Robotic Lamp

ECE 445 Design Document

Dhruv Diddi, James Jia, Saksham Saini
Group 82
TA: Eric Clark

1 Introduction

1.1 Objective

Home automation continues to be on the forefront of technological advancement in 2016 and in 2017. While we have seen software interaction, such as Google Assistant, Siri, and Alexa, and we have seen automation in the form of smart lighting, thermostats, and more, we have yet to see many products that bring automation and interaction together. Specifically, we wanted to look at the office setting - lamps. We spend a lot of time at our office desks working on our computers, scribbling on our papers, and 46% of that time is spent after daylight hours[1]. While lighting in the office typically is through the use of lamps, these lamps are stationary, heavy, out of reach, and take effort to move around when we need them to, especially for those that are moving from one area of the table to another. This can be an annoyance for many office workers - constantly having to move lamps for their intended needs.

Our solution - a robotic lamp - aims to provide automated interactivity for anyone who spends their time at the office. The robotic lamp provides automation - turning itself on as soon as you are near it, and can set the tone of light based on the time of day. It also provides interactivity - following your hands as you write and work, to provide optimal lighting on your workspace, as well as gesture controls.

Through this project, we aim to bring smart technology and interactivity to something as conventional as a desk lamp and hopefully inspire other people to think of ways to bring such interactivity to other conventional devices used in our daily lives.

1.2 Background

There are currently no commercial robotic lamps on the market, but rather only “smart lighting” (such as Philips Hue) and lamps that are easy to adjust. There are many “do it yourself” projects online that have created a robotic lamp, but are not feasible products, as they use expensive, using unnecessary technology such as Arduino variants [2]. Lamps are currently lacking flexibility of use, and a robotic lamp is what many people are looking for [3]. We want to provide a cost-effective solution, while providing fullness of interactivity and automation.

We took inspiration from a Robotics project done at MIT a few years ago [4], which uses OpenCV and Machine Learning to provide the user an interactive desk assistant in the form of a robotic lamp, which keeps the user engaged in their work. The lamp adjusts the gestures based on the user's behavior and mood patterns and has additional functionality such as changing the temperature/color of the light as well. We wanted to use the research done and make a cost-friendly interactive robotic lamp assistant for the users.

1.3 High-Level Requirements

- The lamp must interact with a single person within a 5 ft. distance within it's field of view (greater than 45° by 30°), while never coming into contact with the individual's arm or hand.
- The lamp must identify the following gestures after entering gesture recognition mode (person's palm is facing the camera and all fingers are straight, "stop" position):
    - Hand up (increase brightness of light)
    - Hand down (decrease brightness of light)
    - Hand side (lamp head turns the same side)
    - 1 Finger up, 4 Fingers down (lamp moves vertically upward)
    - 2 Fingers, 3 Fingers down (lamp moves vertically downward)
    - Turn Hand around (lamp toggles motion tracking on/off)
- Entering gesture recognition part should not exceed 150ms.
- The lamp response time should not exceed 2s.
- The lamp power input should not exceed 5V at 1A.

2. Design

2.1 Block Diagram

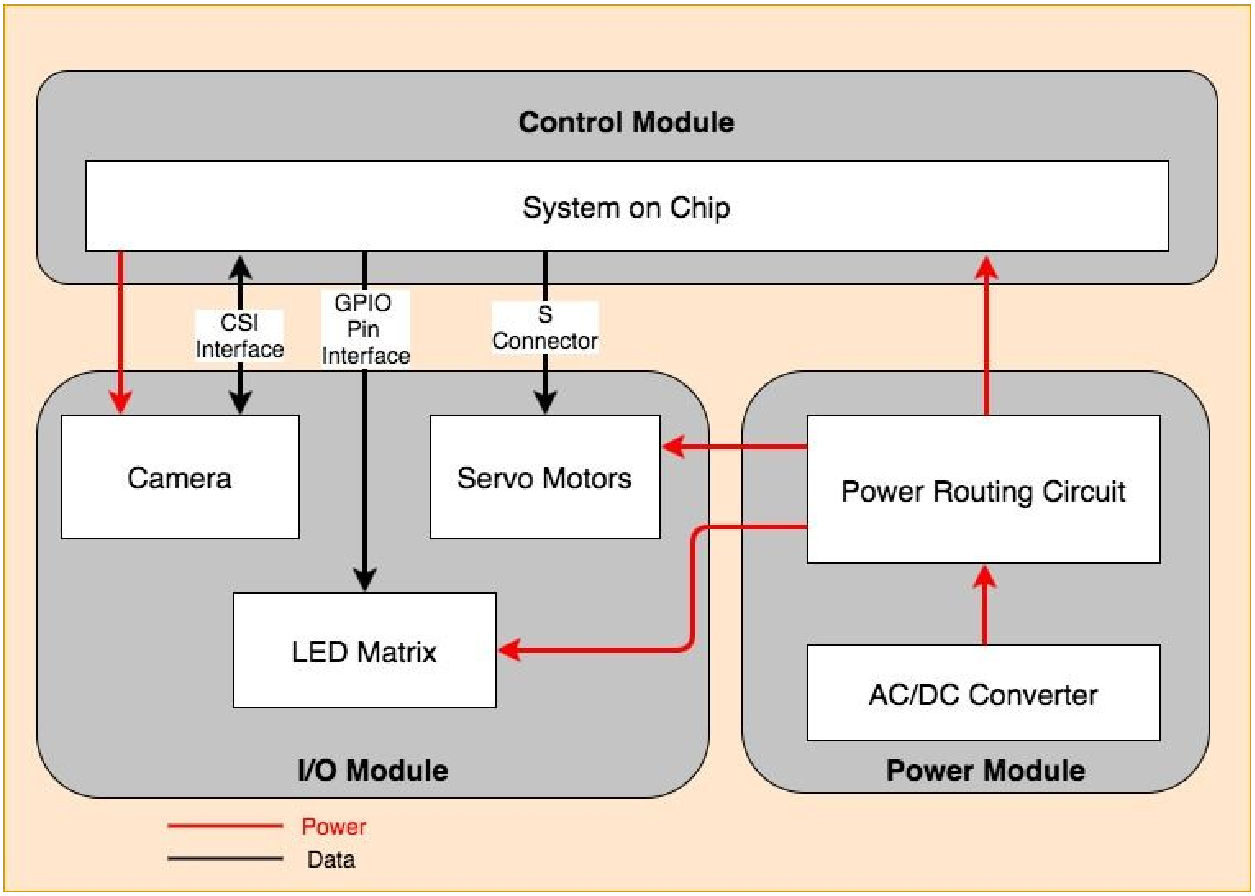

Fig. 1. Block Diagram

2.2 Physical Design

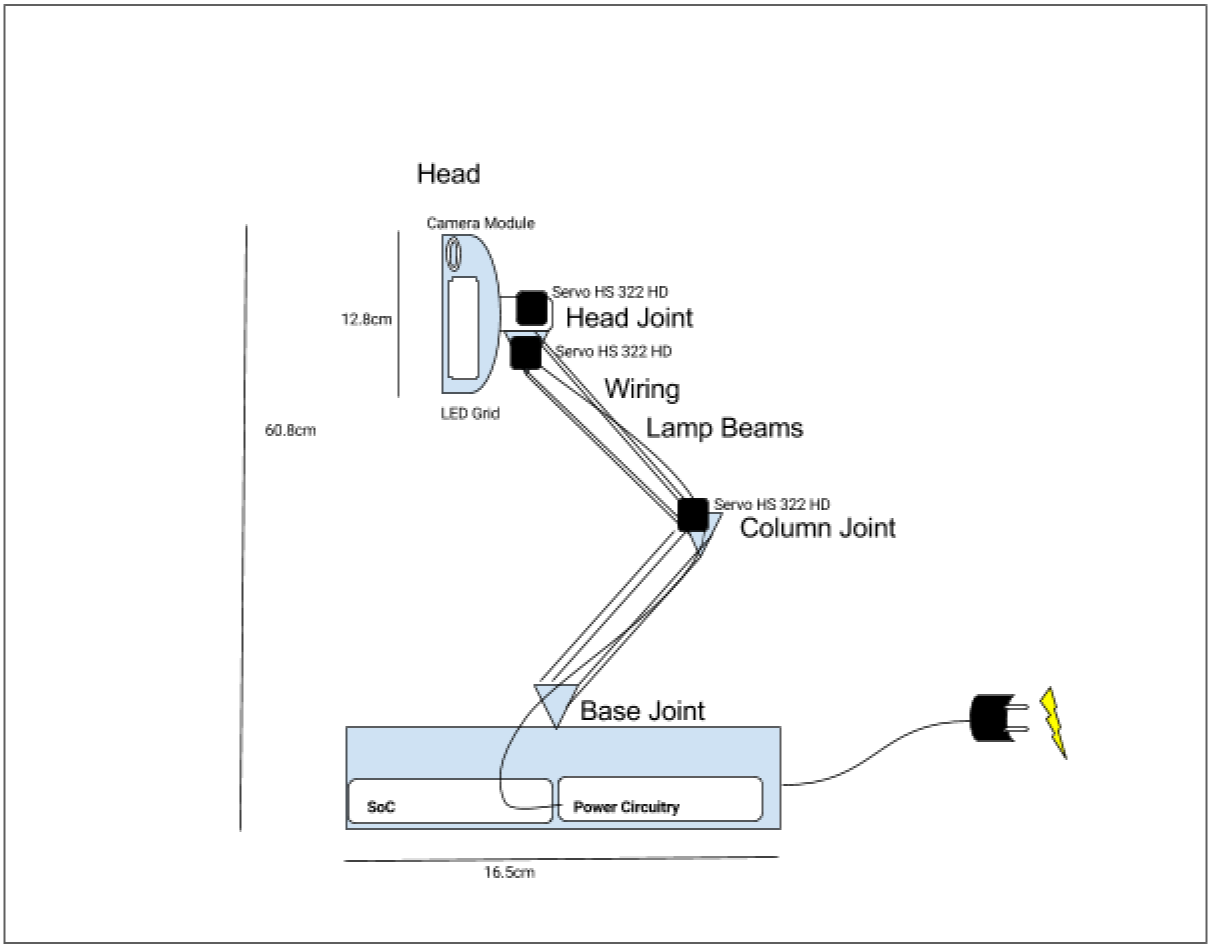

Fig. 2 Physical Design Mock-up

2.3 Control Module

The control module is the part responsible for the gesture recognition and processing the feedback gestures based on the user input. The control module gets input from the camera module in the I/O Module and then sends signals to the LED matrix to adjust lighting and servo motors to adjust the lamp arm respectively.

2.3.1 Raspberry Pi Zero SoC

2.3.1.1 Functional Overview

We will use the Raspberry Pi Zero as our System on Chip solution to take input from the camera, process the visual data, and control the motors and LEDs to provide feedback to the user in the form of light intensity and gestures. The Raspberry Pi Zero SoC is a 1GHz single-core processor with 512MB RAM, which will provide enough computing power to do the video processing and gesture selection. It is powered by the power module and consumes about 5V at 1A of power.

2.3.1.2 Requirements and Verifications

Requirements	Verification
1. Can receive video stream from the camera module at a data rate of at least 0.9216Mbps (for 240p at 12fps) and process it in no more than 150ms.	1. a. Connect the camera module to the CSI interface on the Raspberry Pi and record a 10 second video at 240p resolution. b. Run OpenCV script on the video input. c. (Start timer) when a marker object is displayed on screen. d. (Stop timer) when OpenCV detects the marker. Ensure that the time does not exceed 150ms.
2. Can identify the gesture and starts movement in the lamp arm in 2 seconds.	2. a. Connect the camera module to the CSI Interface on the Raspberry Pi. b. (Start timer) Put a marker image in front of the camera to initialize gesture recognition and action. c. Wait for the motor to start moving. d. (Stop timer) Ensure that the motor started moving within 2 seconds.
3. Can complete response gesture in the lamp arm in 10 seconds.	1. Connect the camera module to the CSI interface on the Raspberry Pi. Put a marker in front of the camera. Wait for the motor to start moving. 2. (Start timer) When the motor starts moving. 3. Wait for the movement to finish. 4. (Stop timer) When the motors stop moving. 5. Ensure that the time taken does not exceed 10s.

2.3.1.3 Supporting Materials

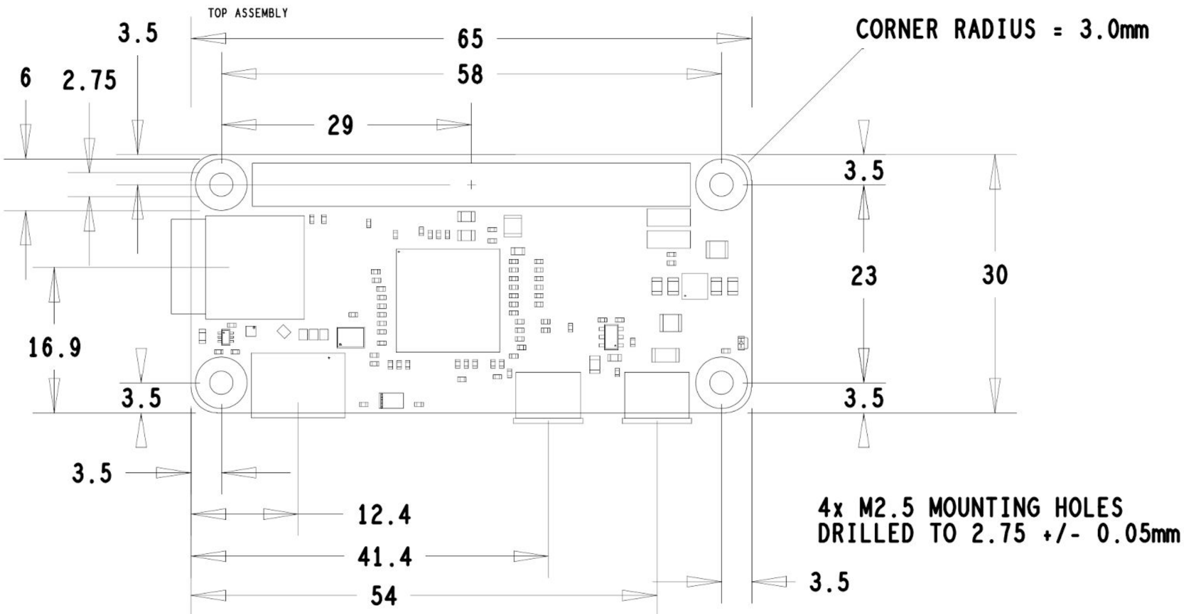

Fig. 3. Block Diagram [5]

2.4 Input/Output Module

The I/O Module are the outfacing, interactive parts of the lamp. While the camera is taking images to input into the control module for processing and recognition, the LED Matrix and Servo Motors outputs are the physical interactions with the user, providing different scenarios and configurations based on recognition from the camera.

2.4.1 Camera

2.4.1.1 Functional Overview

The camera module is in itself one piece - the camera. The OV5647 Camera [6] is a fixed-focus camera that is designed to interface specifically with the Camera Serial Interface (CSI) [7] of the Raspberry Pi Zero, providing the exact I/O of the SoC, fitting well without need for soldering. The OV5647 Camera interfaces with the use of 15 pins through a ribbon cable, while the CSI pins sit on the outer shell of the SoC. Data is transferred through two serial buses though various pins. The camera is also powered through other pins on the CSI.

2.4.1.2 Requirements and Verifications

Requirements	Verification
1. Takes images at a resolution of at least 240p (320 x 240) and a frame rate of at least 12 frames per second	1. a. Connect OV5647 Camera to Raspberry Pi Zero via the CSI connector b. Connect monitor via mini-HDMI to the SoC, as well as a USB Keyboard to the SoC c. Using the Linux Terminal on the Raspberry Pi Zero, run the command "sudo raspi-config" to configure image resolution to a width of 320 and height of 240 d. Using the PiCamera Python Library, run a Python Script to connect to the camera and take an image e. In the same Python Script, use the Python Image Library's wrapper to print out the resolution of the image using "Image.open(imageName).size" to verify the image resolution is equal to 640 x 480 f. In the same script, take a time stamp (startTime), and take 100 photos using the PiCamera Library. Take a second time stamp (endTime) after the 100 photos are taken, and calculate the time elapsed to verify that it took less than 8.3333 seconds to run
2. Provides a field-of-view of greater than or equal to 45° x 30°	2. a. Connect OV5647 Camera to Raspberry Pi Zero via the CSI connector b. Connect monitor via mini-HDMI to the SoC, as well as a USB Keyboard to the SoC

	c. Using the Linux Terminal on the SoC, run the command “sudo raspi-config” to enable the camera and configure image resolution d. Measure and stand 1 meter away from the center of the camera e. Run the command “raspistill -v -o test.jpg” in the terminal to begin a 5-second image preview begin f. Use a tape measure to span the horizontal axis of the photo frame, and record this value. g. Calculate the horizontal angle field-of-view using triangle identities to verify that it is greater than 45 degrees h. Repeat the process by instead measuring the vertical axis of the photo frame to verify that the vertical field-of-view is great than 30 degrees

2.4.1.3 Supporting Materials

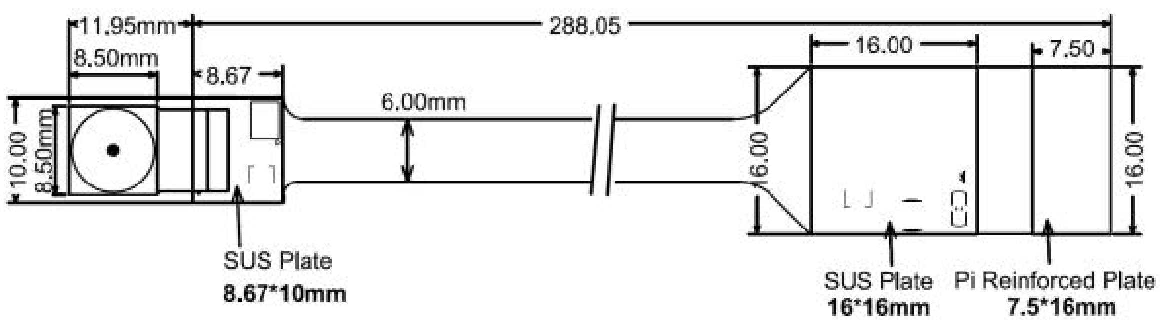

Fig. 4. OV5467 Camera and CSI Interface Dimensions [8]

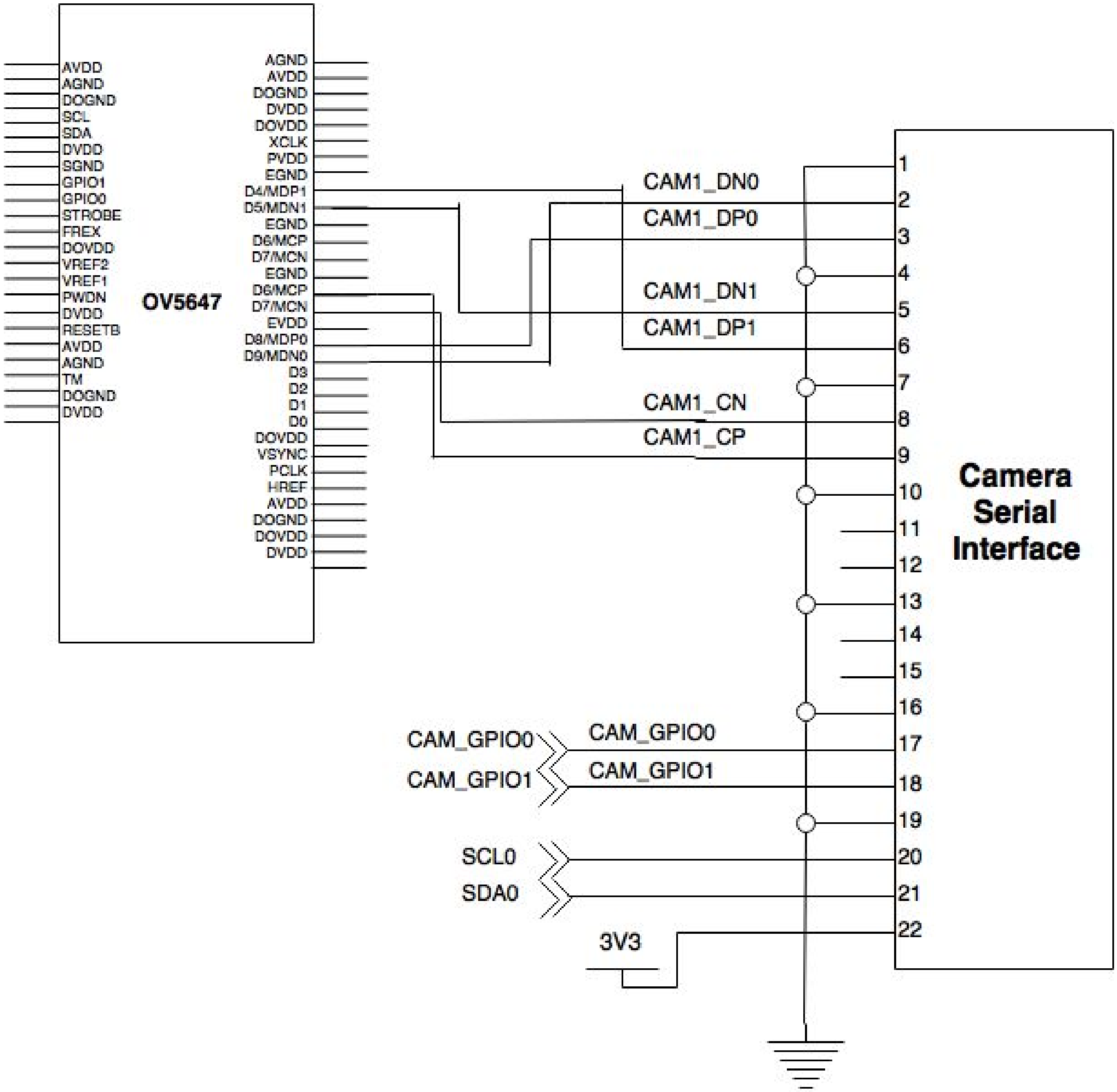

Fig. 5. Camera and Interface Connections

2.4.2 LED Matrix

2.4.2.1 Functional Overview

An 8x8 64 bit Matrix of WS2812B RGB LEDs [9] will be used as our primary lighting. It is powered by the Power Module. Data is transferred from the Raspberry Pi GPIO pins. The LEDs run in series and can be brightness can be changed addressed through the

Raspberry Pi through the use of Pulse Width Modulation. Individually, these LEDs can be addressed by connecting them in a cascading manner and sending instructions of length $64 \times 24 = 1,536$ bits where each 24 bit instruction is used to select the color. Since the LEDs are driven by a 5V Power Supply, the Data must also be driven by a 5V Source, requiring a level converter to convert the 3.3 V Data output pin of the Raspberry Pi [10].

2.4.2.2 Requirements and Verifications

Requirements	Verification
1. At a voltage supply range from 3.5V-5.3V All LEDs light up white at a current draw of 60 mA/LED for at least 30 minutes	1. a. Connect the 5V Power Supply directly to V_{DD} of the LED, as well as Ground to V_{SS} b. Connect GPIO Pin 18 of the Raspberry Pi Zero to D_{IN} on the LED. Send a PWM Signal every 1.75us for 30 minutes that consists of setting all RGB bits to 1, concluding with a 50 us Reset Clock. c. Check and wait to see if the LEDs light up for at least 30 minutes

2.4.2.3 Supporting Materials

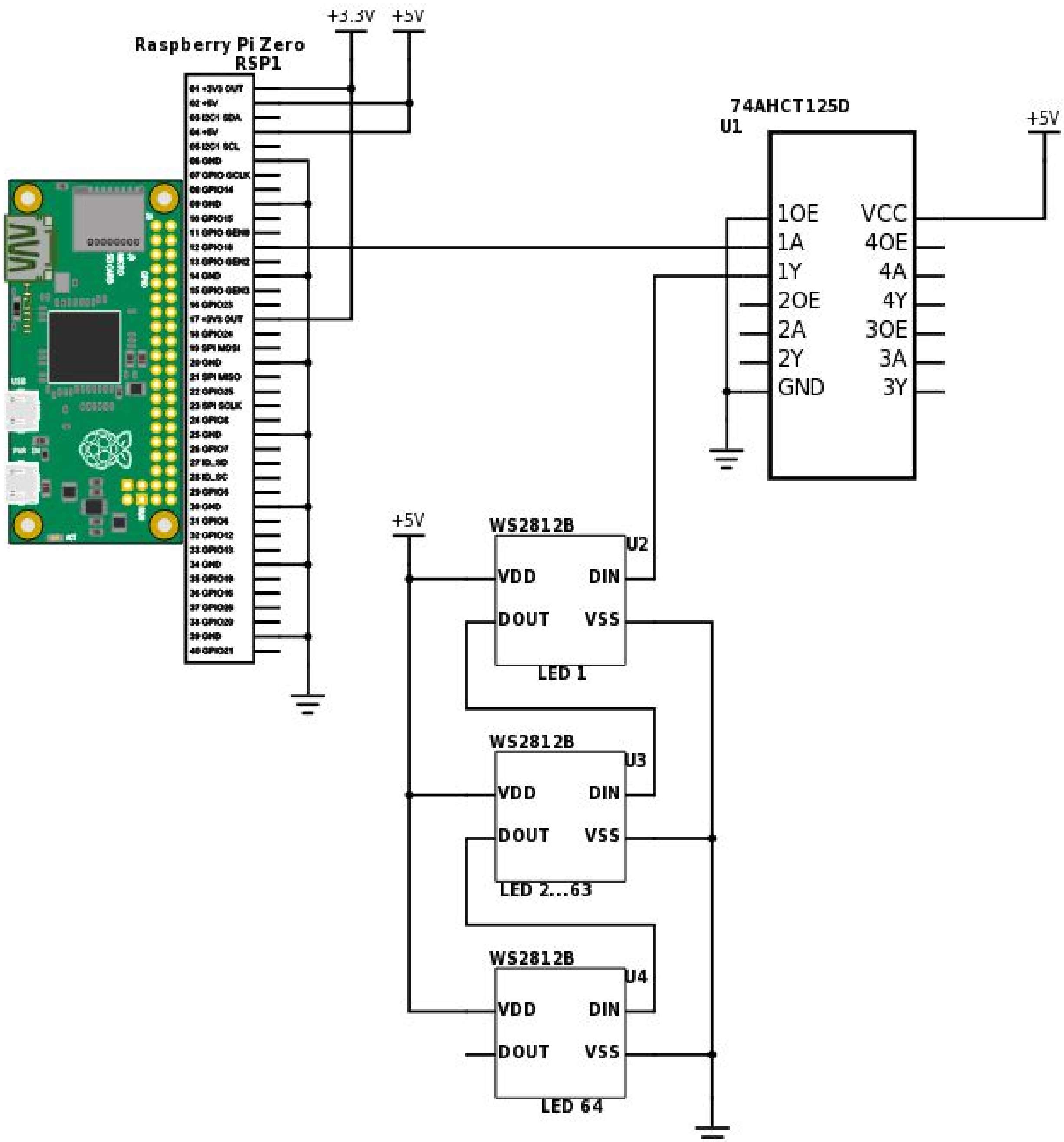

Fig. 6. LED Connections to the Level Converter and Raspberry Pi Zero

2.4.3 Servo Motors

2.4.3.1 Functional Overview

The Servo modules we use are HITEC HS 322HD servos which are used to swivel and turn different parts of the robotic lamp. We will be using 3 servo motors for 3 joints which we'll use to control the motion of the lamp. One servo will be used to swivel the lamp head left and right in the xy-plane. Another one will be used to create the up down yz-plane motion for the lamp head. The third one will be used in the middle joint (in the neck of the lamp) to control the up down motion of the lamp itself. These motors have an operating voltage range of 4.8V-6V and can produce upto 3kg/cm-3.7kg/cm [11]. The servo motors have an analog signal rate of 30 signal inputs per second. The motors will be connected to the PWM Driver[12] which receives duty cycle input from the pins of the raspberry pi using a 3 way connector cable [13][14]. We must make sure that the temperature of the servo motors does not exceed temperature ratings during prolonged use [15].

2.4.3.2 Requirements and Verifications

Requirements	Verification
1. Can lift/control the load of the head of the lamp (50g), the head module(150g) and the load of the upper neck (1kg) of the lamp. The torque requirement for the servo is at least a maximum of 3kg-cm	1. a. Connect the servo module to the power input(4.8V-6V) b. Attach drive gear to the servo end of the motor c. Attach testing rig with load to servo spline with different weights d. Observe if servo motor is able to lift the weight with an equivalent of 3.0kg-cm maximum torque at least

2. Temperature of the servo motors does not exceed 60°C in prolonged usage(10+ continuous minutes) at the expected duty cycle of 80%	2. a. Connect servo motors to GPIO pins of Raspberry Pi SoC b. Create 80% duty cycle PWM signal. We calculate the resistance of the servo motor using a multimeter c. We use performance to get an estimate of temperature and its dependence. d. Use the following formulae to get performance estimate $\Delta T = (R_h - R_c)(234.5 + T)/R_c$ R_c is cold winding resistance(Ω) R_h is hot winding resistance(Ω) T is ambient temperature(C) ΔT is change in temperature e. Note performance drop under different temperature conditions

2.4.3.3 Supporting Materials

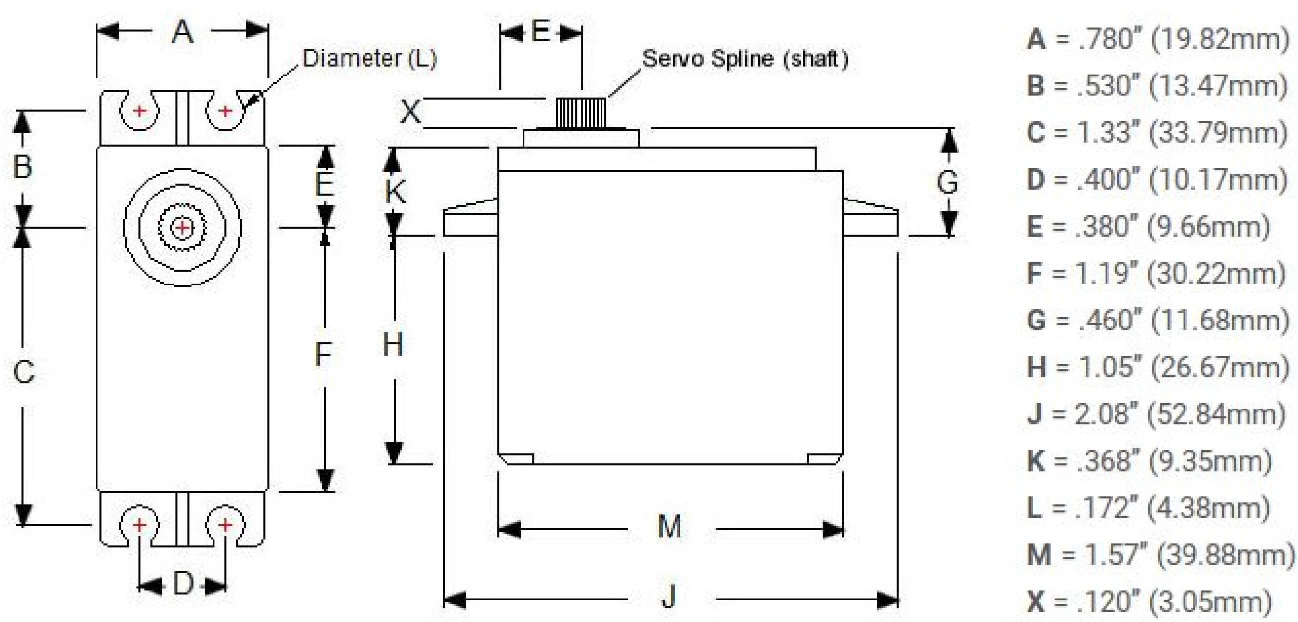

Fig. 7. HS-322HD Servo Motor[15]

The HS-322HD Standard Heavy Duty Servo utilizes Hitec's revolutionary Karbonite™ gear train which is four times stronger than the standard white nylon gears. Karbonite™

gears and Hitec's custom IC make the HS-322HD a great choice for an economical sport servo [17][18][19].

Operating Voltage Range (Volts DC)	4.8V ~ 6.0V
Speed (Second @ 60°)	0.19 ~ 0.15
Maximum Torque Range oz. / in.	42 ~ 51
Maximum Torque Range kg. / cm.	3.0 ~ 3.7
Current Draw at Idle	7.7 mAh
No Load Operating Current Draw	180 mAh
Stall Current Draw	800 mAh
Dead Band Width	5 µs
Dimensions (Inches)	1.57 x 0.78 x 1.43
Dimensions (Metric)	40.0 x 20.0 x 36.5
Weight (Ounces)	1.51
Weight (Gram)	43.0
Circuit Type	HT7003 Analog SMT
Motor Type	3 Pole Metal Brush Ferrite
Gear Material	Karbonite
Bearing Type	Nylon
Output Shaft (type / Ømm)	Standard 24
Case Material	Plastic
Dust / Water Resistance	N / A
Connector Gauge (AWG) / Strand Count	25 / 40

2.5 Power Module

The power module provides proper voltage to the Control and I/O Modules through conversions from 12V AC. A Kill-Switch turns off all power from the whole system.

2.5.1 Power Routing Circuit

2.5.1.1 Functional Overview

The power routing circuit is a module of the project which is going to be used to control the power supply to other elements of the circuit. We get a 12V DC voltage from the AC/DC converter. This goes through an n-channel MOSFET which acts as an override switch for the rest of the circuit. The output of the MOSFET is connected to a step down buck converter module which steps down the voltage from 12V DC to 5V DC. This output of the buck converter is the voltage which is supplied to the PWM Driver and Level shifter which control different components of the circuit [20].
The routing circuit also has a kill-switch(S1) to turn off the whole system if the user wants to turn off the monitoring from the camera module due to privacy concerns.

2.5.1.2 Requirements and Verifications

Requirements	Verification
1. Ensure that override switch turns the voltage off the level shifter and PWM driver when the switch is turned off	1. a. Connect functional generator to power routing circuit module b. Measure output voltage after buck converter using multimeter with switch in ON position. c. Measure output voltage after buck converter of the buck converter using multimeter with switch in OFF position. d. Confirm observation through repetition.
2. Ensure MOSFET works at gate voltage of 5V	2. a. Connect functional generator to power routing circuit module b. Measure output voltage after MOSFET using multimeter with switch in ON position.

	c. Measure output voltage after MOSFET of the buck converter using multimeter with switch in OFF position. d. Confirm observation through repetition.

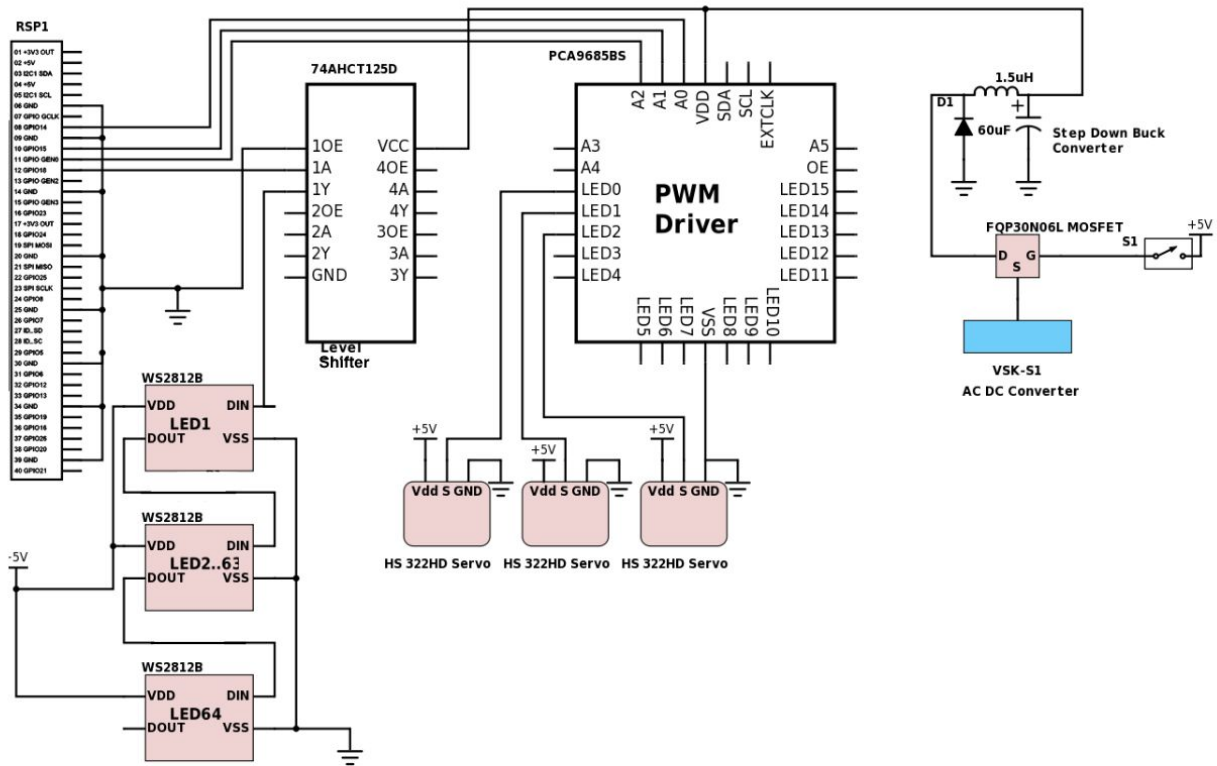

Fig. 8. Power Routing Circuit Schematic

2.5.1.3 Supporting Materials

The FQP30N06L MOSFET has been especially tailored to minimize on-state resistance, provide superior switching performance, and withstand high energy pulse in the avalanche and commutation mode. These devices are well suited for low voltage applications such as automotive, DC/DC converters, and high efficiency switching for power management in portable and battery operated products.[21]

Features

- 32A, 60V, R DS(on)= 0.035Ω@VGS= 5 V
- Fast switching
- 100% avalanche tested
- Improved dv/dt capability

• 175°C maximum junction temperature rating

2.5.2 AC/DC Converter

2.5.2.1 Functional Overview

Since the components of the circuit require DC power supply, we're going to use this module to convert AC power which we get from the wall socket into the DC power required for the components[20].

2.5.2.2 Requirements and Verifications

Requirements	Verification
1. Ensure that AC/DC voltage converted signal is consistent(± 0.15V) and accurate (intended voltage output of 12V)	1. a. Connect AC/DC voltage convertor to wall socket to receive 110V AC input b. Use multimeter to measure output DC voltage and ensure consistency and accuracy of 12V within 3% error. c. Test with resistors(5Ω, 100Ω, 500Ω, 5kΩ) and measure current through it.

2.5.2.3 Supporting Materials

Input Voltage	110V
AC Frequency	60Hz
Output Voltage	12V DC
Weight	1lb

3. Software Flow

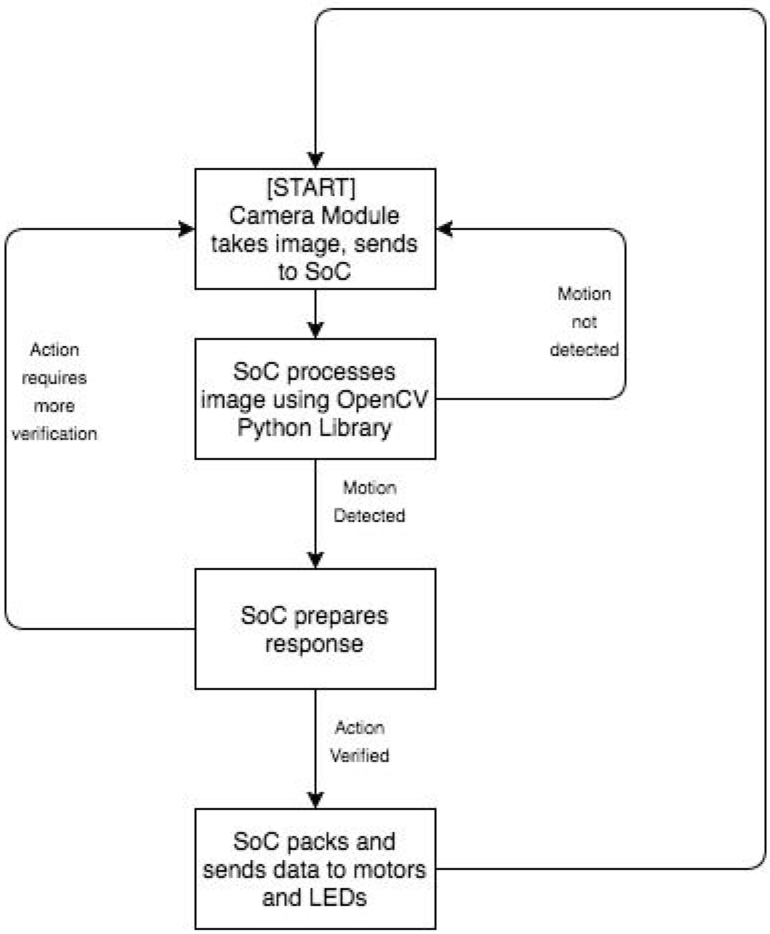

Fig. 9. Image Processing and Action Algorithm

4. Costs

Development costs are based on a $45/hour salary, working 8 hours a week for 3 people. We believe that to be production-ready there are a lot of added features that are out-of-scope for this course, but can be implemented, so only 50% of our final design is appropriated in the labor.

$$3 * \frac{\$35}{1\ hour} * \frac{8\ hours}{1\ week} * \frac{15\ weeks}{} * 2.5 * 2\ = \$63,000$$

The machine shop will be helping to place the servo motors within our pre-bought lamp. The cost are as follows:

$$\frac{\$50}{1\ hour} * \frac{10\ hours}{} = \$500$$

Part	Cost (prototype)	Cost (bulk,100 pieces)
Raspberry Pi Zero	$20.00	$5.00
Camera Module OV5467	$16.99	$6.84
8x8 LED Matrix WS2812B	$15.62	$4.30
Level-Shifter 74AHCT125	$1.50	$1.20
Architect Lamp	$22.00	$5.94
VSK S1 AC/DC Converter	$11.41	$9.78
n-Channel MOSFET	$0.95	$0.85
Servo motors x 3	$43.35	$43.45
AC/DC Converter	$22.04	$11.5
10Ω Resistors	$0.40	$0.10
Diodes	$0.14	$0.08
Inductor (1.5 μH)	$0.22	$0.10
Capacitor (60 μF)	$0.20	$0.12
PCA9685 PWM Driver	$14.95	$11.96
Switch	$1.00	$0.89
Total	**$170.77**	**$102.01**

Total cost of building the prototype: **$63,670.77**

5. Schedule

Week	Task	Delegation
2/20	Create Block Diagrams and physical design parts for mock design review document and design document	Dhruv
	Create camera and LED module as well as ethical and safety issues sections for mock and final design document	James
	Work on microprocessor, introduction and citation sections of the mock and final design document	Saksham
2/27	Order hardware parts for the project and assemble lamp hardware and base	Dhruv
	Develop control software for LEDs	James
	Develop OpenCV code to record and analyse images from web cam interface	Saksham
3/6	Attach servos and finalize position of hardware components	Dhruv
	Add GPIO signal output to control software	James
	Add feature extraction and detection	Saksham
3/13	Test hardware components and movement of lamp	Dhruv
	Test control to all IO signals from the SoC	James
	Test basic contour mapping and object detection using webcam	Saksham
3/20	Solder together the power routing module. Ensure documentation and individual progress reports are upto date	Dhruv
	Create control module which controls servos from routing module and SoC. Ensure documentation and individual progress reports are upto date	James
	Integrate gesture recognition with control module with James. Ensure documentation and individual progress reports are upto date	Saksham

3/27	Combine GPIO outputs with Servo and other parts	Dhruv
	Write code to handle servo motors and functions of interaction	James
	Combine gesture recognition code with SoC camera module	Saksham
4/3	Create function flow for behavior of robotic lamp	Dhruv
	Write function control for LEDs output structure	James
	Test gesture recognition from OpenCV and optimize for device	Saksham
4/10	Add interaction behavior to the servo motors as a reaction to gesture input	Dhruv
	Test gesture recognition quality and LED output quality	James
	Create interesting routines for robotic lamp	Saksham
4/17	Integrate and test hardware and software components for mock demo	Dhruv
	Integrate and test hardware and software components for mock demo	James
	Integrate and test hardware and software components for mock demo	Saksham
4/24	Finish up final papers, presentations and lab notebooks	Dhruv
	Finish up final papers, presentations and lab notebooks	James
	Finish up final papers, presentations and lab notebooks	Saksham

6. Tolerance Analysis

6.1 LED Module

One important tolerance we must maintain is for the LED. The target voltage for the LED Module should be within 3.5-5.3V total (Red: 2.0-2.2V; Green: 3.0-3.4V and Blue: 3.0-3.4V)[6]. We connect a 10Ω to the LED module to make sure the input voltage is withing the operating limit. However, the resistors have a tolerance of 5-20%. Considering the AC/DC Converter might also have a fluctuation of 5% in the 12V output, we cannot connect the converter to the LEDs directly since the maximum voltage (13.2V) is more than the operating limit. Thus, we pass the signal coming out of GPIO18 from the Raspberry Pi is 3.3V signal through a line driver (74AHCT125) which supplies a PWM signal to the LEDs of the length of $64 \times 24 = 1,536$ bits. Since the LEDs are connected in a cascading fashion, each of the LED unit received 24 bits of signal specifying RGB values for the LED. We also add a step down buck converter to the power supply to ensure a constant 5V supply for the line shifter (74AHCT125).

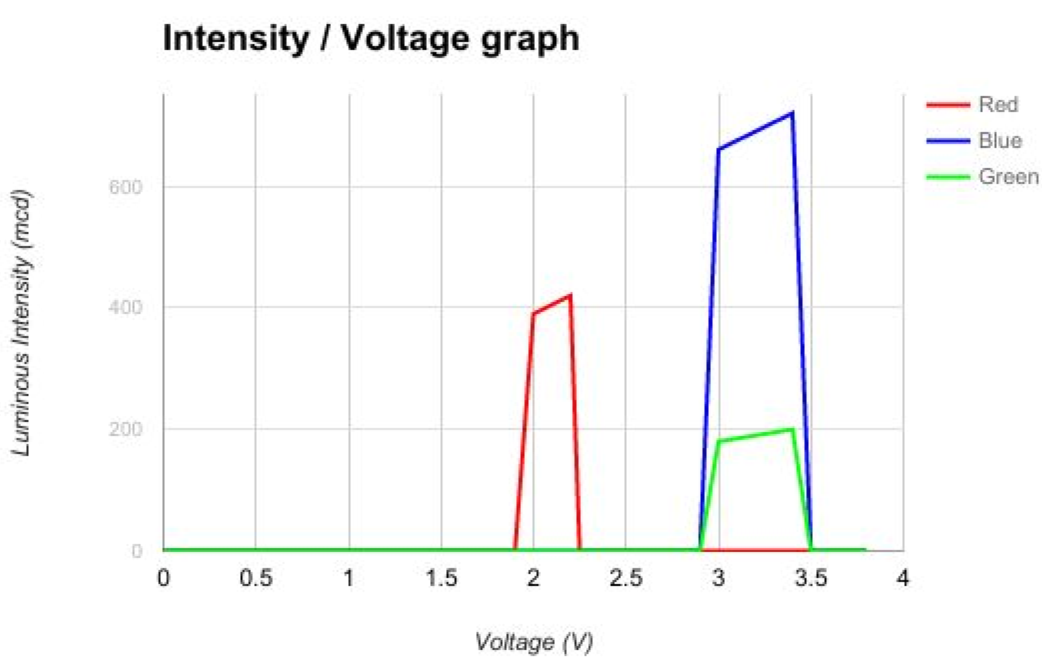

Fig. 10. Intensity vs. Voltage per-RGB Color Graph

6.2 PWM Driver

The PWM Driver is used to provide a PWM signal to the motors without putting the extra computation on the SoC. The Raspberry Pi also has only one PWM enabled GPIO Pin (GPIO18) which gets used in creating the signal for the LED grid. The PWM Driver has an operating voltage of 2.3-5.5V which is supplied by the power routing circuit module. The power routing circuit module provides a constant 5V output from the step down buck converter which is in the tolerance limits for the circuit. A precision error of 10μs in the duty cycle can change up to 90° of position in the servo motors; therefore it's important to maintain high precision through the use of such a PWM driver, as opposed to a software one.

6.3 Power Routing Circuit

The power routing circuit consists of an N-channel MOSFET with an overriding switch connected to it as gate voltage. The drain of the MOSFET is connected to a step down buck converter which is used to DC-DC convert the 12V MOSFET output into 5V DC output required for different modules in the circuit. Both the MOSFET and the buck converter are simple and very efficient electronic devices and we expect low rats of error (3%) from them. In case of malfunction, these devices will stop working entirely giving the user an indication of an issue.

7. Ethics and Safety

There are a lot of ethical issues and safety issues that will come up with a mechanical, robotic lamp. To highlight one of the most severe issues of this decade - privacy. We want to keep the lamp functioning as robotic rather than "smart". As it states in IEEE Ethic #8: "to treat fairly all persons and to not engage in acts of discrimination..." [22]. As spying on cameras have been a bigger problem for personal privacy, we want to prevent engagement in personal and discriminatory acts that may be possible by spying through the lamp's camera. We acknowledged this in our choice of parts, and in particular the Raspberry Pi Zero SoC, which has no networking capabilities built in. This "closed loops" robotic lamp will not have any communication to any signals or inputs from outside the lamp's circuitry, preventing any harm that can be done from attacking personal privacy through the camera.

We also want to emphasize and put efforts towards safety and preventing physical harm, an important IEEE Ethic, #9: “to avoid injuring others, their property…” [22]. As much effort as could to prevent mechanical and electrical malfunction of a robotic lamp, we are implementing a kill switch to immediately disable power to the control and I/O modules (until being turned back on). We will also provide guidelines on the area that the lamp may move, so users will be informed of potential injury or destruction if something were to get in the vicinity of the lamp. This most important includes the degree and range of movement the lamp may move. Additionally, since the lamp will be tracking the user’s motions through the use of arm and hand, algorithms for image processing using OpenCV so that the lamp will never touch the hand for any reason.

Additionally, since we will be working with many pre-built parts such as servos, an LED matrix, and a SoC, we will abide IEEE Ethic #6: “to undertake technological tasks for others only if qualified by training or experience…” [20]. We will not take apart these parts and tamper with them unless we have read the documentation, safety guidelines, and all other paperwork regarding these parts. We will follow instructions on how to properly use these parts.

Safety is one of the most important priorities of this robotic lamp, as stated in IEEE Ethic #1: “to accept responsibility in making decisions consistent with the safety, health, and welfare of the public…”[22]. While we work with many electrical components such as the LEDs, we are aware that these parts will have many limiting factors. This includes particular Voltage and Current regulations. For example, datasheets regarding our LED Matrix[10] provides regulations on max voltage input into the LEDs, as well as working temperatures. Failure to follow regulations such as these will result in blowing out LEDs, or potentially worse. We will abide by each component’s standards, and implement Voltage and Current regulators where necessary.

We are and continue to pay attention to safety precautions we will have to make regarding the servos. In our Requirements and Verifications for our Servos, we are making sure not to exceed torque maximums and temperature maximums. This prevents the lamp from moving excessively, and from potential burning or fires.

8. Citations

[1] National Oceanic and Atmospheric Administration. (2004). “RANKING OF CITIES BASED ON % ANNUAL POSSIBLE SUNSHINE IN DESCENDING ORDER FROM

MOST TO LEAST AVERAGE POSSIBLE SUNSHINE" [Online] . Available FTP: https://www1.ncdc.noaa.gov Directory: pub/data/ccd-data File: pctposrank.txt

[2] arduinoarts, "The IKEA Robot Lamp". [Online]. Available: http://www.instructables.com/id/The-IKEA-Robot-Lamp-Adding-the-servos/

[3] Alvarez, Edgar. (2015, August 17). "This is the desk lamp of your (robotic) dreams" [Online]. Available: https://www.engadget.com/2015/08/17/anodos-robotic-desk-lamp/

[4] Oliynyk, Vitaliy. (2007, May 30). "Robotic Desk Lamp" [Online]. Available: https://www.youtube.com/watch?v=4oCVZTrWrKw

[5] Raspberry Pi. (2015, September 23). "RASPBERRY PI ZERO" [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/mechanical/rpi-zero-v1_2_dimensions.pdf

[6] OmniVision. "OV5647 5-megapixel product brief" [Online]. Available: http://www.ovt.com/uploads/parts/OV5647.pdf

[7] Peter Vis. "Raspberry PI CSI-2 Connector Specifications" [Online]. Available: http://www.petervis.com/Raspberry_PI/Raspberry_Pi_CSI/Raspberry_Pi_CSI-2_Connector_Specifications.html

[8] Arducam. (2015, February). "Raspberry Pi Spy Camera Module ¼-Inch 5-Megapixel Module Datasheet" [Online]. Available: http://www.arducam.com/downloads/modules/RaspberryPi_camera/RaspberryPi_Spy_Camera_Module_DS.pdf

[9] WorldSemi. "WS2812B" [Online]. Available: https://www.parallax.com/sites/default/files/downloads/28085-WS2812B-RGB-LED-Datasheet.pdf

[10] NXP B.V. "74AHC125; 74AHCT125 Product data sheet" [Online]. Available: https://cdn-shop.adafruit.com/datasheets/74AHC125.pdf

[11] Orientalmotor. "Basics of Motion Control" [Online]. Available: http://www.orientalmotor.com/technology/articles/motor-sizing-calculations.html

[12] Adafruit. "PCA9685 Datasheet" [Online]. Available:

https://www.adafruit.com/datasheets/PCA9685.pdf
[13] Monk, Simon. “Using a Servo Motor” [Online]. Available: https://learn.adafruit.com/adafruits-raspberry-pi-lesson-8-using-a-servo-motor/hardware

[14] O’REILLY. “Controlling Servo Motors” [Online]. Available: http://razzpisampler.oreilly.com/ch05.html

[15] The Cowern Papers. “MOTOR TEMPERATURE RATINGS” [Online]. Available: http://www.motorsanddrives.com/cowern/motorterms4.html

[16] SERVOCITY.com. “HS-322HD Servo” [Online]. Available: https://www.servocity.com/hs-322hd-servo

[17] Hitec Multiplex. “HS-322HD Standard Heavy Duty Servo” [Online]. Available: http://hitecrcd.com/products/servos/sport-servos/analog-sport-servos/hs-322hd-standard-heavy-duty-servo/product

[18] ServoDatabase.com. “Hitec HS-322HD - Standard Heavy Duty Servo” [Online]. Available: http://www.servodatabase.com/servo/hitec/hs-322hd

[19] FatLion. “Servos and How They Are Wired [Online]. Available: http://www.fatlion.com/sailplanes/servos.html

[20] Amazon.com. “Sunforce AC/DC Power Converter” [Online]. Available: https://www.amazon.com/Sunforce-AC-DC-Power-Converter/dp/B000FIY08U

[21] Schwartz, Marco. (2013, May 29). “Control a Relay From Anywhere Using the Raspberry Pi [Online]. Available: https://openhomeautomation.net/control-a-relay-from-anywhere-using-the-raspberry-pi/

[22] “SparkFun N Channel MOSFET Datasheet” [Online]. Available: http://cdn.sparkfun.com/datasheets/Components/General/FQP30N06L.pdf

[23] Ieee.org. "IEEE IEEE Code of Ethics". Available: http://www.ieee.org/about/corporate/governance/p7-8.html