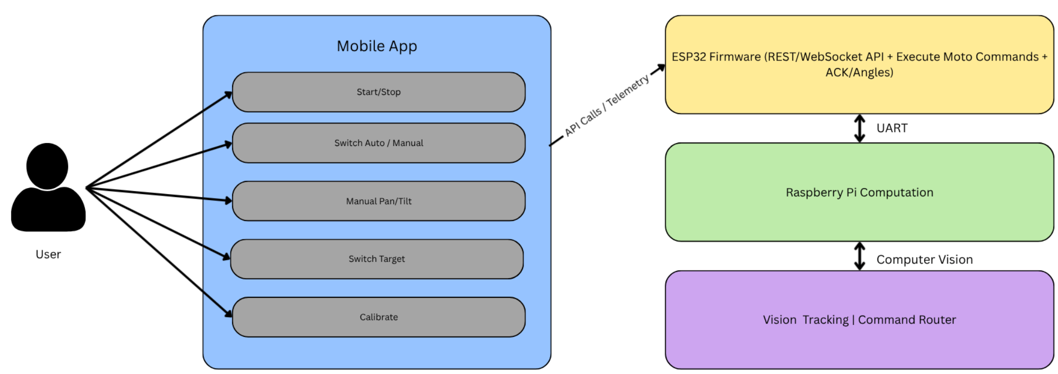


Figure 37: User Interaction Use Case Diagram

Because the ESP32 is now the control-plane endpoint, it coordinates video access rather than the Pi hosting all user-facing networking directly. Depending on implementation, the ESP32 may provide the stream endpoint information to the app, proxy lightweight metadata while the app fetches video directly from the Pi, or maintain a dedicated streaming path while the ESP32 continues to handle commands and telemetry. Regardless of how streaming is implemented, control messages always terminate at the ESP32 to preserve consistent behavior and maintain a single authoritative entry point for actions such as power changes, mode switches, manual control, and target selection.

The app exposes controls such as Start/Stop tracking, Mode switching between Auto and Manual, new selection, manual movement, and calibration routines. When the user taps Start Control, the app sends a start command to the ESP32-S3, which then begins applying motor control commands to the gimbal based on the movement data being continuously received from the Raspberry Pi over UART. When the user taps Stop Control, the app sends a stop command to the ESP32-S3, which immediately halts motor outputs and stops consuming tracking data from the Pi. In Auto mode, the app supports target selection workflows such as tap-to-track when multiple candidates exist; the selection is sent to the ESP32 first and forwarded to the Pi so the vision pipeline locks onto the desired performer. In Manual mode, the app functions as a direct remote controller; movement commands are sent to the ESP32 and applied immediately, while the app receives responsive feedback such as current angles, motor activity, and mode state.

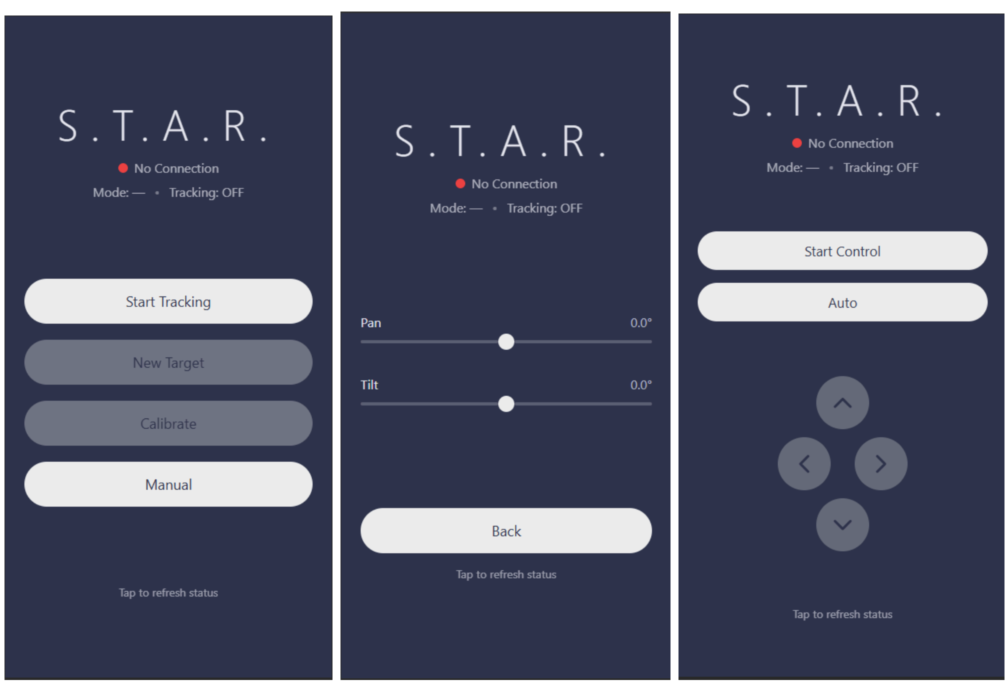


Figure 39: App Layout

8.4 System Modes and State Management

From a control perspective, S.T.A.R. operates in distinct modes that determine which inputs are active and how the gimbal control loop behaves. The primary states are Idle, Auto-Tracking, and Manual Override. In the revised architecture, the ESP32-S3 owns the mode state and enforces mode rules to prevent conflicts between user commands and autonomous tracking telemetry. This ownership is essential for predictable behavior, because it guarantees that only one source has motor authority at a time.

Upon power-up and initialization, the system enters Idle. In Idle, the spotlight is ready but not actively tracking; the ESP32 holds the servos at a safe default or last-known orientation, while the Raspberry Pi may remain idle or run a low-power detection standby depending on configuration. Auto-Tracking begins when the user enables Auto mode and selects a target. The ESP32 then instructs the Pi to begin tracking and publish telemetry at the desired update rate. As telemetry arrives, the ESP32 continuously converts error into smooth servo updates. If the target is lost, the Pi reports a target-lost state; the ESP32 can respond by holding position, performing a slow scan, or transitioning back to a no-target behavior while waiting for reacquisition or user intervention. Manual Override begins when the user toggles Manual mode, at which point the ESP32 immediately stops applying telemetry to the motors and accepts only user commands. Optionally, the ESP32 can request that the Pi pause tracking to reduce compute usage during manual operation. When switching back to Auto, the ESP32 resumes consuming telemetry; if the previous target is still visible, the Pi can reacquire it, otherwise the system remains in an idle-like no-target condition until the user selects a performer again.

State transitions are also protected by safeguards so behavior remains operator-trustworthy. The ESP32 can debounce rapid toggles, ignore commands that do not apply to the active mode, and enforce safe transitions so the gimbal does not whip abruptly when switching states. The guiding principle is that behavior changes only due to explicit user action or explicitly detected conditions such as target loss or telemetry timeout, which keeps performance behavior consistent and predictable.

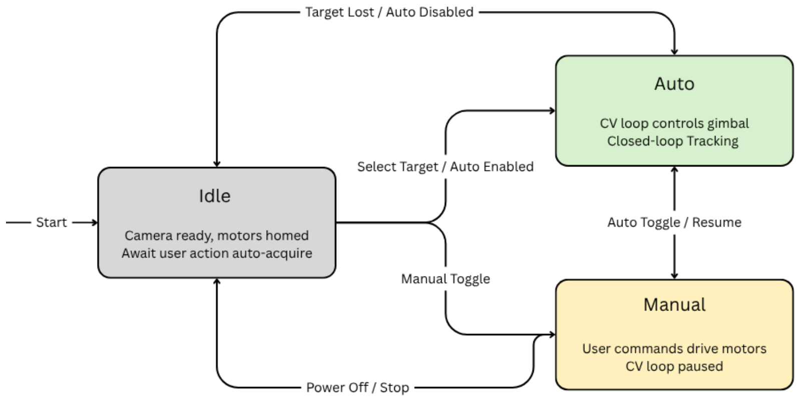


Figure 38: State Diagram of System Modes

8.5 Control Flowcharts and Sequence Diagrams

This section documents the system's behavior in common scenarios using flowcharts and sequences that emphasize the ESP32 as the central coordinator. In each workflow, the app communicates with the microcontroller first, and the microcontroller coordinates vision activity on the Raspberry Pi and actuation on the gimbal. This representation matches the revised architecture goals of deterministic real-time behavior and reduced dependence on Linux network timing during live events.

8.5.1 System Startup and Shutdown Sequence

The startup sequence begins with the operator powering on the Raspberry Pi and allowing the computer vision pipeline to fully initialize. Once the Pi is running, the ESP32-S3 is then powered on, at which point it initializes its servo control outputs to a safe home position and begins listening for data over the UART link from the Pi. With both devices running, the operator opens the mobile app and connects to the ESP32-S3 by joining its Wi-Fi Access Point (AP). Once connected, the app displays the system as ready, and the operator taps Start Control to begin tracking. This sends a start command to the ESP32-S3, which then begins applying motor control commands to the gimbal based on the movement data being continuously received from the Raspberry Pi over UART.

When the user taps Stop Control in the app, the app sends a stop command to the ESP32-S3, which immediately halts motor outputs to prevent unintended motion. The ESP32-S3 then stops consuming tracking data from the Pi and disables motor control, while the Pi stops publishing movement data. This ensures a safe and orderly shutdown of all actuation before the operator powers down the hardware.

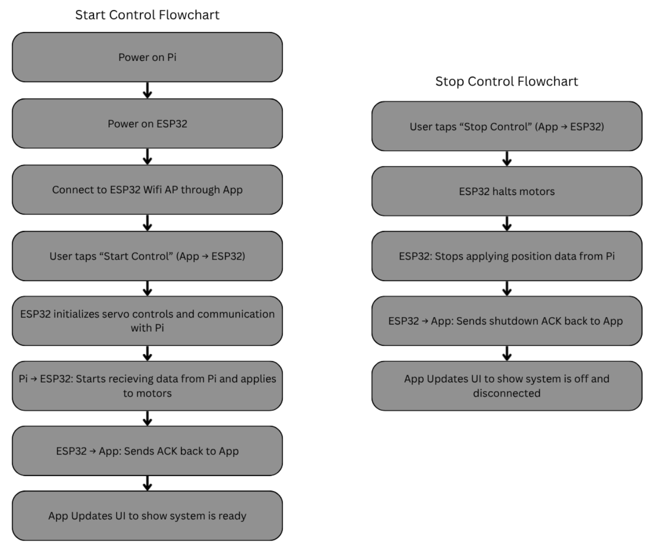


Figure 40: System Power-On and Power-Off Flowcharts

8.5.2 Target Selection and Tracking Update Sequence

Target selection is initiated in the app, but the ESP32 remains the first receiver of the operator's intent. The target selection sequence diagram shows the command path from User → App → ESP32, and then from ESP32 → Raspberry Pi. This clarifies two key software responsibilities: (1) the ESP32 validates and logs the selection as the authoritative state change, and (2) the Pi receives the selection as an input to the vision pipeline, not as the system controller. The acknowledgement returning from the ESP32 back to the app demonstrates that the UI can confirm the new target without needing a direct connection to the Pi.

Once tracking is active, the continuous tracking flowchart illustrates the closed-loop behavior that runs repeatedly during Auto mode. The Raspberry Pi detects the performer and produces telemetry (such as target offset/error), which is transmitted to the ESP32. The ESP32 converts this into servo updates and applies smoothing and motion constraints before commanding the motors, maintaining stable spotlight positioning even when vision updates fluctuate. This diagram helps demonstrate the software's real-time division of labor: the Pi produces perception data, while the ESP32 performs time-critical control decisions and motor actuation, ensuring consistent tracking behavior under variable Pi compute load.

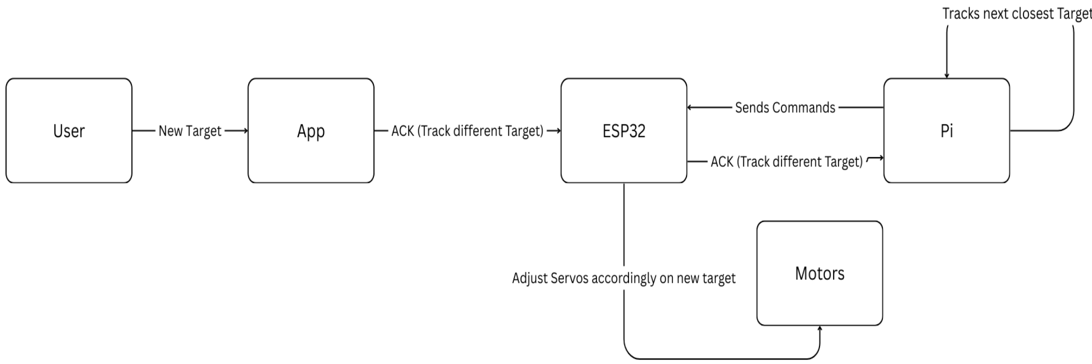


Figure 41: Target Selection Sequence

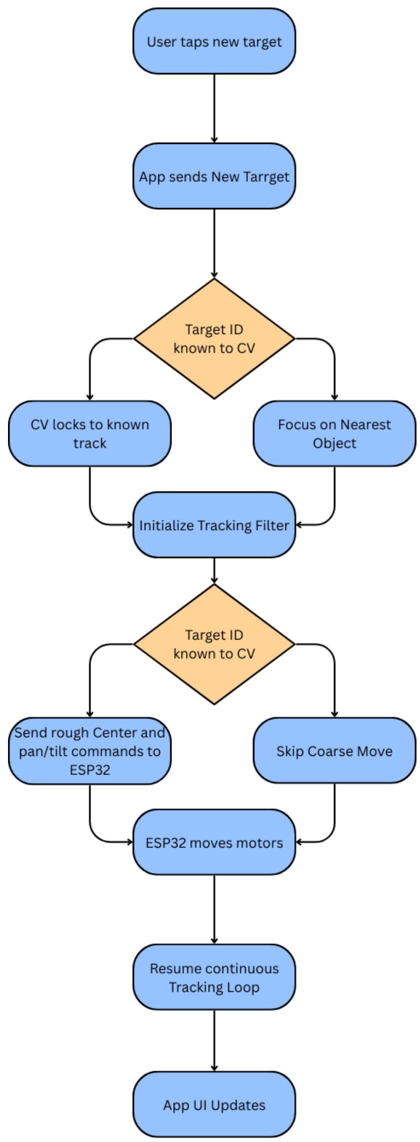


Figure 42: Continuous Tracking Loop Flowchart

8.5.3 Manual Override Control Sequence

Manual override begins when the user toggles Manual mode in the app, sending a mode-change request to the ESP32. The ESP32 immediately switches motor authority to user input and stops applying vision telemetry to servo outputs, ensuring the system does not resist the operator's actions. The user's movement inputs are transmitted directly to the ESP32, which moves the servos accordingly and returns updated angles and status telemetry to the app for responsive feedback. During this time, the Raspberry Pi may continue running the vision pipeline and streaming telemetry, but that telemetry no longer affects motion; optionally, the ESP32 can instruct the Pi to pause tracking to reduce unnecessary compute load. When the operator returns to Auto mode, the app sends a mode-change request to the ESP32, which requests the Pi to resume tracking and then smoothly transitions back to telemetry-driven motion without abrupt gimbal movement.

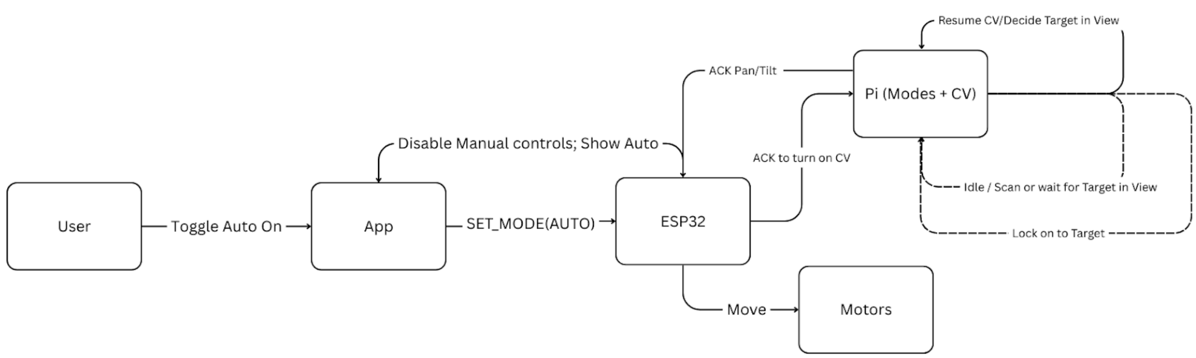


Figure 43: Manual Override Sequence

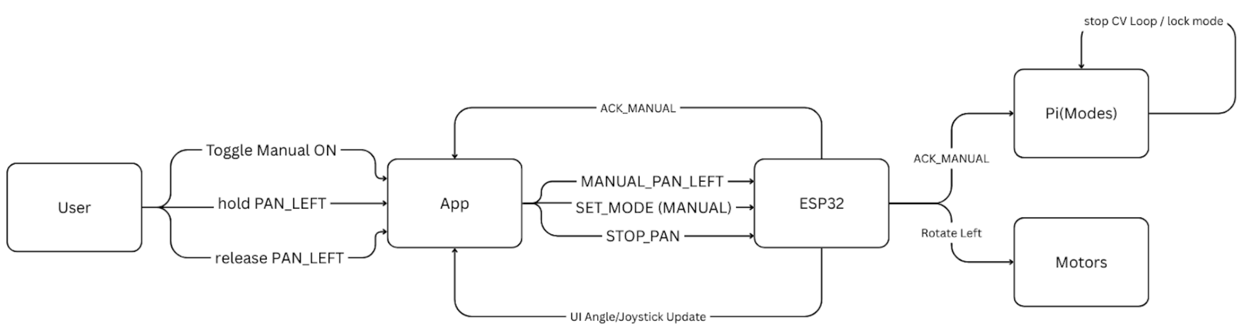


Figure 44: Mode Transition Sequence

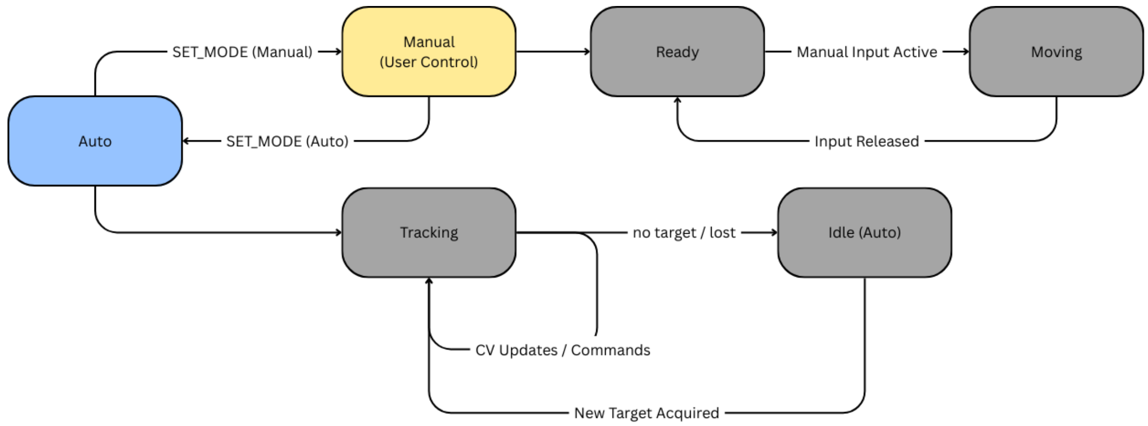


Figure 45: System Modes State Diagram

Chapter 9 - System Fabrication / Prototype Construction

9.1 Camera Housing

The camera housing for the S.T.A.R. system was designed using Onshape to provide a protective and structurally stable enclosure for the imaging subsystem. The housing serves multiple purposes: securely mounting the CMOS camera, maintaining optical alignment with the gimbal system, and shielding the camera from dust, incidental impact, and ambient stage interference such as glare or stray lighting. Special attention was given to the internal mounting geometry to ensure that the camera sensor remained fixed relative to the mechanical axis of the gimbal, minimizing misalignment errors during tracking. Additional features such as cable routing channels, mounting bosses, and removable access panels were incorporated to simplify assembly and maintenance.

The camera housing was designed specifically for additive manufacturing, which introduced several physical constraints into the CAD process. Print volume limitations of the available 3D printer required the housing dimensions to remain within the machine's build envelope, influencing the maximum allowable footprint and wall thickness. Screw hole diameters were oversized slightly relative to nominal screw dimensions to account for dimensional inaccuracies commonly introduced by fused deposition modeling (FDM), such as material shrinkage and layer swelling. Clearance tolerances between mating parts were also intentionally increased to compensate for fitment variability caused by printer resolution limitations and surface roughness. These adjustments were necessary to ensure that assembled parts could fit together properly without requiring excessive post-processing such as drilling, sanding, or reaming.

9.2 Illumination System

The illumination system housing was designed in Onshape to support the optical assembly, including the reflector, lamp source, and focusing lens. The primary objective of the mechanical design was to maintain precise optical alignment between all internal elements while providing sufficient structural integrity to support the thermal and mechanical demands of operation. The housing geometry was modeled to emulate the general optical architecture of an ellipsoidal reflector spotlight, allowing the light source to focus efficiently through the reflector and lens assembly. Mounting features were integrated for secure retention of the reflector and lens while preserving the optical axis throughout pan and tilt movement.

Because the illumination housing contains heat-generating components, additional design considerations were required to account for both thermal expansion and manufacturing limitations. Additive manufacturing constraints such as printer bed size restricted the maximum length of the optical tube, requiring modular segmentation of the housing for assembly after printing. Fastener holes for lens mounts and reflector retention were designed with oversized tolerances to account for the variability of printed dimensions and to ensure consistent hardware fitment. Wall thicknesses were increased in load-bearing regions to improve rigidity and reduce warping during print cooling. Furthermore, interlocking features and alignment tabs were added to improve assembly accuracy and compensate for imperfect print flatness, which is common in large-format printed parts.

9.3 Gimbal Fabrication

The gimbal assembly was designed using both Autodesk Fusion 360 and Onshape to develop a mechanically stable two-axis pan-tilt mechanism capable of supporting the spotlight and camera systems. The design process focused on achieving smooth rotational motion, minimizing backlash, and maintaining structural rigidity under dynamic loads. Servo mounting brackets, bearing interfaces, and cable pass-through channels were incorporated directly into the design to reduce assembly complexity and improve wire management during operation.

Fabricating the gimbal using additive manufacturing introduced significant fitment and tolerance challenges that had to be addressed during the CAD process. The dimensions of the gimbal components were constrained by the available printer’s build volume, requiring larger structural members to be split into multiple printable sections for post-print assembly. Screw hole dimensions were intentionally enlarged to compensate for printer tolerances and to ensure reliable insertion of machine screws without cracking the printed material. Rotating joints and bearing interfaces required additional clearance allowances to prevent friction binding caused by imperfect print tolerances and layer artifacts. These design choices improved manufacturability while preserving the mechanical precision required for accurate spotlight tracking.

9.4 PCB Assembly

KiCad was selected as the primary PCB design software for the S.T.A.R. project due to its robust feature set, open-source accessibility, and strong support for professional-grade schematic capture and board layout. As a widely adopted electronic design automation (EDA) tool, KiCad provides a complete workflow that includes schematic design, PCB routing, footprint management, and integrated design rule checking. Its compatibility with industry-standard file formats and seamless export of Gerber and drill files makes it well-suited for direct integration with manufacturers such as JLCPCB. Additionally, KiCad's zero licensing cost allows the project to scale in complexity without imposing financial constraints, which is especially advantageous in an academic design environment.

From a technical standpoint, KiCad enables fine-grained control over layout parameters that are critical to the S.T.A.R. system, including trace width tuning for high-current power paths, controlled routing of mixed-signal interfaces, and careful component placement to minimize noise and thermal stress. The use of custom libraries and footprint editors allowed the team to accommodate non-standard components such as high-current connectors, buck converters, and servo interfaces. While some software stability challenges were encountered during routing, KiCad's strong community support and frequent updates provide confidence that these issues can be resolved. Overall, KiCad offers the flexibility, transparency, and manufacturing readiness required to support an iterative PCB design process for a complex electromechanical system like S.T.A.R. Below are the PCB layouts for each PCB.

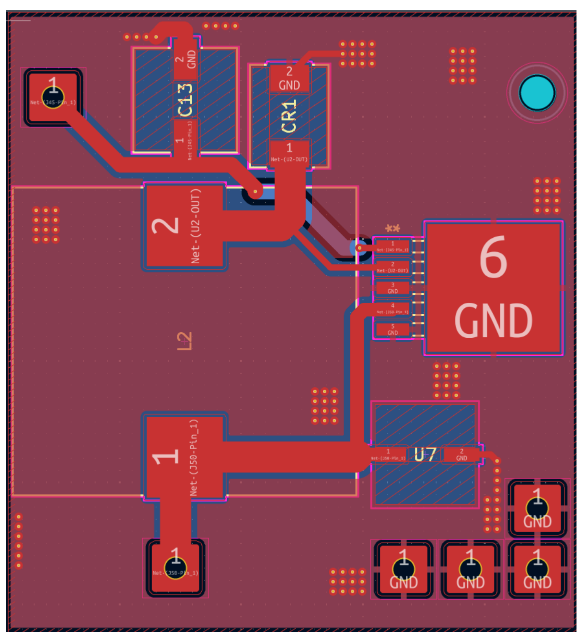


Figure 47: 5V Buck PCB

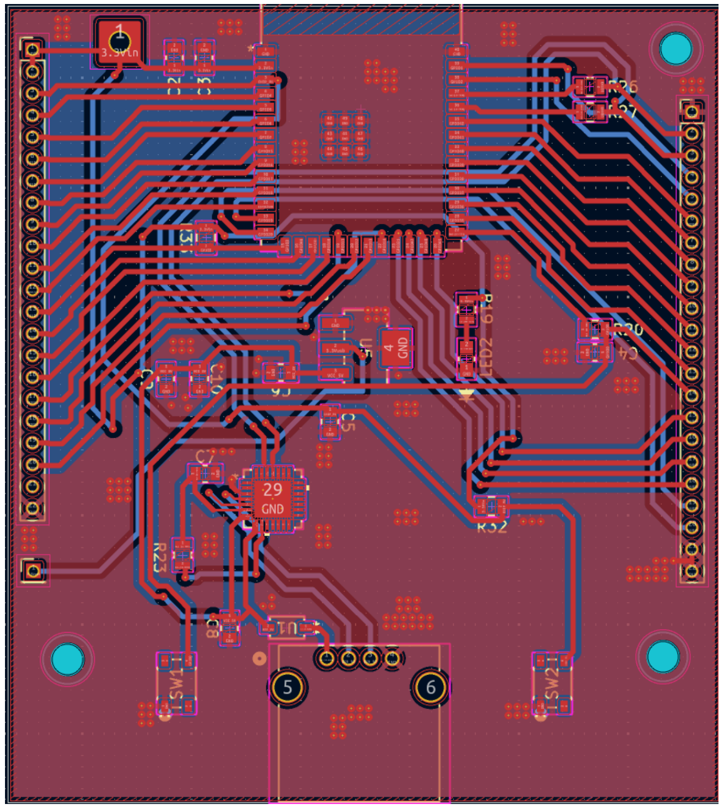


Figure 46: Custom ESP32 PCB

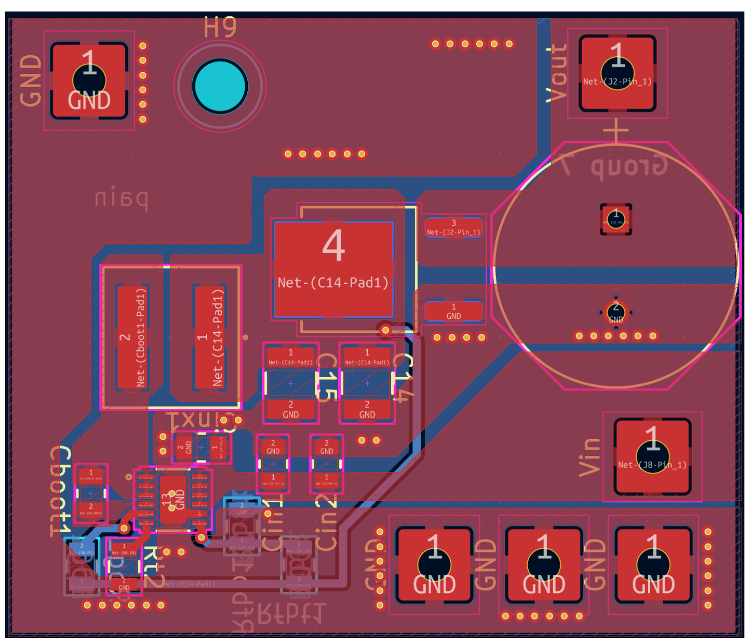


Figure 48: 8.4V Buck PCB

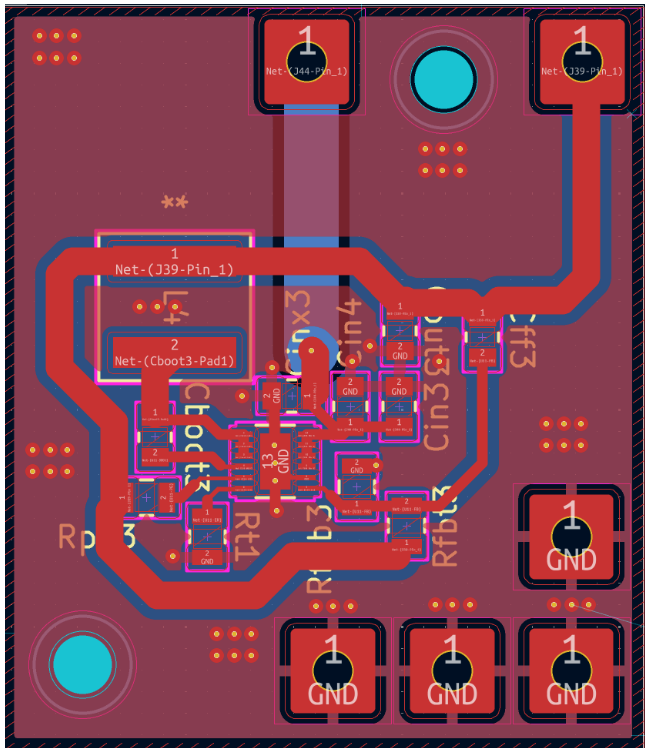


Figure 49: 3.3V Buck PCB

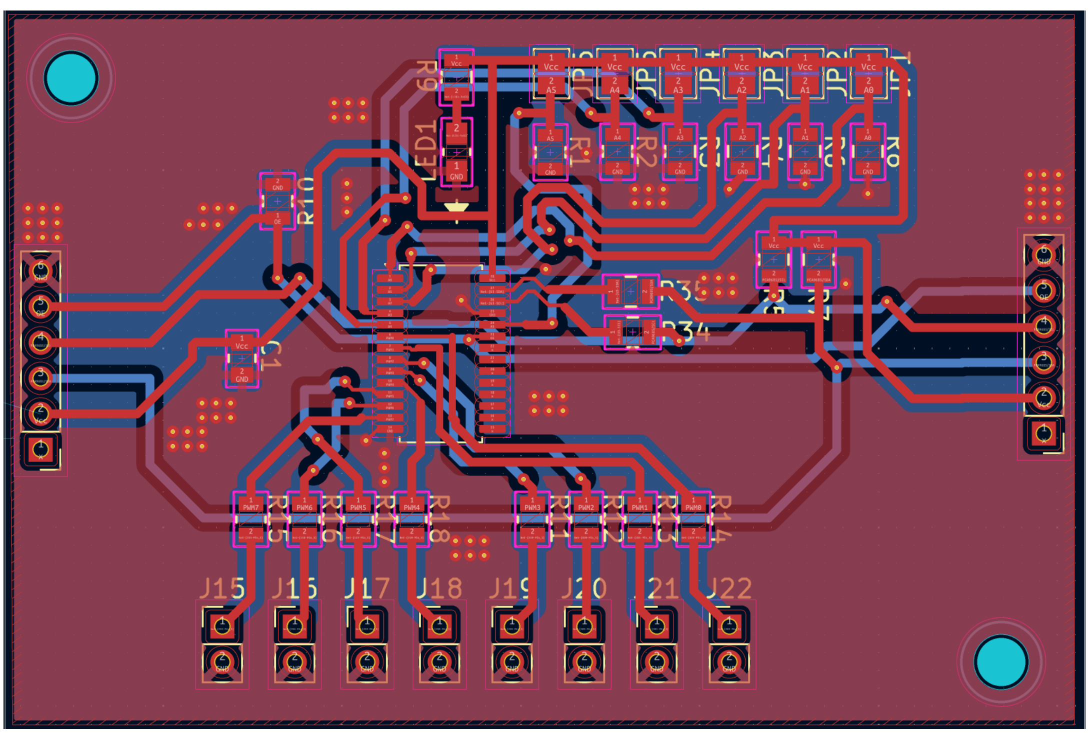


Figure 50: Custom PCA9685 PCB

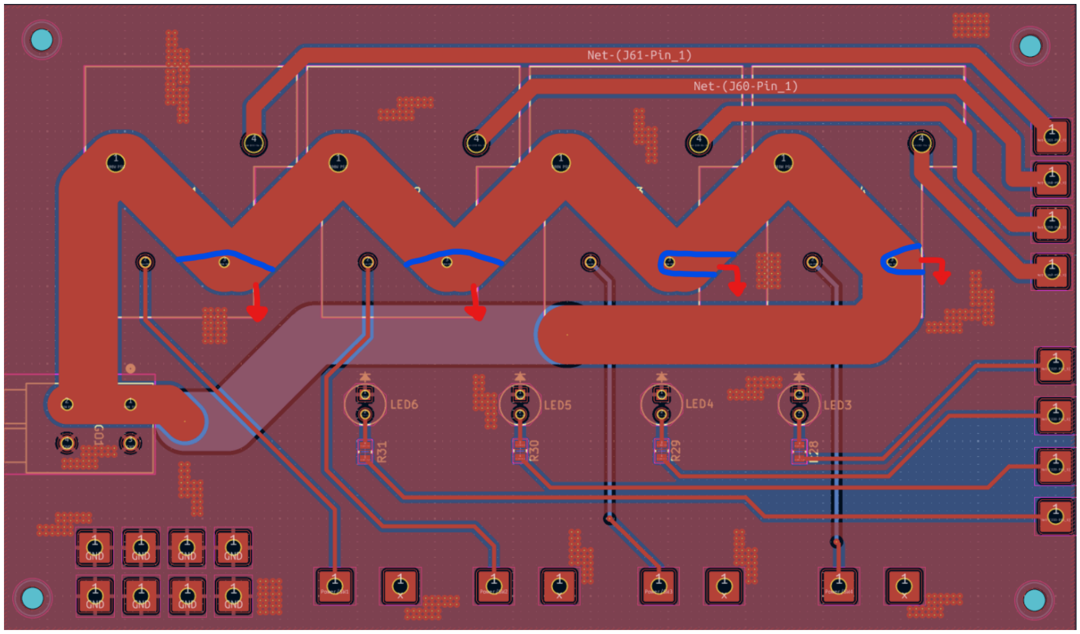


Figure 51: Custom Terminal Board PCB

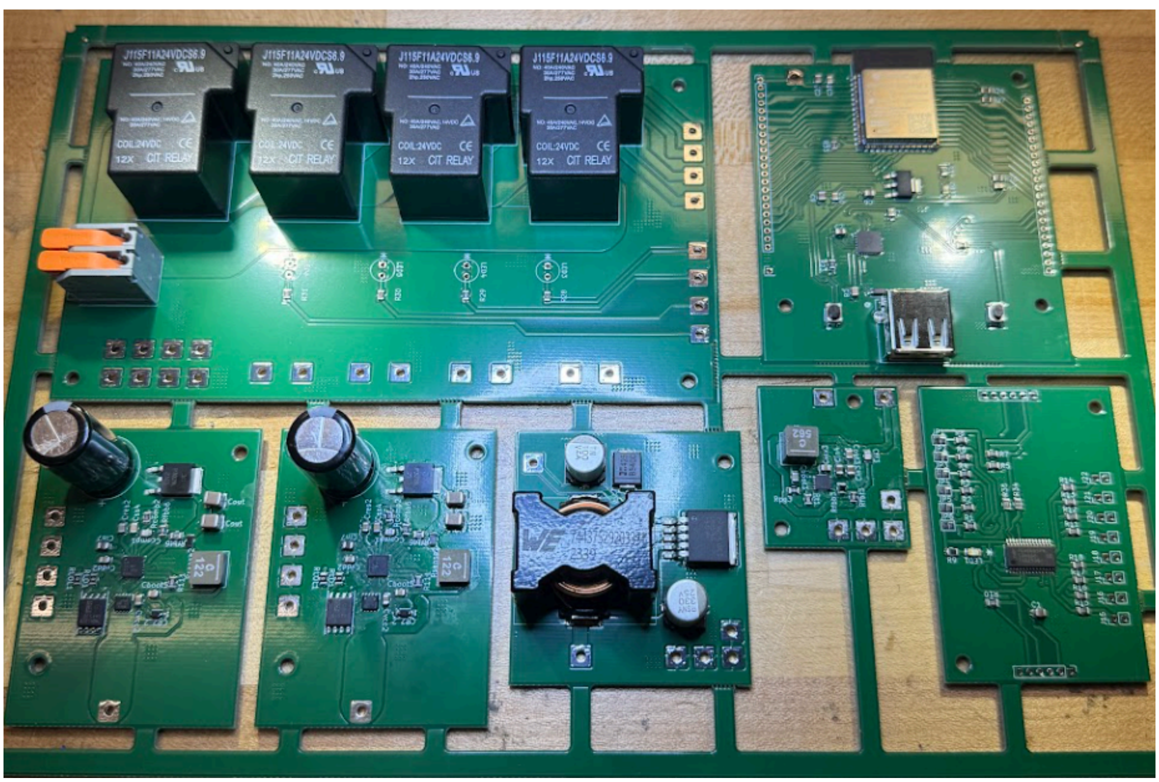


Figure 52: Custom Pannelized PCB Assembly

JLCPCB was selected as the primary printed circuit board manufacturer for the S.T.A.R. system due to its balance of affordability, fast turnaround time, and support for industry-standard fabrication processes. As a student design project with multiple PCB iterations expected, JLCPCB enables rapid prototyping without the high non-recurring engineering (NRE) costs typically associated with domestic or low-volume manufacturers. Their online quoting and design rule checking tools integrate well with KiCad, allowing the design team to quickly identify fabrication issues and iterate on board layouts. This accessibility makes JLCPCB particularly well-suited for senior design projects where schedule constraints and budget limitations are critical considerations.

The PCB designs for S.T.A.R. were developed to comply with JLCPCB's physical fabrication constraints to ensure manufacturability and reliability. Standard two-layer boards were used, with a nominal board thickness of 1.6 mm and 2 oz copper weight, which provides a good compromise between mechanical strength and current-carrying capability. JLCPCB's minimum trace width and spacing of approximately 6 mil (0.15 mm), minimum via drill size of 0.3 mm, and minimum annular ring requirements were taken into account during layout. Additionally, constraints on solder mask expansion, silkscreen line width, and component clearance influenced footprint selection and routing density. Recommendations for minimum trace width and component sizing from Dr. Weeks are well within the limits of JLCPCB's manufacturing capabilities.

Chapter 10 - System Testing and Evaluation

Testing and evaluation are major parts of the Senior Design process because they confirm whether the system works the way it was originally planned and whether any design decisions need to be adjusted before moving into the final build. During Senior Design I, most of the testing centered around the illumination system, the imaging and tracking system, and the mechanical fit of the 3D-printed components. These tests helped verify that the project is moving in the right direction and that the optical parameters, camera performance, and mechanical layout are all practical for a full-stage prototype. This chapter summarizes the tests performed so far, discusses the results, and outlines the next steps that will be taken in Senior Design II.

10.1 Hardware and Software Testing

10.1.1 Reflecting vs. Refracting Optics

The earliest hardware testing involved assembling the LED COB module, ellipsoidal cold mirror reflector, and the focusing lens in order to validate the calculated optical parameters. The goal was to measure whether the beam coming out of the illumination system matched the predictions from the design calculations: the 19° output angle, the expected focal behavior, and the clarity of the projected image at 7.5m.

The reflector and lens were aligned on a test bench using a simple adjustable platform. Once powered, the LED COB module produced a clean, bright beam that was successfully focused through the lens assembly. At a distance of 7.5 meters, the projected spot size matched the calculated value, and the edge clarity was consistent with what would be expected from a single-element lens. The beam uniformity was also acceptable, showing no major hotspots or dim rings. This confirmed that the optical calculations done earlier were correct. From the test bench, we marked the exact distances and translated those into the 3d model, confirming that the printed version will function exactly as expected.

The reflector also performed as expected. The cold mirror surface directed the majority of visible light toward the gate while offloading a portion of infrared energy away from the beam. During testing, the reflector remained stable and did not show signs of warping, peeling, or overheating. This result is important because the reflector is one of the few components that can drastically affect optical performance if it becomes misaligned or deformed.

10.1.2 Mechanical Housing Testing

To ensure that the optical components could be mounted securely, the team 3D-printed an initial version of the illumination housing using PLA. This version was primarily for mechanical validation rather than long-term use. The prints were dimensionally accurate, and each optical element fit properly into its designated slot. The threading and mechanical fasteners also aligned correctly, which confirmed that the CAD models were properly dimensioned.

Even though PLA softens at high temperatures, the test showed that the printed housing was stable when used for short-duration illumination tests. This supports the idea that PLA can be used for the development prototype while heavier use in SD2 may require an upgraded or reinforced shell, such as HT-PLA which performs much better at high temperatures.

In addition to validating the illumination housing, preliminary testing was conducted on the mechanical gimbal assembly to verify structural fitment, servo mounting alignment, and rotational clearance. The gimbal components were 3D printed and assembled to evaluate dimensional accuracy and confirm that the pan and tilt axes could rotate freely without mechanical interference. Particular attention was given to the servo mounting holes, bearing interfaces, and mating surfaces, as additive manufacturing tolerances can often introduce slight dimensional deviations. Initial testing showed that the printed tolerances were acceptable, with only minor post-processing required to smooth rotating interfaces and slightly enlarge fastener holes for proper fitment.

Performance testing of the gimbal demonstrated that the structure was rigid enough to support the illumination housing and camera assembly without excessive flexing or vibration during movement. Under manual articulation and low-speed servo-driven motion, the system maintained consistent alignment and showed minimal backlash in both axes. These results validated the mechanical design and confirmed that the gimbal geometry, mounting strategy, and printed tolerances were suitable for prototype-level testing. Additional stress testing and dynamic tracking evaluations were performed in Senior Design II to assess long-term wear, load capacity, and motion smoothness under full operational conditions.

10.1.3 PCB Testing

PCB testing is a critical phase in the development of the S.T.A.R. system, as it ensures that the manufactured boards meet both functional and reliability requirements before full system integration. We will now go over our testing and some of the issues we faced while working on each PCB.

LMR51450 3.3Vout:

Testing board using Keithley 2230-30-1 triple channel DC power supply, BK Precision 8500B DC Electronic Load, and R&S HMC 8012 digital multimeter.

Initial Testing

Parameter	**Value (Actual)**	**Value (Expected)**
Vout	~5V	3.3V
Iout	~0.050A	~0.150A (ESP32 load)

Final Testing

Parameter	**Value (Actual)**	**Value (Expected)**
Vout	~3.3V	3.3V
Iout	~0.140A (ESP32 loaded)	~0.150A (ESP32 load)
Vout Ripple	~200 mVpp (unloaded)	4.25 mVpp
Operating Frequency	~400kHz	504.68kHz

Issues:

After assembly of this 3.3V regulator board, we went to test it using a multimeter and DC power supply. We were expecting a 3.3Vout from 24Vin, but instead saw a 5Vout. This was concerning and puzzling to us since the board was working, but not dropping down the voltage to the correct output. We looked at the IC datasheet to see if any of our component values were incorrect. Initially we thought our inductor value was wrong, so we tried to recalculate the value according to the datasheet. From there we swapped out the inductor on the board from 5.6uH to 10uH and finally 6.8uH. None of these changes fixed your voltage output issue. We thought about leaving the board as is and fabricating a new 5V to 3.3V LDO using an AMS1117-3.3, however, this was very inconvenient for us as a new board would have to be designed and more money to be spent. we finally decided to take a step back and take another look at the WEBench schematics. Our mistake was right in front of us. The output capacitor was supposed to be 100uF, but we mistook the "u" for an "n" and designed the board with a 100nF capacitor in its place. After finding this out, I removed the 100nF capacitor, scratched off some of the PCB mask, and placed the much much larger 100uF capacitor in its place. We then tested the regulator board once again, and it worked as expected – 24Vin, 3.3Vout.

LM2576SX 5Vout:

Testing board using Keithley 2230-30-1 triple channel DC power supply, BK Precision 8500B DC Electronic Load, and R&S HMC 8012 digital multimeter.

Initial Testing

Parameter	**Value (Actual)**	**Value (Expected)**
Vout	~5V	5V
Iout	~0.500A (0.500A DC Load)	3A MAX
Vout Ripple	~19 mVpp (unloaded)	73.44 mVpp
Operating Frequency	~51kHz	52kHz

Issues:

No issues encountered in the fabrication and testing of this board

LM25141 8Vout:

Testing board using Keithley 2230-30-1 triple channel DC power supply, BK Precision 8500B DC Electronic Load, and R&S HMC 8012 digital multimeter.

Initial Testing

Parameter	**Value (Actual)**	**Value (Expected)**

Vout	~8V	8V
Iout	~0.001A (0.500A loaded)	10A MAX
Vout Ripple	~151 mVpp (unloaded)	6.11 mVpp
Operating Frequency	~3.7MHz	2.2MHz

Issues:
We ran into issues with the regulator's functionality when under load. When unloaded and supplied 24V, the output read 8V on the multimeter. However, once a load was added to the output, the regulator output dropped to 0V and 0A. According to the IC datasheet, the LM25141 has an auto-shutdown function to protect the chip. This explains the zero output. As for why the IC is shutting down when loaded, it turns out our PCB layout was incorrect. The datasheet calls for a 4-layer board with very specific layout guidelines, however, we were using a 2-layer board. Instead of redesigning a board to work with this IC, we decided to choose a new IC and design a simpler 2-layer board.

LMR51450 8.4Vout:
Testing board using Keithley 2230-30-1 triple channel DC power supply, BK Precision 8500B DC Electronic Load, and R&S HMC 8012 digital multimeter.

Initial Testing

Parameter	**Value (Actual)**	**Value (Expected)**
Vout	~8V	8V
Iout	~1.00A (1.00A loaded)	10A MAX
Vout Ripple	~100 mVpp (unloaded)	9.89 mVpp
Operating Frequency	~600kHz	504.68kHz

Issues:
No issues encountered in the fabrication and testing of this board

ESP32 Board:
Testing board using Keithley 2230-30-1 triple channel DC power supply.

I/O Validity:

GPIO Pin	Usage	Working (Y/N)
SCL	PWM control (to PCA9685)	**Yes**
SDA	PWM control (to PCA9685)	**Yes**
3.3Vin	Power to ESP32	**Yes**
GND	Ground	**Yes**

Issues:
No major issues encountered in the fabrication and testing of this board. While testing the functionality of the AMS1117 5V to 3.3V LDO, we noticed that there was no voltage being output. After looking back at our PCB layout, we noticed a trace was not added to the input of the LDO IC. As this does not affect the core functionality of the MCU, we did not opt to fix this. The MCU can still be powered on using the 3.3V input pin and does not need the 5V from the USB-A input.

PCA9685 Board:
Testing board using Keithley 2230-30-1 triple channel DC power supply.

I/O Validity:

I/O Pin	Usage	Working (Y/N)
SCL	PWM control (from ESP32)	**Yes**
SDA	PWM control (from ESP32)	**Yes**
PWM	PWM to motor	**Yes**
Vcc	Power to IC	**Yes**
GND	Ground	**Yes**

Issues:
No issues encountered in the fabrication and testing of this board

Terminal Board:

Testing board using Keithley 2230-30-1 triple channel DC power supply, BK Precision 8500B DC Electronic Load, and R&S HMC 8012 digital multimeter.

Component Validity:

Compnent	Usage	Working (Y/N)
Relays	1 for each Vout	**Yes**
WAGO	Input from PSU	**Yes**
Rocker Switches	1 for each relay	**Yes**
Vout	Power output to regulators	**Yes**
Vled	Power input from regulators	**Yes**

Issues:
There was a lot of troubleshooting done with this board. After initial assembly, it was found that the relays were not functioning properly. We noticed that the rocker switch reference ground pads were not connected to ground on our KiCAD layouts. This didn't cause much of an issue as we just terminated the reference GND wires to the terminal board ground pads. A bigger issue was the incorrect relay pin traces. We accidentally connected the input and ground pins together, which left us without a ground for each relay. To solve this, we had to scratch away at the solder mask, cut away at the copper trace around the GND pins, and solder a connector wire from GND to the now exposed trace. This took a lot of time and patience, but in the end, it solved our problem of non-functioning relays.

10.1.4 Imaging and Tracking System Testing

For initial imaging tests, the Raspberry Pi HQ camera was fitted with an off-the-shelf lens and connected to a Raspberry Pi running object-detection software. The primary goal was to verify that the camera could reliably detect and follow humans within a reasonable field of view and at an adequate frame rate.

Testing showed that the system could consistently detect and track people at approximately 15 frames per second, which is fast enough to maintain a smooth spotlight motion once connected to the gimbal. The camera captured clear images under typical indoor lighting, and the tracking software responded quickly to changes in movement. The algorithm was able to maintain lock on a subject even with moderate motion, confirming that the imaging subsystem is feasible for stage-tracking applications.

When testing with the spotlight enabled on the target, the imaging system is still able to identify and track people despite being brightly illuminated. This verifies that the integration of the two systems can successfully be done, as the illumination system will not interfere with the imaging system. To evaluate detection reliability, testing was conducted over 10 trials in which people walked in and out of the camera's frame to verify that the system was able to consistently detect and respond to individuals entering and leaving the field of view.

To test the latency of our system, two slow motion footage were recorded at 240 frames per second, one recording the subject in motion, and the other recording the gimbal system itself. By measuring the difference in frames between the end of a subject's motion and the end of the gimbal's motion, the latency in time can be determined. We recorded this in 10 different motions, to calculate the amount of latency between the detection of a subject and the mechanics of the spotlight.

10.1.5 Software Stability Testing

Software testing at this stage focused on basic tracking stability, frame rate, and ability to maintain a centered target. The algorithm processed video reliably and did not crash during extended testing. The Raspberry Pi maintained stable performance without thermal throttling, which is important for continuous operation. This gives confidence that the software foundation is strong enough to expand later in SD2, especially once the gimbal and the mobile app are added. The mobile app was thoroughly tested across all functions, and debugging was performed to resolve issues and slight mishaps that occurred when attempting to command the spotlight.

10.2 Performance Evaluation

10.2.1 Illumination Performance

The illumination system met the expected design performance in all major categories. The optical calculations predicted the size of the projected beam at 7.5m and the output angle of roughly 19°, and the real system matched these values closely. The projected image was sharp, confirming that the focal length and entrance pupil diameter were appropriate for the LED and reflector combination.

The LED COB produced a strong, uniform output, and the cold mirror reflector effectively shaped the beam into a usable profile for stage lighting. Because the reflector rejects a portion of the infrared light, the thermal load on the lens and housing was lower than expected, which is especially important when using plastic components.

10.2.2 Imaging Performance

The imaging system demonstrated good performance during early testing. The camera captured clean frames with predictable exposure, and the detection algorithm reliably locked onto people in the frame. The 15 fps tracking speed is fast enough for the gimbal to respond without noticeable lag, as long as the gimbal motors are properly tuned.

The camera also responded well to movement and subtle changes in position, which means the system can track people walking, turning, or shifting direction. These results confirm that the imaging subsystem is practical for real-time stage-tracking. The camera also is able to identify multiple subjects within the same frame, allowing for the potential of selective or mutli-target tracking.

The latency of the system achieved an average of 392 ms, a result which we deemed enough to provide a seamless motion of our spotlight to where the performer is moving. This is necessary as the performer must be able to effortlessly move around our stage without worry of the light trailing behind them.

10.2.3 Mechanical and Thermal Performance

The 3D-printed housing remained stable under moderate heat from the LED module. No visible deformation occurred, and the part maintained structural integrity throughout the test duration. This is encouraging because it shows that HTPLA is sufficient for initial prototyping.

Thermally, the LED and reflector stayed within expected ranges, and the airflow around the system was enough to prevent heat buildup. Final testing in SD2 will likely require a formal thermal management system (vents, optional fans, or heatsinks), but the current results suggest that no major redesign is necessary.

Mechanical performance testing of the gimbal assembly was also conducted to evaluate its behavior under active servo-driven movement and sustained load conditions. The pan and tilt mechanisms were tested through repeated motion cycles to observe positional consistency, structural stability, and rotational smoothness. The servo-driven axes maintained stable operation throughout testing, with no observable structural failure or significant deformation of the printed components. The gimbal demonstrated sufficient rigidity to support the combined weight of the spotlight housing and camera assembly while maintaining acceptable positional accuracy.

From a dynamic performance standpoint, the gimbal exhibited smooth motion with minimal oscillation or overshoot during directional changes. Minor vibration was observed at higher movement speeds, which is expected due to the inertia of the mounted optical assembly and the compliance of printed plastic components. However, these vibrations remained within acceptable limits for prototype operation and did not significantly impact target tracking stability. Future revisions may incorporate reinforced structural members, improved bearing supports, or damping mechanisms to further improve motion smoothness and long-term mechanical durability under continuous operation.

10.3 Overall System Integration

10.3.1 Component Integration

Bringing the components together is one of the most important steps in confirming that the S.T.A.R. system behaves like a single, connected device rather than a collection of individual parts. So far, the focus has been on getting the illumination system, the imaging system, and the early-stage gimbal to communicate in a simple but reliable way. Even though the prototype gimbal is a small-scale test platform, it has been extremely useful in verifying that the tracking coordinates from the camera can be translated into physical movement.

The current integration setup works by having the Raspberry Pi HQ camera detect a person in the frame, calculate the subject's position relative to the center, and then send those coordinates to the microcontroller running the gimbal. The microcontroller converts these coordinates into angular adjustments, and the gimbal moves accordingly. Even with a lightweight prototype, the system successfully demonstrated that real-time tracking is possible. The gimbal could pan and tilt smoothly enough to follow a subject, and the tracking remained stable as long as the person stayed within the field of view. These early tests confirm that the interaction between the software and motors is feasible and that scaling up to a full-sized gimbal should not introduce any major integration issues.

Another important step in subsystem integration is the early planning for user control. The final system will need a simple and intuitive way to let users switch operating modes, override the tracking, adjust brightness, and control other features. To prepare for this, the team has already begun outlining the communication structure for the mobile app that will be added in Senior Design II. The idea is for the app to connect wirelessly (most likely over Wi-Fi) to the microcontroller, which will act as the central hub for all system commands. Once implemented, the app will sit between the user and the gimbal/illumination system, allowing for immediate control without needing to physically access the fixture itself. This will also make testing easier in the next semester because app commands can be sent on the fly without interrupting the hardware setup.

Overall, the current level of subsystem integration shows that the S.T.A.R. system is positioned well for full-scale implementation. The camera, gimbal, and illumination components already communicate effectively in their prototype form, and the planned additions—such as a high-torque motorized gimbal and the app interface—will build on an already functional structure. The work done in Senior Design I confirms that the core interactions between tracking, movement, and lighting are feasible, and the team can now confidently move into Senior Design II with a clear integration pathway.

10.3.2 Alignment and Calibration

Alignment and calibration are essential steps in ensuring that the S.T.A.R. system performs consistently and produces predictable results. Because the system combines optics, imaging, and mechanical motion, small misalignments can lead to noticeable errors in the projected beam, the tracking accuracy, or the gimbal response. During Senior Design I, much of the alignment work focused on the illumination subsystem, since the reflector, LED, and focusing lens all rely on precise spatial relationships. However, early alignment procedures also helped establish a workflow for how calibration will be handled when everything is fully integrated in Senior Design II.

For the optical alignment, the LED COB module had to be positioned at the correct focal point of the ellipsoidal cold mirror reflector. Even a shift of a few millimeters can change how the beam converges or whether the light passes cleanly through the gate. During testing, the team made small iterative adjustments to the LED mounting depth so that the beam exiting the reflector matched the expected geometry. Once the LED and reflector alignment was dialed in, attention turned to the focusing lens. The lens was positioned at the calculated distance based on the numerical aperture and focal length, and then fine-tuned by observing the projected spot at different distances. This process confirmed that the optical path behaved the way it was modeled and that the mechanical housing held each part in the correct position without excessive play.

Alignment of the imaging system was also important because the camera essentially defines how the gimbal knows where to move. The camera's field of view, mounting angle, and relative orientation to the illumination beam all affect how well the tracking coordinates translate into physical motion. For the prototype stage, the camera was mounted rigidly above the illumination system and calibrated by centering a stationary subject in both the camera frame and the projected beam. This basic calibration allowed the team to test whether a detected off-center subject resulted in the correct gimbal movement direction. Even though the current gimbal is a small-scale model, this early calibration confirmed that the coordinate mapping works as expected.

Calibration also involved establishing a baseline for how the gimbal responds to tracking data. Since the prototype motors have limited torque and do not move with the same smoothness as the final version, the team had to adjust the gain and smoothing parameters in the control software to avoid jitter or overcorrection. These adjustments helped demonstrate how the final system will likely need damping algorithms or PID tuning to create fluid spotlight movement on stage. The results of this testing will directly inform the gimbal control design in Senior Design II, especially once the larger motors and full-weight illumination system are assembled.

Overall, the alignment and calibration work in Senior Design I provided a strong foundation for system integration. The optical components were verified to be accurately positioned, the imaging system demonstrated reliable coordinate mapping, and the initial gimbal calibration gave insight into how motion control will scale to the final hardware.

These early steps ensure that as the system becomes more complex in Senior Design II, the underlying alignment and control structure is already stable and well understood.

10.3.3 Prototype Limitations

Even though the early prototype successfully demonstrated the core functionality of the S.T.A.R. system, it still has several limitations that prevent it from operating at full performance. These limitations were expected at this stage because the purpose of the Senior Design I prototype was to confirm feasibility, not to finalize the design. By identifying these limitations now, the team can address them directly in Senior Design II.

One of the biggest limitations is the gimbal mechanism. The current prototype what we are using to test the gimbal system is a smaller scale version of the assembly. This lightweight gimbal is designed primarily for testing coordinate mapping and basic movement responses. While this setup works well for verifying the tracking algorithm, it does not have the torque, stability, or structural strength required to support the full illumination system. The LED module, reflector assembly, and focusing lens are significantly heavier than what the prototype gimbal can handle.

Another limitation comes from the lack of a custom camera lens. Right now, the Raspberry Pi HQ camera uses an off-the-shelf lens, which does not perfectly match the field of view or depth characteristics needed for optimal tracking. While the camera can still detect and track humans reliably, there are small differences in the optical prescription of the actual intended system. This becomes more noticeable when the camera must capture wide movement ranges, such as people walking across a stage. Designing and optimizing a dedicated imaging lens will improve detection accuracy, reduce edge distortion, and allow the camera to operate more consistently in various lighting environments.

The prototype also does not include a motorized iris, which limits control over the beam size. Without an adjustable iris, the spot size remains fixed, and the illumination system cannot dynamically adapt to different distances or subject heights. This feature is important for creating a professional stage-lighting effect, and its absence in the prototype means that certain lighting characteristics cannot yet be tested. Adding a mechanized iris in SD2 will allow full control of the beam diameter and improve the overall image quality of the spotlight.

In addition, the current prototype does not implement a wireless user interface. All testing has been controlled directly from the camera and microcontroller, which is fine for early development but not practical for real operation. Since stage lighting systems need quick and flexible control, the absence of a mobile app or operator interface is a temporary limitation. Once added, the app will allow users to toggle tracking, adjust brightness, move the gimbal manually, and access other system settings.

Finally, the prototype’s 3D-printed housing, while mechanically accurate, is not designed to withstand extended use or high thermal loads. HTPLA is suitable for mechanical prototyping, but long-duration operation of an LED COB module requires higher heat tolerance and better airflow. Although the current enclosure performed well during short tests, it will eventually need reinforcement or material upgrades. This limitation will be addressed once the team creates the final enclosure design with improved cooling and durability.

Chapter 11 - Administrative Content

11.1 Budget and Financing

The projected budget for this project is $1,700, which reflects both the technical complexity and the specialized hardware requirements necessary to achieve reliable real-time spotlight tracking. A significant portion of this cost comes from PCB manufacturing, as custom-designed boards are required to integrate the MCU, ICs, etc. into a compact, stage-ready system. Not to mention the cost of international shipping. Additionally, the optical components, including a high-quality LED emitter, reflector, and focusing lenses, contribute heavily to the expense, since precision glass optics and lighting elements for professional-grade illumination are costly. The motorized gimbal system also represents a major investment, as it must use high-torque, low-noise servo motors to provide smooth, precise, and silent operation suitable for performance environments. Beyond hardware, funds must also be allocated for testing, prototyping, and iteration, as initial prototypes may require redesigns or component replacements. Compared to off-the-shelf lighting systems, this budget remains reasonable, as it accounts for both performance-level functionality and the flexibility of a custom-engineered design. The group has agreed to split the total cost of this project equally between all 4 members.

Table 27: General BOM

Part Name	Vendor	$/unit or lot
ESP32 & RPi 5	Amazon	$152.47
Raspberry Pi Camera and Lenses	Digikey	$166.48
Servo Motors & Peripherals	Amazon	$303.44
Filament	Amazon	$96.83
Housing Hardware	Amazon	$81.33
Lens System	Edmond Optics	$268.37
PCB Components & Boards	JLCPCB	$633.31
Approx. TOTAL		**$1702.23**

The general Bill of Materials (BOM) for the S.T.A.R. system lists all major electrical, optical, and mechanical components used in the design. Core control hardware includes an ESP32-S3 microcontroller and Raspberry Pi Compute Module 5 units, which handle system processing and communication. The optical assembly consists of components sourced mainly from ETC and Edmund Optics, such as the HPL757 lamp, illumination lens, Source Four reflector, and precision achromatic and plano-concave lenses for imaging. The system also includes several servo motors for movement, a portable monitor for visual interface, and an RP Camera HQ for image capture.

Electronics and fabrication costs account for a significant portion of the budget. Miscellaneous PCB components and fabrication services from Digikey and JLCPCB were estimated at around $450 combined. The total projected cost of the S.T.A.R. system is approximately $1,700, though the team anticipates that final expenses may exceed this estimate due to unforeseen costs such as replacement parts, additional wiring, or updated module versions. Overall, the BOM demonstrates a balanced mix of off-the-shelf components and specialized optics to meet the system's illumination and imaging requirements.

11.2 Project Milestones

The milestones for the S.T.A.R. project outline the planned timeline and deliverables for both semesters of the Senior Design sequence. These checkpoints help the team stay on schedule by dividing the project into specific phases, each with measurable goals. During Senior Design I (Fall 2025), the main focus is on concept development, research, and preliminary prototyping. Early weeks involve group formation, selecting a project idea, and drafting the Divide and Conquer (D&C) document. Senior Design II begins with PCB testing to verify functionality of the control systems and electrical components. This is followed by code testing to ensure the application communicates properly with the PCB and supports tracking and gimbal control. Once functionality is confirmed, PCB assembly begins, where the prototype hardware is fully constructed. After assembly, additional PCB testing and debugging are performed to resolve any issues and ensure reliable system operation. The project concludes with the final presentation, where the fully assembled S.T.A.R. system is demonstrated and the final product is presented.

Table 28: Milestones Fall 2025

Senior Design I - Fall 2025			
Week	Milestone	Dates	Deliverable
1-2	Group Formation and Project Idea	8/18-8/29	Finalized Project Idea with Required Components
3	D&C Document	9/1 - 9/5	Finished D&C Document
4	D&C Revision	9/8	Revised D&C Document
5-6	Research on CMOS Camera and MCU	9/9 - 9/21	Selection of CMOS and MCU

Senior Design I - Fall 2025			
5-6	Research on ML model for object tracking and how to interface it with MCU	9/9 - 9/21	Demo application of of object tracking
7-8	Research motorized GIMBAL, PTU, and PCB Design	9/22 - 10/5	Demo application of Spotlight Tracking Person
9	Research on lowlight optimization	10/6 - 10/19	Demo application of Spotlight Tracking Person in near dark
10	Research on Wireless and UI Interface	10/20-10/26	App/web control demo
11-15	Integration testing	10/27 - 11/17	Prototype built
16	Final Report and Demo Video Due	11/25	Full Report Due

The Senior Design II (Spring 2026) milestones will focus on hardware implementation, testing, and system integration. Major tasks include PCB fabrication and testing, assembling the prototype, and refining system software. Once integration is complete, the team will conduct full system debugging and prepare for the final presentation. These milestones ensure that each stage of the project builds toward a complete and functional product, keeping development organized and on track for the end-of-semester demonstration.

Table 29: Milestones Spring 2026

Senior Design II - Spring 2026			
Week	Milestone	Dates	Deliverable
TBD	PCB Testing	2/20	Test PCB
TBD	Integrated Code testing	2/28	Have working application for PCB
TBD	PCB Assembly	2/30	Assembled Prototype
TBD	PCB Testing and Debugging	3/1	Assemble Device
TBD	Final Presentation	TBD	Present final product

In order to ensure an efficient workflow, we have assigned specific tasks to each member of the group based on their area of expertise. The photonics engineers, Travis and Gage, are responsible for the optical and imaging aspects of the project. Their work includes lens design, optical calculations, selecting and integrating the image sensor and camera module, and developing the illumination system. They also manage the system display output and production of the demo video. The computer engineer, Lucio, is responsible for the software portion of the system. His tasks include developing the computer vision algorithms, designing the user interface, coding the system logic, and managing motor control for the gimbal. He also maintains the Senior Design website. The electrical engineer, Jerison, focuses on PCB design, electrical circuitry, and gimbal motor control. He is also in charge of 3D modeling and printing mechanical components for housing and support. This distribution of work allows each team member to specialize in their field while ensuring smooth integration across the optical, electrical, and software systems.

Table 30: Work Distribution Table

Photonics Engineers	**Responsibilities**
Travis and Gage	Lens design
	Optical calculations
	Image sensor / Camera module
	System display output
	Administrative Work
	Demo video production
Computer Engineer	**Responsibilities**
Lucio	Computer vision algorithms
	System UI
	System logic coding
	Motor command code
	Senior Design website
Electrical Engineer	**Responsibilities**
Jerison	PCB design

	Motor logic for gimbal
	Administrative Work
	3D design & printing
	Circuitry & electrical components

Chapter 12 - Conclusion

The S.T.A.R. (Spotlight Tracking with Automated Recognition) system represents a multidisciplinary senior design project developed at the University of Central Florida. This project integrates principles of photonics, electrical engineering, and computer engineering to create an autonomous spotlight capable of detecting and tracking human subjects in real time. Building upon the legacy of previous UCF projects such as CHROMA, and existing products such as the ETC Source Four, S.T.A.R. extends the concept of intelligent lighting into the live performance environment by combining optical precision, embedded control, and computer vision into a single self-contained unit. The motivation for this design stemmed from the need to reduce reliance on manual spotlight operation in stage productions and presentations, offering a more affordable and accessible alternative to large-scale automated lighting systems used in professional theaters.

Summary of Design and Integration

The system merges multiple engineering disciplines into a cohesive prototype. The optical subsystem was modeled after the ETC Source Four ellipsoidal reflector spotlight, a well-established standard in stage lighting. Through analysis of reflecting versus refracting optics, the design implements an ellipsoidal cold mirror reflector paired with a biconvex focusing lens to achieve an output beam angle of approximately 19°. The use of N-BK7 glass provides high transmission efficiency in the visible spectrum, while internal baffles minimize stray light for improved beam contrast. This optical design ensures that the illumination maintains professional quality while remaining compact and thermally stable.

The imaging subsystem employs a Raspberry Pi High Quality Camera with a Sony IMX477 CMOS sensor, chosen for its high frame rate, sensitivity, and affordability. The imaging optics use commercial off-the-shelf lenses to achieve the required magnification and minimize aberrations. Through this system, the camera captures continuous video for human detection and tracking. On the computational side, a Raspberry Pi Compute Module 5 executes the tracking algorithm using OpenCV and a trained YOLO model, while an ESP32-S3 microcontroller handles gimbal motor control and communication. Together, these subsystems form an integrated pipeline, from image capture and recognition to motorized actuation that achieves tracking accuracy within 5° RMS at 10 m and latency under 300 milliseconds.

The mechanical and electrical systems were designed for reliability, heat management, and modularity. High-torque servo motors provide smooth two-axis pan and tilt movement. The PCB design integrates motor drivers, logic control, and communication interfaces within a compact footprint. HT-PLA was selected as the housing material for its lightweight strength and heat resistance, allowing the fixture to remain durable without the need for metal fabrication. The modular construction of the system supports easy maintenance and future upgrades, including potential LED light-engine integration.

Interdisciplinary Collaboration

Development of S.T.A.R. will require consistent coordination between team members specializing in photonics, electrical engineering, and computer engineering. The photonics engineers will focus on optical design, sensor integration, and illumination system analysis; the computer engineer will develop the vision algorithm, system interface, and embedded control logic; and the electrical engineer will design the power systems, PCB layout, and mechanical housing. This division of labor allows the team to efficiently address each subsystem's challenges while maintaining consistent system-level communication. Integration testing, simulation, and modeling ensure that each component will reliably function within the complete assembly.

This collaboration reflects the interdisciplinary nature of modern engineering projects. Optical engineers must understand electronic control systems, while software developers must account for real-world optical constraints such as focus, field of view, and latency. The success of the S.T.A.R. project demonstrates how these specialties intersect to produce a practical, automated system capable of meeting both technical and creative requirements.

Research Contributions and Learning Outcomes

Throughout development, the team explored and compared multiple technologies to determine the most effective balance between performance, cost, and complexity. Research into illumination systems evaluated halogen, LED COB, and RGBW light engines, ultimately identifying the LED COB module as the most efficient choice for future iterations. Optical design research applied principles such as Snell's Law, the Lensmaker's equation, and numerical aperture calculations to determine lens geometries and focal lengths. Similarly, research on sensor technologies compared CCD and CMOS architectures to assess trade-offs between noise performance, speed, and power consumption.

Beyond technical understanding, the project provides significant hands-on experience with professional design tools and fabrication techniques. The team will gain practical knowledge in optical simulation, PCB layout, 3D modeling, and embedded programming. The experience of managing procurement, budgeting, and interdisciplinary collaboration will also reinforce professional engineering practices, which are skills that extend beyond the classroom into industry applications.

Challenges and Future Improvements

As with any complex engineering system, the development of S.T.A.R. will involve numerous technical and logistical challenges. Optical alignment, heat management, and signal synchronization between the imaging and motor subsystems will require careful calibration. Latency reduction will be particularly demanding, as delays could arise at multiple stages—image processing, command transmission, or mechanical response. Further optimization of software algorithms and microcontroller communication protocols may yield even faster and smoother tracking.

For future iterations, several enhancements have been identified. The first involves replacing the halogen light source with a high-power LED COB module to improve efficiency and longevity while maintaining brightness. The second focuses on refining the gimbal mechanics with brushless motors and encoders for higher precision and reduced noise. Expanding the computer-vision model to support multi-subject tracking and user selection would further enhance versatility. Additional goals include implementing DMX/Ethernet control for compatibility with professional lighting consoles, improving the mobile UI for real-time calibration, and developing thermal management solutions such as passive heat sinks or forced air cooling.

Broader Impact and Future Applications

S.T.A.R. demonstrates how automation and computer vision can enhance accessibility in live performance technology. The same principles applied here—autonomous motion tracking, intelligent illumination, and wireless control—can be extended to other environments such as lecture halls, film studios, and industrial inspection systems. By leveraging affordable components and open-source software, the project lowers the barrier for integrating intelligent lighting into small-scale venues or educational settings. In doing so, it bridges the gap between professional automation systems and entry-level stage technology, enabling a broader range of users to benefit from advanced optical engineering.

The project also contributes to the growing intersection between photonics and computer science. As imaging sensors and embedded processors become increasingly powerful, autonomous optical systems like S.T.A.R. will continue to evolve. The team's work provides a foundation for future UCF students to build upon, whether by refining the optics, optimizing code performance, or expanding functionality through artificial intelligence and networking.

Final Remarks

In conclusion, the S.T.A.R. project represents the integration of optical design, embedded electronics, and computer vision into a unified system capable of dynamic, real-time human tracking. It demonstrates the practical application of engineering in solving real-world problems while emphasizing affordability, modularity, and user accessibility. Through detailed research, structured milestones, and collaborative teamwork, the project will achieve its foundational objectives and establish a platform for future innovation in automated lighting. S.T.A.R. not only showcases the technical abilities of its designers but also serves as an example of how multidisciplinary collaboration can transform a conceptual idea into a functional, impactful engineering solution.

Appendix A - References

[1] T. B. Greenslade and S. G. Heisler, “Stage Lighting Instruments,” *The Physics Teacher*, vol. 13, no. 9, pp. 548–551, 1975.

[2] A. P. Kshirsagar and H. Azath, “A Comprehensive ATM Security Framework for Detecting Abnormal Human Activity via Granger Causality-Inspired Graph Neural Network Optimized with Eagle-Strategy Supply-Demand Optimization,” *Expert Systems with Applications*, vol. 272, 2025.

[3] R. Holzer, “YOLO — Object Detection,” *OpenCV Tutorial*, 2019. [Online]. Available: https://opencv-tutorial.readthedocs.io/en/latest/yolo/yolo.html

[4] “The Ultimate Guide to Lens Design Forms: The Types of Optical Systems in a Lens Designer’s Toolbox,” *Pencil of Rays*. [Online]. Available: https://www.pencilofrays.com/lens-design-forms/

[5] M. Kaur, et al., “Security Camera with Frame Detection,” in *Proc. 2024 Int. Conf. on Emerging Smart Computing and Informatics (ESCI)*, IEEE, 2024, pp. 1–5.

[15] D. Korsch, *Reflective Optics*. Boston, MA: Academic Press, 1991.

[16] Z. Djafar, A. Z. Salsabila, and W. H. Piarah, “Performance Comparison Between Hot Mirror and Cold Mirror as a Beam Splitter on Photovoltaic-Thermoelectric Generator Hybrid Using LabVIEW Simulator,” *Heat and Technology*, vol. 39, no. 5, pp. 1609–1617, 2021.

[17] A. Q. Gadhban, H. M. Roomy, and S. K. Taha, “Design High Performance Multilayers Cold Mirror,” *J. Phys.: Conf. Ser.*, vol. 1879, no. 3, p. 032086, 2021.

[18] Newport Corporation, “Paraboloidal and Ellipsoidal Reflectors.” [Online]. Available: https://www.newport.com/f/paraboloidal-and-ellipsoidal-reflectors

[20] I. G. Priest and U.S. National Bureau of Standards, *A New Method of Determining the Focal Length of a Converging Lens*. Washington, DC: U.S. Dept. of Commerce and Labor, Bureau of Standards, 1909.

[21] Electronic Theatre Controls Inc., “Gobo Holder Comparison for Source Four.” [Online]. Available: https://support.etcconnect.com/ETC/Fixtures/Source_Four/Source_Four_ERS_and_HID/Gobo_Holder_Comparison_for_Source_Four

[22] “Landscape Lighting Assembly Having Stackable Gobo Sections,” *U.S. Patent Application 20190078761*, 2019.

[23] "LED Light Fixture for Indirect Lighting with Adaptable Baffle Structure," *U.S. Patent Application 20190309917*, 2019.

[24] J. A. Jimenez, "Enhanced NIR Emission from Nd^{3+} Ions in Plasmonic & Dichroic Cu Nanocomposite Glass," *Chemical Physics Letters*, vol. 824, 2023.

[25] A. S. H. Md Yasir, et al., "Effect of Heat Treatment on Mechanical Properties and Dimensional Accuracy of 3D-Printed Black Carbon Fiber HTPLA," *Heliyon*, vol. 10, no. 11, 2024.

[26] E. Juntunen, et al., "Copper-Core MCPCB with Thermal Vias for High-Power COB LED Modules," *IEEE Trans. Power Electron.*, vol. 29, no. 3, pp. 1410–1417, 2014.

[27] R. D. Araguillin, et al., "Spectral Power Distribution of the Tungsten-Halogen Lamp (Blue) Compared with the Blackbody Spectrum with the Equivalent Color Temperature (Red)," *ResearchGate*, 2025. [Online]. Available: https://www.researchgate.net/figure/Spectral-power-distribution-of-the-tungsten-halogen-lamp-blue-compared-with-the_fig2_382400303

[28] Gigahertz-Optik GmbH, "ISS-15-RGBW." [Online]. Available: https://www.gigahertz-optik.com/en-us/product/iss-15-rgbw/

[29] Electronic Theatre Controls Inc., "Source 4WRD Color Tech Specs." [Online]. Available: https://www.etcconnect.com/Products/Legacy/Lighting-Fixtures/S4WRD-C/Tech-Specs.aspx

[30] Flutter, "Build UIs." [Online]. Available: https://docs.flutter.dev/ui

[31] Flutter, "Performance Best Practices." [Online]. Available: https://docs.flutter.dev/perf

[32] Meta, "React Native — Getting Started." [Online]. Available: https://reactnative.dev/docs/getting-started

[33] Meta, "React Native — Optimizing Performance." [Online]. Available: https://reactnative.dev/docs/optimizing-performance

[34] Google, "Dart Language Tour." [Online]. Available: https://dart.dev/guides/language/language-tour

[35] Meta, "React Native — Architecture (JSI, TurboModules)." [Online]. Available: https://reactnative.dev/architecture

[36] R. T. Fielding, M. Nottingham, and J. Reschke, "HTTP Semantics," *RFC 9110*, Jun. 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9110

[37] M. Thomson and C. Benfield, "HTTP/2," *RFC 9113*, Jun. 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9113

[38] I. Fette and A. Melnikov, "The WebSocket Protocol," *RFC 6455*, Dec. 2011. [Online]. Available: https://www.rfc-editor.org/rfc/rfc6455

[39] OASIS, "MQTT Version 3.1.1," *OASIS Standard*, Dec. 2015. [Online]. Available: https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

[40] OASIS, "MQTT Version 5.0," *OASIS Standard*, Mar. 2019. [Online]. Available: https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html

[41] Eclipse Foundation, "Eclipse Mosquitto — An Open Source MQTT Broker." [Online]. Available: https://mosquitto.org/

[42] S. Tiangolo, "FastAPI User Guide." [Online]. Available: https://fastapi.tiangolo.com/

[43] Pallets, "Flask Documentation." [Online]. Available: https://flask.palletsprojects.com/

[44] Python Software Foundation, "Asyncio — Asynchronous I/O." [Online]. Available: https://docs.python.org/3/library/asyncio.html

[45] OpenCV Team, "OpenCV Documentation and Tutorials." [Online]. Available: https://docs.opencv.org/

[46] Ultralytics, "YOLOv8 Documentation." [Online]. Available: https://docs.ultralytics.com/

[47] Raspberry Pi Ltd., "Cameras — Documentation (libcamera)." [Online]. Available: https://www.raspberrypi.com/documentation/computers/cameras.html

[48] Espressif Systems, "ESP-IDF Programming Guide." [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/

[49] Espressif Systems, "Arduino Core for ESP32 — Documentation." [Online]. Available: https://docs.espressif.com/projects/arduino-esp32/en/latest/

[50] Adafruit, "PCA9685 16-Channel PWM Servo Driver." [Online]. Available: https://learn.adafruit.com/16-channel-pwm-servo-driver

[51] Flutter, "Navigation and Routing." [Online]. Available: https://docs.flutter.dev/ui/navigation

[52] Meta, "React Native — Networking and WebSocket." [Online]. Available: https://reactnative.dev/docs/network

[53] Google, “Material Design 3 — Design System.” [Online]. Available: https://m3.material.io/

Appendix E - ChatGPT Prompts

[1a] ***Prompt:*** “*Explain the differences between a COB lighting module and an RGBW engine. I understand that they are both LED driven, but what are some key differences? Please explain in one paragraph*”
[1b] ***Response:*** *“A COB (Chip-on-Board) lighting module and an RGBW engine are both LED-based systems, but they serve different purposes and are built differently. A COB module packs many small LED dies tightly together on a single substrate to create a large, uniform light-emitting surface. This design produces very high brightness and smooth, even output, making COBs ideal for applications like stage fixtures, spotlights, and general illumination where consistent white light and strong punch are needed. In contrast, an RGBW engine uses separate red, green, blue, and white LEDs—often arranged in clusters or mixed through optics—to create a wide range of colors, including white light through color mixing. RGBW engines offer precise color control and dynamic effects, but typically have less uniformity and may show color separation without proper optics or diffusion. In short, COBs prioritize brightness and uniform white output, while RGBW engines prioritize color flexibility and tunability.”*

[2a] ***Prompt:*** *“Explain the reasoning behind putting baffles in a lighting fixture. I understand it helps with reflection and stray light, but please provide a quick summary on the benefits”*
[2b] ***Response:*** *“Baffles are added to lighting fixtures to control and shape the light more precisely by blocking stray rays and reducing internal reflections. They help prevent light from spilling where it isn't wanted, improving beam definition and maintaining clean edges, which is especially important in stage and architectural lighting. Baffles also reduce glare by shielding the light source from direct view at certain angles, making the output more comfortable for audiences or occupants. Additionally, they enhance contrast and image clarity in optical systems by limiting light that would otherwise wash out the beam or reduce sharpness. Overall, baffles provide better beam quality, visual comfort, and more professional, controlled lighting performance.”*

[3a] ***Prompt:*** *“What is the heat output of an ETC Source Four tungsten-halogen lamp?*
[3b] ***Response:*** *“The ETC Source Four (halogen version) emits a total heat output that can be calculated as* ***3.412 BTU per hour per watt*** *of electrical input. Electronic Theatre Controls Inc*

- *For the 575 W HPL lamp: ~1,962 BTU/hr. Electronic Theatre Controls Inc*
- *For the 750 W HPL lamp: ~2,559 BTU/hr. Electronic Theatre Controls Inc*

If you tell me the exact wattage of your lamp I can compute the exact BTU/hr for your specific fixture.

[4a] ***Prompt:*** *"I want to expand on reasons why you would want to use a lens instead of a reflector. Can you write a paragraph or so explaining this?"*
[4b] ***Response:*** *"While reflectors are highly efficient for collecting and redirecting light, they alone cannot achieve the level of* ***beam precision and focus control*** *required in professional illumination systems. Reflective surfaces shape light primarily through geometric reflection, which is ideal for collimation or spreading light broadly, but they lack the fine spatial control necessary to form a sharply defined, adjustable beam. A* ***refractive lens****, on the other hand, allows for precise manipulation of light convergence and divergence by altering the optical path through changes in refractive index. This enables the user to fine-tune the focus position and beam diameter, producing a crisp edge and uniform intensity distribution across the illuminated field. Unlike a reflector, a lens provides a continuously variable focus range simply by changing the axial distance between the lens and the light source or gate plane. For the S.T.A.R. system, this flexibility is crucial— it allows the same optical assembly to adapt to different stage sizes and performer distances without mechanical reconfiguration of the reflector. Additionally, lenses can correct minor beam aberrations introduced by the reflector and improve overall image quality when projecting through gobos or irises, making them indispensable for applications that demand precision and visual consistency."*

[5a] ***Prompt:*** *"What does magnification do for a lens design?"*
[5b] ***Response:*** *"In a lens design, magnification describes how the size of the image formed on the sensor relates to the size of the object being viewed. For an imaging system like one capturing a theater stage, magnification tells you how much smaller the stage appears when projected onto a camera sensor. It is defined as the ratio of image size to object size, often expressed as* $M = \frac{\text{image size}}{\text{object size}}$*, and is typically negative for real inverted images. When the magnification is less than one in magnitude — that is, when* $|M| < 1$ *— the image is a demagnified version of the scene. The magnification also determines the field of view (FOV): a smaller magnification allows more of the scene to fit within the frame, while a larger magnification narrows the FOV and zooms in on smaller details. In practical terms, magnification bridges the geometry of the object, the sensor, and the focal length — guiding how lenses are selected and spaced to ensure that the desired area of the stage is imaged sharply onto the sensor.*

[6a] ***Prompt:*** *"Can you elaborate on what demagnification indicates?"*
[6b] ***Response:*** *"Demagnification is the process by which a lens system forms an image that is smaller than the actual object being viewed. In optical design, it refers to a* ***magnification magnitude less than one (|M| < 1)****, meaning the image on the sensor is a scaled-down version of the original scene. This occurs when the* ***object distance*** *(the distance from the lens to the scene) is much greater than the* ***image distance*** *(the distance from the lens to the sensor). In stage or theater imaging, demagnification is essential because a very large object — such as a 25-foot-wide stage — must be captured onto a small camera sensor only a few millimeters across. To achieve this, lenses are chosen or combined to create a* ***short effective focal length****, which allows the system to project a wide field of view while keeping the image sharply focused on the sensor. The degree of demagnification directly controls how much of the scene fits within the image: stronger*

demagnification means a wider portion of the stage is visible. However, designing for extreme demagnification introduces challenges, such as maintaining image sharpness across the field and controlling aberrations, since rays from wide angles must still converge precisely onto the small sensor plane.

*[7a] **Prompt:** "What type of connector would you recommend specifically for 25A at 24V—are there any brands or models you suggest?"*
*[7b] **Response:** "For a 25A, 24V application, you'll want to use connectors that are specifically rated for high current. Here are a few types and brands that are commonly recommended:*

*1. **Anderson Power Products**: Their Powerpole connectors are popular for high-current applications. They are modular and can handle up to 45A, making them suitable for your needs.*

*2. **Amphenol**: Known for their robust industrial connectors, Amphenol offers various models that can handle high current, such as the Amphe-Power series.*

*3. **Molex**: The Molex Mini-Fit Jr. series is another option, designed for high-current applications up to 9A per circuit, but they can be used in parallel configurations to handle higher currents.*

*4. **TE Connectivity**: Their Dynamic Series connectors are designed for high-current applications and offer reliable performance.*

*5. **WAGO**: WAGO's lever-operated connectors are easy to use and can handle high currents, suitable for quick and secure connections.*

When selecting a connector, ensure it is rated for at least 25A and is compatible with your wire gauge. Additionally, consider factors like ease of use, environmental resistance, and mechanical durability based on your specific application needs."

*[8a] **Prompt:** "How do I calculate the required PCB trace width for 25A current at 24V, and are there any online tools you recommend for this?"*
*[8b] **Response:** "Calculating the required PCB trace width for a specific current involves considering factors such as the current carrying capacity, temperature rise, and the thickness of the copper. The IPC-2221 standard provides guidelines for trace width calculations. Here's a basic approach:*

Calculation Steps:
*1. **Determine Copper Thickness**: Common copper thicknesses are 1 oz/ft² (35 µm) and 2 oz/ft² (70 µm).*

*2. **Temperature Rise**: Decide on an acceptable temperature rise, typically 10°C to 20°C above ambient.*

3. **Use IPC-2221 Formula**: The IPC-2221 standard provides formulas to calculate trace width based on current, copper thickness, and temperature rise.

Online Tools:
Several online calculators can help you determine the trace width:

1. **Saturn PCB Design Toolkit**: A popular tool that provides comprehensive calculations for trace width, impedance, and more.

2. **Advanced Circuits Trace Width Calculator**: Offers a simple interface to input current, copper thickness, and temperature rise to get the trace width.

3. **EasyEDA Trace Width Calculator**: An easy-to-use tool for quick calculations based on IPC standards.

Example Calculation:
For a 1 oz copper thickness and a 10°C temperature rise, you might find that a trace width of approximately 200 mils (5 mm) is required for 25A. However, this can vary based on specific conditions and standards.

Using these tools and guidelines, you can accurately calculate the trace width needed for your PCB design to safely handle 25A at 24V."

[9a] **Prompt:** "how large would the PCB traces be to handle 25A and 8A"
[9b] **Response:** "To determine the appropriate PCB trace width for handling specific currents, you can use the IPC-2221 standard as a guideline. This standard provides formulas and tables for calculating trace width based on current, copper thickness, and allowable temperature rise. Here's a general approach:

Factors to Consider:
1. **Copper Thickness**: Common copper thicknesses are 1 oz/ft^2 (35 µm) and 2 oz/ft^2 (70 µm).
2. **Temperature Rise**: Decide on an acceptable temperature rise, typically 10°C to 20°C above ambient.

Calculating Trace Width:
For a rough estimate, you can use online calculators or the IPC-2221 formula. Here are typical values for 1 oz copper thickness:

1. **25A Current**:
   - **External Layers**: Approximately 400 mils (10 mm) for a 10°C temperature rise.
   - **Internal Layers**: Approximately 800 mils (20 mm) for a 10°C temperature rise.

2. **8A Current**:
   - **External Layers**: Approximately 100 mils (2.5 mm) for a 10°C temperature rise.
   - **Internal Layers**: Approximately 200 mils (5 mm) for a 10°C temperature rise.

Online Tools:
- **Saturn PCB Design Toolkit**: Offers comprehensive calculations for trace width, impedance, and more.
- **Advanced Circuits Trace Width Calculator**: Provides a simple interface to input current, copper thickness, and temperature rise to get the trace width.

Considerations:
- **Safety Margin**: It's often wise to use a slightly wider trace than calculated to provide a safety margin.
- **Thermal Management**: Ensure adequate cooling and consider using thicker copper if space is limited.
- **Testing**: Validate the design with thermal and electrical testing to ensure reliability.

These calculations provide a starting point, but always verify with detailed design tools and testing to ensure your PCB can handle the required currents safely."