Automated Lens Parameter Metrology
Lens In, Datasheet Out.

Kiva McCracken[1], Ollie Mueller[1], Zachary Kassner[2], and Daniel Gomez[3]

CREOL Group 4 – University of Central Florida, Florida, FL 32826 USA
[1]PSE, [2]EE, [3]CPE

Abstract

The Automated Lens Metrology system uses a fleet of sensors to measure lens properties such as diameter, center thickness, front radius of curvature, back radius of curvature, effective focal length (at different wavelengths), reflective spectra, and transmissive spectra. These properties are used to derive additional parameters and build a datasheet that provides the user with information about the inputted lens.

Advisors

Dr. Aravinda Kar & Dr. Sreeram Sundaresh

Reviewers

Dr. Stephen Eikenberry, Dr. Justin Phelps, & Dr. Nazanin Rahnavard

CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. System Overview

A. Motivation & Background

Optical laboratories suffer from a chronic component organization issue. Specifically, it is hard to label and keep track of lenses that perform vastly different tasks because they tend to look very similar. As a result, there are tens of thousands of dollars of unsorted optics in our department that are not utilized effectively.

As optical engineers, we have experienced this problem first-hand. A typical sight in our labs is a teaching assistant standing over a set of lenses to compare their focal lengths by eye alone. This is an imprecise and time-consuming sorting method that is not suitable for research. Our design project aims to remedy this issue by autonomously creating a comprehensive datasheet for a given test lens. This design's intended users are students, researchers, and professors in a lab setting.

Current organization solutions on the market focus on the storage portion of inventory. Companies like ThorLabs [16] sell pouches, cassettes, and large storage drawers for lenses. However, these products do not help in identifying an unknown or misplaced lens. The lens metrology systems currently available are generally geared towards production and high-accuracy single data point measurement. Our system serves as a middle ground, providing researchers with data they otherwise might not have about their lenses in a timely and convenient manner. It is designed to work on a macroscopic scale, listing lens characteristics rather than data on quality control.

Typical datasheets in the optics industry take the form of tables detailing the geometric and optical data of a particular lens. While some datasheets are more thorough than others, most include specifications for diameter, focal length, index of refraction, radii of curvature, lens material, coating, and Abbe number [11] [12] [15]. Many also provide a transmission or reflection graph over the visible spectrum [11] [12].

The design does feature one major compromise. It doesn't measure surface or curvature quality. This is because the easiest way to do so is a white light interferometer, and based on the time constraints for the project, isn't a realistic goal. Furthermore most lab optics are bought already having undergone a quality control process. So our device just focuses on identifying and producing a datasheet rather than quality assurance.

B. Existing Products/Projects

There are several existing products that automate parts of the lens metrology process. While their scopes vary due to their intended applications, these systems usually function in the UV and visible range and measure foci, dimensions, and transmission spectra. The examples covered in our analysis are:

1) Automatic Lens Analyzer

The SOLOS Automatic Lens Analyzer by Tropcon Healthcare is a fully automated lensmeter with a spectrometer measuring UV-A and visible light from 315 nm to 800 nm. It is intended for use in optometry and automatically detects, measures, and marks the centers of single vision, progressive, and multifocal eyeglass lenses. It also measures UV-A, blue light, and visible light transmittance. Its data can be either printed on an internal printer or transmitted wirelessly over LAN or Wi-Fi. This machine is relatively small, fitting easily on the table of a lab or office [7].

2) Optical lens thickness gauge

The OptiSurf Optical lens thickness gauge by Trioptics uses an interferometer to take non-contact center thickness measurements of single lenses and flat optics. The description listed on company's website notes the product's ease of use, even for "inexperienced operators" [17]. It includes custom software that automatically aligns the lens in the machine and conducts a statistical analysis of the measured data. It, too, is relatively small and can fit on the table of a lab or office [17].

3) Micro Lens Process Metrology Inspection Equipment

The Micro Lens Process Metrology Inspection Equipment by Zygo measures lens data, analyses it, and visualizes it as graphs. Although the exact lens featured measured by this machine are unavailable on its catalog page, it is described as supporting “a wide range of parts and metrology parameters” [18]. It is a relatively large machine, and would likely be stored on the floor of a lab.

4) Software Components

For the software portion of this project, we will be utilizing available open-source libraries as well as existing micro-controllers such as the Raspberry Pi and STM32. These libraries differ in content and scope depending on the technologies they relate to and the purposes those technologies serve.

1) **Raspberry Pi** The existing libraries that will be used on the Raspberry Pi will be split among four tasks:
   a) **Hardware Communication**
   For hardware communication between the Raspberry Pi and the STM32, *pyserial* [9] is an existing library that enables UART communication between the two. This library enables Python to interact with devices that communicate via UART.
   b) **Data Handling**
   CSV [4] is a standard Python library used for making CSV formatted files from data. This will be crucial in transferring raw test data into CSV format. *Pandas* [8] provides higher-level data handling, enabling structured row/column storage for CSV files. *Openpyxl* [6] enables the generation of Excel files in Python, extending compatibility for XLSX files if required.
   c) **Image Processing**
   Opencv-python [14] captures and stores images from the imaging system at defined time intervals. *Picamera2* [5] enables the camera module in a Raspberry Pi by providing camera control, exposure settings, and frame capturing. *Pillow* [1] provides additional image processing capabilities such as saving, converting, and manipulating images in Python.
   d) **System Utilities** *Asyncio* [2] supports concurrent execution of serial communication, data logging, and image acquisition tasks. *logging* [3] assists with structured runtime logging for debugging and test traceability.
2) **STM32**
   The STM32 micro-controller will be handling real-time control of hardware subsystems such as the tray motion, lens clamping, lens flipping, etc. For these actions, the STM32 will be using some already existent Arduino Core libraries. *AccelStepper* [10] is useful to use along side high-level stepper motors, which will be used in actions such as the tray movement and caliper positioning, *Servo.h* [13] provides simple control over servo motors for some of the motions used in this procedure.

C. Scope: Goals & Objectives

1) Basic Goals

- The system will measure the focal length of the test lens.
- The system will measure the transmissive and reflective spectra of the test lens.
- The system will manually measure the test lens’ geometric features.

2) Basic Objectives

- The white light source will be built.
- The white light source will be collimated.
- The monochromator assembly will work with at least one of the mirrors.
- The image acquisition system will record an image.

3) Advanced Goals

- The data produced by the system will be output in a plain text format.
- The data produced by the system will be recoverable with a flash drive.

4) Advanced Objectives

- The monochromator assembly will work with both mirrors.
- The system will derive optical values from the measured geometry.

5) *Stretch Goals*

- The system will have a graphical touch interface.
- The system will generate a datasheet as a PDF.
- The datasheet output by the system will be accessible by QR code or printed document.
- The code will estimate the test lens' product number by comparing the specs to a known product database.

D. *Features/Functionality*

For the advanced scope the following inputs will be recorded or calculated:

- Geometric Properties
  - Diameter
  - Center Thickness
  - Front Radius of Curvature
  - Back Radius of Curvature
- Optical Properties
  - Effective Focal Length
  - Normalized Transmission
  - Normalized Reflection
- Controlled Variables
  - Light Wavelength
  - Lens Presence

The above will be measured using a mechanical measuring system, monochromator assembly, plane imaging system, and photodiodes.

From these values & related calculations the following will be derived:

- Geometric Properties
  - Edge Thickness
  - Lens Shape
- Optical Properties
  - Index of Refraction
  - Front Focal Length
  - Back Focal Length
  - Power
  - Spectra
  - Substrate
  - Abbe Number
  - Coatings

The system will then generate either a plain text document or datasheet from this derived data and output it to the user.

E. *Table of Specifications*

Tables I & II show the required minimum scope expectations for our device. In the future, the device may exceed the tables' current scope and accuracy.

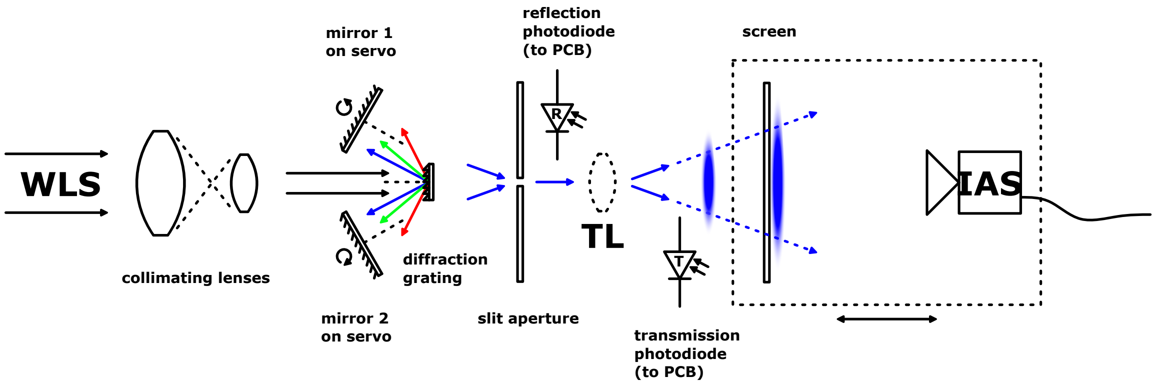


Fig. 1: Optical Schematic

Spec	Minimum	Maximum	Accuracy
Diameter (mm)	20	30	±2
Center Thickness (mm)	5	20	±4
Front Radius of Curvature (mm)	±10	±∞	±2
Back Radius of Curvature (mm)	±10	±∞	±2
Eval Wavelength (nm)	400	700	±10
Effective Focal Length (mm)	±10	±∞	±10
Index of Refraction	1.5	2	±.1
Front Focal Length (mm)	±15	±∞	±10
Back Focal Length (mm)	±15	±∞	±10
Power (1/m)	.1	3	±.3
Abbe Number	30	60	±8
Coatings (nm)	400	700	±10

TABLE I: Numerical Specs.

Spec	Allowed	Prohibited
Lens Types	Plano Cocave, Plano Convex, Double Concave, Double Convex, Negative Meniscus, Positive Meniscus	Aspheric, Achromatic, Compound,
Substrates	Glass, Plastic	Metal, Ceramic, PTFE, etc...

TABLE II: Categorical Specs.

F. Hardware Block Diagrams

The Automated Lens Parameter Metrology (ALPM) project requires the precise actuation of mechanical components. This is a task that is best achieved through the use of stepper and servo motors. These motors have been chosen due to their ability to provide fine control over angular displacement and torque, which is critical in optical measurement systems where accuracy and repeatability directly influence experimental validity. For this task, a dedicated Printed Circuit Board (PCB) will be designed and fabricated. A custom PCB not only consolidates the necessary electronics into a robust form, but also provides a level of noise immunity and ease of integration with the rest of the ALPM system. At a high level, the PCB can be broken down into four elements, shown in Figures 2 & 3:

1) Main Control Unit (MCU)
2) Stepper Motor Driver PCB
3) 3.3V Digital Voltage Supply
4) 12V-20V Motor Voltage Supply

The MCU serves as the central element that generates the command signals required to operate the motors. Its primary role is to issue step and direction pulses to the motor driver. For this project the STM32F103C8Tx series will be used for its high processing power and low cost (<$2) when manufactured through JLC PCB, a China-based PCB manufacturer. The MCU will manage higher-level control of the driver through pins specific to the a4988 stepper motor driver breakout PCB such as sleep, enable, MS1, MS2, and MS3. Sleep controls whether the stepper driver creates a continuous resistive torque or not. This is useful for limiting power output, since it later leads to heat output. Enable controls whether the stepper driver is on or off, a feature typically used in systems that implement polling for power control. It likely will not find use in our design. However, the PCB design will still feature traces from the enable pin to the MCU in case the function becomes useful later on. The pins MS1, MS2, and MS3 control the micro-stepping capabilities. The specifics of which will be detailed in the stepper motor driver section.

The A4988 stepper motor driver serves as the hardware interface that converts the lowpower logic signals issued by the MCU into the high-current control necessary to operate the motor coils. It achieves this through its integrated dual H-bridge circuitry, which allows it to rapidly switch current between windings in the precise sequence required for stepper motion. The driver also provides current limiting, which is configured through an external reference voltage and sense resistors, ensuring that the motor receives adequate torque without exceeding safe thermal or electrical limits. Additionally, the A4988 supports multiple micro-stepping modes that are selectable through the MS1, MS2, and MS3 pins. These modes enable a stepper normally capable of 200 step revolution to reach up to 3200 step revolution. This does come at the cost of increased power consumption, so the MCU should be capable of toggling these pins. With the a4988 maintaining smooth motor operation, the MCU will be able to manage higher-level control logic with little issue.

The stepper motor supply will be provided through a USB-C input that supports the USB Power Delivery (PD) protocol. Through PD negotiation, the PCB can request voltage profiles in the range of 5–20 V at currents up to 5 A. These levels are sufficient to drive the A4988 stepper driver and energize the motor coils. The MCU and driver logic will be powered by a regulated, low-voltage rail. By isolating the motor supply in this way, the system can benefit from the flexibility of USB-C PD negotiation for high-power delivery without compromising the stability of the control electronics.

Both the A4988 and the STM32F103C8Tx require a clean and stable 3.3 V supply to operate reliably. Because the USB-C PD input can deliver a voltage anywhere up to 20 V depending on system configuration, dedicated regulation circuitry will be necessary to step the input down to logic-safe levels. To accomplish this, two stages of regulation will be employed. The LM7805CT/NOPB, a 5 V linear regulator, will be used to produce a stable 5 V intermediate rail from the variable PD supply. This device will be useful for its easy integration and ability to regulate a large range of voltages. From this 5 V rail, the AMS1117-3.3 low dropout regulator will be used to generate the final 3.3 V supply for the MCU and stepper driver logic. The AMS1117 is capable of excellent line regulation at a steady voltage level, even under fluctuating load conditions. 22 μF decoupling capacitors will be placed at both regulator stages on the inputs and outputs to minimize ripple and suppress transient noise. This dual-regulator design guarantees that the control electronics will remain stable and isolated from the motor supply, protecting the sensitive components.

The ALPM process requires the accurate collection of lens dimensions, which in turn necessitates that each lens be securely lifted and stabilized during measurement. The mechanical design, shown in Figure 5, accomplishes this using three arms that will extend outward to clasp around the lens. It was designed by the team and will be 3-D printed to ensure its dimensions and tolerances meet the needs of the project.

The user will position the lens at the center of the mechanism and adjust a potentiometer embedded in a voltage divider to open or close the arms. The resulting voltage output will be proportional to the arm displacement. It will then be read by the MCU through an Analog to Digital Converter (ADC). From this measurement, the system will calculate the lens diameter while simultaneously maintaining a firm grip on the lens for further analysis.

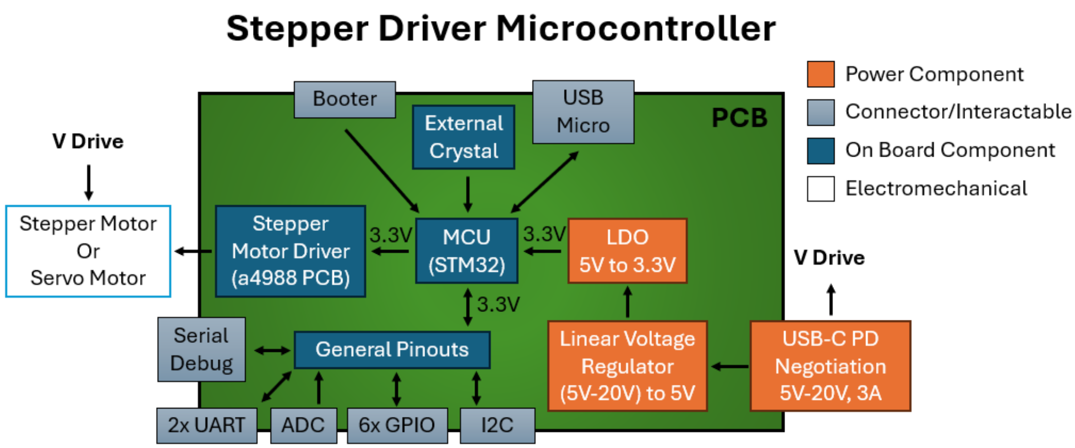


Fig. 2: Stepper Driver Overview

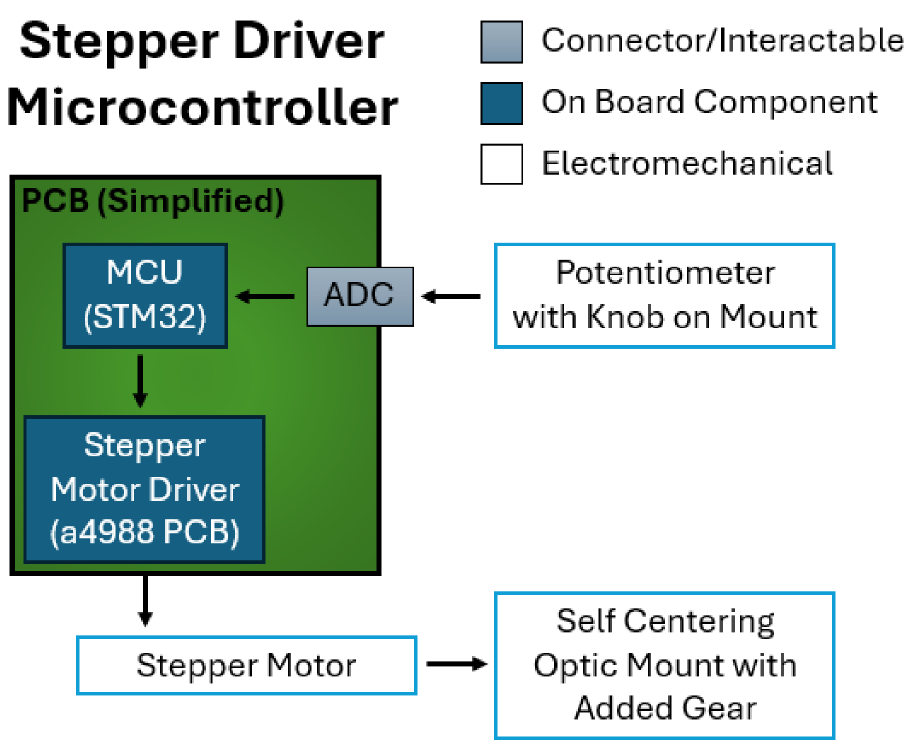


Fig. 3: Stepper Driver Detail

G. Optical Block Diagrams

Figures 1 & 4 depict the different optical systems/components:

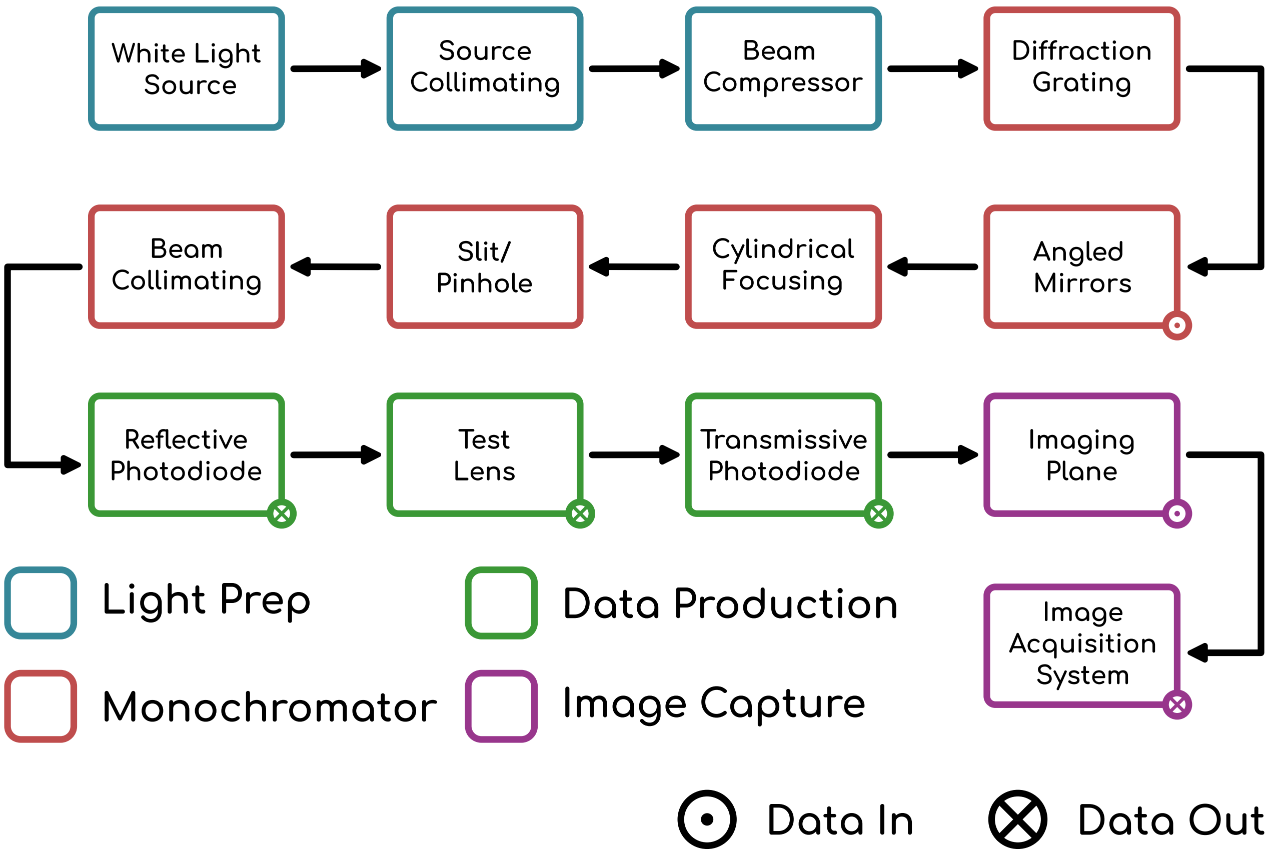


Fig. 4: Optical Block Diagram

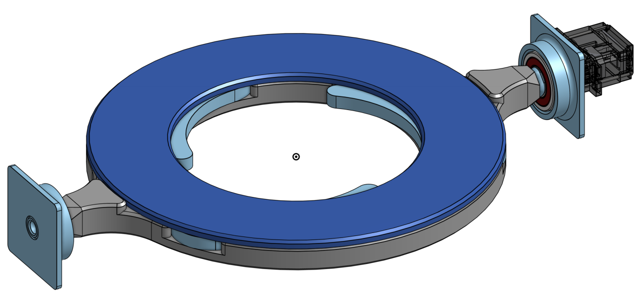


Fig. 5: Self-Centering Mount Rendering

White Light Source: A high power Xenon gas discharge tube source adapted from a car headlight. The light source itself is a previous project by Kiva McCracken. The system will be adapted & integrated into the optical assembly.

Source Collimating: A Xenon bulb that is effectively a point source. In order for the source to be optically worthwhile, it needs to be collimated so that all the rays are parallel. The least expensive &

simplest way to achieve a high-quality beam is with a parabolic reflector, a converging lens, an aperture, and a collimating lens. A diagram can be found in fig. 6.

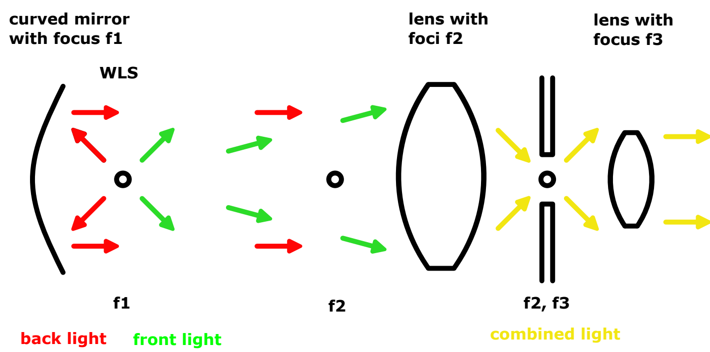


Fig. 6: Collimation Diagram

Beam Compressor: The beam will be fed into two lenses, reducing the beam to a 10mm diameter.

Diffraction Grating: A diffraction grating will split light into its spectrum through interference, creating a rainbow pattern.

Angled Mirrors: Two mirrors are mechanically coupled so that when they are rotated, the maxima of a given wavelength are directed towards the slit/pinhole.

Cylindrical Focusing: A cylindrical lens is used to compress the vertical spread produced by the diffraction grating and focus as much light as possible into the slit/pinhole.

Slit/Pinhole: The slit/pinhole blocks the unwanted wavelengths and acts as a monochromatic point source.

Beam Collimating: A lens placed one focal length away from the slit/pinhole collimates the light into a tightly collimated monochromatic beam.

Reflective Photodiode: A PCB consisting of multiple photodiodes (with different wavelength ranges) will allow broadband evaluation of intensity. The sensor will be positioned off-axis with regard to the lens so that any light reflected by the test lens can be evaluated.

Test Lens: This is the test optic. The supported lens types can be found in Table II. It will be held by a self-centering holder which will also measure the diameter of the optic.

Transmissive Photodiode: This is similar to the reflective photodiode, but positioned after the test lens. It measures the intensity of transmitted light through the lens.

Imaging Plane: The imaging plane is effectively a projector screen that can be viewed from the opposite side. It sets up the macro-scale evaluation of the lens spot size by using a relatively long focal length and simple image acquisition assembly. The imaging plane will jog toward or away from the test lens to evaluate the spot size over distance.

Image Acquisition System Also known as IAS. The system consists of a series of imaging lenses and a CMOS/CCD sensor to the main board. The IAS is fixed in relation to the imaging plane.

H. Software Block Diagrams

The software flow for the optical testing machine has been designed to provide clear sequencing of all measurement and data-handling tasks. The process will begin with an initialization step in which the Raspberry Pi 3 will orchestrate system startup and the STM32 will send the first command to extend the tray. Once the lens is placed and confirmed, the tray will retract and secure the lens with the self-centering mount. Displacement data from the mount will then be read by the STM32 subsystem and transmitted over UART to the Raspberry Pi, where the lens diameter and radius will be calculated.

Following lens placement, the caliper system will measure the center thickness and then the radius of curvature on the first surface of the lens. The self-centering lens mount will subsequently be rotated 180 degrees to allow for measurement of the second surface curvature. These results will be transmitted to the Raspberry Pi for aggregation.

The next phase will involve optical characterization using a light source and imaging system. The imaging plane and camera will be positioned at the starting point of the test track. At each position, a servo mechanism will adjust the light source to transmit different wavelengths sequentially through the lens. For each wavelength, the system will record the spot measurement, photo diode readings, and capture an image using the image acquisition system (IAS). After data capture, the imaging system will advance 10 mm along the track and repeat the loop until the full lens area has been tested.

Captured images will be processed in Python on the Raspberry Pi to extract spot size and other relevant optical properties. All measurement results, including lens geometry, curvature data, light transmission characteristics, and image-derived features, will then be consolidated and formatted into a CSV file. This file will represent the complete dataset for the test run. In the current implementation, the CSV will be stored locally on the Raspberry Pi for further analysis or manual transfer.

The software architecture is divided into functional domains: kinematics/servo control, sensor acquisition, data handling, and communication. The STM32 microcontrollers will manage real-time kinematics and sensor data collection, while the Raspberry Pi will aggregate the data, process the image, and store the output. Communication between devices will be done over UART, which will ensure reliable data exchange.

This structured workflow provides a clear sequence of operations and a modular software design. By separating low-level hardware control from high-level data handling, the system will achieve both precise measurement control and robust data management, creating the foundation for future expansions such as automated networking and integration with external applications.

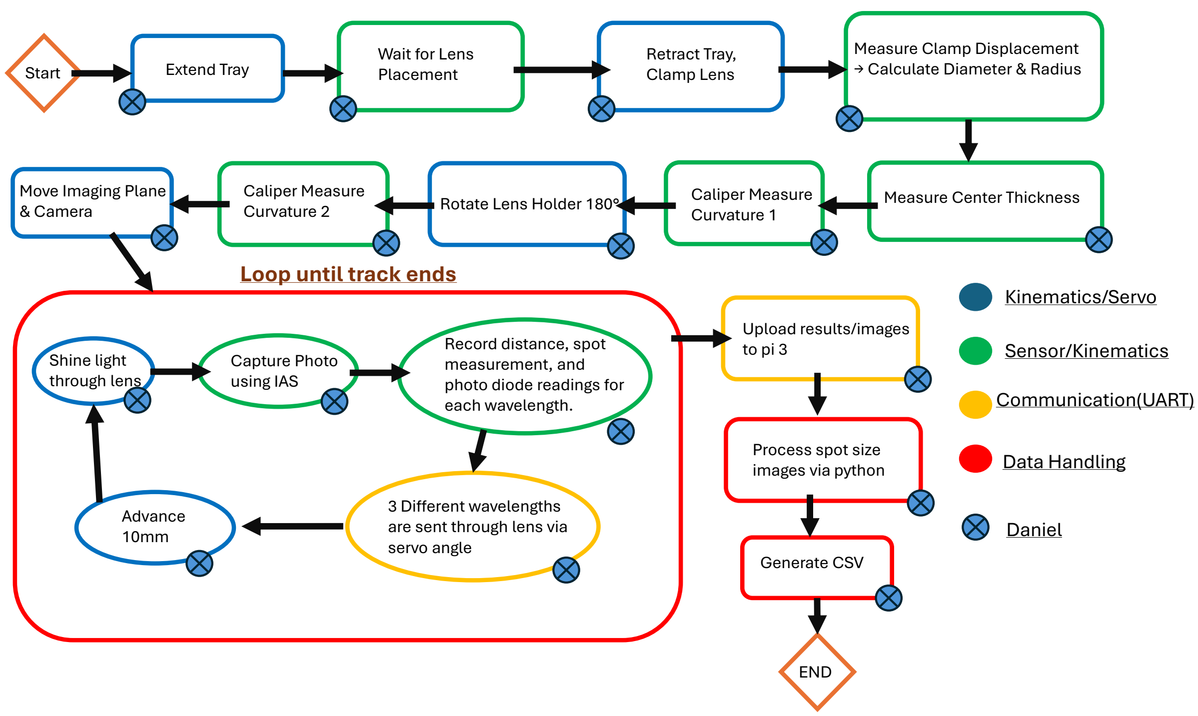


Fig. 7: Software Block Diagram.

I. Prototype Illustrations

The illustration visible in Fig. 8 shows a concept of how the final design may look. It features an aluminum extrusion enclosure to keep out light, the lens holder, the beam & screen, and the IAS. In the back of the enclosure lies the WLS & monochromator.

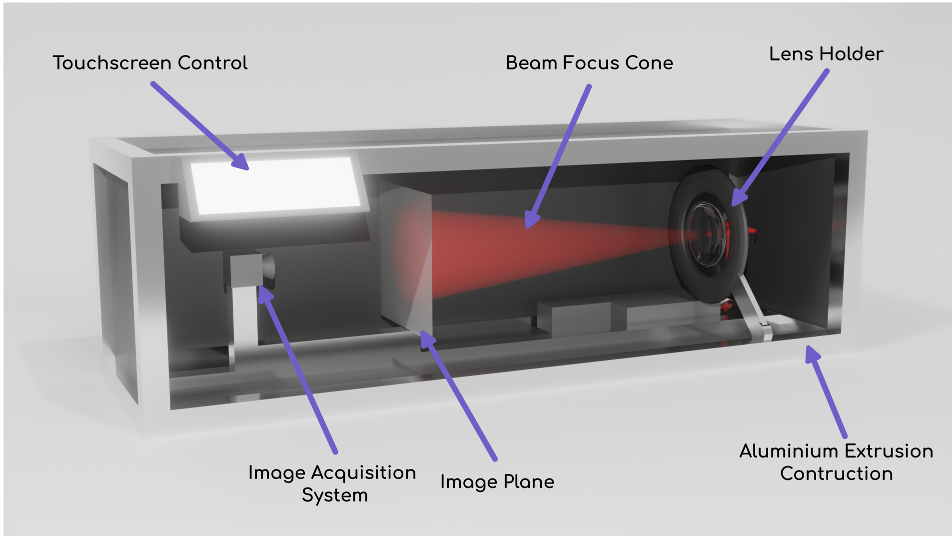


Fig. 8: Concept Rendering

II. Administrative Content

A. House of Quality

A house of quality (visible in Fig. 9) is a planning system that allows the comparison of customer needs to engineering requirements, and helps in balancing goals to fit.

B. Budget

We are aiming for a sub $1000 system. These numbers do not include already owned materials, or those provided by sponsors. The breakdown by category is below:

- Electronics $400
- Mechanics $200
- Optics $300
- Replacement parts and other misc. $100

C. Bill of Materials

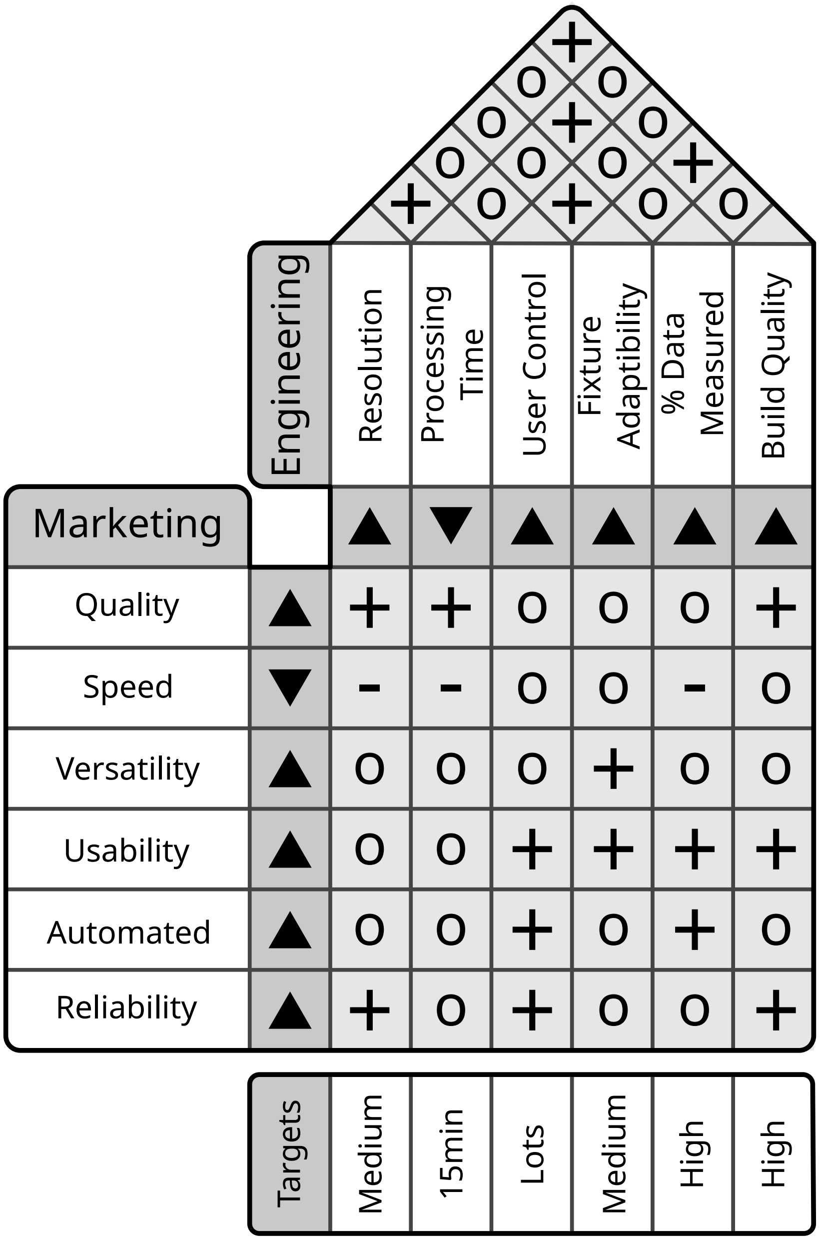
Tab. III is an overview/estimate of the materials needed.

Optics	
...	$...
...	$

Electrical	
...	$...
...	$

Mechanical	
...	$...
...	$

TABLE III: Bill of Materials



+/- indicates correlation, ▲/▼ indicates desire

Fig. 9: House of Quality

D. Milestones

The milestones are visible in Tables IV & V.

E. Task Distribution

Our team consists of two Photonic Science and Engineering (PSE) majors, an Electrical Engineering (EE) major, and a Computer Engineering (CPE) major. The tasks are divided by specialization in Table VI.

Milestone	**Start Date**	**Completion Date**
Group Formation	8/19/25	8/28/25
Project Idea	8/19/25	8/28/25
Committee Formation	8/28/25	9/5/25
D&C Completion	8/28/25	9/5/25
D&C Meeting	9/10/25 5pm	9/10/25 5pm
D&C Document Revision	9/10/25	9/12/25
Research Components	9/12/25	TBD
Midterm Milestone Report	9/13/25	10/31/25
Midterm Report Revision	11/3/25	11/7/25
Final Report		11/25/25

TABLE IV: Project Initialization (SD-I)

Milestone	**Start Date**	**Completion Date**
Order Initial PCB	TBD	TBD
Fabrication of Tray and Lens Mount Assembly	TBD	TBD
STM32 Firmware Development and Raspberry Pi Software Development	TBD	TBD
PCB Assembly	TBD	TBD
Optical Light Source and Photodiode Assembly	TBD	TBD
Imaging System Integration	TBD	TBD
System Integration and Debugging	TBD	TBD
Final Prototype Build	TBD	TBD
Final Testing	TBD	TBD
Final Presentation	TBD	TBD

TABLE V: Project Fabrication (SD-II)

Major	PSE		EE	CPE
Group Member	Kiva	Ollie	Zachary	Daniel
Tasks / Responsibilities	WLS, IAS, Optical Data Processing & Mechanical Systems	Diffraction Grating & Monochromator	System Supporting Electronics & PCB	Embedded Development & Data Collection
Alternates	Ollie	Daniel	Kiva	Zach

TABLE VI: Task Distribution

References

[1] Alex Clark and contributors. Pillow: Python imaging library. https://pillow.readthedocs.io/en/stable/, 2024. Accessed: 2025-09-05.
[2] Python Software Foundation. asyncio — asynchronous i/o. https://python-pillow.org/, 2024. Accessed: 2025-09-05.
[3] Python Software Foundation. logging — logging facility for python. https://docs.python.org/3/library/logging.html, 2024. Accessed: 2025-09-05.
[4] Python Software Foundation. Csv file reading and writing. https://docs.python.org/3/library/csv.html, 2025.
[5] Raspberry Pi Foundation. Picamera2 python library. https://datasheets.raspberrypi.com/camera/picamera2-manual.pdf, 2024. Accessed: 2025-09-05.
[6] Eric Gazoni and contributors. openpyxl: A python library to read/write excel 2010 xlsx/xlsm files. https://openpyxl.readthedocs.io/, 2024. Accessed: 2025-09-05.
[7] Topcon Healthcare. Solos automatic lens analyzer, May 2024.
[8] NumFOCUS Inc. Python data analysis library. https://pandas.pydata.org/, 2024. Accessed: 2025-09-05.
[9] Chris Liechti. Python serial port extension. https://pyserial.readthedocs.io/, 2024. Accessed: 2025-09-05.
[10] Mike McCauley. Logging facility for python. https://www.airspayce.com/mikem/arduino/AccelStepper/, 2024. Accessed: 2025-09-05.
[11] Newport. Kbx022ar.14 bi-convex lens, 2025.
[12] Edmund Optics. Eco-550 glass datasheet.
[13] Arduino Project. Servo library. https://www.arduino.cc/reference/en/libraries/servo/, 2024. Accessed: 2025-09-05.
[14] OpenCV team. Opencv-python: Open source computer vision library. https://opencv.org/, 2024. Accessed: 2025-09-05.
[15] ThorLabs. N-bk7 plano-convex lenses, 2023.
[16] ThorLabs. Storage, 2023.
[17] Trioptics. Stationary thickness gauge, 2015.
[18] Zygo. Micro lens process metrology inspection equipmnent, 2025.