The ArticuLight

Group 9

Scott Bell, EE

Keng Chu, EE

Benjamin Simms, CpE

Michael Taub, EE

Motivation

Current Problems with light sensor systems:

- Light fails to turn on
- Lens of sensor becomes obstructed, dirty or old
- Expensive cost for detecting motion and installation (Approx. $400)

Solution:

- To create a motion-controlled lighting system that will be controlled through hand gestures to move, rotate, and adjust brightness levels of the light for the user to work on projects on workbenches at any time of day.
- Help eliminate shadowing, as well as creating a hands-free device, so the user can use both hands while working on their project.
- We felt this needed to be invented because we feel this will allow for users to finish and produce final projects at a faster rate.

Goals and Objectives

ArticuLight offers:

- Hands-free light for work benches
- Light direction movement through hand gestures in view of camera
- A hand following algorithm and a move-to-hand algorithm for articulating arm with light attached
- Allows for less shadowing when working on a project

Market Requirements

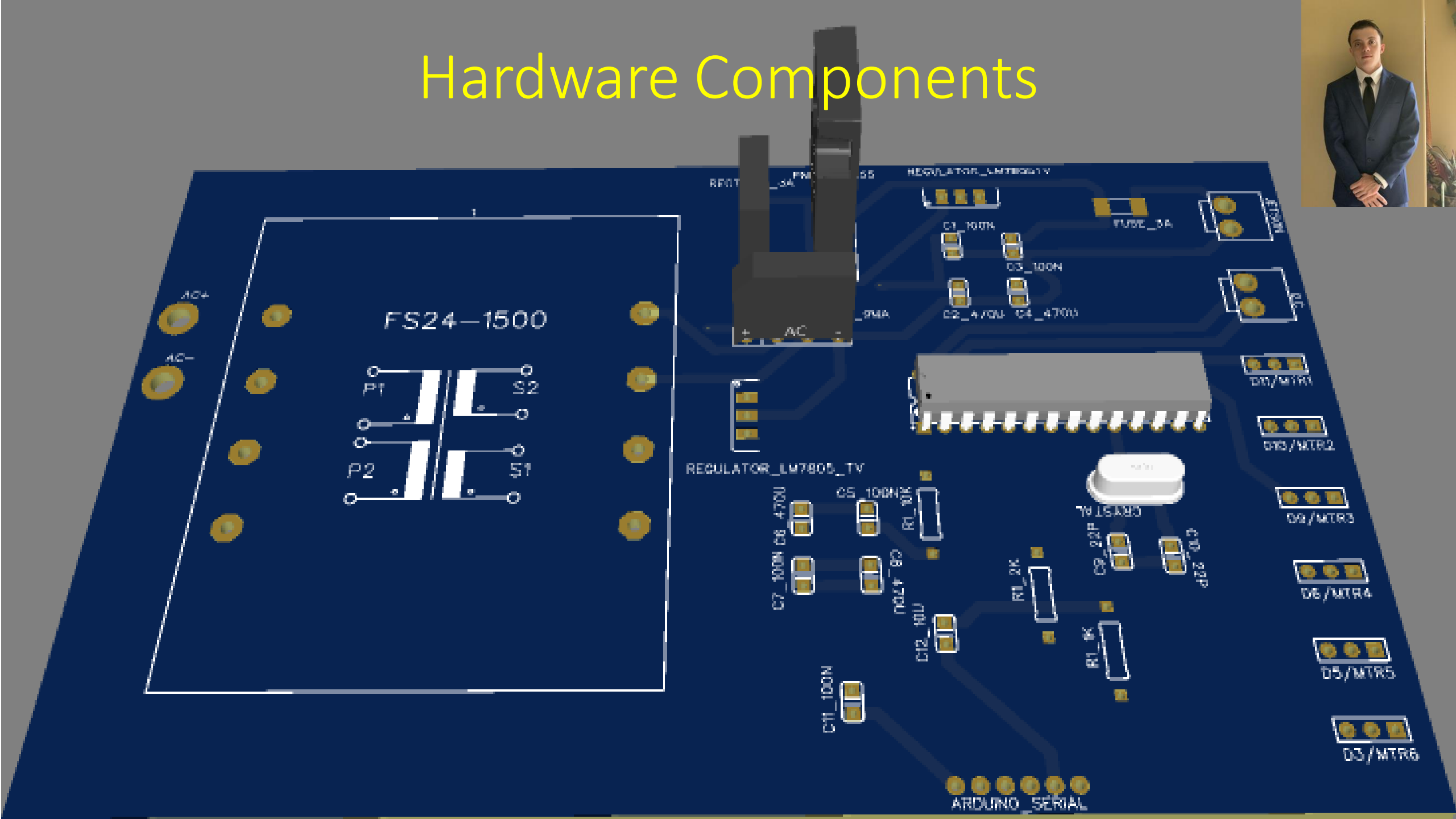
Specifications	Description
Pitch/Yaw and Angle/Degrees of Freedom	- Up to +/- 90° of movement left and right (yaw) - Up to +/- 90° of movement up and down (pitch)
Robot Vision	Will use a camera to identify hand gestures to control the actuation
Stand	Can be mounted vertically or horizontally
Environmental Robustness	Able to handle temperature and humidity variation: - 20° to 100° temperature range - Waterproof and dust proof
Response Time	Software to hardware interface will have a response time less than 3 seconds each
Manual Reset/Zeroize	Device will be able to manually stop and reset the system

Key Specifications

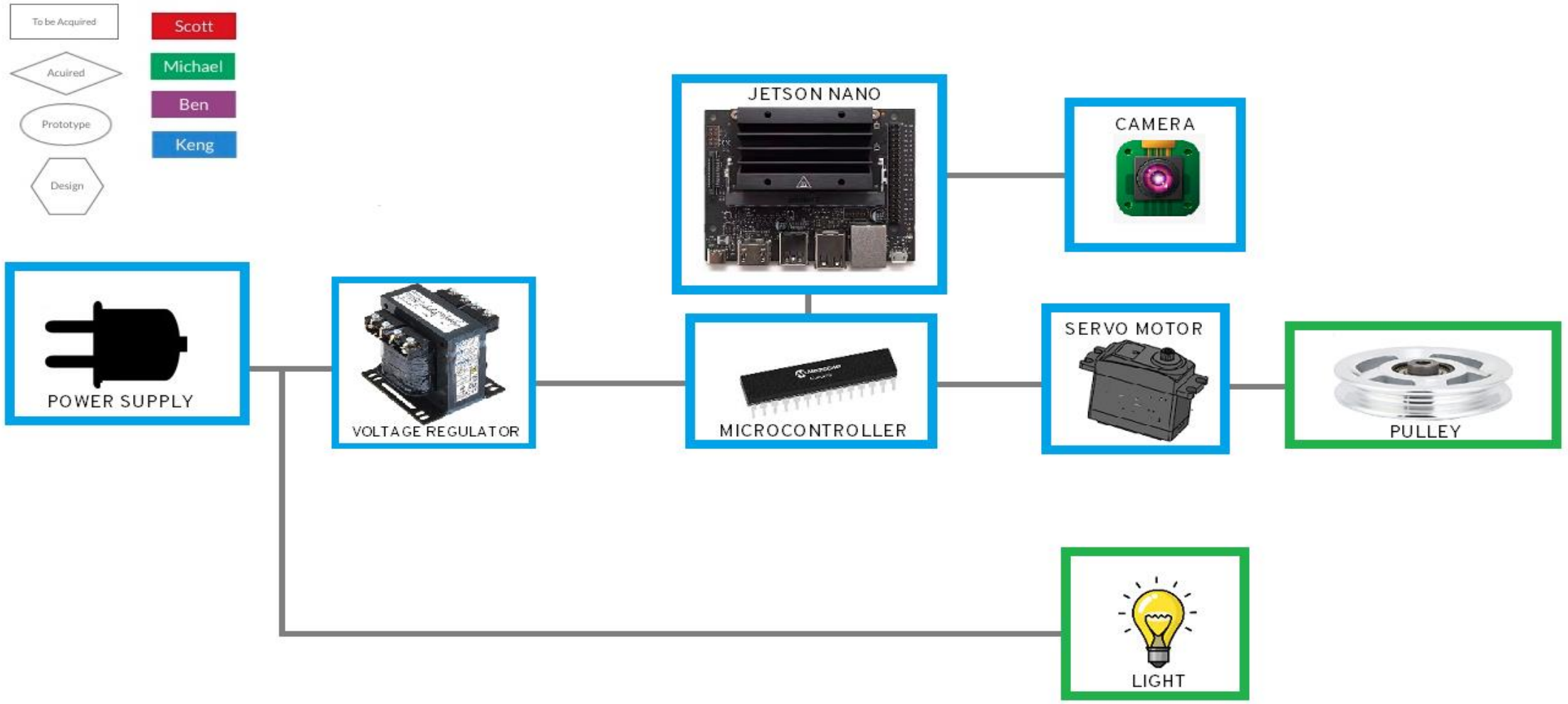
- Accuracy: how close the device will move the light source to focus on specific location.
- Durability: Number of uses and sustained usage of device
- Actuation Time: time taken when computation is finished to complete the movement
- Computation Time: Time it will take the system to read a gesture from the camera and send a stimulus to perform actuation

Engineering Specifications	Value
Accuracy	<= 1 cm
Durability	1000+ uses
Actuation Time	Between 3 – 5 seconds
Power Input	120V AC
Weight	<= 7 lbs
Computation Time	<= 2 seconds
Arm Length	24 inches
Light Voltage	120V

Hardware Components



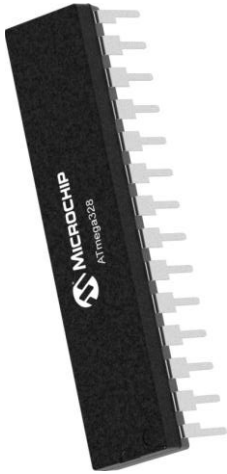
Hardware Block Diagram



Microcontroller

- We chose the ATmega328P microcontroller because we felt it had the best specs for our project and was the most convenient for servo control
- Some of these specs include but are not limited to:
  - Input voltage: 1.8 - 5.5 V - DC
  - Operating voltage: 1.8 - 5.5 V - DC
  - Digital I/O pins: 27
  - PWM channels: 10

MCU	ATmega 328P	MSP430G 2553	STM32G0 31F8P6
Operating Voltage (V-DC)	1.8 - 5.5	1.8 - 3.6	1.7 - 3.6 V
Digital I/O pins	27	83	18
PWM channels	10	8	18
Flash Memory (KB)	32	16	64
Price (USD)	~5	~3	~3.5



Embedded Computer Options

NVIDIA Jetson Nano

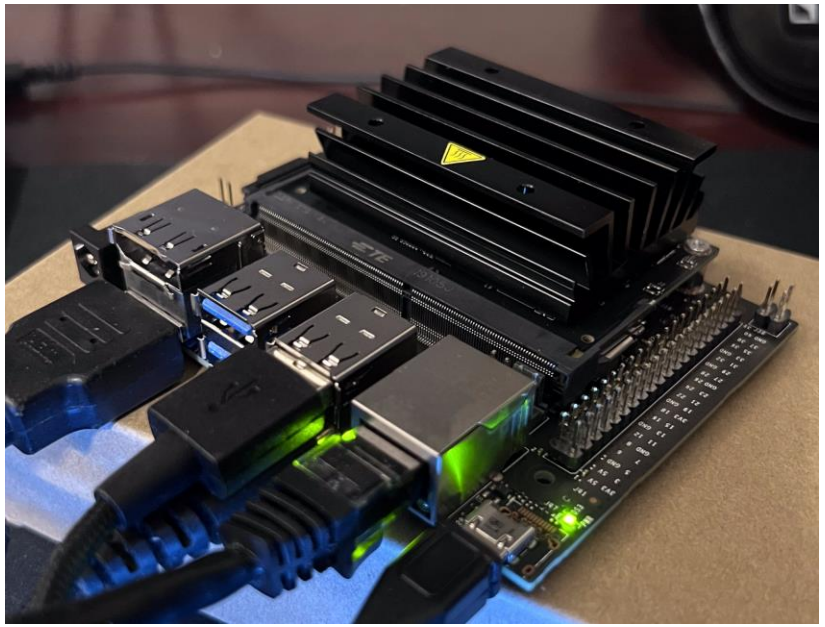
- 4 GB of RAM
- Designed for embedded deep learning such as computer vision
- Parallel AI tasks aid in this goal
- Runs modified version of Ubuntu Linux (Jetpack)

Raspberry Pi 4

- 4/8 GB of RAM
- Can run OpenCV and Mediapipe
- No architecture specifically for deep learning
- Runs modified version of Debian Linux called Raspberry Pi OS

Jetson Nano Developer Kit

- Implementing deep learning such as vision requires a decent amount of processing. The Jetson Nano is marketed for embedded deep learning and was chosen used in this project.
- Features:
  - "Jetpack" - A modified Ubuntu Linux installation
  - 128-core Maxwell GPU for parallel AI tasks
  - ARM A57 CPU
  - CSI camera connection
  - 40 pin GPIO header
  - 5V 2A input, 3.3V pins

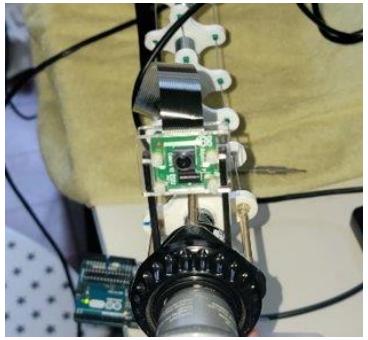


Motor Selection

- We decided on the D956WP servo motor
  - High torque
  - Meets rotation requirements
  - Waterproof

Motor	Operating Voltage (V)	Rotation Angle (degrees)	Max Torque (kg)	Price
HPS-3527SG	5-7.4	270	35	$33.99
DS3218MG	4.8-6.8	270	20	$15.99
DS3225	4.8-6.8	180	25	$20.99
D956WP	4.8-7.4	180	29	$54.52
HST-35H	9-12.6	240	35	$23.99

Camera



Camera	Raspberry Pi Camera Module 2	Raspberry Pi High Quality Camera	Arducam IMX219 Visible Light Fixed Focus Camera	Arducam IMX219-AF Programmable/ Auto Focus Camera	Arducam IMX219 Low Distortion M12 Mount Camera
Image sensor	IMX219	IMX477	IMX219	IMX219	IMX219
Maximum Resolution	8 megapixels	12 megapixels	8 megapixels	8 megapixels	8 megapixels
Horizontal FOV (degrees)	62.2	Depends on lens sold separately	62.2	65	75
Full Resolution Frame Rate	N/A	N/A	21 fps	21 fps	21 fps
720p Frame Rate	60 fps	60 fps	120 fps	120 fps	120 fps
Price	$25	$50	$19	$30	$32

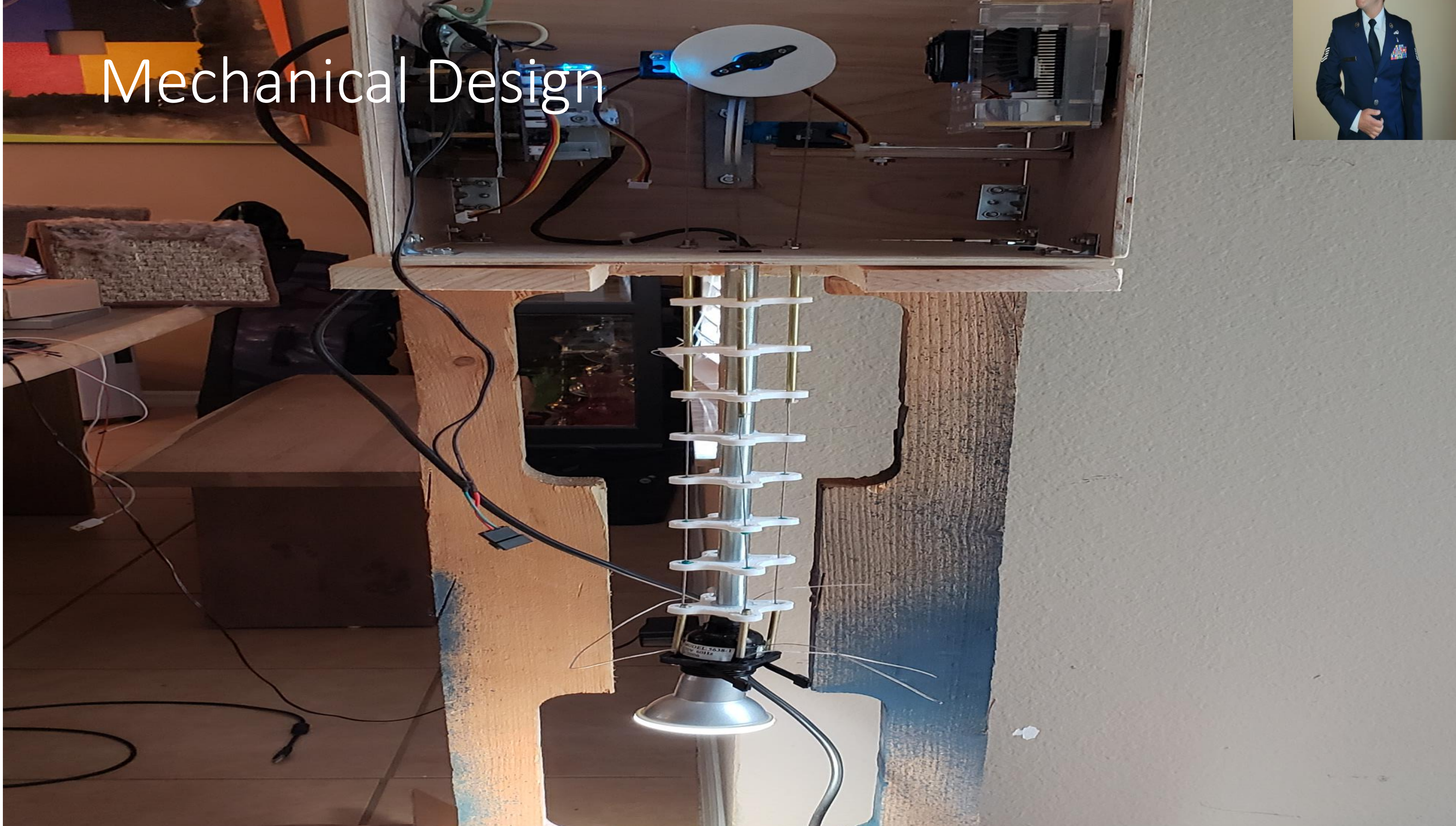
Socket Specifications using an LED Bulb

Light Socket	E-12	E-19	E-26	E-39
Power (W)	75	60	60	660
Voltage (V)	125	125	250	240
Luminosity (Lumen)	400	840	800	36000
Size (mm–diameter)	12	19	26	39
Weight (oz)	6.4	2.04	0.564	8.2

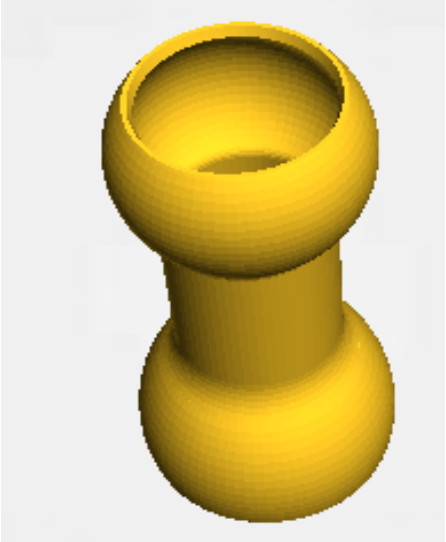
Voltage Regulator

Type of Regulator	Linear Voltage Regulator	Switching Voltage Regulator
Pros	• Low output ripple voltage • Fast response time to load or line changes • Low electromagnetic interference (EMI) and less noise	• More complex design, which allows for handling higher power efficiency • Able to provide output, which can be greater or less than or invert the input voltage
Cons	• Efficiency is low • Requires large space – heatsink is needed • Voltage is above the input cannot be increased	• Higher output ripple voltage • Slower transient recovery time • EMI produces very noisy output • Very expensive

Mechanical Design



Rapid Prototyping - Initial Design of Frame



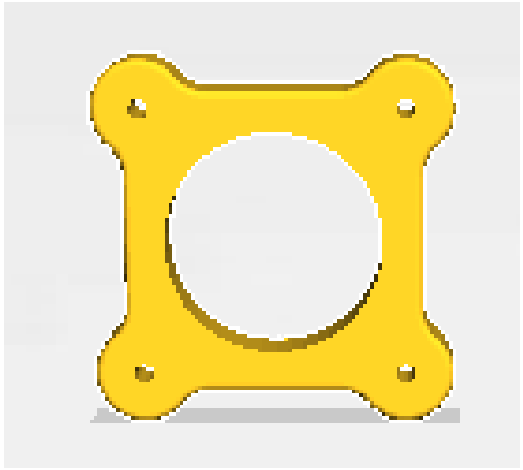
Initial Design of interlocking segments

High degree of mobility

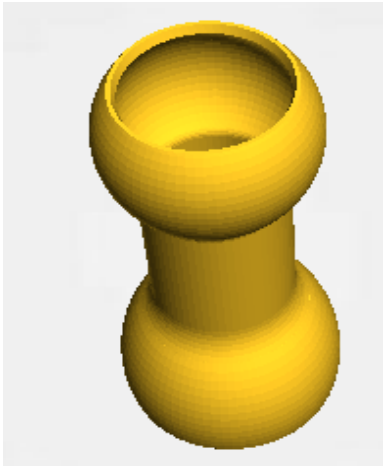
Rigid structure

Light weight

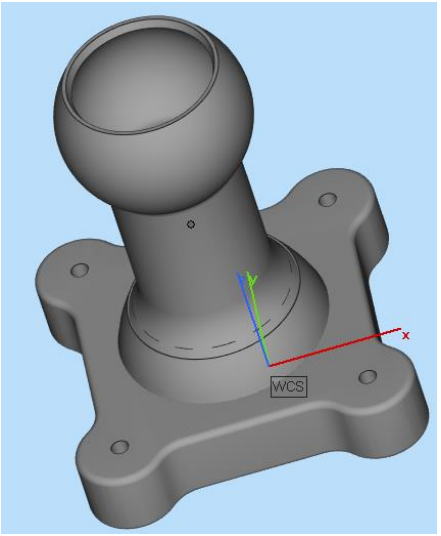
Rapid Prototyping – Wire Guide



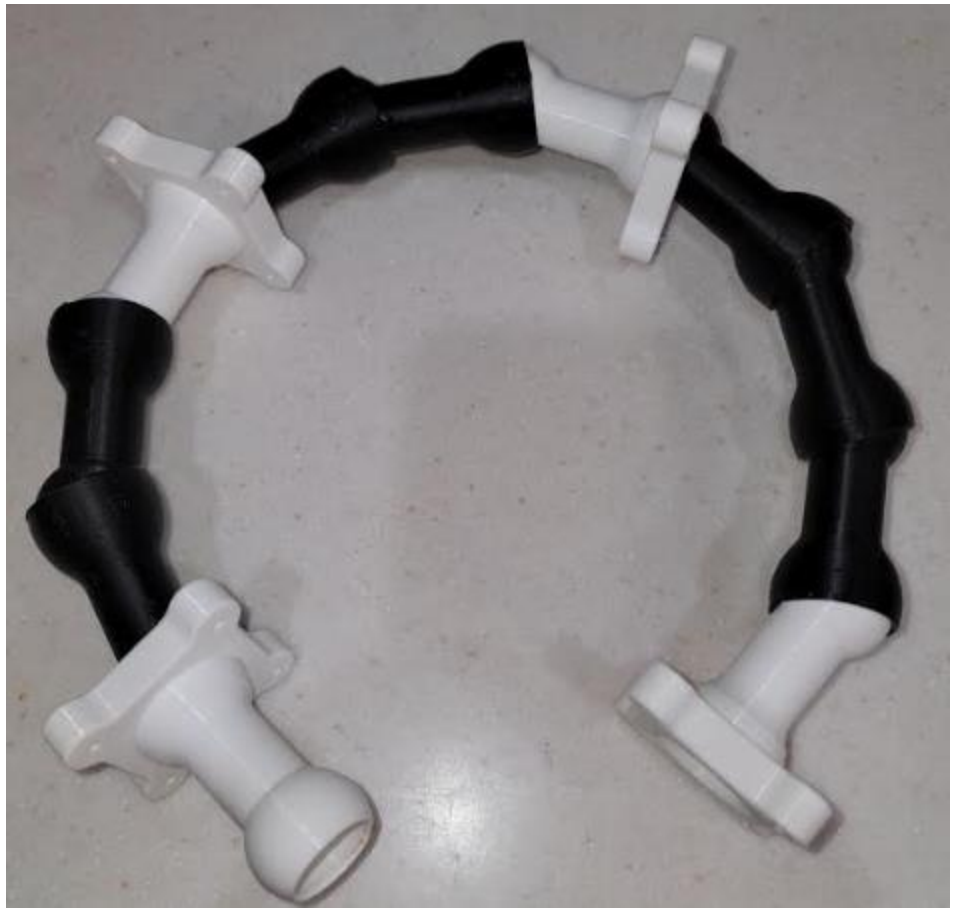
Wire Guide



Interlocking Segments



Rapid Prototyping – Fully Assembled Frame

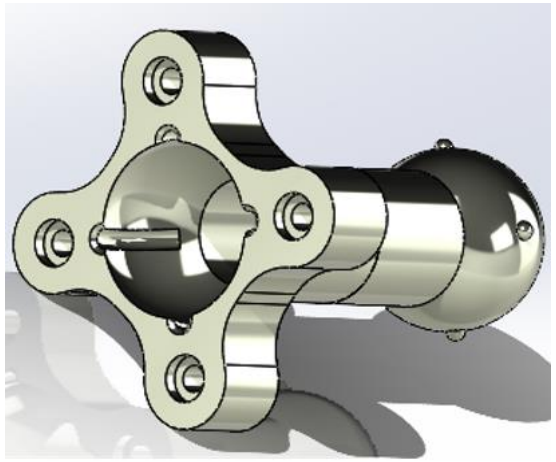


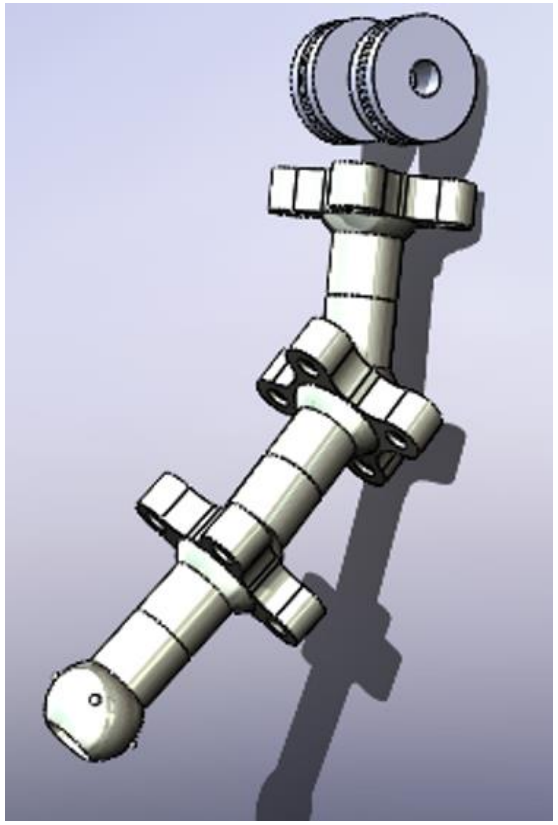
Design Flaws:

- System was too rigid which
- would cause it to crimp when actuated

- System would also rotate during movement, causing the guide wires to wrap around the frame

Rapid Prototyping – New and Improved Design

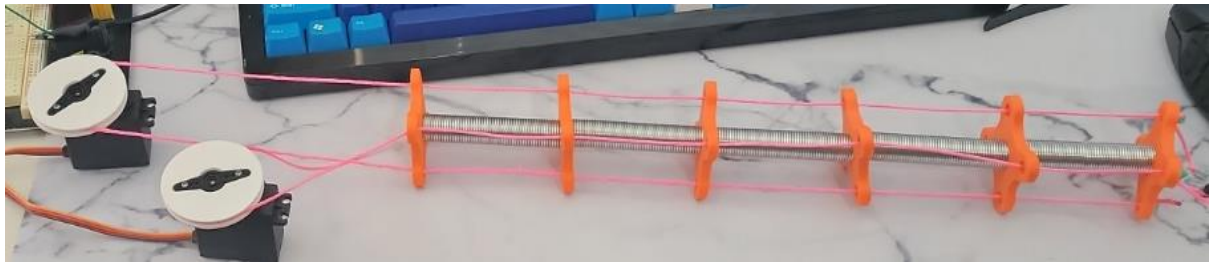




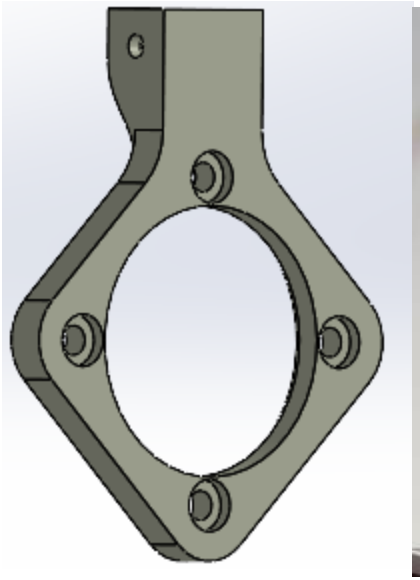
- Dimensions adjusted to allow the connecting pieces to be looser
- Channels added to prevent rotating

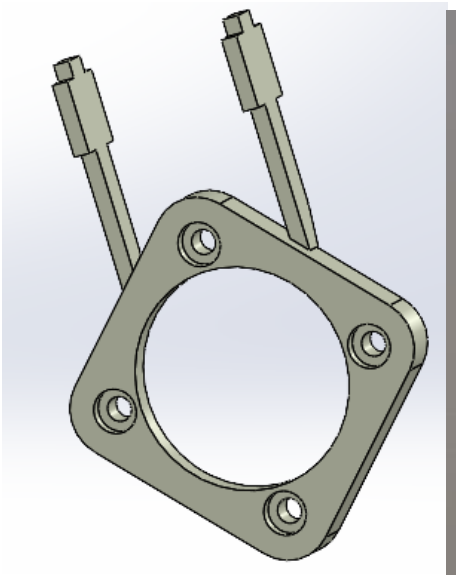
Alternate Design

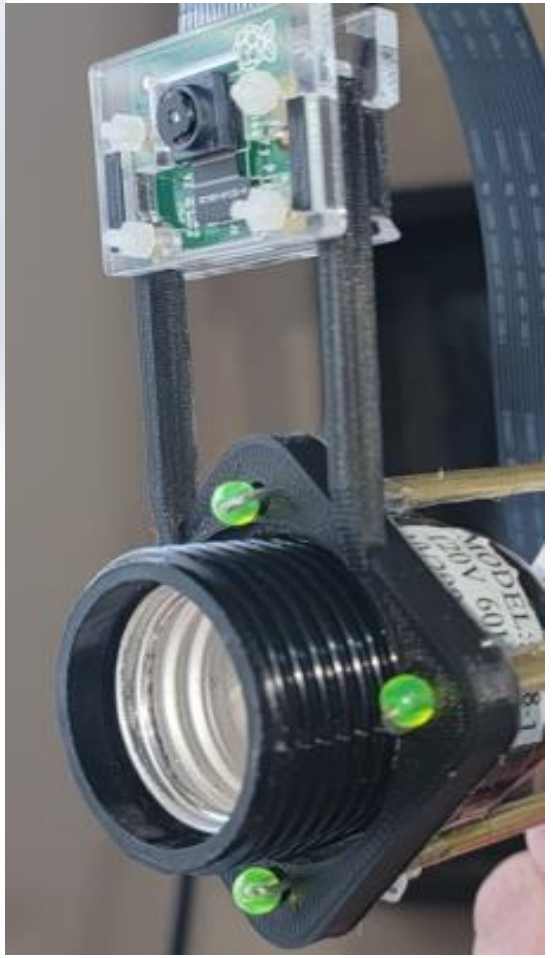
- Modelled as a continuum robotic arm, allows for very flexible movement
- Will allow us to add 3rd degree of freedom which is one of our stretch goal, extension and contraction



Camera Mount

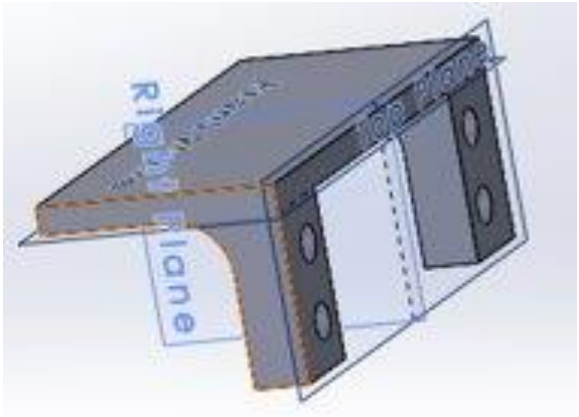


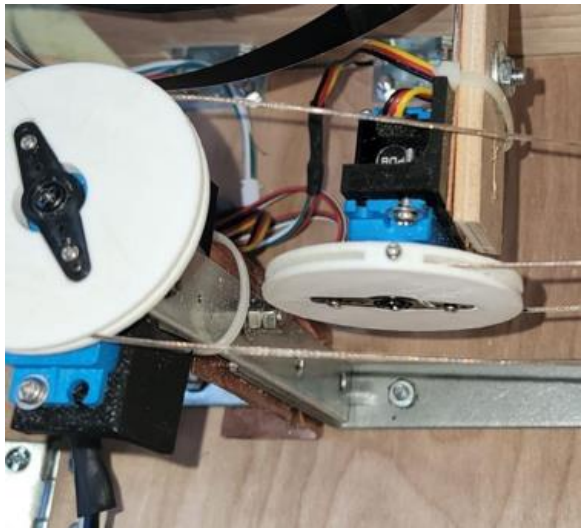


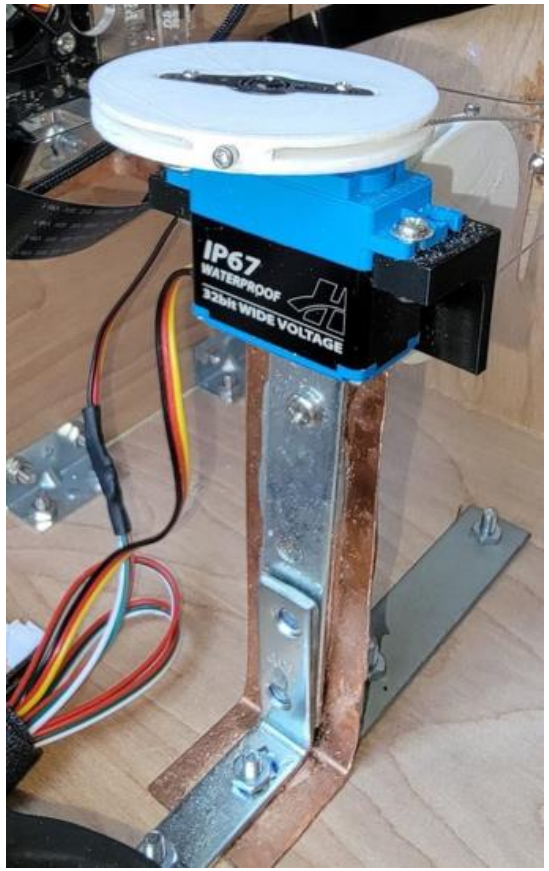


- 2 Versions Created
  - USB Camera
  - Raspberry Pi Camera
- Connects the camera to the light socket

Servo Mount and Stands

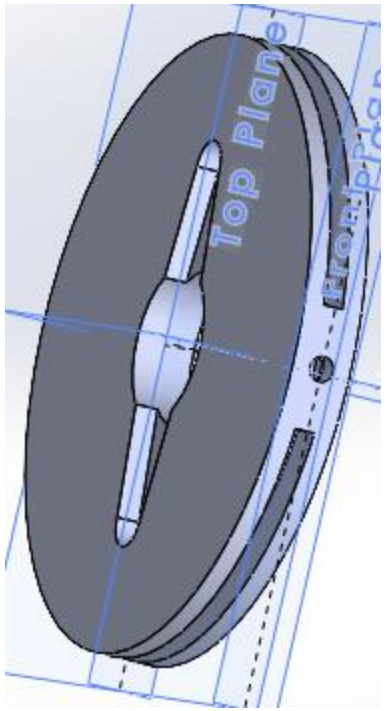




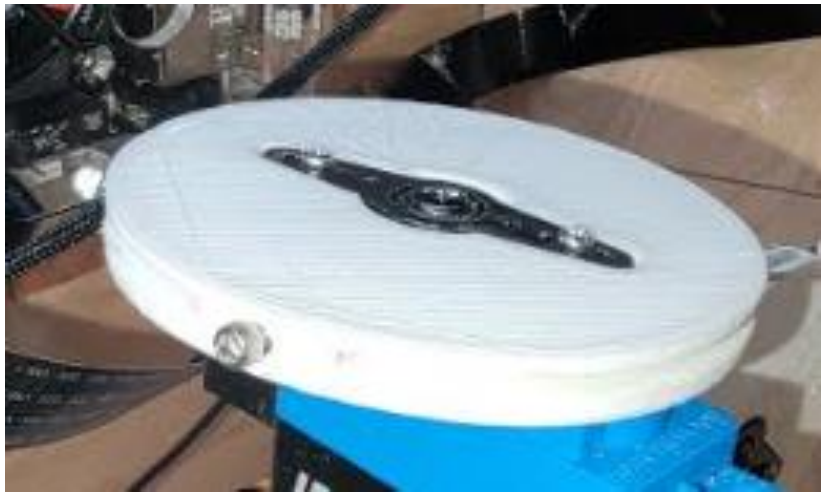


- Mount designed specifically to fit this motor
- Stand use L brackets and copper
  - Provides stability- prevents flexing when actuating
  - Acts as Heat Sync

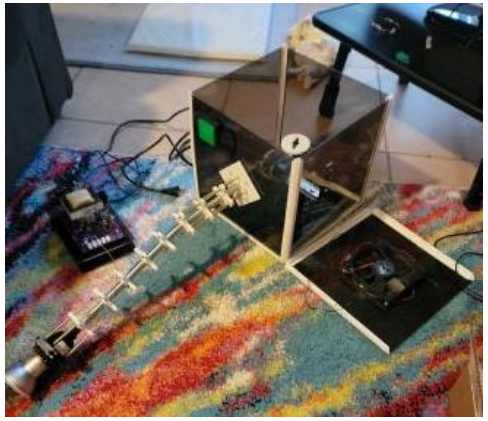
Pulley Design and Cable Selection



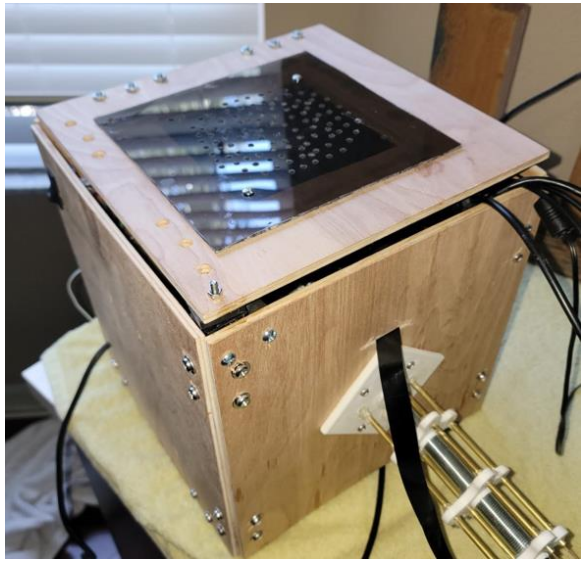
- Has inset articulation arm mounting hole
- Deep channel for wire
- Set screw to hold wire

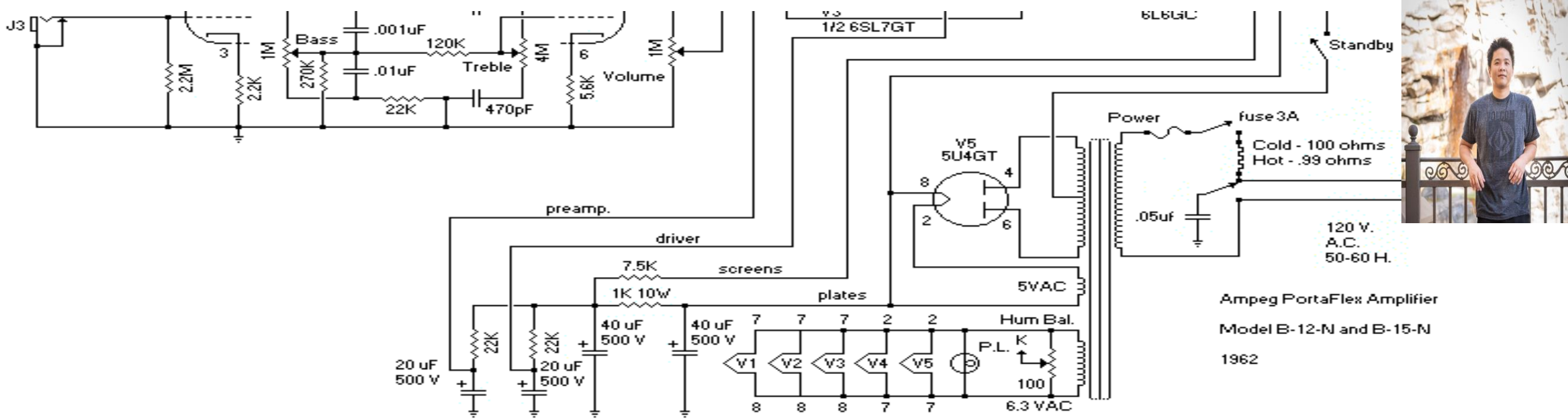


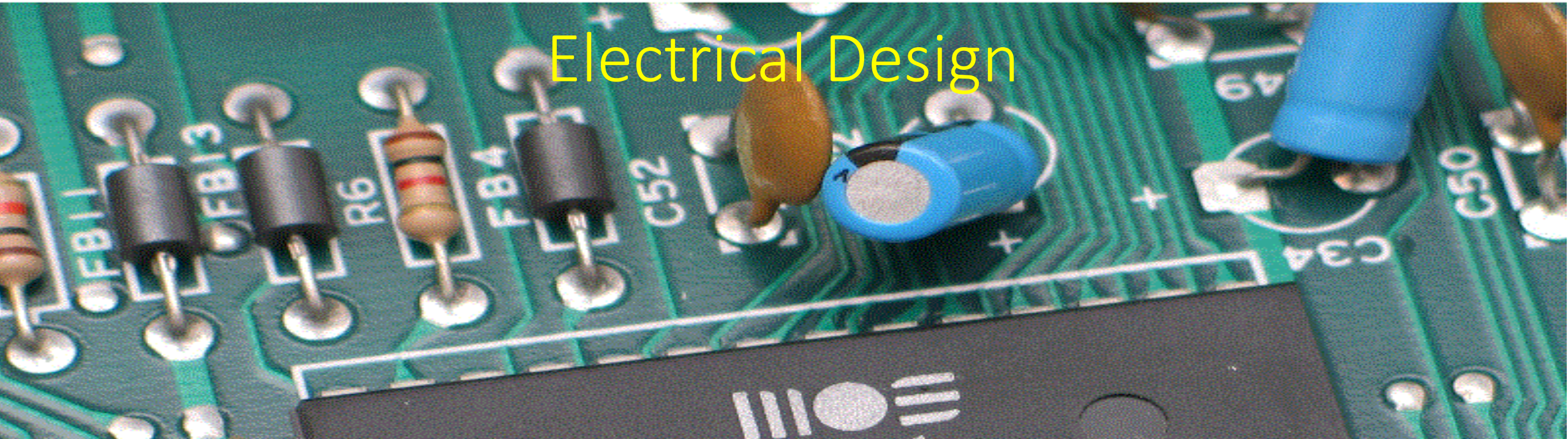
Case/Housing



- Sturdy material
- Double hinge for easier access to components

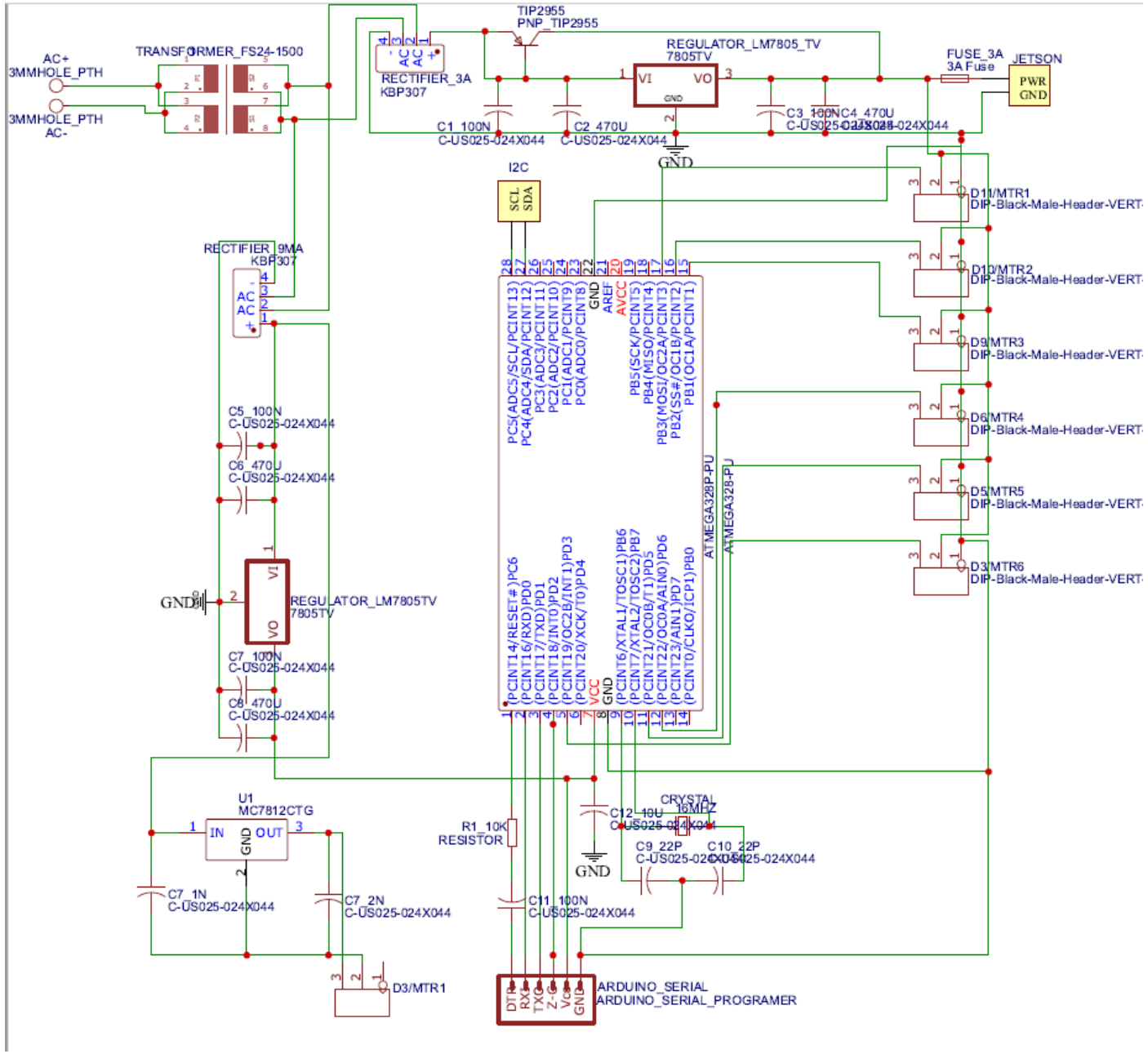




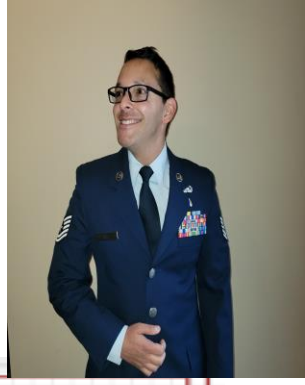


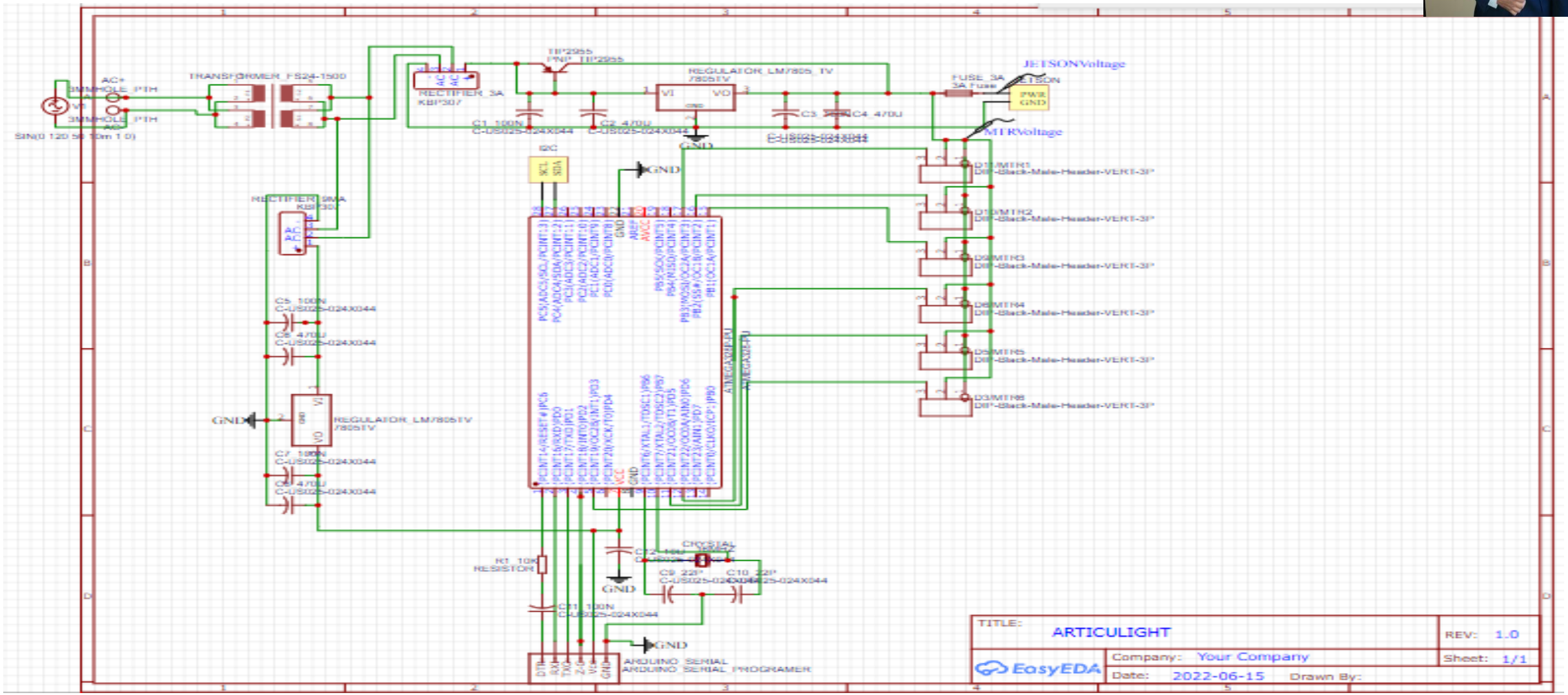
Electrical Design

Overall Schematic

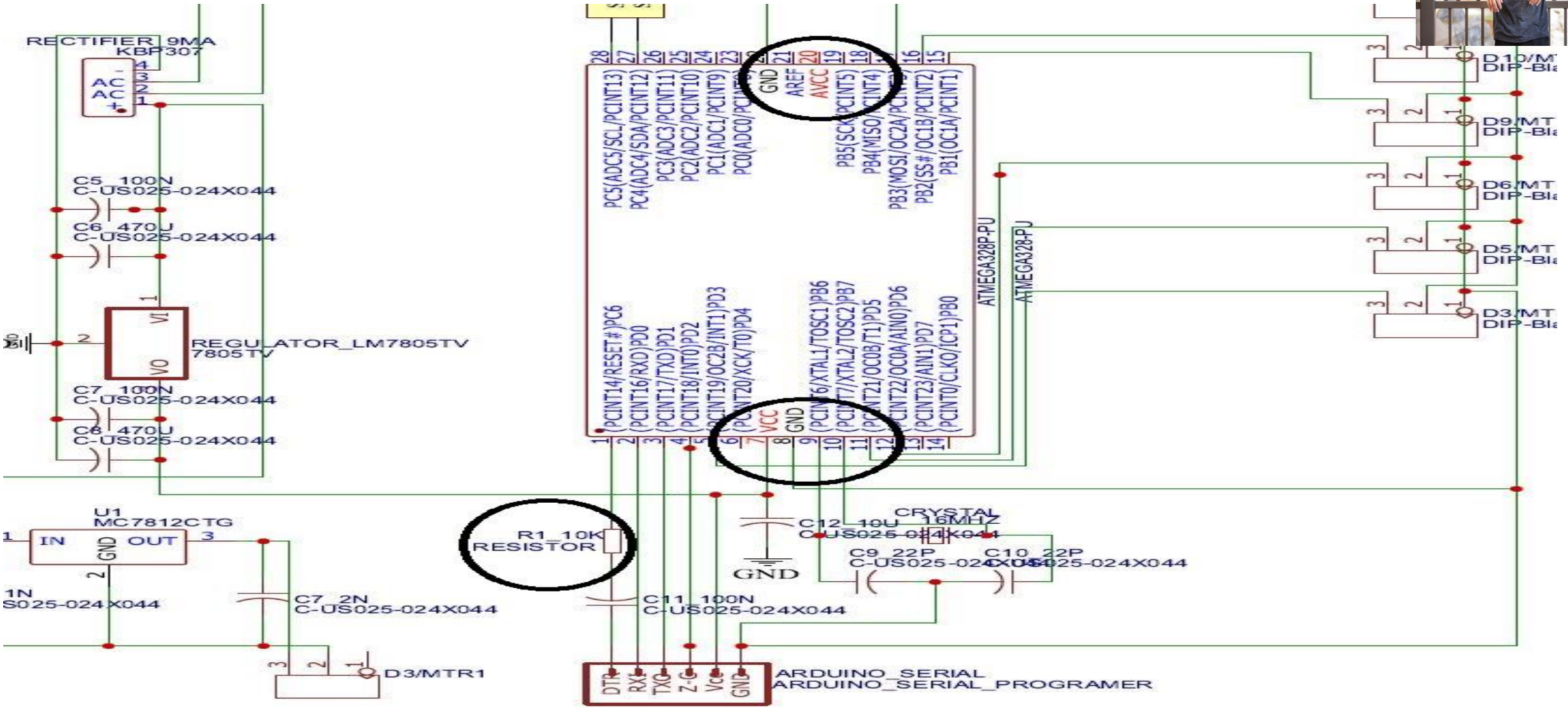


Schematic view of Microcontroller

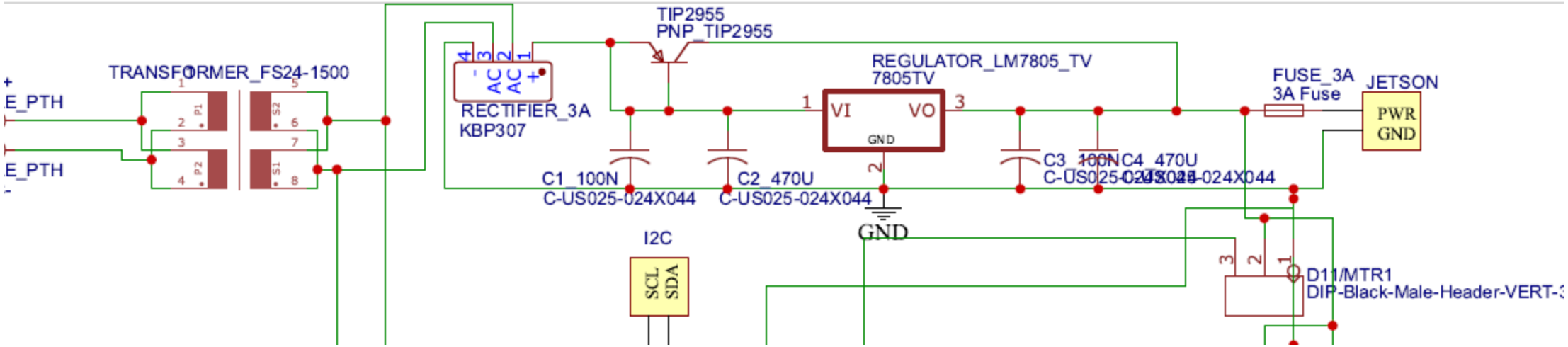




Schematic view of Microcontroller

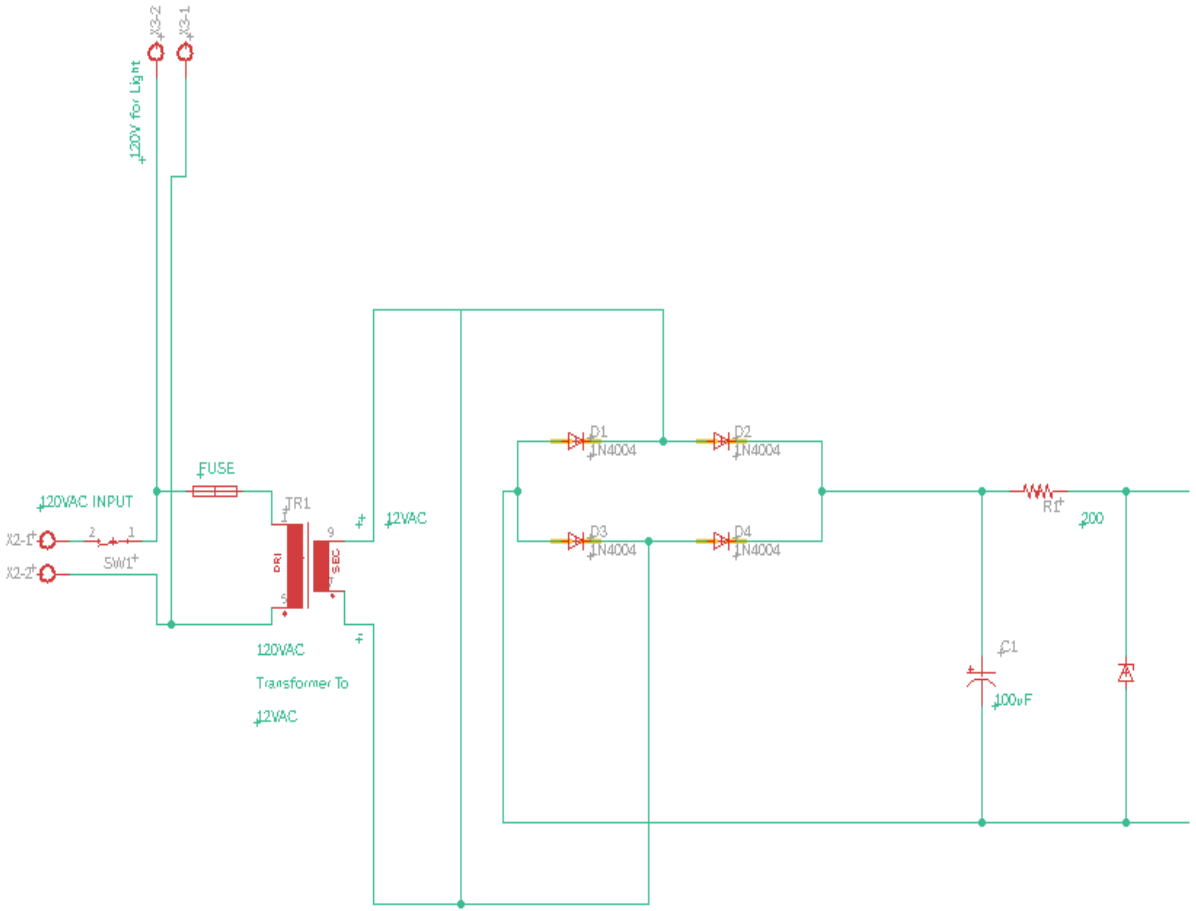


Schematic view of Power Supply/Voltage Regulator



Functionality Hardware (starting from power to voltage regulator)

- Reducing 120V AC to lower voltage
- Converting AC voltage to DC Voltage
- Control Amperage



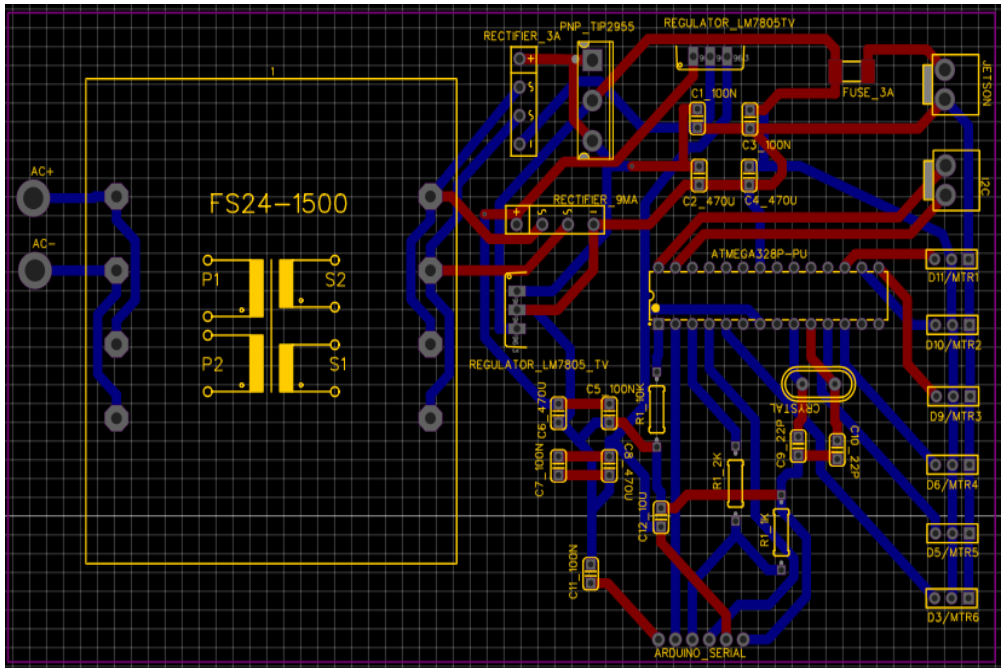
Voltage Regulator Functions

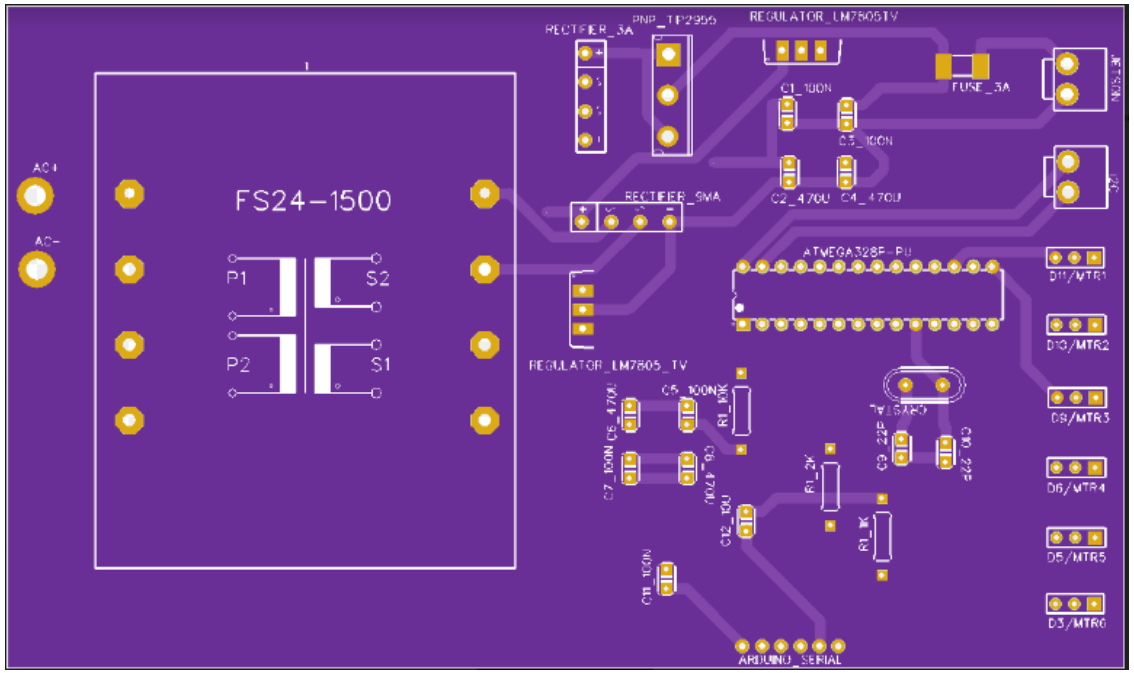
From Transformer

- 2 Voltage regulation circuits
  - High amperage circuit
    - Uses PNP Transistor
  - Low amperage circuit

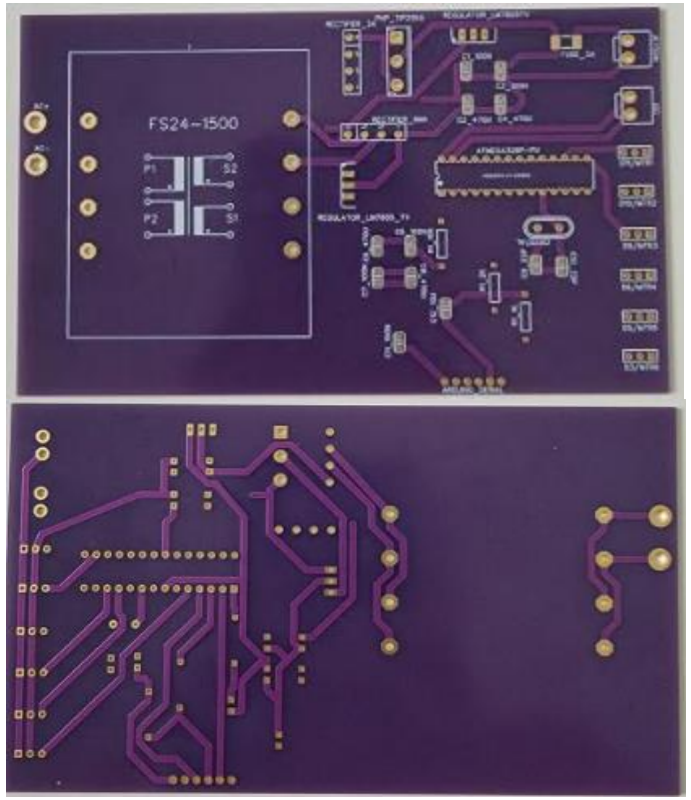
PCB Design

- Designed in EasyEDA and printed at JLCPCB





PCB Design cont.



- Once received, all components were mounted and tested

Software Design

Machine Learning Software

- Trt_pose_hand: NVIDIA made visual hand detection in real time using Jetson Nano
- Uses TensorRT with Python
- Modified and extended to fit the needs of this project
- Supplies retrained Resnet18 convolutional neural network model for hand to return list of key point locations in image
- SKLearn Support Vector Machine

Machine Learning Options for Gesture Detection

Model	Training our own CNN	ResNet18-Hand + SVM
Architecture	Standalone. Would use convolutional neural networks with global average pooling.	Requires use of SVMs for gesture detection. Uses convolutional neural networks.
Accuracy	Very accurate	Depends on implementation
Return type	Classification label only	Vector key points for each part of hand
Time to implement	Long	Relatively quick

Computer Vision Software

- Jetcam: A library for implementing a camera in python
- USB and CSI camera support
- Instantiates camera as an object with image accessible through camera.read()
- OpenCV: Image manipulation and operations
- Displays, saves, rotates, interpolates images + more

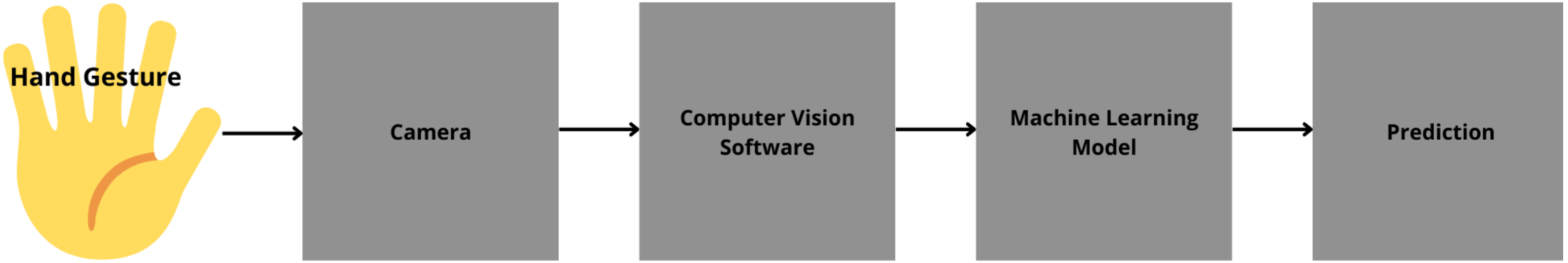
What's the difference?

Computer Vision Software

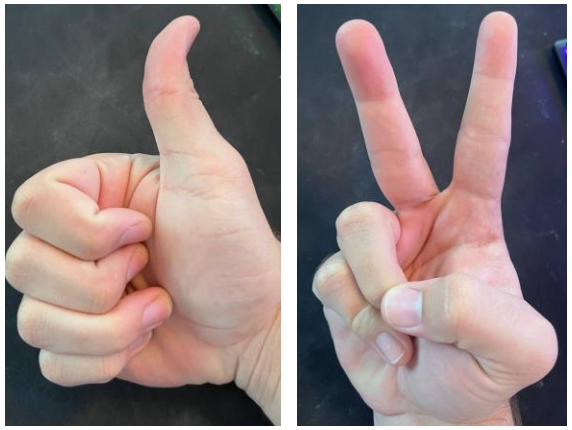
- Converts image from camera to useful data for machine learning use

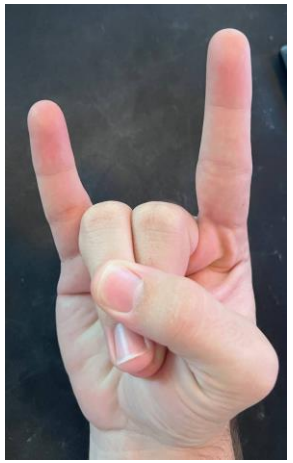
Machine Learning Software

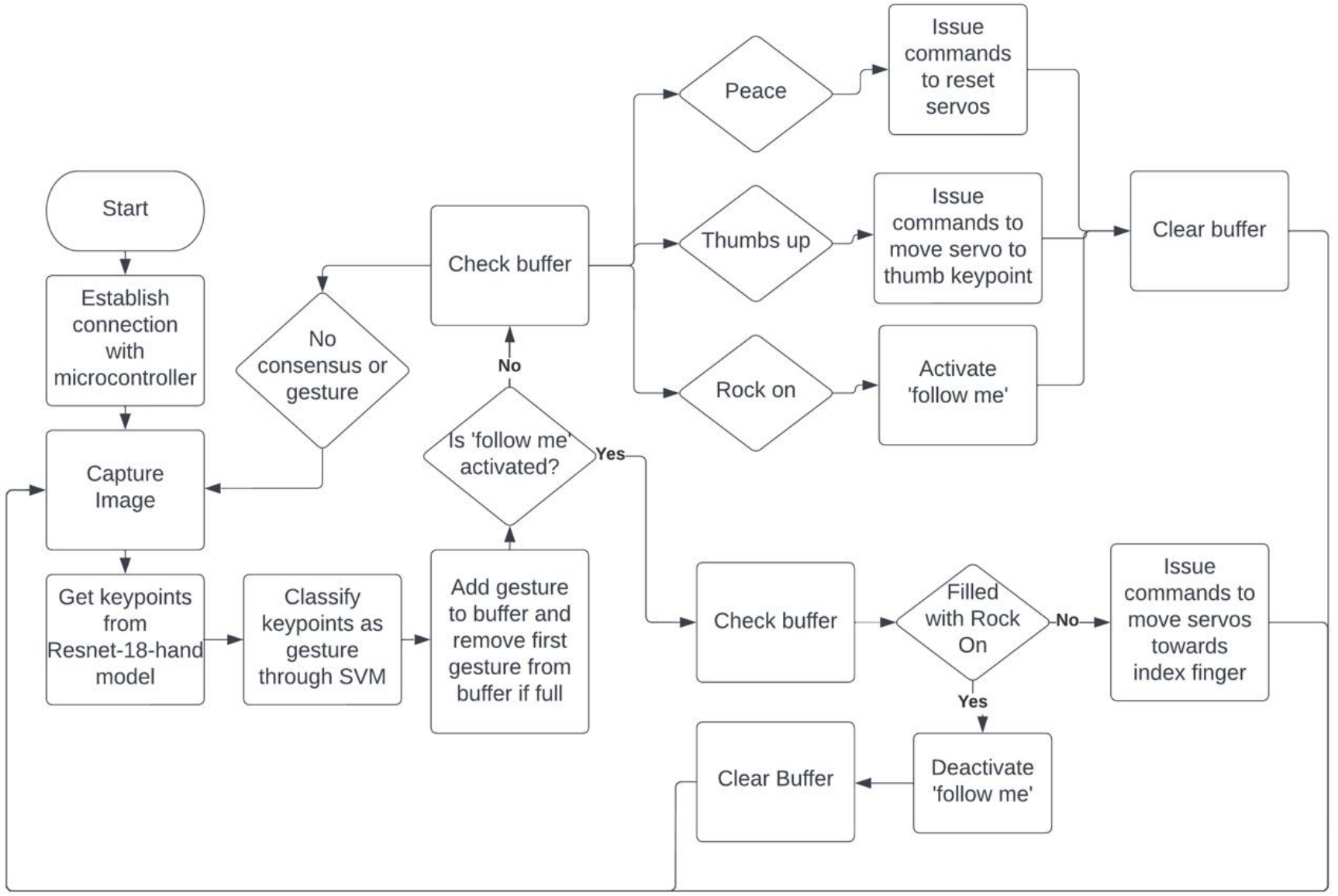
- Utilizes models for classifying images from data supplied by computer vision software



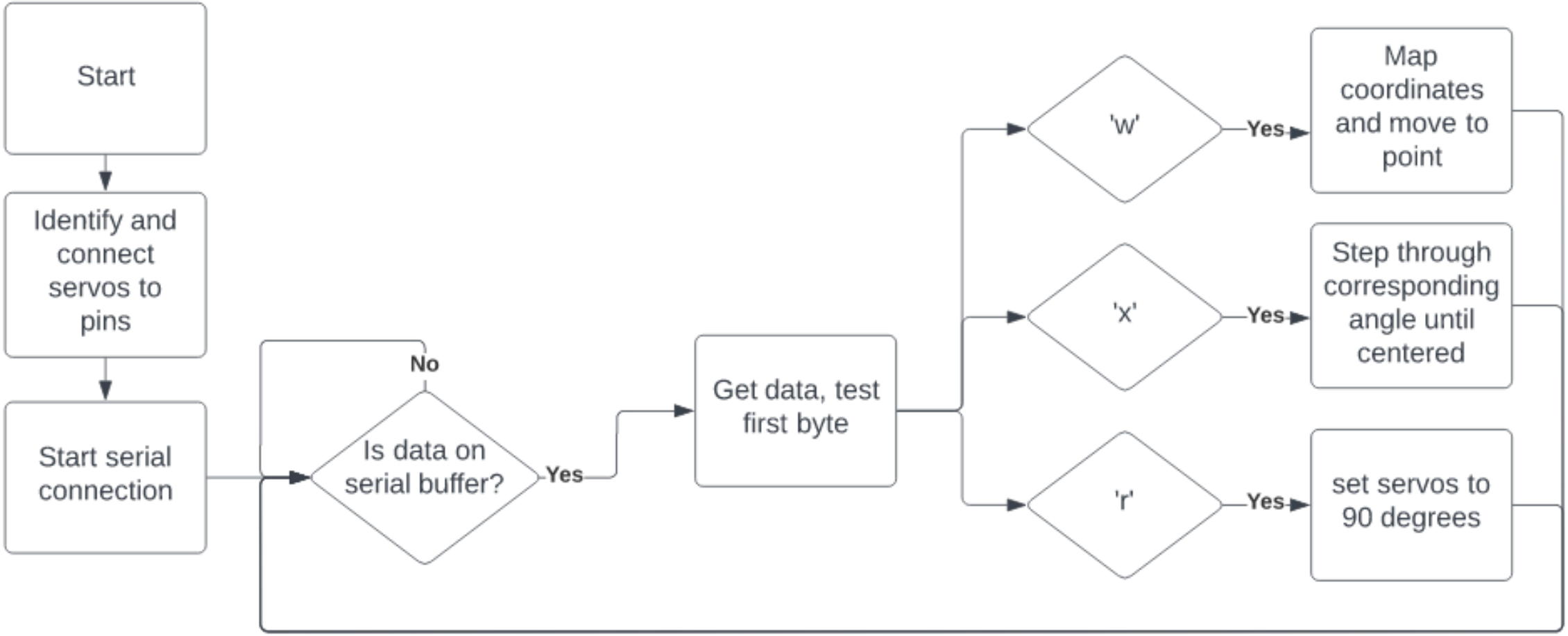
Jetson Nano Project Software Block Diagram





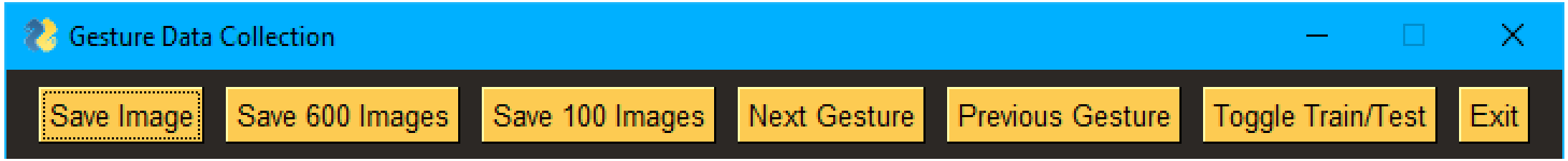


Microcontroller Software Block Diagram



Dataset Collection Software On Jetson Nano

- Image_collection.py
  - Gathers images from camera to be used in training SVM model
  - Saves images on a per gesture basis in their own directories
  - Saves JSON files to map expected outputs (the gesture label) for the SVM to load images and learn the gestures based on outputs from the CNN model
  - Contains GUI for navigating dataset creation
  - Saves hundreds of training images in the click of one button



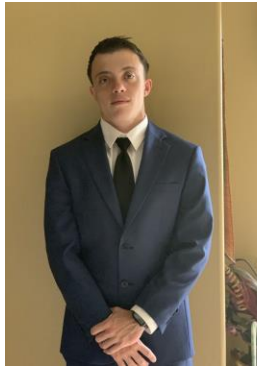
SVM Training Software on Jetson Nano

- Svm.Py
  - Trains a SVM and saves it as a .sav file for use in main program
  - Uses dataset selected by the user – loaded through JSON constructs
  - Passes each image into resnet18-hand model to gather location of key points in each image – key points passed to the SVM for training
  - Saves JSON file mapping SVM outputs to name of gesture
  - After training loads camera feed with live gesture classification for testing performance of SVM

ArticuLight Software On Jetson Nano

- ArticuLight.py
  - Uses trained SVM to classify gestures in real time
  - Issues commands to ATmega328 Microcontroller based on detected gestures
  - Gestures must be held for one second
  - Implements 'go here', 'follow me', and 'reset' commands
    - 'Go here': Activated by thumbs up gesture. Sends single command to ATmega MCU to actuate motors to center tip of thumb if it is outside of dead zone center of frame
    - 'Follow me': Activated by 'rock on' (horns) gesture. Continuously sends commands to ATmega MCU to center tip of index finger if it is outside of dead zone center of frame until it is deactivated
    - 'Reset': Activated by peace sign gesture. Issues command to ATmega MCU to reset the position of servos to 90 degrees resulting in a neutral arm position

Power Options



- Wall Plug-in
- 12 V Power Supply
- Cord length is 12 feet, however actual length from project to wall outlet is 8 feet.

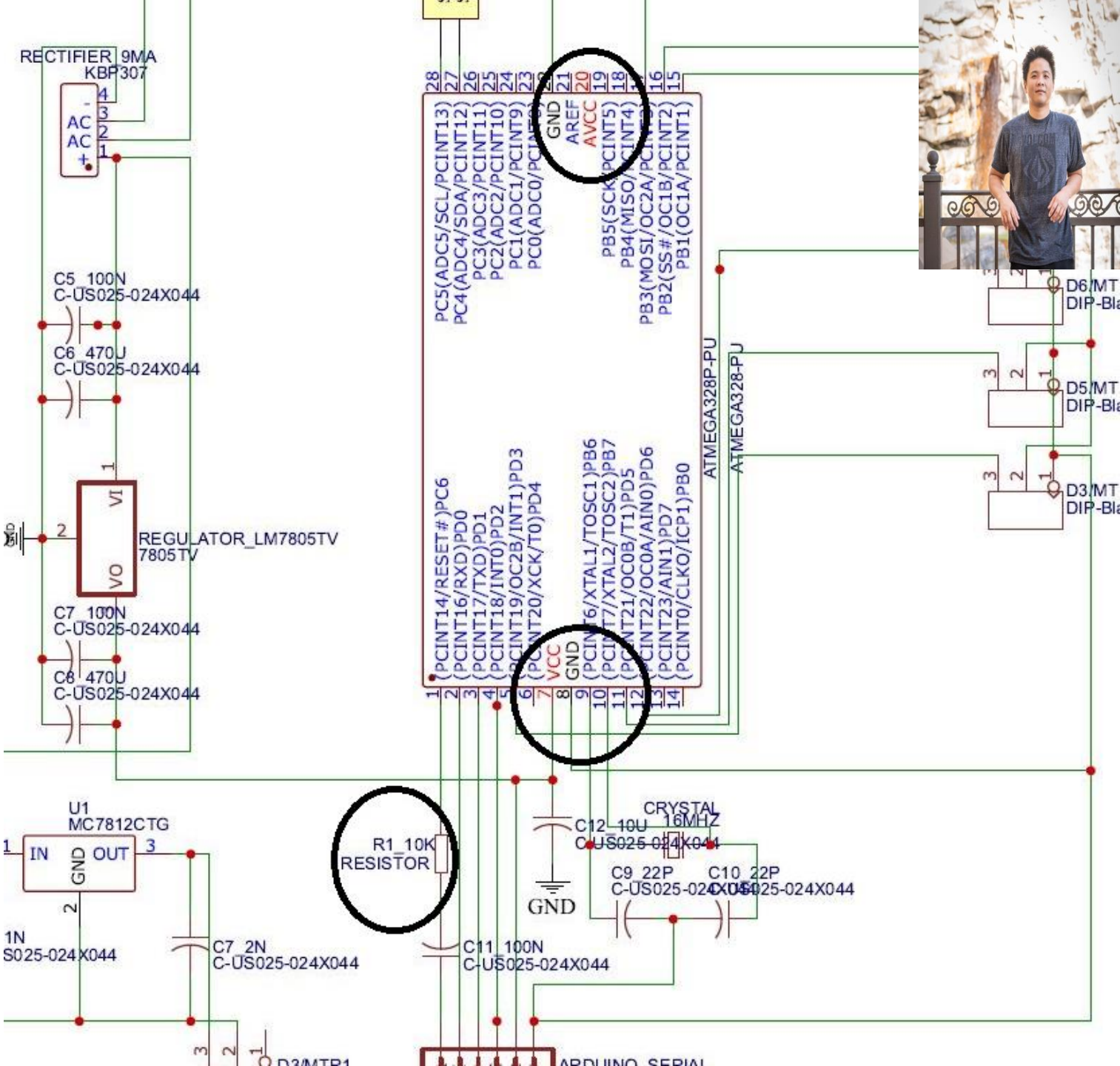
Specifications
Stable voltage
Able to plug into extension cord for extra length
No discharge

- Battery Pack Power Supply
- Allows for portability
- Stable and clean power
- Different types of batteries that can be used

Type of battery	Voltage (V)
9V	9 for non-rechargeable 8.4 for rechargeable
Flat pack	3.7 Rechargeable
Car battery	12 Rechargeable

Circuit Testing

- Not receiving a PWM signal from the Jetson due to a 10K Ohm pull-up resistor. Removing this component allowed for a clean PWM signal.
- Needed to add decoupling capacitors between Vcc, AVcc, ARef and grou nd to generate a cleaner PWM signal



Administrative

Work Distribution

	PCB Design	Voltage Regulation	Computer Vision Software Programming	Servo Control	Mechanical Design
Ben			P	S	
Keng	P	P			S
Michael	P	S			
Scott	S	S	S	P	P

Budget and Financing

Part	Quantity	Cost	Seller
NVIDIA Jetson Nano Developer kit	1	$99	NVIDIA
CanaKit 5V 2.5A Raspberry Pi	1	$9.95	Amazon
PNY 128GB Micro SD card	1	$13.99	Amazon
Raspberry Pi Cam V2	1	$26.57	Amazon
Servo motor D956WP	2	$109.04 each	Amazon
ATmega 328P Microcontroller	1	$25.99	
E-26 Light Socket	1	$8.99	Amazon
LED Bulb	1	$2.50	Amazon
Filament (PLA)	1	$30	Amazon
Frame/Case Material	1	$15	Lowes
Total		$450.07	

Constraints

- Number of PWM channels ATmega328 can support (6)
- No direct method to load code to ATmega328
- 10 x 10 x 10 inch requirement to fit components in case
- Had to add a fan due to regulators and motors becoming hot during articulation
- Length of arm for motors to handle torque requirements
- FPS of camera – processing too many frames within one second causes lag in gesture classification worsening system performance

THANK YOU