PTM Dome For Preserving Our Past

ECE 445 Design Document - Fall 2024

Project #6

Stephanie Leigh, Austin Meissner, Philip Xie-O'Malley

Professor: Arne Fliflet

TA: Jason Jung

Contents:

1 Introduction

1.1 Problem

We as a human race have always been interested in preserving and reflecting on our history. It can be a beautiful way to connect with our ancestors and gain a deeper appreciation for all the advancements we have made throughout time. Artifacts are an important part of preserving this history, but they are sometimes inaccessible, for those unable to visit the artifact, or unable to be understood completely without technology like Polynomial Texture Mapping, or PTM. PTM is a technique that allows for artifacts to be preserved digitally with a high level of detail and shared with researchers around the world and with future generations of humanity. The Spurlock Museum at The University of Illinois at Urbana-Champaign would like to participate in this preserving and sharing of artifacts. However, this process can be quite extensive; it requires a high quality camera as well as a light source, or dome of LEDs, that must be moved repeatedly to allow photos with different angles of light to be captured. With thousands of artifacts to put through the PTM process, the museum needs an efficient solution. Previous solution attempts were made during the Fall 2021 semester of ECE 445, but unfortunately issues arose and the solution failed eventually. The Spurlock Museum is in need of a cost-effective, robust, and long-lasting solution, which our group hopes to provide during our time in Senior Design.

1.2 Solution

To assist the Spurlock Museum in the preservation of their many artifacts, our group will redesign the existing PTM Dome solution while reusing the physical framework that the machine shop manufactured for the previous design team. We will implement an entirely different electrical design, redesigning the wiring design as well as utilizing a different microcontroller and different LED drivers to control the LEDs. We plan to create a simple but effective user interface via Python code on an external computer, where the user will be able to select which mode the system operates in: automatic or manual. The automatic mode will be designed to capture 32 images, each corresponding to a different LED in the dome. The manual mode will allow the user to select specific LED lights to turn on (up to 5 at a time) and will be mainly used for circumstances where a 2D, non-PTM, image output is desired by the user.

1.2.1 Picture of Physical Design

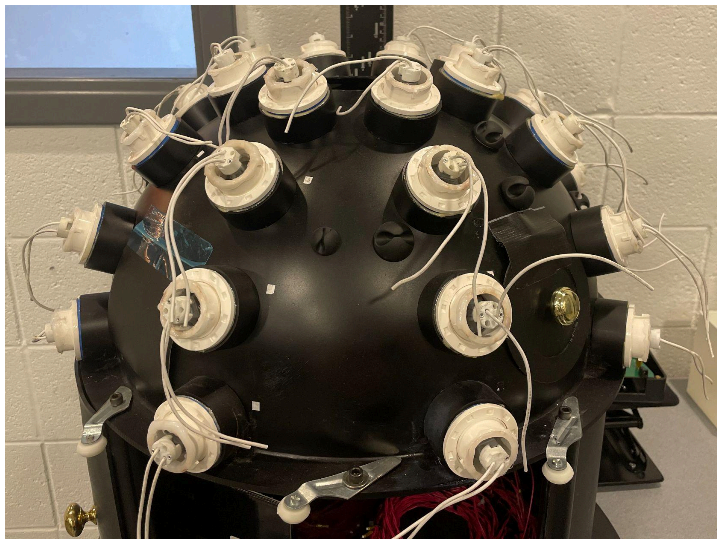


Figure 1: PTM Dome created by machine shop

The picture above depicts the existing plastic shell and the LEDs in their respective placements. It allows for good visualization of what the Dome looks like and shows the mechanical details that will largely remain the same. For instance, the black shell that the system is housed in will remain the same, as well as the cutouts for the LEDs and the LEDs themselves. However, the wiring system that the previous group utilized has been completely scrapped and will be redesigned to improve the organization and simplicity of the design.

1.3 High Level Requirements

To consider our project a success, our team must demonstrate compliance with the following requirements:

1. The PTM Dome System shall have 2 mode states that are determined based on user input: one mode to provide an output of 32 images, with each image corresponding to a

particular LED light, and the other mode to provide a lighting configuration corresponding to any specific selection of up to 5 LED lights that is designed to act as the flash for a single picture output.

2. The PTM Dome System shall interface with the Canon EOS-1D camera, an external computer, external 12VDC power supply, Arduino ATMega640, and 32 LEDs to produce the output.
3. The design of the PTM Dome System shall be modular and repairable, with components such as the 32 LEDs being interchangeable, and the team shall provide a detailed user-interface manual to document such modularity and how to fix any problems that may arise.

2 Design

2.1 Physical Design

For the physical design of our project, we will mainly be utilizing the existing plastic shell created by the machine shop in the UIUC ECEB building. This shell will house the 32 LEDs and will allow for a plastic box containing the PCB and its connection to be mounted to it. The dome itself will contain wire organization pieces as well as a support system for a common ground line that will wrap around the middle of the dome.

2.2 Block Diagram

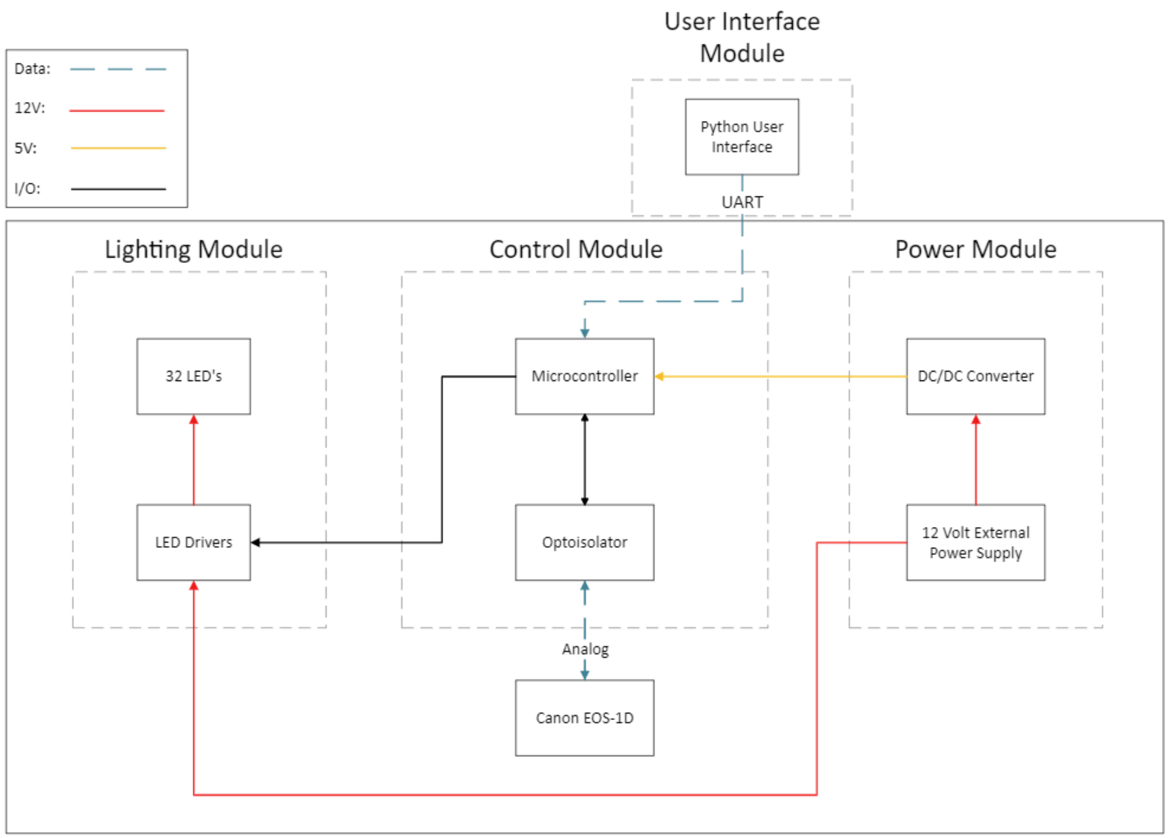


Figure 2: System Level Block Diagram

2.3 Functional Overview and Block Diagram Requirements

2.3.1 Lighting Subsystem

The lighting subsystem consists of the LEDs and the LED drivers. This subsystem is responsible for illuminating the artifacts and will operate in different modes, depending on the outputs from the microcontroller. The LEDs we will be using are the MR11 LED Landscape Light Bulbs from the manufacturer "superbrightLEDs". They draw 208mA of current and operate at +12VDC. The LED drivers chosen are of part number TLC59213 from manufacturer Texas Instruments. These drivers operate at a supply voltage of up to +15VDC and can support up to 500mA of current through each of its 8 output pins. These drivers were chosen because of their large current handling capability and the simplicity of the i/o interface. Each input pin on the driver can simply be linked to a GPIO pin from the microcontroller, allowing for straightforward firmware code and wiring. This subsystem will interact with the power subsystem, by drawing power from it, as well as the control subsystem, by obtaining i/o data from the microcontroller.

Requirements	**Verification**
The LED subsystem shall support two modes of operation- automatic mode, where each of the 32 LEDs will sequence in sync with the camera's shutter, and manual mode, where up to 5 of the LEDs can be selected by the user to be turned on.	Verify by test: 1. Select automatic mode on the user-interface and ensure that each of the LEDs turns on, one after the other. 2. Then, select manual mode on the user interface and select each individual light to be turned on, one at a time. Verify that each of the 32 lights responds to the user input and turns on when commanded and off when commanded. 3. Ensure that 5 LEDs can be turned on at once using the user interface.
The LEDs shall operate at a voltage of 12V +/- .5V.	Verify by inspection and test: 1. Verify via the schematic that the LEDs are receiving power from a +12VDC source. 2. Verify, using a physical probe or scope, that the voltage across each individual LED is within the limits 12V +/- .5V when the LED is in the "on" state.
The LED module shall provide required voltage to the LEDs and shall support a 210mA current draw for each LED.	Verify by document inspection and by test: 1. LED drivers should be chosen in the design stage to be capable of comfortably supporting at least 210mA * 1.2. Verify using LED driver

	datasheet. 2. Perform analysis mentioned in the requirement above to ensure the LEDs are being provided 12V +/- .5V.

2.3.2 Control Unit Subsystem

The control system is an essential piece of our design as it serves as an interconnect between all of the submodules as well as provides code-generated i/o signals to the drivers and to the Canon EOS-1D. The control system consists of the microcontroller and optoisolators which allow us to safely send signals to the Canon EOS-1D. The microcontroller, chosen to be the Arduino ATMega640, was selected for its large number of GPIO pins and its low power consumption. The microcontroller acts as the brains of the PCB and generates outputs to other subsystems based on what is communicated by the python user interface through the UART connection. For instance, each input pin on the LED drivers is mapped directly to a GPIO pin on the ATMega640, and based on the mode selected by the user, the GPIO pins will send the appropriate signals to the LED they correspond to. Another important process to note is the communication between the microcontroller and the Canon EOS-1D. The physical connection between these elements is an audio jack to N3 type connector. This allows us to send two analog signals to the N3 input port on the camera, with one signal being for the shutter and the other dictating the focus of the camera. The way the microcontroller will send these signals to the connector without damaging the expensive Canon camera is through optoisolators (P.N. 4N35 from manufacturer "Vishay"). The optoisolators ensure that no unwanted power surges affect the camera, but the signals are still able to be transmitted effectively.

Requirements	**Verification**
The ATMega640 shall successfully receive automatic/manual mode command signals from the Python user interface	Verify by test: 1. Using an oscilloscope, probe the RX/TX pins of the microcontroller during operation. 2. Ensure the signals are successfully being

	transmitted from the user interface via a USB-A to Serial adapter cable.
The ATMega640 shall successfully send high/low signals to the optoisolators and the LED drivers at a value of 5V +/- .1V for high signals and .1V +/- .1V for low signals	Verify by test: 1. Using an oscilloscope, probe the GPIO pin on the microcontroller. 2. Ensure the pin is at voltage 5V +/- .1V for high signals and .1V +/- .1V for low signals
The Canon EOS-1D shall respond to both focus and shutter signals received from the optoisolators, with a picture being taken when the shutter signal is received and the focusing adjusted when the focus signal is received.	Verify by test: 1. If the design process is fully complete, utilize the Python user interface to select the automatic mode and allow the mode to run fully. 2. Ensure that pictures (a) have been taken, (b) are successfully stored on the computer for operator review, and (c) are properly focused, with artifact details clear rather than blurry. This requirement should also be tested before the Python interface is complete. Perform the following steps: 1. Define the i/o pins on the ATMega640 that send data to the optoisolators as outputs. 2. Write code that sends a focus signal and subsequently a shutter signal to the camera and flash the microcontroller with the code. 3. Verify by inspection that the pictures have been taken and are at or above the level of quality aligned with the pictures provided by the Spurlock Museum representatives.

2.3.3 Power and Wiring Subsystem

The power and wiring system is an essential hardware component, requiring a robust and durable design to meet the Spurlock Museum's needs. An issue the museum encountered with the previous solution was that the solution failed more and more over time, eventually reaching the point where it didn't function at all. Our team believes that this was likely due to the unorganized and cluttered wiring and connector solution. Our team will be designing the power and wiring system to require 1/8 the amount of PCB connectors that the previous design required. This will greatly reduce the amount of space required for our PCB. We will be accomplishing this by running a common ground line around the dome to eliminate 32 pins needed, where one was previously required for each of the 32 negative terminals of the individual LEDs. For the positive terminals of the LEDs, we will be consolidating the 32 wires necessary into 4 groups (connectors) of 8 wires, thus providing a more organized design foundation. Our team plans to use 8-pin connectors to get the signals from the PCB to wires, which will then be spliced and shielded to connect to the input power and common ground wires existing on the 2-pin LED headers. This subsystem will interact with the lighting subsystem and the control module subsystem, as it will provide power to the LEDs and their drivers as well as the microcontroller.

Requirements	**Verification**
The external power supply shall provide 30W of power at 12VDC +/- .1V.	Verify by test: 1. Using an oscilloscope, probe the output of the 12V external power supply. 2. Ensure that the supply is at 12V +/- .1V
The wiring system shall be modular and organized, such that someone in a non-technology related field could understand the design and fix the system based on a detailed user's manual, if needed.	Verify by inspection and by test: 1. Ensure that the design is visually appealing and the wires can be easily isolated or removed if necessary. 2. After writing a detailed user's manual, seek out a peer in a non-technology related field and observe if they are able to use the manual to fix

	the design if problems arise.
The linear regulator shall provide 5V +/- .3V when given the external power supply's 12VDC input	Verify by test: 1. Using an oscilloscope or voltmeter, probe the input of the regulator to ensure it is receiving the full 12VDC 2. Probe the output pin of the regulator to ensure it is receiving 5V +/- .3V
The wires shall be labeled with the number corresponding to the LED number and the 32 wires themselves shall be of at least 2 colors to further distinguish them for debugging purposes.	Verify by inspection: 1. The user may observe the dome and verify that all 32 positive LED power wires are each labeled with the number of the LED they correspond to. 2.

2.3.4 User Interface Subsystem

The physical dome will be paired with a simple and user-friendly Graphics User Interface(GUI) developed in Python using the Tkinter framework for the interface design and PySerial for establishing a UART serial connection. PyInstaller will bundle the entire package and make it executable across multiple platforms. The GUI will allow users to control the dome's operation from any computer using a USB-to-UART connection with our microcontroller, the ATMega 640. When the program is launched, the software will automatically scan for available communication(COM) ports to identify the ATMega640 microcontroller's unique vendor ID, which will establish a serial connection. The GUI window will offer two primary operation modes: "Automatic Imaging" and "Manual Imaging". In the "Automatic Imaging" mode, Pressing the "Start" button will send a signal to the microcontroller to initiate the pre-programmed imaging process, capturing 32 images with corresponding individual LEDs activated. This allows the user to efficiently capture a complete set of images with minimal user input. In the "Manual Imaging" mode, the user can choose to enable/disable a specific LED and capture a single image. The amount of LEDs enabled at once will be limited to five, preventing insufficient supplied power. This mode will be commonly used for troubleshooting and testing, as well as scenarios in which the user wants a singular well-lit picture of an artifact. Overall, this subsystem will interface with the control module as signals are sent from the computer to the microcontroller for further processing.

Requirements	**Verification**
The GUI shall be functional in "Automatic Imaging" and "Manual Imaging" modes, and correctly communicate with the microcontroller	Verify by test: 1. Clicking each button click triggers the corresponding serial command 2. Python code detects the correct UART port for the microcontroller automatically 3. Error messages displays properly when communication fails
The user experience shall be intuitive and could be operated by a user with little to no training	Verify by inspection and by test: 1. A friend without engineering background can successfully operate the GUI and complete an imaging cycle without assistance

2.4 Hardware Design

2.4.1 Operating Voltage and Power Calculations

For our circuit, our input operating voltage is 12 volts DC, where this 12 volt DC signal will go straight to the LED drivers to power the LEDs based on the mode the user chooses and the flash sequence of the lights. Then, we also have a regulator that will take this 12 volt DC signal and buck this signal down to a 5 Volt DC source to supply power to the microcontroller and all of the LED drivers that it controls. In order to properly power our system with this 12 Volt DC source, we must first calculate the maximum amount of power draw the circuit will have at any given time. We know from the datasheet of the ATmega640 that the maximum amount of current that can flow through it is when it is in active mode. Here the maximum current draw is approximately 15mA. Then, since we know the maximum current draw is 15mA and the voltage at this point is 5 Volts, we can calculate the maximum power draw of this microcontroller using the formula: $P = V*I = 5*0.015 = 0.075$ Watts. Now knowing this, we know that our 12 to 5 Volt DC Buck converter should be able to handle at least 0.075 Watts of power. Through our research, we found the onsemi LM1117MPX-50NOPB 5V linear regulator, which has a power rating of approximately 6.25 Watts. This particular model was chosen because it is small in size so it will not take up much space on our PCB, but it still has a high enough power rating to satisfy our needs and is still large enough that it can be fairly easily soldered onto our PCB. In addition to this, we had to choose LED drivers that were also sized correctly for the power/current draw to the LEDs as well whenever they are turned on. In doing this we know that our lights have a maximum power draw of 2.5 Watts at 12 Volts. This means that they will be drawing a current of $I = P/V = 2.5/12 = 208.333$ mA. Through our research we found that our LED drivers have a current rating of 800 mA which will be more than enough to power our lights. Then, in knowing that the maximum power draw of each one of our lights is 2.5 Watts, we can calculate the maximum approximate power draw of 10 light bulbs being turned on at the same time. We can calculate the approximate power draw of 10 lights being turned on at once through the calculation $P(10 \text{ Lights}) = 10*2.5 = 25$ Watts. We chose to make this calculation based on 10 lights being on at a single time because that was an agreed upon number between us and the representatives of the Spurlock Museum. In adding these two values together, we can see that the maximum approximate power draw of the whole circuit is 25.075 Watts. Then, in adding in a little bit of power for the losses in the wire and non-ideal components, we get a maximum approximate power draw of 25.5 Watts. This additional power consumption is far higher than any true losses in our system from non-ideal wires and components, but we chose to add in this extra bit of consumption to err on the side of caution. Now, in seeing this maximum power consumption, we know that our 12 Volt DC power source must be rated to provide at least 25.5 Watts of power to our circuit. This means that it must be able to provide a current calculated as $I = P/V = 25.5/12 = 2.125$ Amps. We then researched any power sources that provide at least this amount of current at 12 volts DC, and we found the Power Supplies L6R30-120 power source that can be plugged into any standard 120V, 60Hz wall outlet, and output a 12 Volt DC signal that has a power capacity of 30 Watts and therefore a current rating of 2.5 Amps.

2.4.2 Schematic Design

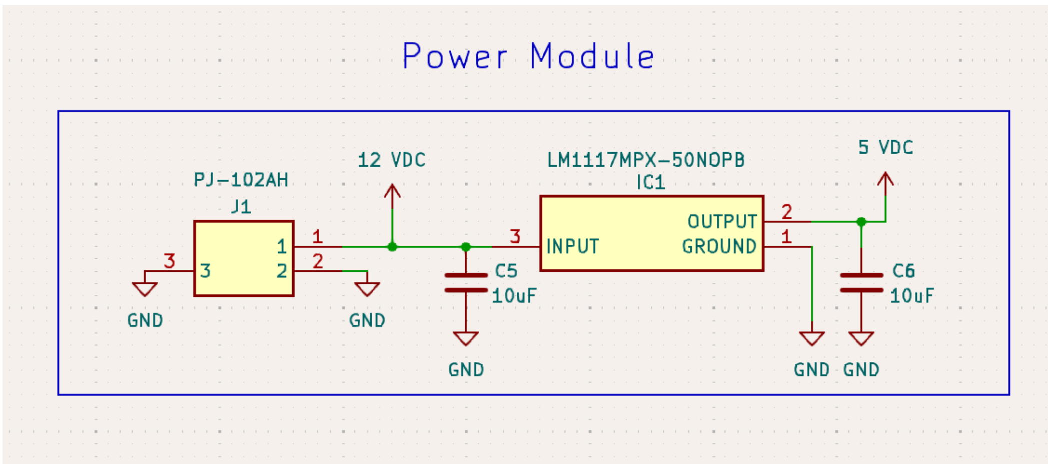


Figure 3: PCB schematic power module

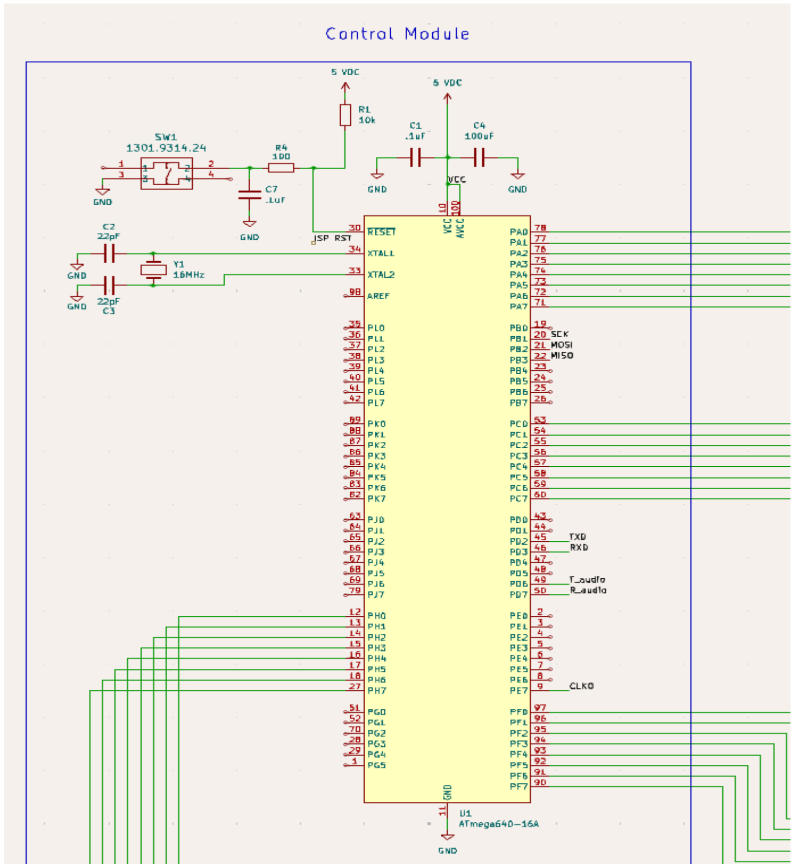


Figure 4: PCB schematic control module

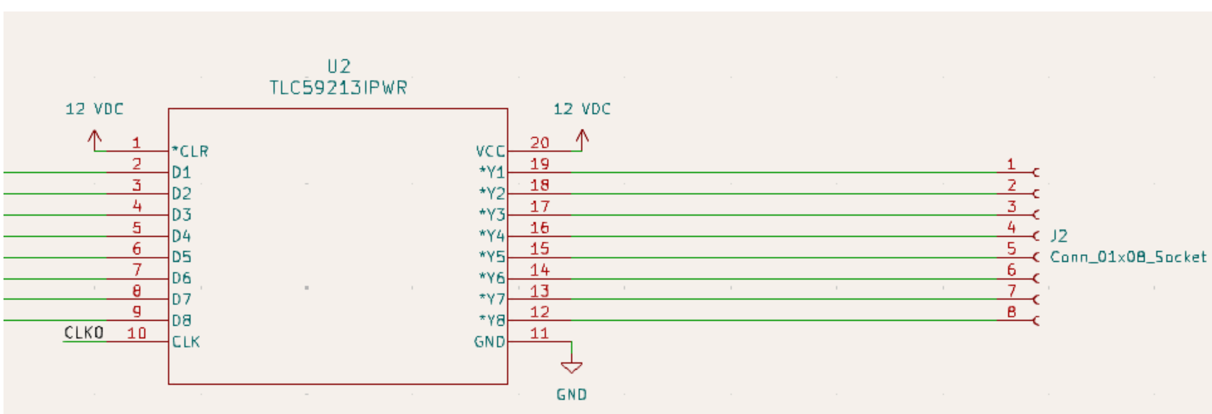


Figure 5: Portion of PCB schematic lighting module

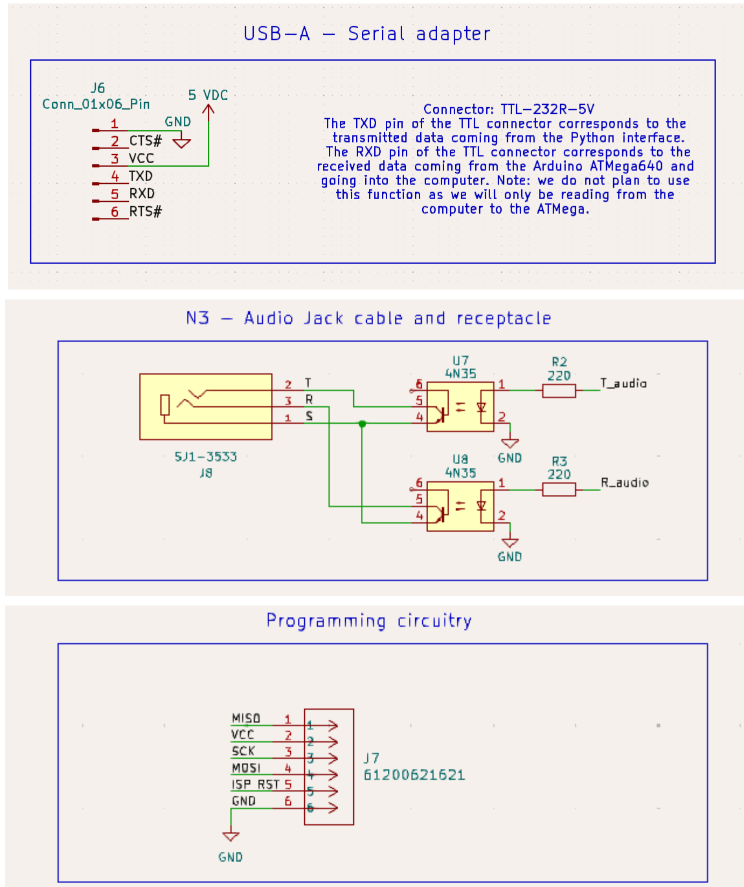
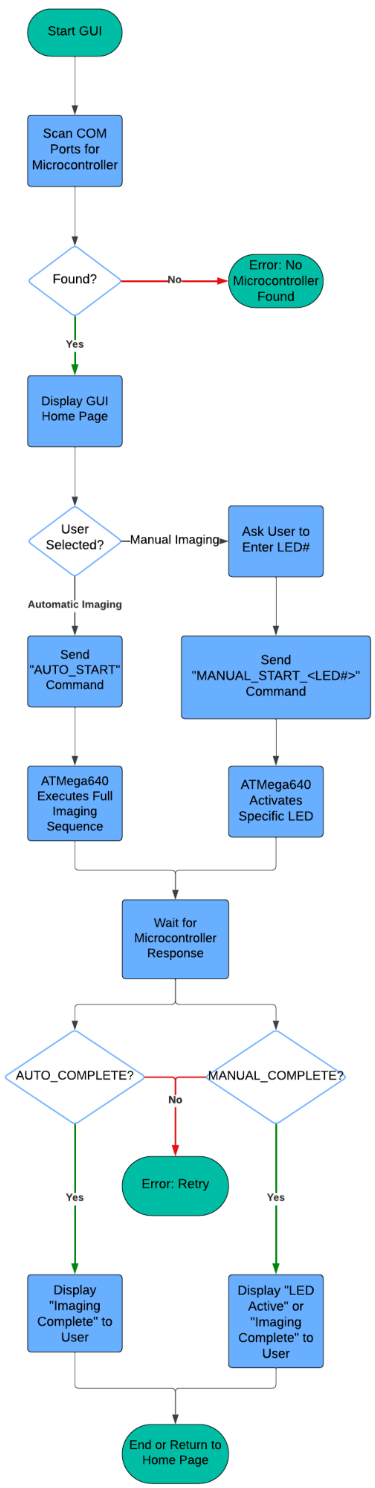


Figure 6: PCB schematic interface subcircuits

2.5 Software Design

2.5.1 System Control, as Determined by User Selection



An important aspect of our project is the software decision making enabled by the ATMega640. It responds to user commands and executes the corresponding illumination sequence of the dome's 32 LEDs. The micorcontroller governs the system by processing external inputs and selecting the appropriate operational state.

The algorithm running on the microcontroller receives the following inputs and determines the appropriate operational state for the dome:

- START: The microcontroller waits for user input without sending any signals to the LEDs. The dome system is in standby mode until an imaging mode is selected.
- AUTO_START: The microcontroller activates all 32 LEDs in sequence, controlling the timing for each LED flash and camera shutter. Once all 32 images are captured, the system enters to the AUTO_COMPLETE state.
- MANUAL_START: The microcontroller turns on a specified LED based on the user's command from the GUI and the system enters to the MANUAL_COMPLETE state.
- AUTO_COMPLETE: Displays "Imaging Complete" to user and returns to the START state.
- MANUAL_COMPLETE: Displays "LED Active" or "Imaging Complete" to user and returns to the START or MAUAL_START state.
- ERROR: The microcontroller stops the imaging process and turns off all LEDs to prevent hardware damage and notifies the GUI.

PySerial is used to establish communication between the Python-based GUI and the ATMega640 through a USB-A connection. The Python program sends serial data (command strings) over the USB connection, which is received by the ATMega640's UART interface. The microcontroller processes the commands and responds with status updates or error codes.

2.5.2 Firmware

This portion of the design consists of the microcontroller and its associated firmware. The main tasks of the microcontroller are to receive the automatic/manual mode signals from the Python user interface, determine the output values to send to each of the LED drivers, and determine the output values of focus and shutter to send to the Canon EOS-1D. As mentioned above, the ATMega640 was selected based on its large number of GPIO pins. We are able to wire all of the 8 input pins on each of the 4 LED drivers with a bank of 8 GPIO pins on the microcontroller. The firmware must first define these pin banks as outputs and then must define their values at each stage of the specified mode of operation. Simultaneously, two other GPIO pins designated for the shutter and focus inputs to the Canon EOS-1D must output correct values for the mode of operation that the user specified. The firmware will be written in the Arduino IDE and will be programmed to the microcontroller via the USBTinyISP component. The user will connect the USB-A end of the adapter cable into their computer and the USB-B end of the adapter cable into the USBTinyISP board. Then, a 6 pin ISP cable connector will route from the USBTinyISP onto the PCB into a 6 pin ISP receptacle connector. Then, since the appropriate ISP pins are connected to the MOSI, MISO, and SCK pins of the ATMega640, the user will be able to flash the code onto the microcontroller.

2.6 Component Selection and Supplier Information

2.6.1 Parts List and Manufacturers

Color code (note: different shades indicate the connector and its associated mating component)

Power system

Connectors & associated cables

Microcontroller & programming elements

Other circuit elements

Description:	Manufacturer:	Part number:	Quantity:	Total price:
12V power supply connector for board	Same Sky	PJ-102AH / CP-102AH-ND	1	$0.82
12V 30W wall mount power supply	Power Supplies	L6R30-120	1	$8.82
5V linear regulator	onsemi	LM1117MPX-50NOPB	1	$0.67
6-pin Male/male connector for USB to	Molex	0022232061	1	$0.52

Serial adapter input				
USB to 6-pin Serial adapter cable	FTDI	TTL-232R-5V	1	$24.90
8-pin connector female receptacle	TE Connectivity	3-640440-8	8	$3.52
8-pin connector to mount to PCB	TE Connectivity	640456-8	8	$3.76
3.5 mm Audio jack to N3 cable	PIXEL	N/A	1	$9.99
3.5mm Audio jack receptacle	Same Sky	SJ1-3533	1	$2.41
ATMega640 Microcontroller	Arduino	ATMEGA640-16AU	2	$20.56
USB-A to USB-B cable	Qualtek	3021001-03	1	$2.78
Programmer for ATMega640	Universal Solder	3647-USBtinyISP-ND	1	$7.79
Programming header for ATMega640 programming cable	Würth Elektronik	61200621621	1	$0.47
Optoisolator	Vishay	4N35	2	$1.48
16MHz crystal oscillator	CTS	ATS16A	1	$0.30
Push button (for	SCHURTER	1301.9314.24	1	$0.37

microcontroller reset)				

Each of the parts above has been verified to be in stock at the links in the table.
Total cost: $89.16

2.7 Tolerance Analysis

From section 2.4.1 above, we know that the maximum approximate power consumption of our circuit is 25.5 Watts and that the maximum power rating of our 12 Volt DC power source is 30 Watts. Therefore in calculating the difference in maximum power provided by the power source and the maximum power draw of the circuit, we can see that this will give us a power buffer of approximately 4.5 Watts, which will give us plenty of extra power as a safety net if it is needed. Also seen in section 2.4.1 above, each one of our LED drivers will have a maximum current rating of 800 mA. However, our light bulbs will only ever draw a current of 208.333 mA through them at 12 Volts, which means that we will have a difference of approximately 591.666 mA between the current rating of the LED driver and the maximum current draw of the LED.

2.8 Cost Analysis

The total cost of our project is $89.16. This is well under the project budget of $150, leaving room to purchase additional parts or expand our design to have increased capabilities in the future, if deemed necessary. Part of the reason why we have so much flexibility in our budget is because we will be reusing the LEDs and associated sockets that the previous group used. This allows us to save money without compromising functionality or sacrificing quality, as these components were deemed to be satisfactory by the Spurlock Museum representatives.

2.9 Schedule

Week	Philip	Austin	Stephanie

9/30-10/4	Research Python libraries and interface with microcontroller	Remove and discard old wiring system	Continue schematic design
10/7-10/11	Design graphical UX using Python tools	Order PCB parts	Finish schematic design and take it to layout
10/14-10/18	Test communication method success between the computer and microcontroller	Assist with power portions of layout design	Ensure PCB passes audit by 10/15
10/21-10/25	Create and/or adapt existing code to get manual mode working	Implement common ground wiring system around the center of the dome	Test the TLC59213 drivers using a breadboard and 5 LEDs from the existing setup and begin drafting microcontroller code
10/28-11/1	Solder components on the breadboard.	Implement positive voltage wiring system for each of the 32 LEDs and splice the wires together	Continue microcontroller code and attempt to program the ATMega640 to ensure a successful connection
11/4-11/8	Finish soldering and set up interface and physical connections between all elements. Complete individual progress report	Work with Philip to interface the power system wiring with the completed, soldered PCB. Complete individual progress report	Finish microcontroller code and complete individual progress report
11/11-11/15	System integration and testing. Debug python/interconnect	System integration and testing. Debug hardware issues as necessary	System integration and testing. Debug microcontroller issues as

	issues as necessary		necessary
11/18-11/22	System integration and testing. Debug python/interconnect issues as necessary. Start documenting the Python User-interface and how to use and debug it in the detailed user manual	System integration and testing. Debug hardware issues as necessary. Start documenting the wiring and power system and how to debug issues in the detailed user manual	System integration and testing. Debug microcontroller issues as necessary. Start documenting the microcontroller processes and how to run the system in the detailed user manual
11/25-11/29	Thanksgiving Break		
12/2-12/6	Test manual mode with many different combinations of LEDs turned on, and with different total amounts of LEDs turned on	Test the automatic mode and ensure all 32 LEDs light up and sequence properly	Review images produced by the system and communicate with Spurlock representatives to ensure the photos are of desired quality and the system functions as they would desire
12/9-12/13	Final presentation and final paper		

2.10 Risk Analysis

While there are no significant risk factors in our design, we have taken action to mitigate risks of smaller proportions in our design process such as: simplicity of our design, faulty user interfaces, and damage to equipment due to incorrect power, voltage, and current ratings. Some of the actions we have taken to mitigate these risks include: making sure our 12 Volt DC power source and 12 to 5 Volt DC Buck Converter have adequate power ratings to handle the power needing to be provided to their connected devices based on our calculations made in sections 2.4.1 and 2.7, making slight overestimations on calculations of power consumption of our circuit to err on the side of caution for the true power rating needed by the 12 Volt DC power source and the 12 to 5 Volt DC Buck Converter, choosing a wire size with a current rating high enough that the wire will not fail/burn up during use, choosing a microcontroller that has a larger amount of i/o pins than needed to ensure that the design is as simple as it can be and that it will be easier to debug if any issues arise, and choosing an LED driver that has a large enough current rating to support/supply the LEDs with adequate power without failing.

3 Ethics and Safety

There are a few ethical and safety concerns that are relevant to our project. One priority is keeping the safety of the user in mind and avoiding any injury to the user due to faulty instructions or faulty equipment. Another priority is to preserve the artifacts without significant degradation. To ensure this priority is met, we must keep the lights at an intensity level of at or under 100 lux for artifacts made of highly sensitive materials, such as textiles, paper, photographs, and more. For other less sensitive artifacts, such as those made of wood or undyed leather, it is recommended that the amount of light exposure be kept at or under 200 lux. Another important aspect to consider is safety to the user. Keeping this in mind, we are going to construct our project in compliance with the guidelines provided from OSHA 1910.303 that detail general electrical safety, and we will also be providing an instruction manual for how to use our project. Using said instruction manual will help explain how to replace any faulty equipment that could pop up through repeated use, and this will help the user troubleshoot any program errors that could arise as well. To ensure that the light intensity is kept at a safe level, we will confirm that the light bulbs used output a light intensity that is lower than the artifact-safe threshold and that the power being used by the LED’s is under their rated power consumption. This will ensure that we will not burn up any lights, create a light intensity higher than the ratings of the bulbs, or create a light intensity higher than the rated damage levels of the artifacts. As far as overall ethical concerns go, our team does not have any additional concerns about noncompliance with any guidelines mentioned in the IEEE code of ethics, and we will ensure that our project solution remains in line with these ethical standards throughout the duration of our project.