

Motorizing Optical Mounts With 3D Printing



Bo Blackburn

Abstract

As part of an ongoing effort to replace costly optomechanical equipment with 3D printed parts, this project aims to implement the ability to control the positioning of a motorized optical mount with Arduino-driven software.

This goal was accomplished by using three stepper motors to control the vertical and horizontal deflection angles as well as the position of a kinematic mirror mount on the optical axis. These motors were attached via 3D printed attachments connected to two manually controlled screws situated diagonally on the mirror and to the micrometer head of a linear translation stage. These two pieces of equipment were acquired secondhand, and all three stepper motors (5V 4-phase stepper motor 28BYJ-48, 1/64 reduction gear, step angle $5.625^\circ \times 1/64$) were controlled with the use of two sketches (programs) and Arduino microcontrollers. The 3D printed mounts were designed with the CAD software Fusion 360, and simultaneous control of the kinematic mirror motors was achieved using a joystick as a human interface.

The complete motorized mount was tested to assess the function of all three degrees of movement. All three motors driving both the mirror's tilt as well as the linear translation stage work as expected, with a smooth and responsive control interface. The increment of the mirror's deflection angle was found to be 0.05° per step at a three-foot distance, measured by reflecting a collimated laser beam off the mirror to a perpendicular surface. This experiment was also used to test the system's stability and repeatability by shifting the mirror's tilt to its maximum and returning it to its starting position; this did not result in any observable displacement of the laser.

The successful implementation of a motorized mirror mount with three axes of movement for less than 150 USD demonstrates the capability of 3D printing technologies to create complex optical systems at an economic cost. The measured precision of this system is sufficient to control optical lasers/waves and suggests its potential application in interferometric experiments.

Introduction

This project's history begins with the creation of a white light Michelson interferometer in the work titled "*Creating an Economical Interferometer Through CAD Engineering*," (AJAS 2020)¹. This was a simplified system implemented to demonstrate the feasibility of replacing conventional optomechanical components with 3D printed alternatives while still relying on industrial-grade moving parts for precision. This led to a focus on creating kinematic mounts using the same economical methods the following year, titled "*Optical Engineering with Economical 3D Printed Components*" (AJAS 2021)². The overarching goal of both these projects was to create inexpensive alternatives to what is available on the market through professional-grade sources for lower-budget educational settings. The current investigation maintains this goal while adding a new focus, the motorization of the kinematic mount and the automation of the measurement process. By integrating mechanics such as stepper motors into kinematic designs, a new level of consistency can be achieved that is impossible to replicate by hand due to the variability of human error. Furthermore, this would facilitate systematically taking rapid measurements at regular intervals as well as the creation of an active control system to keep the system aligned under stress. The experimental system for the work reported here is a white-light Michelson interferometer. This was chosen as the target application because the amount of precision required is at the scale of wavelengths of visible light ($\sim 500\text{nm}$), which provides a meaningful assessment of the specificity of the mirror's control system. Michelson interferometers superimpose separated halves of an optical wavefront to form a visible interference pattern. White light interferometry requires precise control over three axes of motion to observe an interference pattern; the superimposed wavefronts need to travel in the same path of travel (Figure 1). For a Michelson-type interferometer, this implies the need to control the tilt of one of the mirrors in the system as well as its position along the optical axis.

The following details the combination of inexpensive mechanics, 3D printing, and second-hand optical components for the purpose of creating an economical motorized interferometer system that is sufficiently accurate enough to control the operation of the applied optical system.

¹ Blackburn, J. (2020, February 13-16). *Creating an Economical Interferometer Through CAD Engineering* [Conference presentation]. AJAS Annual Meeting, Seattle, WA, United States. <https://onedrive.live.com/?cid=6AA47BA455714236&id=6AA47BA455714236%211807&parId=6AA47BA455714236%21170&o=OneUp>

² Blackburn, J. (2021, February 8-11). *Optical Engineering with Economical 3D Printed Components* [Conference presentation]. AJAS Annual Meeting, Held Online/Remotely. [file:///C:/Users/jobob/OneDrive/Documents/BlackburnAJAS2020\(21\)Poster\(ACTUAL\).pdf](file:///C:/Users/jobob/OneDrive/Documents/BlackburnAJAS2020(21)Poster(ACTUAL).pdf)

Materials / Methods

item desc.	manufacturer	man. part #	unit price	qty ordered	total price
Stepper Motor, Drive Module Board	Aoicrie	28BYJ-48, ULN2003	\$ 4.50	3	\$ 13.50
UNO R3 Board	elegoo-shop	EL-CB-001	\$ 14.00	2	\$ 28.00
Joystick Module	SainSmart	101-50-136	\$ 7.00	1	\$ 7.00
20 mm Mirror Mount	WaveDynamic	N/A	\$ 30.00	1	\$ 30.00
Linear Micrometer Stage	New Focus	N/A	\$ 60.00	1	\$ 60.00
PLA Filament	prusa3d	FLM-PLA-175-WHT	\$ 23.00	N/A	\$ 2.02

This system is primarily made up of two parts: the kinematic mirror mount and the translation stage. The former is meant to change the tilt of a mirror on the x/y axes to direct the reflection of light. The latter is meant to drive the mirror forward through a spectrum to help automate measurements of a wave's displacement.

Two pieces of the interferometer's calibration componentry design were to be automated. One was the micrometer and the other was the tilting mirror. 3D printed mounts were designed to allow the adjustment screws (which were previously manipulated by hand) to be controlled by stepper motors and drivers through programmed Arduino microcontrollers (Fig. 2). The CAD software Fusion 360 was used to sketch out then create full-scale models of what was then uploaded to PrusaSlicer and exported as an STL file to be replicated on a Prusa Mini 3D optics plate on detailed print settings (Fig. 3). Coding of the Arduinos was undertaken to move the micrometer forward and back with one motor and to control the x/y movement of the tilting mirror mount with two additional motors which change the mirror's tilt from opposite corners (Figs. 4 & 5). These pieces were then wired and mounted to an optical plate and the motors attached with 3D printed mounts to allow the movement of the micrometer back and forth to allow for the ability to take precise measurements by shifting the mounted mirror through a spectrum using the micrometer and for the angle of the tilting mirror to be controlled using a joystick.

Two methods were used to test the quantitative results of this investigation, both to test the mirror's smallest possible degree of tilt as well as the consistency with which it can be adjusted. The first was done by reflecting a laser against a wall then shifting one motor 1000 steps, marking the start and end points. Using trigonometry, the angle of that movement can be found as well as that of one step (dividing by 1000). The second is done by moving the mirror over several sequences of 1000 steps, reversing each action so that the laser ideally ends up back where it was. The results of these methods are discussed below.

Results / Discussion

This investigation is the culmination of three years of optical research through which an economical, mechanically driven translation stage and kinematic mirror mount were created to combine the precision of programming and the inexpensive nature of 3D printed and secondhand optical mounts in a test interferometry system. The primary qualitative result is that the motors driving both the mirror's tilt and the micrometer's spin work as expected. The movement achieved is smooth and of high quality. Many inexpensive motors tend to get stuck, wobble, or even have some slippage (within the gears or design), and the fact that this is not happening is a testament to the quality of the system's control. The primary quantitative result of this investigation is the precision of the mirror's deflection, as the smaller degree of tilt this system can achieve, the better. Through the method detailed above, it was found that the smallest possible degree of movement that this motor could achieve was roughly 0.05346° per step (the minimal interval using this model of stepper motor). This arrangement of motors also increases the consistency of movements with a simple command, such as moving a motor a certain number of steps. The method of such testing is listed above. As well as such direct commands, a joystick was implemented to control both motors (screws) at once, with its 'x' axis controlling one screw and its 'y' axis controlling the other (Figure 6). The fact that the screws are placed in opposite corners of the mirror means that the mirror cannot move on either axis with solely one screw, so controlling both at once was the ideal solution. As for the overarching goal of conserving the expense of optical systems, most of this system's cost comes from the secondhand optical components and the mechanical components listed above. The included Bill of Materials provides the individual price(s) and details of each item included in the system. The total cost of creating this optomechanical system is \$140.52, but the cost alone provides little information without a frame of reference. Professional-grade systems can be found online that offer a similar function with higher quality mechanics, though lacking the customizability that a 3D printed design offers. This investigation contributes a system that can capitalize on that same flexibility, combining inexpensive components with 3D printing and customizable programming to create an economical alternative that can approximate the quality of professional-grade equipment.

As for future work, the primary path forward is to complete the collection of quantitative data by integrating the system into an interferometer to test the inter/intrareliability of the system, which was predicted to increase with the addition of the motorized controls. Additionally, the motorization of this optical system could serve to move 3D printed optics out of the classroom and into more serious applications such as amateur investigations. Integrating this system with the interferometer would allow for a more thorough evaluation of this system's fine control and flexibility.

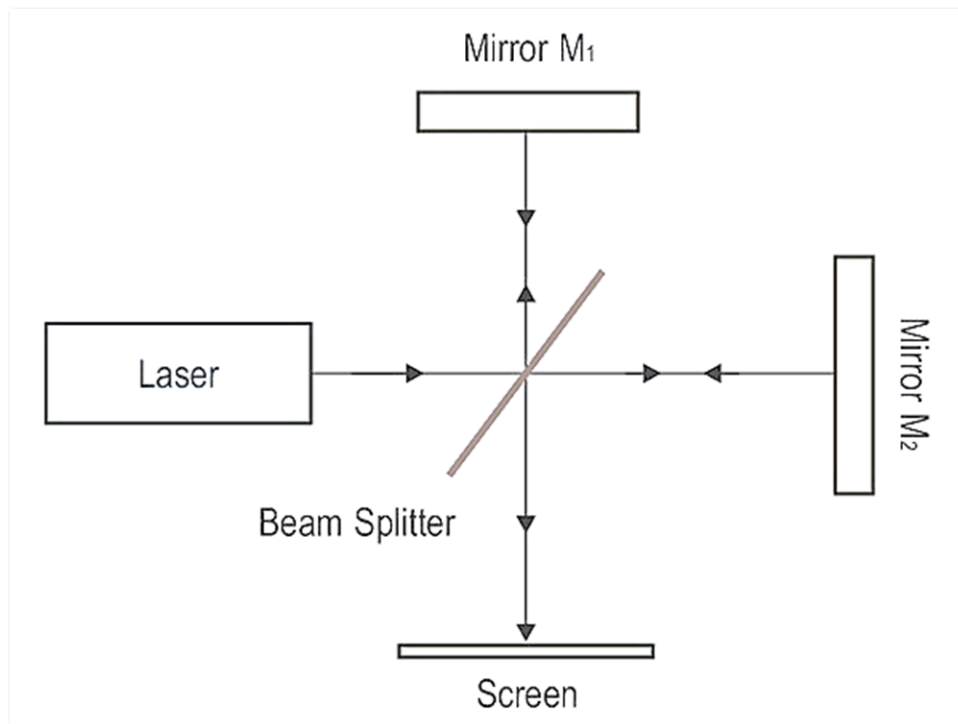


Figure 1

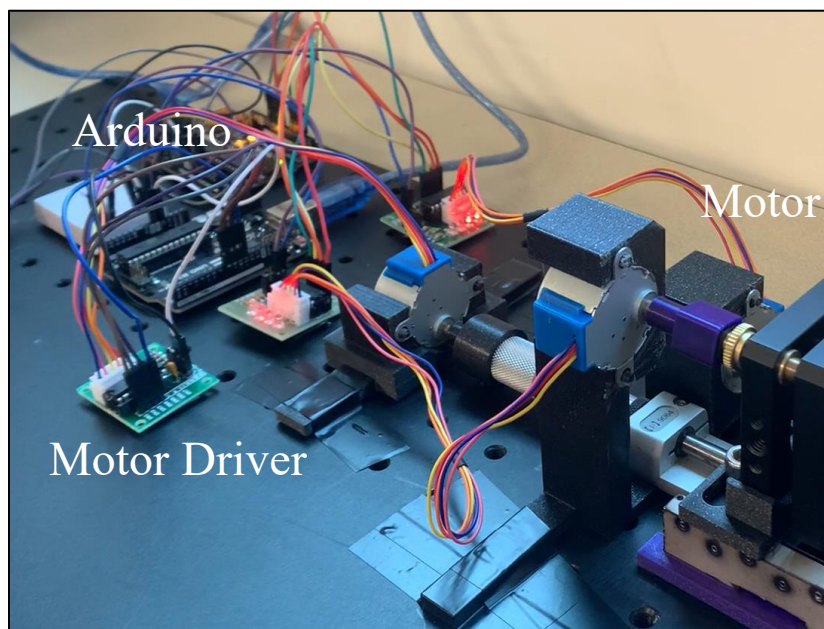


Figure 2

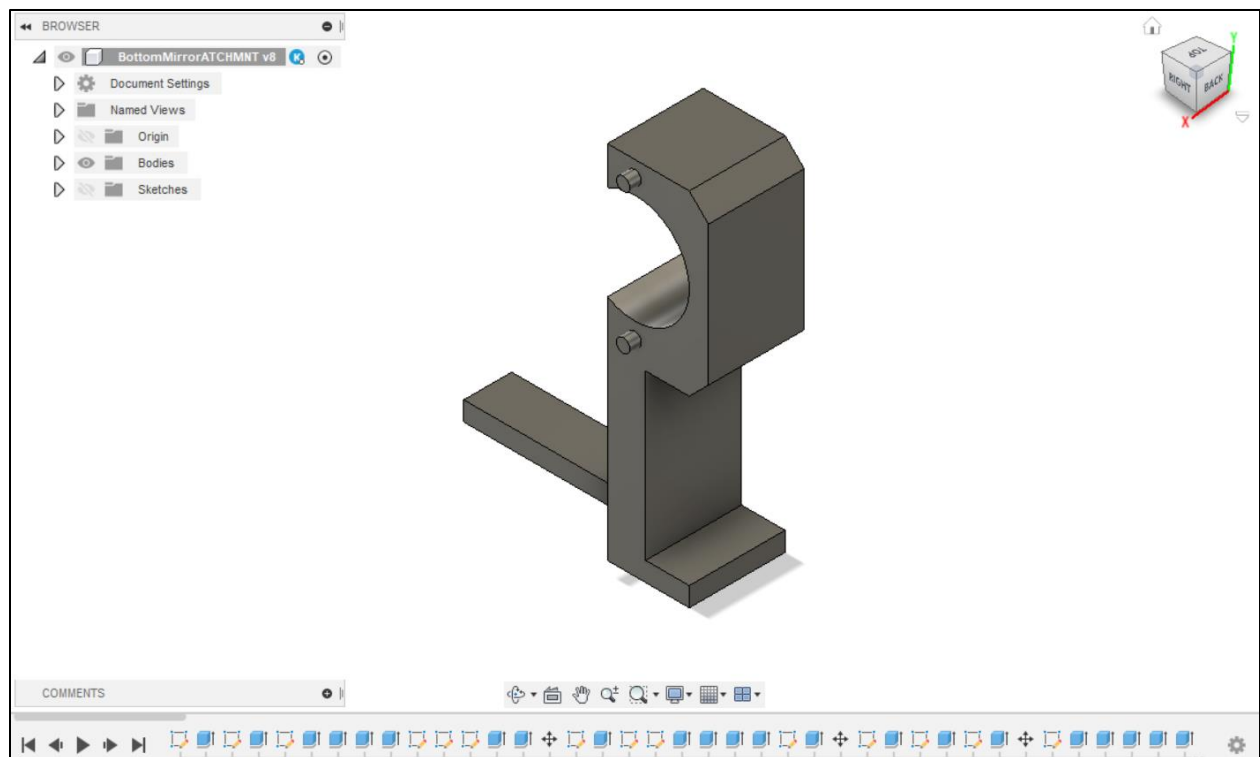


Figure 3

```

#include <Stepper.h>

const float stepsPerRevolution = 40.76; // is 1/50th, total step count is
2038. 'float' allows decimal, 'int' rounds down. 40.96 for 2048 steps.
// ((steps/rotation)/360)*(desired degrees) = # of steps needed.

// initialize the stepper library on pins 8 through 11:
Stepper myStepper(stepsPerRevolution, 2, 4, 3, 5);
int x = 1;
int stopNumber = 50;

void setup() {
  // set the speed at 60 rpm:
  myStepper.setSpeed(60);

  // initialize the serial port:
  Serial.begin(9600);
}

void loop() {
  Serial.println("counterclockwise");
  Serial.println(stepsPerRevolution); // the number from 'stepsPerRevolution'
  myStepper.step(stepsPerRevolution);

  if (x > 0)
  {
    myStepper.setSpeed(60);
  }
  if (x == stopNumber)
  {
    myStepper.setSpeed(0);
  }
  Serial.print("steps:"); // Serial Monitor Command: CTRL + SHIFT + M.
  Serial.println(x); // turn count, prints after the step.
  x = x + 1; // x++; // adds 1 to the print script.
  delay(500); // pause for 0.5 seconds between turns.
}

```

Figure 4

```

#include <Stepper.h>
#define STEPS 2038 // the number of steps in one revolution of your motor
(28BYJ-48)
Stepper stepper(STEPS, 8, 10, 9, 11);
Stepper stepper1(2038, 2, 4, 3, 5);
int potState = 0;
int pot1State = 0;
void setup() {
    Serial.begin(9600);
}
void loop() {
    potState = analogRead(A0); // y
    pot1State = analogRead(A1); // x

    Serial.println(pot1State);
    stepper.setSpeed(5);
    stepper1.setSpeed(5);

    if (potState > 600){
        stepper.step(10);
    }

    if (potState < 400){
        stepper.step(-10);
    }

    if (pot1State < 400){
        stepper1.step(-10);
    }
    if (pot1State > 600){
        stepper1.step(10);
    }
}

```

Figure 5

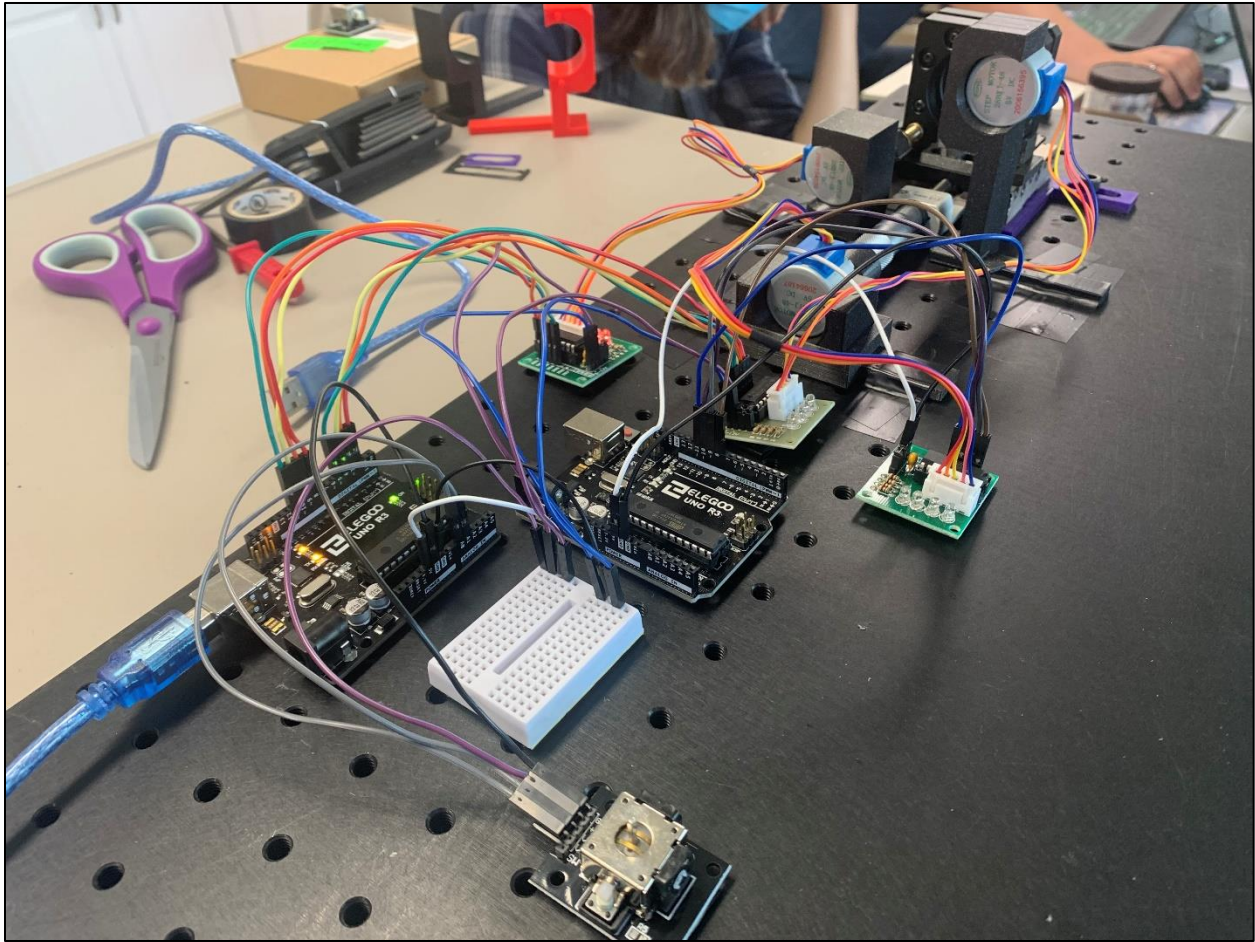


Figure 6