Kermotiva

An IQP in Interactive Public Art

Advisor: Professor Joshua Rosenstock

Students: Elliot Borenstein IMGD Art ’11, Yilmaz Kiymaz IMGD Art ’09,

Bradford Lynch ME ’10, Cynthia Weiler ME ‘10

Table of Contents

Statement of Intent and Goals

Kermotiva is an interactive sculpture that uses light, text, and sound to display emotion. The emotion displayed by the sculpture is a function of its base emotion, combined with input collected from members of the community. The base emotion is created from the analysis of financial, weather, and RSS feeds, providing a global perspective to the overall output. The second half of the emotion comes from a web interface that collects data on how users are feeling and the colors and words they associate with their emotions. This data is then displayed on the sculpture using RGB LED's, LED screens, and a xylophone. The result is a snapshot of the current emotional state of the community in which the sculpture has been installed.

The goal of the project is to utilize this sculpture to provoke thought on how we treat one another and raise awareness on the effects of our actions. If people walk past the sculpture and see that it appears sad, they might recognize that their community is in a slump. They may then be inspired to go out of their way to raise the spirits of their community.

Group Member Roles

Areas of Focus

Due to the breadth of technologies that must be employed to successfully meet our goals for the project's social impact the areas of focus and design will be distributed to the team's specialists. Those areas include:

- How to get the input
- How to parse the input into a stream we can use
- Visualization of the model
- Building the model
- Designing/implementing the moving parts
- Implementing electronics in the model (lights/patterns/gobo-rotators)
- Sound design and implementation
- LED screen design
- Design and implementation of the online interface
- Design and implementation of the kiosk interface
- Circuiting and programming of an interface between the data stream(s) and the creature

Team Members' Specialties and Interests

Each member of the team brings unique skills to the picture and these will need to be utilized in order to be successful. In addition to these specialties many members of the team have specific areas of interest that they wish to learn more about and therefore have taken on tasks in those areas. Here is an outline of the areas that each team member will be responsible for:

- Brad
  - Sculpture pre-visualization
  - Solid Modeling
  - CAM Programming for machined parts
  - Machining
  - Assembly and Construction of the piece
- Cynthia
  - Sculpture pre-visualization
  - Building the model
  - Designing/implementing moving parts
  - Sound design
- Elliot
  - Sculpture pre-visualization
  - Programming
  - Circuitry
  - LED implementation
  - Lighting design

- Yilmaz
  - Sculpture pre-visualization
  - Programming
  - Interaction design (website, kiosk etc.)

Acknowledgements

We would like to thank all of those who made this project possible through their contributions.

- Professor Rosenstock
- WPI PC Shop
- WPI Network Operations
- DelSignore Electric
- Humanities & Arts Department
- Arduino Development Team

Abstract

Kermotiva is the visualization of the emotional state of the community in which it is installed. This was achieved through the use of data analysis on RSS feeds, weather statistics, and financial data, resulting in a base emotion. This is then combined with user data to create the final emotion displayed by the sculpture. The data was visualized by projecting varying colors, displaying words provided by users on LED screens, and playing songs associated with the emotion being displayed. This provides the members of the community interacting with Kermotiva information about how they are behaving and how they are treating one another. By providing the population with this information, increased awareness of the effects of an individual's actions on the community, as a whole, is created.

Executive Summary

The goal of Kermotiva was to create an installation of interactive public art that would invoke discussion on the impact of an individual's behavior and how it affects the community they are a member of. After researching many other installations and assessing the methods of interaction employed, a list of the top interactions we wanted to incorporate was developed. The next step involved the consideration of different methods of data collection, input, and analysis. After determining the factors most significant to an individual's emotional, personal, professional, financial, and global outlook, we focused on creating a model to visualize this data. The group concluded that the form be reminiscent of the human figure so that members of the community could relate to it. It was also in consensus that the sculpture should be abstract enough that it wasn't just a simple model of a human. The final design is an amorphous blob with three torsos emerging from it.

Collection of financial, news, and weather data was performed to build a base emotion of the sculpture. A web interface was developed to collect information from users on their present emotional state; this is based on word and color association. The resulting data from the web interface is then combined with the base emotion to create a snapshot of the community. This ensures that even with no user input the sculpture will operate and change.

This data is then visualized on the sculpture through the use of RGB LEDs, LED screens, and sound. The RGB LEDs bathe the skin of the sculpture in light, shifting from color to color depending on emotion. LED screens are used to display the words gathered from the web interface. Finally, sound production is used to create tones and progressions that are associated with specific emotions.

The result is a creation that displays the current state of health of a community at any given instant. The resulting awareness of the emotional state of one's colleagues is invaluable if they choose to act on the information provided to them.

Preliminary Research

Summary of Process

In order to develop our ideas for the Interactive Public Art project, we began by researching various websites to find relevant projects that had been done in the past. Each of us was given a particular website related to data visualization, and we found three projects on our sites that we found most interesting. Professor Rosenstock also went through a box of gadgets such as fans, microcontrollers, and solenoids to get us thinking about what cheap resources were available to create animated visualizations. This helped to give us some basis for what to think about when we tried to come up with our own ideas.

The next step was to brainstorm on our own and come up with 10 ideas or “metaphors.” This meant coming up with ideas for different data streams that might be usable, and possible ways to make them into something visual. After sharing our favorite ideas with the rest of the group, we met as a group to come up with more fleshed out ideas. We took four categories (weather, RSS feeds, financial data, and environment), and for each of them we came up with three ways to visualize different data related to that category. In addition to this, each of us spoke with people from around campus to find out what data might be available to us at WPI. We met with people from Network Operations, Plant Services, the Library, and Chartwells.

At the following meeting, we talked about which ideas we liked the most from our group brainstorm, and talked about what we found out from the people we met with. The Library was unable to release any data to us, Plant Services never responded to our requests to meet with them, Chartwells had a limited amount of information that they keep track of, and Network Operations offered up an almost unlimited amount of data that could be used. However, due to new ideas proposed at this meeting, we chose not to make use of these resources.

During the meeting, the idea of a life form which could express emotion through the use of shape, color, and sound was introduced. Its base mood was to revolve around “Real World” events; the current state of various financial markets, weather information, and selected RSS feeds. Members of the WPI campus were to be able to interact with it via email. People could write to it and, based on the tone and structure of the letters, its mood would change.

The group met again for another brainstorming session and we explored this idea in great detail. There was disagreement on visualization; some thought it would be best to keep the form abstract and others believed that no one would empathize with it unless it had a face. On our own, we attempted to brainstorm what this life form could look like.

At our next meeting, we discussed our ideas for the look of the life form. In addition to this, there was some doubt about whether to continue with exactly what we were doing. The opinion surfaced that nobody would care enough about something that was ultimately a robot, and another opinion was that it was too much like a life sized Tamagotchi or Gigapet. However, the idea of having a life form that would be cared for was not completely left behind. Some discussion was made on the idea of representing emotions of the people on campus, rather than giving the thing its own emotions. The website EmotionallyVague (http://www.emotionallyvague.com/) provided inspiration. There was interest in creating a website where people could anonymously write about how they felt at any moment and what made them feel that way. Then, the visual element would display the feelings from each entry and there could possibly be some scrolling text that let the viewer know what it was that made the person who submitted that particular entry feel that way. There was concern that the piece was going to lack interaction.

We thought some sort of kiosk could help provide on-site interaction. This way, a person could possibly look at past entries and maybe enter in some of their own information through simple menus. If a keyboard was included, the user could additionally write a response to an entry, or maybe even write their own if they felt comfortable in public. It was also discussed that security is a much greater concern once a user has access to a keyboard.

Assessment of Existing Interactive Art

There are two installation of interactive art that are of particular interest to us. The first is the installation *Blinkenlights*. This is an installation in memory of a famous German hacker, Wau Holland. The project converted the top 8 stories of an office building into a giant monochrome screen. The public could interact with it by submitting animations online, playing *Pong* via cell phones, or by submitting love letters through cell phones. This project served as inspiration for the LED screens that were implemented in Kermotiva. The web interface that allowed people to paint and animate the building also served to provide us with direction for Kermotiva's web interface.

The second installing is called Vectorial Elevation by Rafael Lozano-Hemmer and was built in Zocalo Plaza, Mexico City. The project converted the sky into a giant user controlled light show by using robotic spotlights. Users could log in to the web interface and design lightshows and then submit them to the installation. The web interface again provided us with some direction.

Proposed Types of Interaction

In order to create the social link between the sculpture and the community in which it is installed, there must be interaction between the two. The proposed methods of generating this interaction are:

- Submit words/phrases that are then parsed for output
    - scroll common words on the LED screen
    - keywords can affect its mood
- Users submit entries on their mood/emotions
- Being near it/staying near it by virtue of a motion sensor
    - getting attention makes it happy
- Talking to it
- Provide a kiosk near it with just a screen and a trackball for direct interaction

- Provide a website interface similar to the kiosk, but with certain unique features added and removed to provide a slightly different input experience
- Sliders and buttons on the website and kiosk
    - users can control how it is affected by the data
    - users can control colors, sounds, movement, and how it interprets various feeds
- Provide optional login ability for people to track their interaction

Proposed Methods of Output

- Embed a scrolling LED screen into it
    - potentially in the head
    - potentially wrapping around the it in a circular fashion
    - shows words from user input and feeds
    - shows facial feature(s) when people are nearby/follows them with its 'eye(s)'
- Place lights within it to project color, glow, and/or gobo patterns onto its skin
- Produce some sort of audio output
    - music
    - noise, as if to represent a voice
- Vary the speed of its physical movement based on input
- Setup a webcam that streams to the website for people to see it remotely
    - will LED scrolling that uses persistence of vision fail when viewed over a webcam?

Potential Types of Movement

- Breathing
- Swaying (ex, boredom)
- Up and down/bouncing (ex, excitement)
- Use fans to blow some part of it with varying speed
    - fans from within if it has a cloth exterior
- Use simple cams to make pieces move cyclically

Potential Materials for the Base Form

- Store bought (unpopular choice)
- A wire structure with a fabric covering
- A vacu-formed plastic shell

Parts and Resources Needed

- Solenoids
- LED matrices
- LED scroller electronic parts and controller
- Motion or location detector
- Webcam(s)
- Cams or material to make them
- Kiosk parts
    - materials for the outside container
    - computer, screen, and trackball

- Server or server space for our use
- Choose the feeds to be used for the baseline
- Find out if there is a vacu-form machine on campus
- A place to install the final product

Design and Development

For the technical backend of this installation, we used the following technologies:

- Web Server – A Site5 web server that runs the Kermotiva site and the Web Interaction applet.
- Client Machine – An old Pentium 3 machine, generously donated by the WPI PC Shop. It runs Ubuntu Intrepid Ibex (8.10) as its operating system.
- Arduino – see Microcontrollers section.
- Processing – A framework that allows for easy interaction with FTDI-based serial devices such as the Arduino. Using this technology, we can interface with the microcontrollers in the sculpture by running multiple code sketches on the client machine.
- Java – Powers the data gathering application which oversees the high-level state of the installation and makes sure it has access to the latest data available.
- Perl – Used on the Web Server to store user data.

Web Server and Sculpture Communication

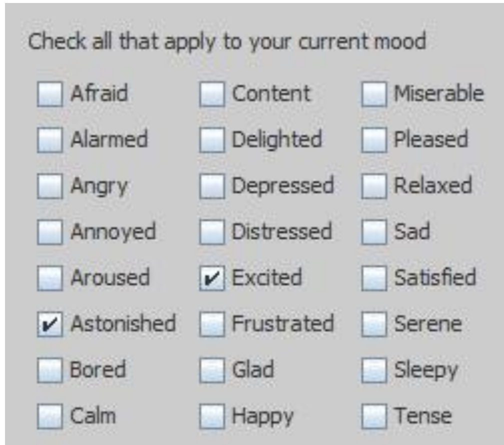
As an Interactive Art Installation, our project needed a means of collecting emotional information from users. The two ideas we had from the start were to use a web-based interaction model and to have an on-site kiosk. We later dropped the kiosk idea for the amount of complexity it would bring to the project and the costs associated with such a setup. This allowed us more time to polish the web interaction and we ended up with a cross-platform interactive Java applet that works on all major browsers.

Web Interface and Server

At its core, the web interaction applet consists of a series of "panels," each with its own set of components to allow for a specific type of interaction. Having multiple panels allows us to collect a wider range of emotional data while also enabling us to give the users the option to not participate in some of the panels. In the initial version of the web interaction model, all panels were on the same page. Later on, we decided that separating each panel into their own pages would make it easier for the user to focus on a certain type of interaction at a time. In the final version, we have five panels. Let us look into what each one is responsible for:

- Mood selection panel

This panel allows users to select one or more moods that they are in right now. There are 24 moods listed with checkboxes next to them.



The reason we did not allow for users to type in their own moods was the difficulty of having to deal with profanity since the data that they enter will in some form be displayed on the sculpture itself. Users can choose not to select any moods in this panel and still continue with the rest of the interaction scheme.

- Color selection panel

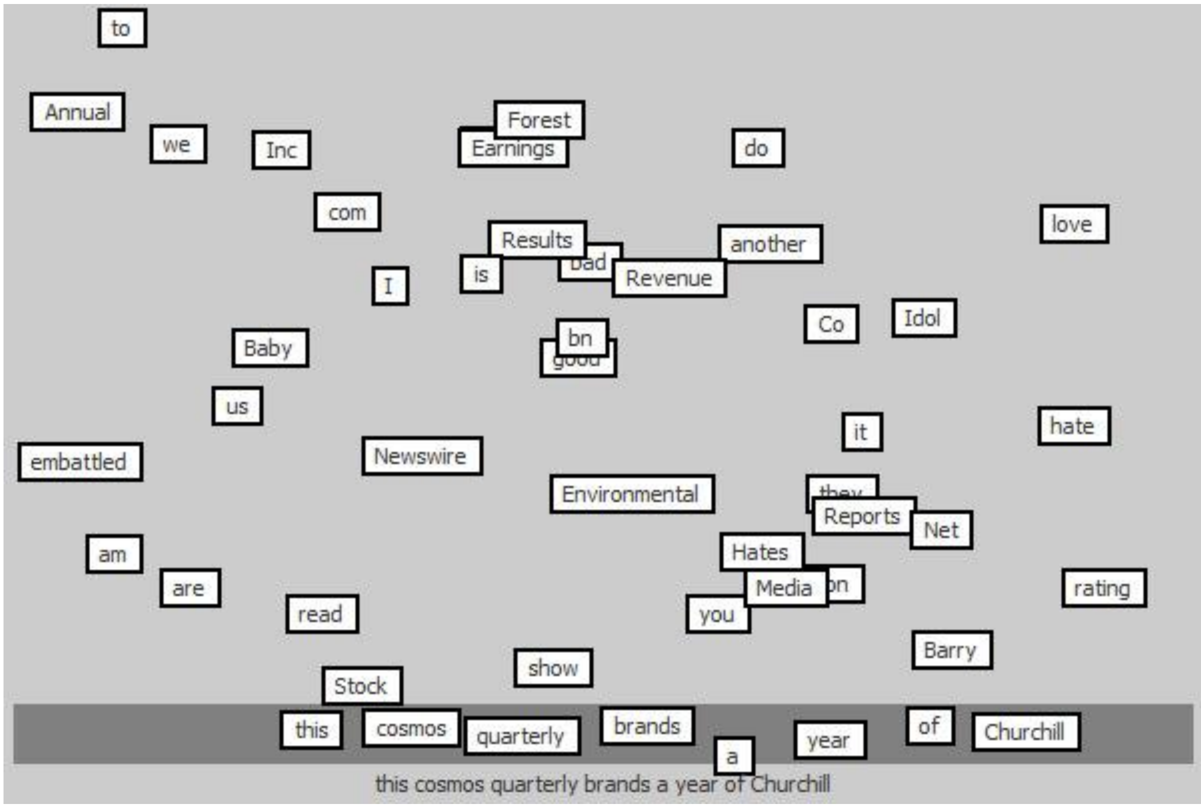
This panel allows users to select a color that applies to their current emotional state. Ten fundamental colors are provided for the sake of quantification of emotional data. Colors are chosen by clicking on the colored rectangles and only one color can be chosen. Users can choose not to select any color.

- Care selection panel

In this panel users can select up to three topics they care about. When three topics are chosen, all other checkboxes are disabled until one of the chosen topics is unchecked. According to the users' selections, certain Yahoo! RSS feeds are accessed and random words from the Title and Description fields of these feeds are grabbed to be used for the next panel. Obtaining the words from Yahoo! RSS feeds makes it easy to avoid profanity. The connection to the feeds is initiated on a separate thread to prevent the user interface from being blocked. This allows us to show a loading screen to indicate that the applet is still functioning. Users must choose at least one item to continue.

- Sentence forming panel

This panel is populated with the words gathered from the RSS feeds in the previous panel. Users can drag the words to the sentence forming area (darker grey) to create their own sentences. The resulting sentence is formed by these words, as read from left to right. Since the gathered words are chosen at random and the exact words that the users are looking for might not be available, this interaction cultivates creativity in the users' input. However, some default words such as pronouns and auxiliary verbs are inserted along with the words from the feeds to make it easier to form sentences with a meaning. Users can choose not to form a sentence.



- Feeling drawing panel

In this panel users can draw their emotions on a seven by thirty matrix of squares. Each square represents one LED on the sculpture LED screens. Since the LEDs we are using to display these drawings are single-color, the drawings are black and white. Users can hold down left mouse button to draw in black and hold down right mouse button to draw in white. Users can also undo and redo their drawing strokes using the Undo and Redo buttons on the panel. Both the Undo and Redo stacks have a history depth of ten. A Random button on the panel fills in all squares randomly to give users inspiration and a starting point. Users can choose not to draw anything.

Once the users complete the drawing, they can click the Submit button to submit their input to installation's data stream. When the button is pressed, the web interaction applet communicates with a Perl script residing on the same server to store the users' input into a text file. A separate text file exists for each day the installation is running. Along with all their input, the users' IP addresses are also collected to prevent bots from spamming the web interaction applet with automated input and from affecting the data stream of the installation.

Client Machine

The installation houses a client machine that oversees all the subsystems of the sculpture. A Java application that is running on this machine is responsible for gathering the necessary data for the subsystems. We can break down the execution of the Java application into four stages; data gathering, data analysis, state management, and processed data output. Let us talk about the details of each stage:

- Data gathering

The application gathers two types of data; user data and external data. User data is the data collected through the web interface. External data is data collected from outside sources such as Yahoo! Weather and the stock market. The application continuously polls the data sources to keep itself updated with the latest data available. When data cannot be gathered from a specific source (due to network failures or other unforeseen issues) the application recycles old data if available. If old data is not available, the specific data source is not included in data analysis until new data from it becomes available.

Getting specific bits of information from external data sources can be difficult. For example, the Yahoo! Weather feeds have a lot of XML data that we do not need. To facilitate the extraction of desired data, helper classes were written to pick out bits of data from a large string. The functionality of these classes are generalized enough to allow for adding more external data sources to the system without having to write custom data extraction code.

- Data analysis

Both user and external data is analyzed to calculate the overall emotion of the installation. User data provides information such as color and moods chosen by the users of the web interface. External data provides information such as the high and low temperatures of the week and the state of the weather such as sunny, cloudy, rainy etc. All these variables are mapped to a two-dimensional emotion graph. At the extremes of the X and Y axes, we have emotions such as happiness, excitement, frustration, and sleepiness. Then, depending on the source of the data, the mean or the mode of all entries are calculated. Each data point has a weight associated with it to fine-tune the result. This allows us to give more importance to certain inputs (such as color choice) than others if necessary. Once the analysis is finished, the data is given to the state management system.

- State management

The application is also responsible for managing the high-level state of the installation. This means deciding which types of data to display on the sculpture, in what form, and when. In the current implementation, there are four states that the installation can be in: pure user data, pure external data, user data compilation, and user and external data compilation. User data states are the most preferred since they have the highest potential richness in information. Depending on the availability of user data, the application cycles between these four states and outputs data accordingly. Each state also has a preset time limit which forces the application to change to another state. We believe this will make the installation more interesting for viewers by switching the represented data around to provide a variety of visual states.

- Processed Data Output

Once the data has been analyzed and modified according to the current state of the installation, the application outputs this data to a text file on the client machine that can then be read by a Processing sketch. The reason for this is that Processing is the most suitable framework to use when communicating with an Arduino from a desktop application. Using Processing, we create a link from the data we have on the client machine to the data that is made available to the master Arduino that controls all subsystems using serial communication. One exception to this model is the Arduino that controls the LED screen subsystem. That Arduino is directly connected to the client machine and is communicated with using a Processing Sketch as well.

Master Arduino

The master Arduino is connected to the client machine through a USB connection. Since the client machine is an old Pentium 3 computer and only has 2 USB ports, it was necessary to use an Arduino as master and overseer of all other subsystems that are controlled by slave Arduinos. A serial communication protocol was designed to implement the communication between the master and slave Arduinos. Details on this protocol and the communication scheme can be found in the Serial Communication Protocol section. More details on the Arduinos can be found in the Microcontrollers section.

Emotion

Emotions are not easy to understand using computing algorithms. Even as humans we have trouble telling what another person is feeling at a given moment. This difficulty in deducing emotions from inputs such as sentences convinced us to quantify emotions in a more basic fashion. As you have seen in the explanation of the web interaction applet, we give users a limited range of options when asking for input. This allows us to assign pre-computed values to each of these inputs and then use these values to calculate an average emotion for each submission we receive. Aside from mapping the inputs to a two-dimensional emotion graph, we also came up with 7 cardinal emotions. These emotions and their corresponding values (in the -1 to 1 range, with accuracy to two decimal places) are as follows:

Emotion	X value	Y value
Pleased	1	0
Excited	0.33	0.76
Afraid	-0.74	0.54
Frustrated	-0.92	0.06
Sad	-0.76	-0.74
Sleepy	0	-1
Calm	0.83	-0.63

This table provides ease of selecting an output when there is only a handful to choose from. For example, we use these emotions when deciding which of the several tunes to play in the Sound Controller system. If, on the other hand, we need to provide a system with wide-ranging values, the -1 to 1 range of values for each average emotion becomes more useful.

Electronic Subsystems

From the start of the project, it was clear that several electronic systems would be required. As the team had no ECE majors, implementation of these systems was slow in the beginning. However, as more research was done into the design of embedded systems and microcontrollers, the full system of control and output became clear.

Microcontrollers

Initially, we evaluated two types of microcontroller: The PIC by Microchip[1], and the Arduino[2] development platform.

PIC

The PIC microcontroller initially seemed to be the best choice; they were very inexpensive, and seemingly easy to program. It was anticipated that we would need several different systems, each being controlled separately, making the PICs and attractive option. However, early attempts to use these chips were unsuccessful. Any attempt to create anything but the most basic circuits seemed to fail, and a large portion of B term was used attempting to integrate these chips into our system. The problems were exacerbated by the lack of good compilers for the PIC16 chips we were using. No official free C compiler exists for the PIC 16 series, meaning the choice was between assembly and crippled versions of more expensive compilers. Over winter break, it was decided that the PIC was not viable, given the time left.

Arduino

Alongside work with the PIC in B term, tests were done using an Arduino. Rather than just a microcontroller, the Arduino is a development platform with an Atmel[3] microcontroller at its core. It

---

[1] http://www.microchip.com

[2] http://www.arduino.cc

comes pre-assembled on a circuit board complete with broken-out headers, integrated clock, built in pulse-width-modulation (PWM), and a bootloader to allow for easy code uploading over a standard usb port. The most useful feature, though, is an open source programming environment that uses a simple C-like architecture, making programming very intuitive and fast. Included with this environment is a vast wealth of knowledge and pre-written libraries for many different applications; several of these libraries and code-fragments were used in the final project.

The main downside to using the Arduino platform was cost. While a single PIC or Atmel microprocessor can be purchased for und $5, an Arduino board costs $30-$40. This was deemed to be acceptable, and the final system uses a total of 4 Arduinos: two standard Arduinos and two 16Mhz Arduino Pros by Sparkfun Electronics[4].

Serial Communication Protocol

To communicate between the microcontrollers within the sculpture, it was necessary to develop a simple, but robust, serial communications protocol. We were constrained to 8-bit messages, and we knew at least one subsystem would need seven of those eight bits for the data it was being passed. So, we decided that all command data would begin with a high binary bit of 1, such that all commands would be of decimal number 128 or higher. We eventually determined that we had few enough commands to keep them at or above decimal 248. The commands are (not that not all were used in the final project implementation):

248 – (Common) START

249 – (LED Screen) CONTINUE

250 – (Common) SPEED

---

[3] http://www.atmel.com/

[4] http://www.sparkfun.com

251 – (RGB LED) COLOR

252 – (RGB LED) ALT COLOR

253 – (RGB LED) DIRECTION

254 – (RGB LED) PATTERN, (LED Screen) CHUNK, (Music) SONG

255 – (Common) END

A basic communication for the RGB LED system would look like the following, using “I” as the initiator of the connection, and “S” as the receiver:

I sends a START, and waits

S receives START

S sends START back and goes into receiving mode

I receives START confirmation

I sends COLOR

I sends 2, representing the color chosen

I sends PATTERN

I sends 3, representing the pattern to play

I sends END, and waits

S receives COLOR, and prepares to be given a color choice

S receives 2, and changes the display color accordingly

S receives PATTERN, and prepares to be given a pattern to display

S receives 3, and sets the output pattern accordingly

S receives END

S sends END back and resumes normal operating mode, using new parameters

I receives confirmation of END from S

This protocol and its implementation work well enough for the project, but it does have its flaws. Primary among them is that each controller enters a blocking loop during the communication. This means that the system being updated stops running for as long as the communication occurs. This also means that, should there be an error in the communication, one or both microcontrollers may become locked in the loop waiting for input, requiring a manual reset to resume normal operation. A better approach would be to have the controller continue to operate normally, and to maintain variables as to the state of a serial communication. Such a system would allow the controller to continue its primary function, only stopping momentarily to receive new data, and never falling into a loop that would require a manual reset.

PCB Design

Due to the lack of electronics experience in the group, it was necessary to quickly learn how to design printed circuit boards (PCBs). To this end, a series of tutorials by Sparkfun Electronics were used to gain the basic knowledge of PCB design. The software for the design was Eagle by Cadsoft[5]; this software is available for free, with only restrictions on size, making it ideal for use on the project. The actual fabrication of the PCBs was through AP Circuits[6], who provide rapid prototype creation with a two to three day turnaround time. The only restriction was that all boards must be ordered in multiples of two, which was easily worked around by designing the boards to work in pairs. All boards were designed to use only two layers.

LED Subsystems

In total, there are two LED based subsystems: scrolling LED screens, and an array of RGB LEDs.

LED Screens

The LED screens consist of 2 boards attached below each of the three heads of the sculpture, along with a board to scroll through the columns and an Arduino. Each of the boards on the heads consists of two single-color 5x7 LED matrices, for a total grid of 20x7 on each head (*figures B-1 & B-2*). The text being displayed is mirrored across all three heads due to design problems that cause noticeable display

---

[5] http://www.cadsoft.de/

[6] http://www.apcircuits.com/

problems when text is scrolled across more than 2 of the boards. Each board receives two sets of control signals, one from the Arduino and one from the control board.

The control board consists of a series of five 4017 decade counters, each stepping through ten outputs (*figures B-3 & B-4*). A sixth 4017 chip is used to step through each of the other five chips to allow for up to 100 outputs, with 50 from a single board. The board was designed such that a second board could be added, and the signal from the sixth chip on the first board would take the place of that chip on the second board. Each board receives three signals from the controlling Arduino: a master-clock, a slave-clock, and a reset signal. When the slave-clock ticks, the currently active chip moves forward by one output. When the master-clock ticks, the board moves on to the next slave chip. Finally, when the reset ticks, the board resets to the first output on the first chip.

In addition to controlling the clock and reset pins, the Arduino outputs seven signals to represent the seven rows on the display. As each column is activated by the control board, the Arduino outputs the data for that column for a fraction of a second, before moving on to the next column. This process is repeated many times a second to give the illusion of a static image. This method is known as persistence of vision (POV).

The controlling Arduino is linked directly to the on-site server through Processing. The server converts any data to be displayed into an array of bytes, and stores it in a text file. Processing then reads this text file and sends the bytes over a serial connection to the Arduino. Due to the size of the serial buffer on the Arduino, processing limits the output to the Arduino to 256 bytes, including one byte representing the length of the data. This byte array is stored in the Arduino's on-board EEPROM, and read back in sequence, with the seven high bits of the byte being displayed.

This system for multiplexed POV display, using 4017 chips, works well for small display arrays. However, issues arose the longer the size of the display. It was common for the scroll speed to slow drastically as more of the screen space was used, and flickering of the image was common. The only solution was to speed up the duration that each column is displayed in a given pass to 1 millisecond. This decision came with its own problems, in the form of the screen becoming far dimmer than it had been. Even with the speed benefits of this change, the maximum displayable size was two boards (20 columns), meaning that each head had to display exactly the same image. We believe that the issue is due to the intensity of the matrices, coupled with the relatively low speed of the Arduino's microprocessor; however, more research and testing would be required to verify this.

RGB LED Array

The RGB LED array consists of an Arduino, two 32 channel pulse-width-modulation (PWM) boards, and a series of 36 5mm RGB LEDs attached in rings throughout the body of the sculpture. The LEDs are common cathode, sold by Futurlec[7]. Due to constraints by the circuitry of the control boards, it was necessary to use only common cathode LEDs. These were the lowest priced RGB LEDs found that had mostly even output across all colors, while remaining sufficiently bright to be seen in a lit room. The LEDs were originally intended to be connected in large rows, each containing 10-20 LEDs. Unfortunately, due to choices made in the design of the control boards early on, it was unfeasible to power this many LEDs. Instead, they are run in rows of six LEDs, which are further divided into three groups of two LEDs. Each of these groups requires 3 PWM channels, one for each color, meaning that each row requires 9 channels for complete control. As the control boards output a maximum of 64 channels together, this meant we were limited to seven rows, though we eventually chose to utilize only six due to time.

For mounting, the LEDs are hot-glued to the fiberboard structure of the sculpture. Those on the main body are connected in pairs back to their control signal, directly. The three torsos of the body, however,

[7] http://futurlec.com/

need to be detachable to facilitate movement through smaller doors. To this end, the LEDs on the torsos are wired into RJ-45 jacks, which then connect to the control boards through broken-out cat-5 cables.

Control for the LEDs is provided by a pair of boards based around the Texas Instruments TLC5940 16-channel, 12-bit PWM chips (*figures B-5 & B-6*). Each board has two of these chips, allowing for 4096 step precision PWM for all 64 channels. These chips are designed to sink relatively large currents, but not enough for the number of LEDs that were planned. For that, each board has four 8-channel ULN2803 Darlington arrays, which are designed to sink significantly larger currents and voltages than the PWM chips. Finally, since the PWM chip outputs are sinks, and the Darlington arrays require sourcing signals, 32 PNP transistors are on each board to convert the PWM signal to a form usable by the arrays. All 32 channels are output through four RJ-45 jacks.

Control for the boards is provided by an Arduino. This Arduino has a series of pre-programmed patterns to be used in conjunction with the LED rows, along with pre-programmed PWM values for various colors. The Master sends a signal to the Arduino, containing a color and a pattern. The Arduino then runs the pattern, using a free Arduino library for the TLC5940 chips, developed and maintained by Alex Leone[8]. The library utilizes several low level clock functions to generate the necessary signals to operate the TLC chips, and sends the output data over an SPI serial connection.

The TLC chips can be quite useful, especially when used with an Arduino and the free control library. Unfortunately, the designs for this project were not ideal. While the Darlington arrays can sink large currents, they cannot cope with as much power as was needed for the number of LEDs we needed to use. This meant that we were required to split each row into three sub-groups, and use nine channels per row where only three should have been used. Instead of placing the Darlington arrays on the control boards, they should have been placed on separate boards; in this way, a single PWM signal could be sent

---

[8] http://code.google.com/p/tlc5940arduino/

to several arrays at once, allowing one channel to control as many LEDs as may have been needed. Designs for these boards were eventually created (*figures B-7 through B-9*), but the control boards had already been made and it would have been too costly to fabricate additional boards for this purpose.

An even more optimal solution would have been to remove the arrays entirely, and replace the small PNP transistors with large power transistors, possibly in the same configurations just described for the arrays. This design would be much simpler and less prone to catastrophic failure. It would have been modular; allowing for more LEDs to be added as needed, and also allowing for faster, cheaper and easier repairs should anything break.

One other issue to be resolved was that of the LEDs themselves. It was difficult to find good quality RGB LEDs at reasonable prices; they tended to cost a significant amount per unit, only be available in large quantities, or be of such low quality as to be unusable. The LEDs from Futurlec are the best mix that we could find, but they still have problems. Mainly, the green emission chip, while bright, is overpowered by both the red and the blue. Similarly, the only other supplier found to meet our supply needs provided LEDs whose red chip was severely underpowered when compared to the other two. In the future, it would be necessary to either find a better supply of LEDs, or to do research into grouping together individual LEDs of each color in a way that still mixes the colors into one cohesive output. It would also be useful to look into brighter and more diffuse lighting methods in general.

Sound Controller

The Sound controller consists of a simple signal chain. The system is robust enough that it could work with many different types of sound producing systems, which was important for the project. Initially, the sound would be created by striking custom tuning forks with solenoids controlled by this system. The final version has a similar setup, but the tuning forks are replaced by a modified xylophone.

The signal chain starts at an Arduino, controlled by the Master Arduino. The Arduino is pre-programmed with various songs and tones for each of the 8 notes that can be played. Any given tone or song is made up of a byte or series of bytes. Each bit in the byte represents a note on the music device, with a 1 signifying that the note should be played and a 0 signifying otherwise. To create a song, a series of bytes is written, along with a delay between each. Each song or tone is then assigned a number, and the Master Arduino sends that number to the system, causing it to be played.

When a byte needs to be played, the Arduino shifts it out over a basic serial connection to a 74HC595 8-bit shift register (*figures B-10 & B-11*). The connection is simple, using a clock pin and a data pin to send the data. Once the entire byte is sent, the shift register is sent a latch command, causing it to output the byte. The eight outputs then activate some number of relays on a pair of boards, each containing four relays (*figures B-12 & B-13*). These relays connect the solenoids of the system to a 12V power supply rail, causing them to strike their respective notes.

The sound system was the most reliable and simple system of the three and it is easily extensible to larger systems using cascading shift registers. The only significant issue comes from the inherent power drain caused by striking one or more of the solenoids. The peak drain, with a maximum of over 2A if all eight strike, causes disruptions to the other systems. On the LED screens, the disruption manifests itself as a flickering each time a solenoid fires. The RGB LEDs, however, are disrupted to the point of not working when the sound system is allowed to play for an extended period. The ideal solution would be to include an isolating circuit of some kind, or a sufficiently large capacitor to even out the power drain. Due to time constraints, however, we opted to include a second power supply for the sole purpose of supplying a 12V rail to the solenoids, thus isolating them from the rest of the systems in the sculpture.

Sound Creation

Ball Machines

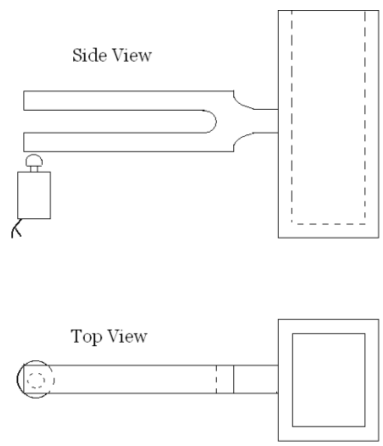
One of the originally favored ideas for putting a sound element into the project was a roller ball track. This would have been a steel ball that would descend on a track, hitting various sound-producing features along the way. Examples would be wooden blocks, chimes, or plastic tubing. It would have been interesting to have several tracks for various emotional states we wished to represent. The track would have been made out of some heavy gauge wire that could have been soldered together. To bring the steel ball up to the beginning of the track, a belt and electric motor hooked up to a gondola like carrying system would have been implemented.

Tuning Forks

The initial plan for the sound production system was to utilize tuning forks. Tuning forks have a nice tone to them and come in every musical note. Since purchasing quality tuning forks is expensive, the intent was to machine them in WPI's shop using CNC machines. To make the forks, they first had to be designed and modeled. Tuning forks work by vibrating at a particular frequency, which produces sound at a particular musical note. Their vibration is a function of the dimensions of the tines and the elastic modulus of the material they are made out of. A limited amount of information found on the internet from research papers implied that the frequency depended on the cross-sectional area and the length of the tines. The equation is $f = \frac{1}{l^2} * \sqrt{((E * A)/\rho)}$ f is frequency, *l* is length, E is the Elastic Modulus of the material, A is the area of the cross-section, and ρ is the density of the material. Models based on these dimensions were sent to the shop and it took about a month to get the first fork back. Not too long after getting the first part back, a conflicting equation was found that was based on more legitimate looking research. The new equation was $f = 0.161614 * t * v/l^2$ where *t* is the thickness of the tine normal to the striking side, and *v* is the velocity of sound within the chosen material. After deriving this calculation from looking at the research it was based off of, it was determined that it has an error that can throw the pitch off by an entire note. In the meantime, more models had been sent out using the earlier equation, but with much thinner tines, which turned out to be a mistake because it takes away from the purity of the note. The manufacturing of the remainder of the forks was halted when it was discovered they weren't right. There was a struggle to get help from various administrators in the machine shop, and the time came when it was no longer feasible to wait so long to get prototypes machined. It was necessary to learn to use Computer Aided Machining software so that the tooling paths and programming for machining the forks could be developed. This was done successfully and tooling paths and programming were completed for 9 forks- a full major scale plus an octave and a minor third.

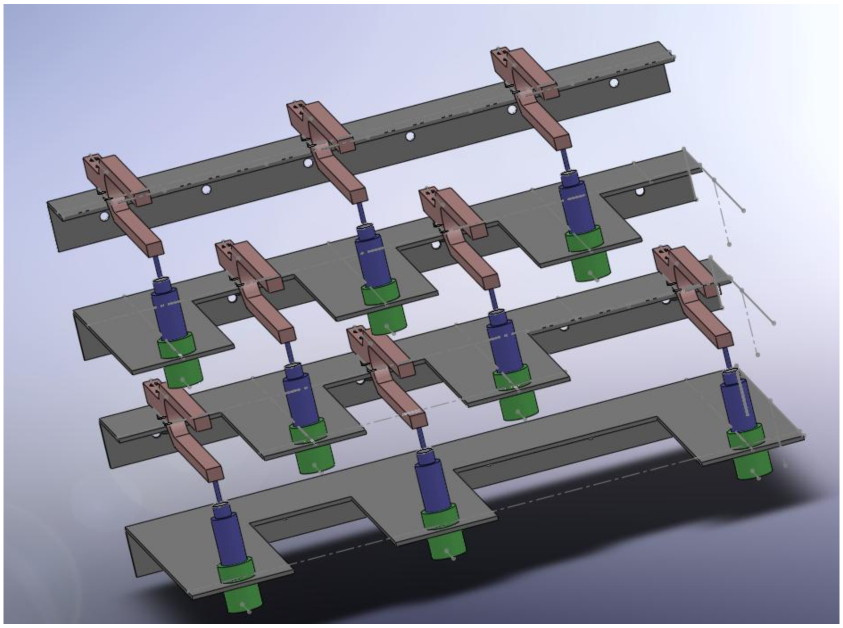
The forks require something to strike the tines in order to produce a note. Having a mechanical system was the best way to carry it out because then it would not require active participation from the user. After researching various ways to have the forks play mechanically, it became apparent that the best idea was to use linear pull solenoids to directly strike the tines. The bars within the solenoids would have been topped with small rubber caps to give the optimum impact and reduce clang (the high pitched note one hears when striking a metal object with something else hard.) These could have been easily and cheaply purchased at a local tool supply shop. The solenoids would have been wired up to a board which would give them commands saying when to fire.

Mounting the forks was another issue that needed to be dealt with because they would not play if the tines were lying against anything else. The first plan was to mount them on a series of simple amplification boxes made from a hard wood such as oak. A diagram of what this might look like can be seen in **Error! Reference source not found.** below.



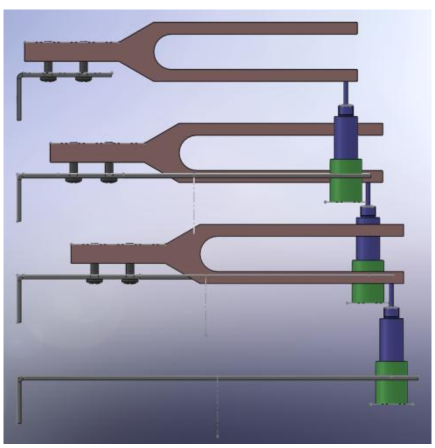
Tuning Fork, Amplification Box, and Solenoid Arrangement

Since the boxes could also experience damping, they could be placed on a soft surface such as felt. The amplification boxes may have been unnecessary because they may not have had a large amplifying effect, but would have provided a little volume and somewhere to mount the forks. This idea was abandoned in favor of using one big amplification box which would house all of the forks and solenoids within. At first this seemed much simpler because then only one box would need to be used instead of 8, but as it turned out it was quite complicated to find a way to nicely mount all of the components within the box. In order to model this system it was necessary to learn to use some features SolidWorks independently within a couple of days to a level equivalent to taking a whole course on the program. Because of this, there was a good amount of trial and error that came along with getting accustomed to the software that slowed down the completion of the assembly significantly. Eventually a completed assembly of the mounting system within the box was designed. A screenshot may be seen in **Error! Reference source not found.**. The forks would be mounted on aluminum brackets (these were drawn out on raw material but never cut,) with the solenoids placed in holsters. The solenoids are shown in blue and the holsters are shown in green.



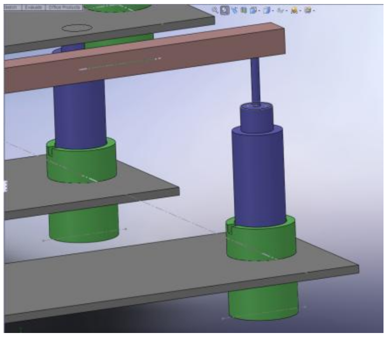
Entire Assembly

To allow for optimum flow, the forks would be mounted in a tiered manner as seen in **Error! Reference source not found.**.



Tiers

Since the solenoids need a place for the striker to rest below, custom holsters were designed to be machined out of aluminum. These would have allowed the striker to return to a rest position without falling out of the solenoid. This arrangement is shown in **Error! Reference source not found.**.



Solenoid in Holster

The tuning forks would have been attached to the brackets using two holes through the ends of each fork. A bolt would be placed through the hole loosely with rubber grommets supporting the fork in the distance between the bracket and fork handle. Rubber grommets would have allowed more vibration for the forks than just bolting the ends down.

Xylophone

Due to time constraints, the xylophone was eventually implemented as the form of sound creation. Utilizing solenoids an electrical system was developed to allow for the autonomous playing of the xylophone.

Motion Elements

Along the road to conveying emotion we felt that it would be vital for our sculpture to exhibit movement. The major design element of the sculpture is that it is an amorphous blob with three torso/head combinations emerging from it. This being the case the movement of the sculpture needed to be more slow, fluid, and soft as opposed to the typical quick and abrupt motions of humans. Since the emotion of the sculpture was dependant on feeds that change with time, this sort of motion would appear to almost slow time. In order to create this movement it was decided that bladders strategically placed throughout the sculpture, which could then inflated or deflated would be the best option. The key to this system would be a constant supply of air pressure. Due to time and budgetary constraints, none of the proposed and researched motion elements were implemented in the final sculpture.

Air Pressure

In order to get the air pressure needed a compressor along with a tank of suitable size so that the compressor could run for a short period of time and then remain off until all of the pressure in the tank was used. The compressor then feeds to rails; a high volume low pressure one and a low volume high pressure rail. The HVLP rail would allow for larger expansions to take place that don’t need a large amount of pressure while the LVHP rail would allow for smaller expansions requiring higher pressures.

Bladders

The main emotions that the sculpture will display are anger, joy, love, and sorrow. The bladders can play a role in conveying each of these through converting air pressure into motion. The actual movement that correlates to a given emotion requires different designs implement both low and high pressure air.

Low Pressure

Each of the torsos would have a bladder system to create the appearance of breathing. We established two variables in order to help quantify the breathing action; breathing rate and breath volume. By altering the values for these parameters we can control the emotion that the sculpture is exhibiting; fast and small for anger or slow and large for sorrow.

High Pressure

Sometimes when we are feeling sad we feel as though we have put on a lot of weight or when we are happy we feel skinnier than we really are. Bladders around the base of the sculpture will allow us to increase the size or relative “blobbiness”. If we are simply changing the size it will also help to key viewers into the fact that something is changing causing them to examine it closer. Light and other features can then be used to help convey the emotion or change as well. These would still be HVLP but would require higher pressures due to the increased stretching of the cloth.

Surgical Tubing

Connected to the LVHP rail will be ancillaries that need higher pressures to run. One example would be veins running through the sculpture that could pulse to illustrate a heart beating and depending on the speed and tenacity of the pulses it could help to convey anger or joy.

Motor Driven

Cam Elements

As an alternative to using airbladders to simulate breathing in the sculpture it was thought that we could utilize motors to power a cam motion system. Due to time constraints this avenue was not developed very far.

Skeleton Design

After deciding to use a stretchy fabric as the skin of the sculpture a skeleton needed to be designed and built to give the fabric the shape desired and to support all of the internal subsystems. The skeleton needed to be rigid enough to support the stress being applied to it by the fabric and subsystems but without being too bulky of heavy. Several different methods of construction were discussed.

Wire Construction

The wire construction method was desirable due to the possibility of a very light skeleton. It was also thought that it would be easy to fabricate due to the formability of the wire. In the end this method was not used due to its lack of strength.

MDF Ribs

By taking slices or cross-sections of our sculpture's design and then cutting the profiles out of MDF it was thought that we could loft the cloth over the profiles to get our desired shape. This method was desirable due to low material costs, durability, and ease of fabrication. Despite producing the heaviest skeleton of all of our construction options this was the chosen fabrication method. We chose it because the benefits of using MDF far outweighed the extra heft of the wooden construction.

Vacuum Forming

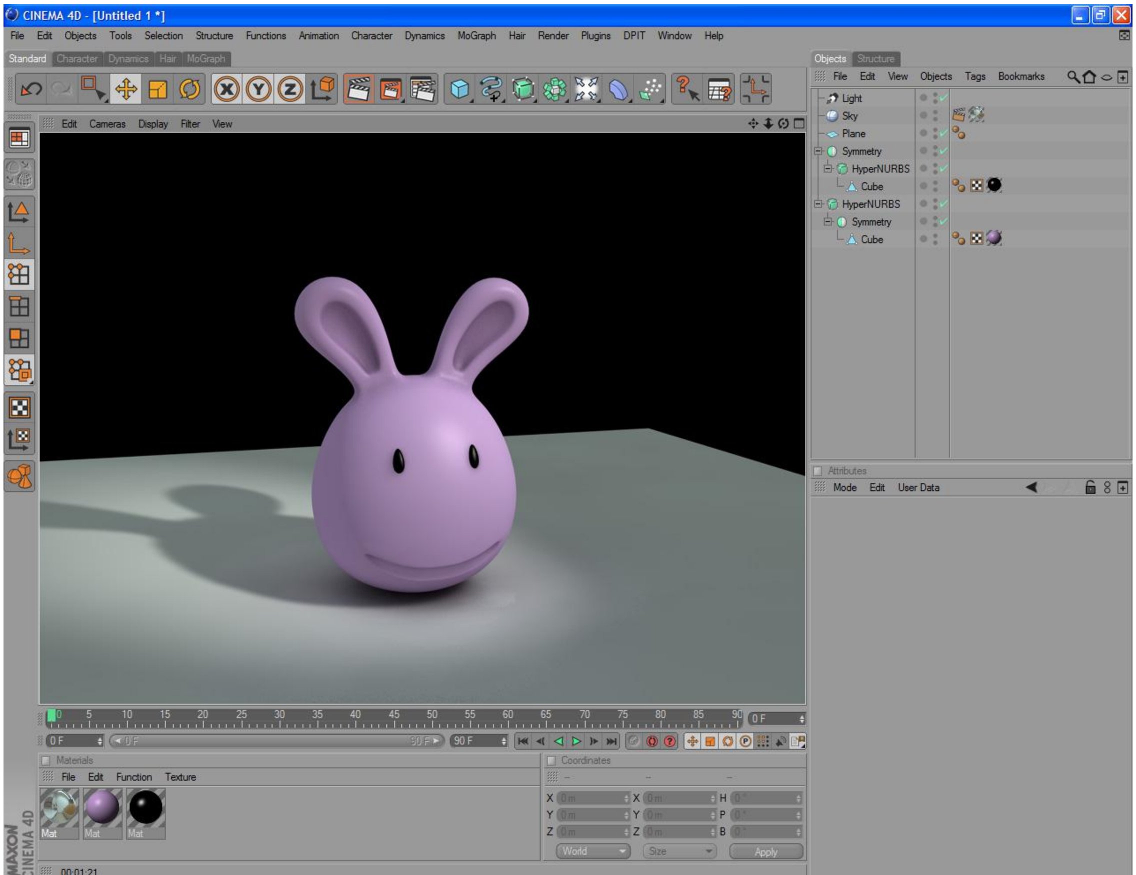
The vacuum forming method was never considered as a possibility for fabrication of the entire skeleton; only select portions of it. In the end we did not utilize this forming technique as there was no need for its implementation.

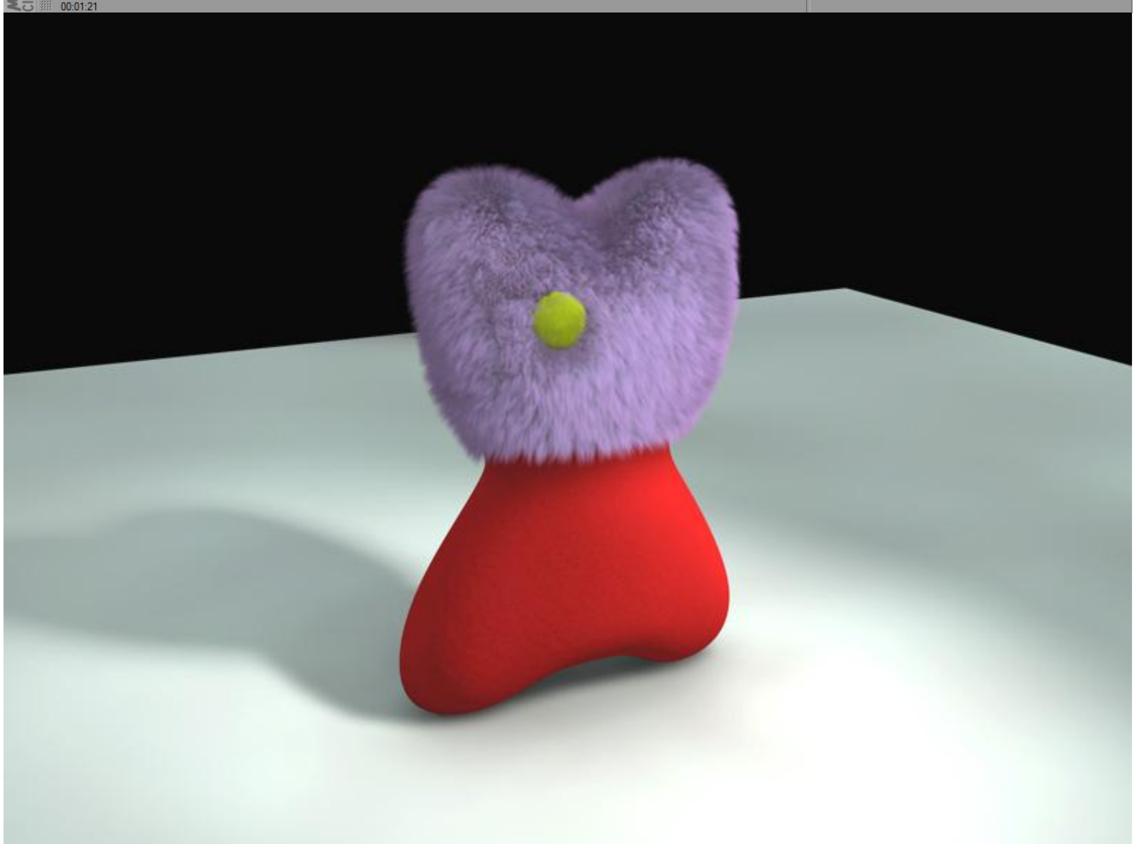
Store Bought

After deciding that we wanted to have a humanoid sculpture we considered a couple of prefabricated options from various stores. They each took the form of a human, some more abstract than others, and had semi opaque material that we felt would be good for projecting light onto and creating a diffuse effect.

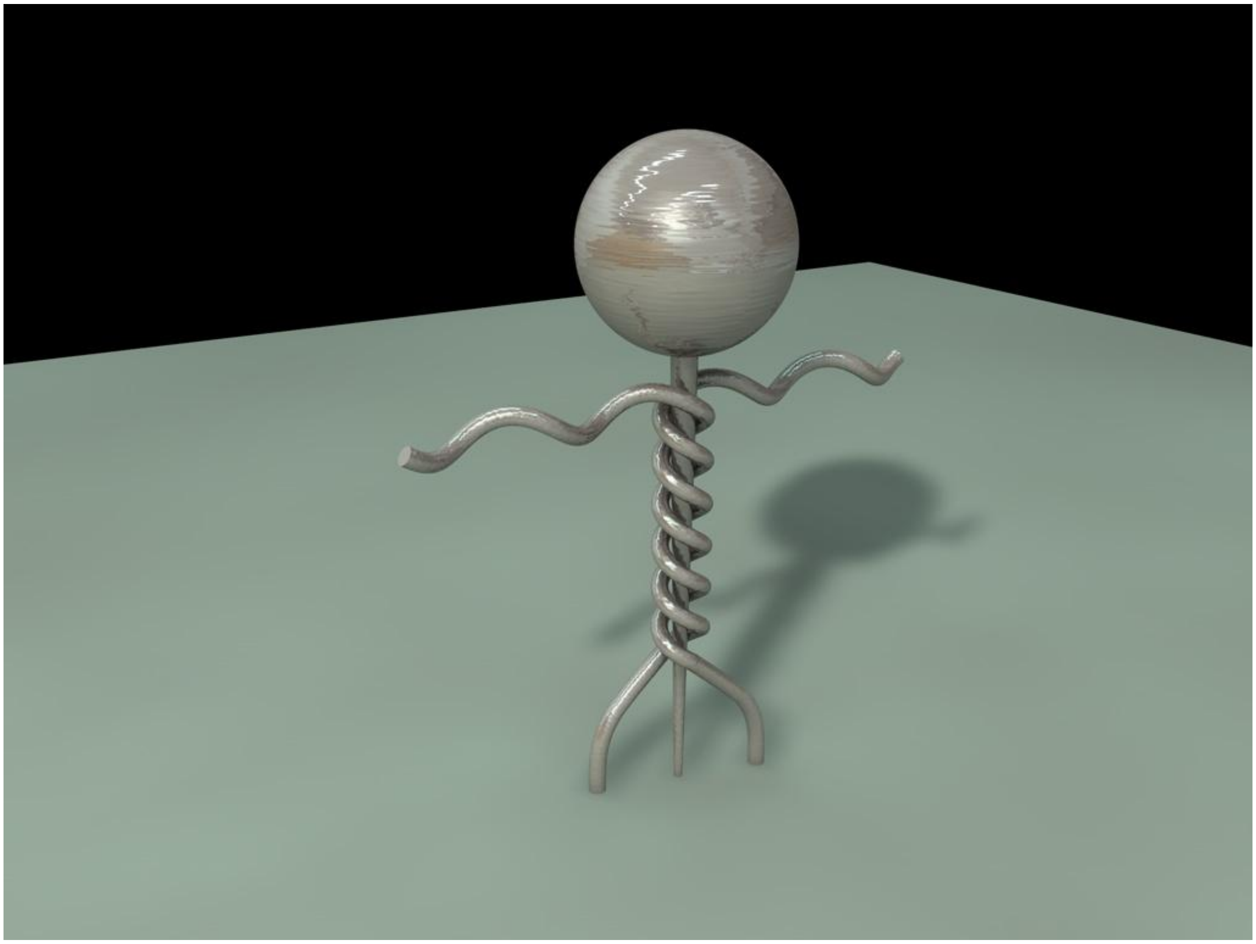
Evolution of Physical Form Design

The initial desires for the form of the sculpture were to have an abstract sculpture that people could relate to and would be an effective form for conveying emotion. Some of the first design considerations are shown in here:

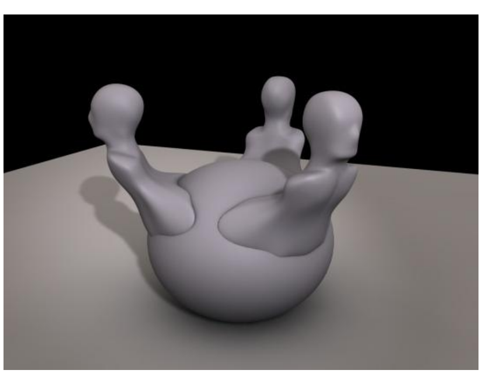

CINEMA 4D - [Untitled 1 *]
File Edit Objects Tools Selection Structure Functions Animation Character Dynamics MoGraph Hair Render Plugins DPIT Window Help
Standard Character Dynamics Hair MoGraph
Edit Cameras Display Filter View
Objects Structure
File Edit View Objects Tags Bookmarks
Light
Sky
Plane
Symmetry
HyperNURBS
Cube
HyperNURBS
Symmetry
Cube
Attributes
Mode Edit User Data
Materials
File Edit Function Texture
Mat
Mat
Mat
Coordinates
World
Size
Apply
MAXON CINEMA 4D
00:01:21



Eventually it was decided that these were either too animal like or too abstract, respectively. With further discussion of the form a more human like figure was created.



While moving closer to the final design goals it still needed refinement. These refinements led to the development of an amorphous blob with three torsos and heads. The following is a progression to the final form of the sculpture:



This form has been met with much enthusiasm. Some find it creepy and others feel engaged by it. The general consensus is that the project's goals for the design of the form have been met.

Construction

Implementation of Web Server and Web Interface

The web server and space has been provided by Elliot Borenstein; the web interface will be hosted there. Provisions have been made so that there is a safe location within the sculpture for the server that is localized at the installation.

Assembly of LED Subsystems

Assembly of the RGB LED system was a laborious task requiring many man-hours due to the need to individually mount each LED at a specific location within the sculpture. Each RGB LED required 4 wires to be run to it, which furthered the work required to install the LED's and documentation of the system and wiring.

The LED screens, while more condensed than the RGB LED's, required a significant amount of work to complete the wiring and the runs back to the controllers. This was in part due to the complexity of any LED screen and also due to the design requirement that the location of installation, the torsos, be easily removable from the rest of the sculpture.

Conversion of Xylophone for Automated Sound Creation

The decision to move from a set of fabricated tuning forks to a purchased sound creation device was made due to time constraints. An 8 note xylophone was purchased to fill this need. It was then disassembled and then solenoids were mounting to the base so that they would strike a note and it would ring. The use of a single solenoid per note allows for chording and minimized the mechanical complexity.

Skeleton Fabrication

Thanks to the material choice of MDF the fabrication of the sculpture was carried out without the extensive use of machine tools. A render of the final design of the sculpture was used to create drawings for each rib. Using a scroll saw the ribs were then cut out of the MDF sheet. Pine 2 by 4's were used to connect each rib and also to attach the ancillary systems. The use of wood allowed for ease of mounting and adjustment of each individual piece of the sculpture.

Sculpture Skinning

Cloth is the medium of choice for covering the sculpture. A spandex mix was chosen from a variety of stretchy materials that were sampled because it had the best mechanical properties in terms of flexibility and ability to diffuse light from the LEDs. The purpose of the cloth was to give the sculpture more of a feeling of unity, while still being able to give the three entities their own definition. The cloth is one piece at the bottom, then sewn so that the upper bodies and heads are jutting out by themselves.

Smooth sweeps between the entities make an easy an aesthetically pleasing transfer from one body to three separate ones. The gatherings on either side of the heads eliminate as many wrinkles as possible while at the same time creating a transition into the sweeps between the bodies. This might even imply a sort of mind connection between the three figures. White coloring lends to the neutrality the sculpture should possess before it is activated to show various data. A picture can be seen below.

The lower half is attached at the bottom with Velcro, is stapled at the top, and may be unzipped. The upper half is attached to its base with Velcro. For the bottom half, the sewing was done entirely using a sewing machine to attach the zipper and all the pieces of Velcro. The top half was more difficult because of the amount of material that had to be put together. It was completed with a combination of machine sewing, hand sewing, and fabric glue. Sewing with spandex is quite a process and it took a while to find settings where the machine would sew any seams that would not fall apart and the machine would not jam and require complete re-threading. After doing some reading on other people's experience with the fabric, the process became somewhat easier with the thread only breaking about every 4-6 inches and hardly any jamming. The entire center piece of fabric was attached using super fabric glue which dries very quickly and can also be washed without coming undone. A considerable amount of tweaking was done after everything was put together and the glue was used for all of these modifications since it was the best option. After the cloth was put together and tweaked as best it could be, the excess fabric was trimmed away in order to give it a cleaner look when light would be shown through it. In the end the whole thing should be able to withstand being washed if it gets dirty from being on display in a public area, and can be removed enough so that any components within the structure may be easily accessed.

Conclusions

The primary objective of our project was to create an interactive sculpture that would generate awareness about the emotions and the effects of an individual's actions throughout the community in which it is installed. The conclusion of the project has marked the achievement of a solid portion of this goal. In its current state, the sculpture successfully represents the emotional input of the community and factors in the live feed data to the base emotion being displayed. The web interface for user data collection is fully developed and robust enough for public use. All subsystems are known to work, and are fully capable of displaying the output necessary to represent the emotional state of a community. Finally, the sculpture itself is an accurate representation of the initial conceptual art created prior to construction.

This project is the first of its kind on the WPI campus; no other project group has attempted to create an interactive public art piece before. While not perfect, the project has succeeded in its initial intentions, and is a good example for further project work in this area. We have made great strides and have implemented a complex and interesting system of components capable of provoking discussion wherever it is installed. It is our hope that the sculpture will enrich the community, and encourage the creation of more interactive art projects on campus.

Post Mortem and Recommendations

Though the sculpture could be installed as it is, significant development work is still needed before it can be presented or installed in an ideal state. While improvement is needed across the board, much of the work is concentrated in two areas.

The web interface has been developed technically and is ready for full feature use. One area of improvement is further refinement of the visual design. The web interface is only one section of a entire web page; the other sections detailing what the project is and our goals also need further development.

The embedded systems have proved their functionality but need further refinement due to some issues that only presented themselves when the whole system was assembled. The first major issue is due to the lighting control system failing and in turn resulting in over heating of the entire embedded system. This caused the majority of the chips to need replacing and delayed any further testing of the project. The sound subsystem was a major contributor to this failure due to the peak power draw of the solenoids. In order to fix this issue they must be isolated from the other systems through the use of a second power supply. The recognition of these issues and the solutions derived from this knowledge, once implemented, will fix a majority of the barriers preventing prototype demonstration.

The major issue currently associated with the skeleton is the attachment of subsystems to the main cylindrical base. The first weak attachment is the casters to the base. More screws of larger diameter and longer length should be installed to strengthen this area. Another area is the connection of the torsos to the base. This could be improved by increasing the number of fasteners in that area. While the skeleton itself is mature there are some issues with it that are innate of the design. The first one is associated with the hard edges of the ribs. Foam was attached to these in an attempt to soften the edges but it wasn’t completely effective and the resulting appearance is less organic than desired. During the development stage there weren’t set points for installation of the subsystems so it was thought that the use of wood would facilitate the eventual mounting of the systems. While it is easier, and actually feasible, to mount the systems to the wood than metals this area could be improved. One solution would be the incorporation of dedicated mounting areas designed for specific systems.

The cloth is the second area that is in need of significant development. The number of seams in the cloth needs to be reduced and the seams need to be aligned to the flow of the contours of the sculpture. Difficulties in sewing the cloth led to the use of glue to bond the pieces of cloth to one another. While this is acceptable for a prototype the cloth needs to be redone, without the use of glue, so that the seams blend with the sculpture and stretch tightly around the skeleton preventing wrinkles in the cloth. Further work should also be done to optimize the diffusion of light on the skin being emitted rather directionally from the LEDs.

Once these refinements have been completed the sculpture will be ready for professional installation. This, along with recognition and interaction from the community, would mark the successful completion of our goal. The next step is to get the sculpture installed on campus at WPI so that the development and testing of the systems can commence.

PQP

This projected was extended to a fourth term, D-Term, for 1/6 of a credit. The amount of work done during A-Term was equivalent or greater than a PQP and it has been the concensus that a PQP would have been beneficial. A major effect of a PQP would be to get the creative gears in motion before sitting down and jumping right into the project. This would have allowed for more direction from the start and more time to think about the goals of the project. Creative thought is a finite process and therefore takes time to occur. Getting the process underway earlier in the development of the project would only have increased the quality of the final product.

Define better from the start the web and server technologies that will be used

In the research phase of the project, understanding the technical requirements of the interactivity model will greatly benefit the team when time comes to implement the system. Looking into several different solutions and technologies for backend systems and comparing them against each other for ease-of-use, functionality, and costs is important in order to not stumble later on. An example for this in our case was the use of Perl scripts to store data on the web server. We did not know that we would need to use them until halfway through the project and when it was time to use them we had no prior experience with the language and had to hack a lot of things together to make it do what we wanted it to do.

Research how emotional data can be properly analyzed

Reading research papers on the subject of emotion analysis will help the team grasp the concept much more easily and will directly help with the implementation of analysis systems. Also, it is important to start this research at an early phase for any meaningful processing to be realized later on.

Make sure someone on the group has good ECE experience

Due to the nature of the project, many embedded systems were necessary. The fact that the group did not contain an Electrical Engineer was a significant hindrance to the project's progress. While we were able to learn the appropriate material and move forward, the inclusion of an ECE major would be greatly beneficial to any further work in this area. Not only will that person have the knowledge to design systems on hand, but he will also be better able to troubleshoot issues as they arise.

Try to consolidate micro-controllers

The project contained 4 Microcontrollers for three systems. Initially, we had planned to implement more systems, but these were cut due to budgetary and time constraints. Also, the initial plan to have a single Arduino connected to the server, controlling the slave Arduinos was made less useful due to the need to provide a direct connection between the LED screen Arduino and the server. It might be possible to make do without the master. Another possibility is to consolidate the simple functions of the music system onto another Arduino. Either method could result in the removal of at least one Arduino from the system, thus reducing both cost and complexity of the final project.

Plan out wiring and exact power usage before any board fabrication or installation

With a project of this size, planning ahead is essential; it can save both time and money as time goes on. There were many occasions during the project when stopping to calculate power usage and the exact specification of how the system would work would have greatly aided our progress. Decisions were made about board specifications before the entire system was planned, which forced us to be locked into those decisions when it came time for assembly. Issues such as the flickering and dimness of the LED screens, and the power requirements of the RGB system could have been avoided with proper planning. Most important, particularly given the lack of an ECE major in the group, is having someone else who understands the basics of electrical engineering check the plan for circuit boards and wiring before committing to a given design. Had such steps been taken, the project might have progressed more quickly, and potentially less expensively.

Eliminate the need for an onsite computer by running the tasks on a microcontroller.

The need to have an on-site computer, as well as an off-site server, makes the system significantly more complex. It is very likely that the computations being done by the computer could just as easily be completed by a microcontroller with an on-board Ethernet stack. In fact, such a system exists for the Arduino, providing a full TCP/IP stack. Removing the on-site computer would certainly make setup easier, and it is possible that some systems could be consolidated to be controlled directly by this Arduino rather than through the use of an individual controller for each system.

Reduction of weight of the skeleton

No work was done in reducing the overall weight of the structure and this would be a good area to focus on in improving the portability of the sculpture. The easiest thing to do would be to remove large sections of unneeded material. This would have been done during the fabrication process but due to the indecision in where subsystems would be mounted and how many subsystems there would be, no optimization was done in this area.

Utilizing the Machine Shop facilities to the best extent

For an IQP it is too much to expect anyone in the machine shop to help out in any significant manner because it is not in their department and it is not a big enough project in their eyes to warrant the attention. If someone wants to have something machined at the school I have few things to say about how to get it done. At the very least, it is necessary to learn the basics of a Computer Aided Machining program such as ESPRIT which is used by many people who help out in the shop. Even if it is only possible to muddle incorrectly through a few steps in the process of creating a tool path for the part, this is the minimum amount of effort needed to garner some attention from the people who work in the shop. I recommend being slightly more competent than this if possible however because it will save you a lot of time; basically the only people who are proficient at the software who can assist you in a short amount of time are the people who run the shop and don't have more than a few minutes to spare. Hopefully if you want something machined you are an ME major and have taken the introductory Manufacturing course WPI offers to acquaint you with the CNC machines. It will then be possible to schedule a time for yourself to use one of the machines to produce your part. However, beware that if you want someone to supervise you because you are unsure about using the machine, you must be prepared to wait a week for the help.

Appendix A: Kermotiva Presentation to the IMGD Department

Kermotiva

An IQP in Interactive Public Art

Advisor: Professor Joshua Rosenstock

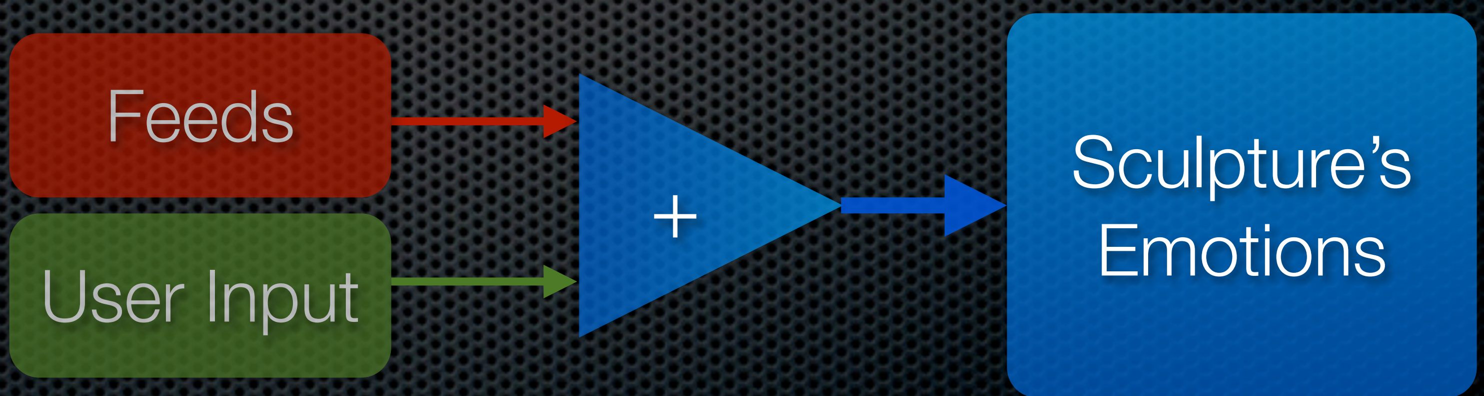
Students: Elliot Borenstein IMGD Art ’11, Yilmaz Kiymaz IMGD Art ’09, Bradford Lynch ME ’10, and Cynthia Weiler ME ’10

Goals

- Create an installation that would engage society to interact with the technology
- Allow people to display emotion through this interaction
- Use the sculpture as a form of data visualization for societal issues
  - Health
  - Economy
  - Food
  - World Issues
  - Nature
  - Personal
  - Entertainment
  - Work

Social Implications

- Emotion displayed by the sculpture
  - Has a “Base” emotion that is a function of various general data feeds
    - Financial
    - Weather
    - RSS News Feeds
  - Through a web interface users can interact with the sculpture, contributing to the displayed emotion



Social Implications (Cont.)

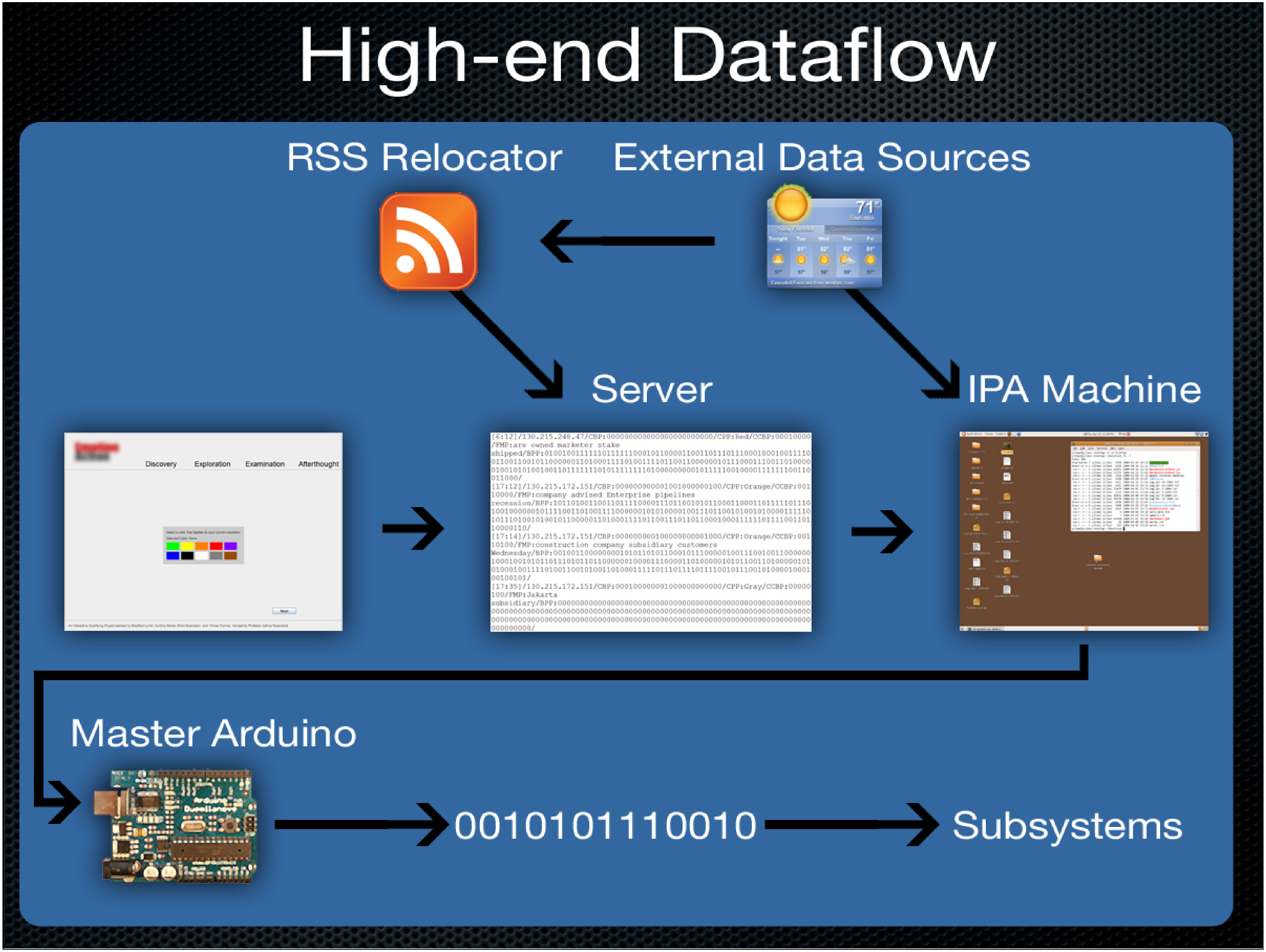
- The sculpture then becomes a snapshot of the state of the community that it has been installed in
- Increase awareness of the current state of the community and inspire people to improve their interactions with one another

Other Interactive Public Art

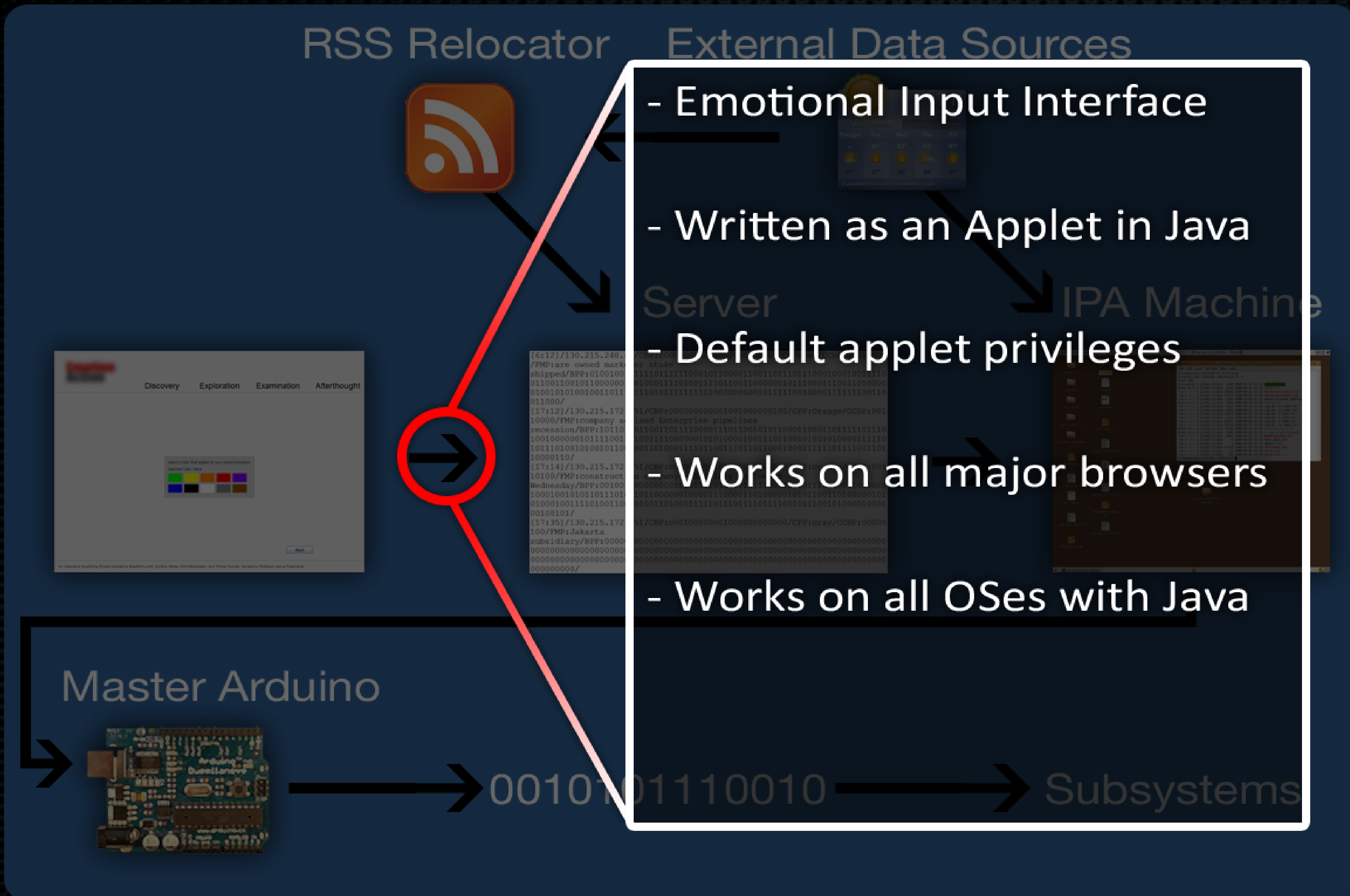
- Project Blinkenlights
  - Transformed a office building into a giant monochrome screen
  - Engaged the community and world through online and cell phone interactivity
- Vectorial Elevation
  - Transformation of the sky over Zocalo Plaza, Mexico City into a light show using 22 robotic spot lights
  - Over 800,000 participants from more than 89 countries

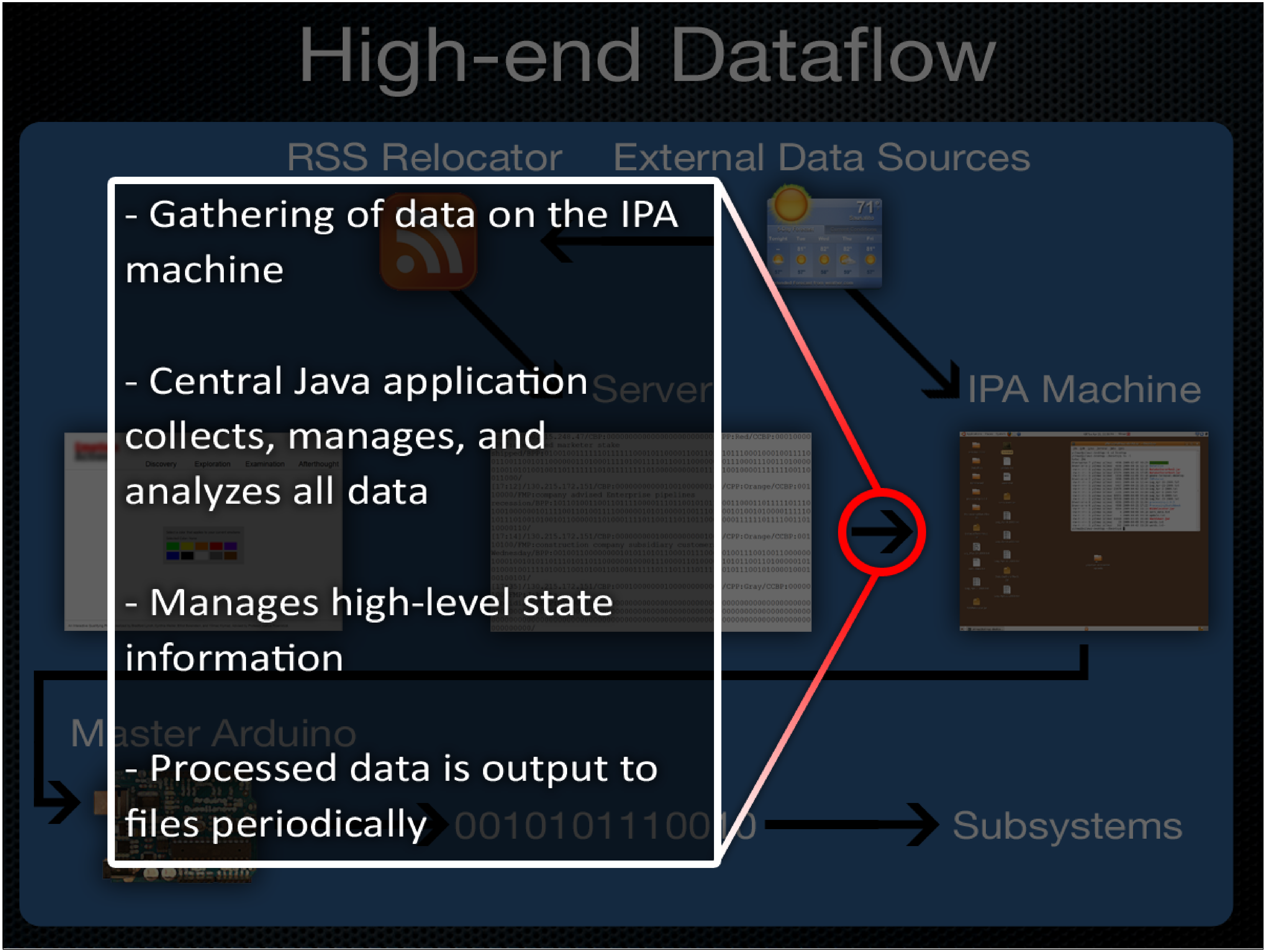
Blinkenlights

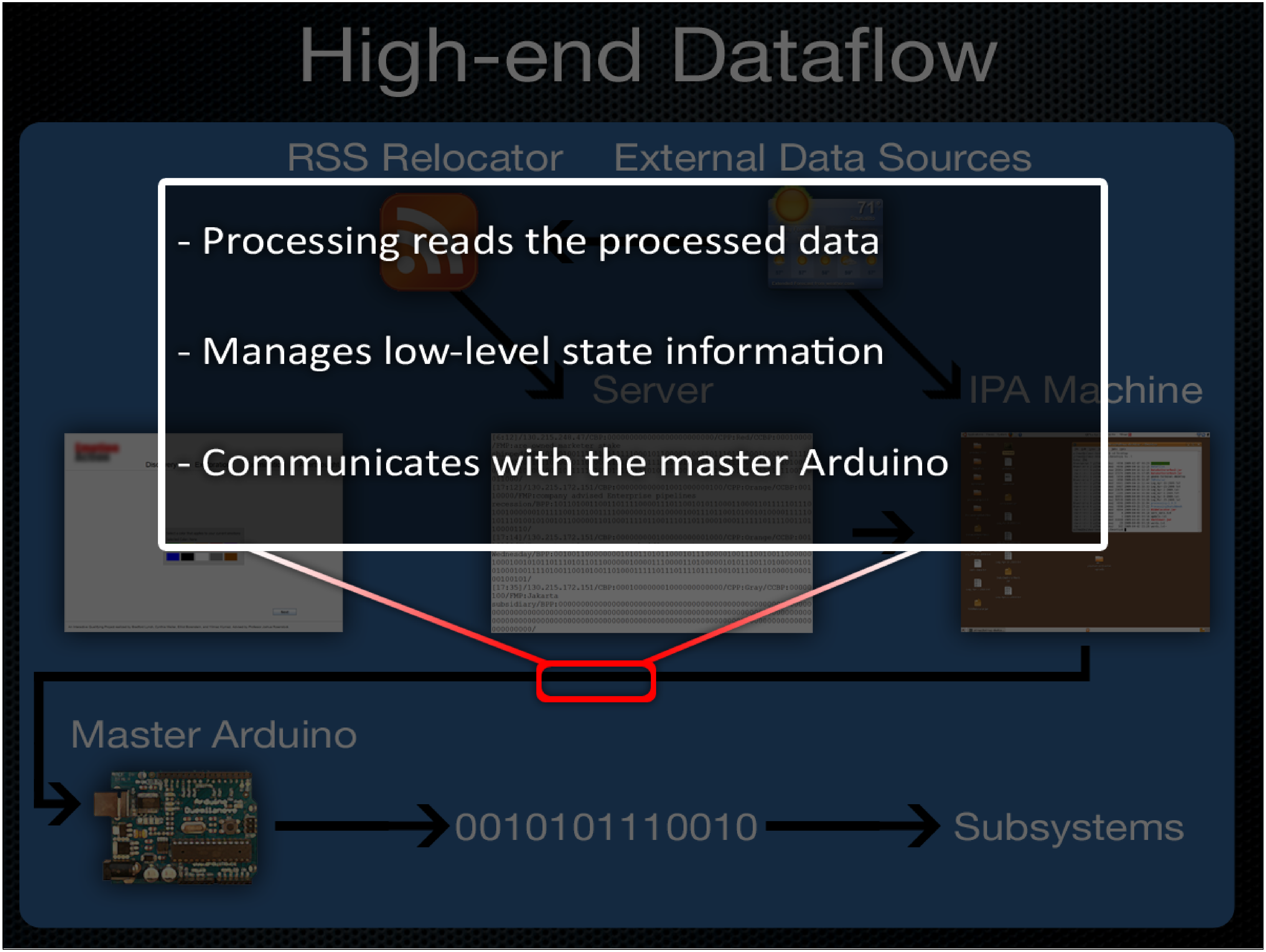
Vectorial Elevation


High-end Dataflow
RSS Relocator
External Data Sources
Server
IPA Machine
Discovery
Exploration
Examination
Afterthought
Master Arduino
0010101110010
Subsystems

High-end Dataflow



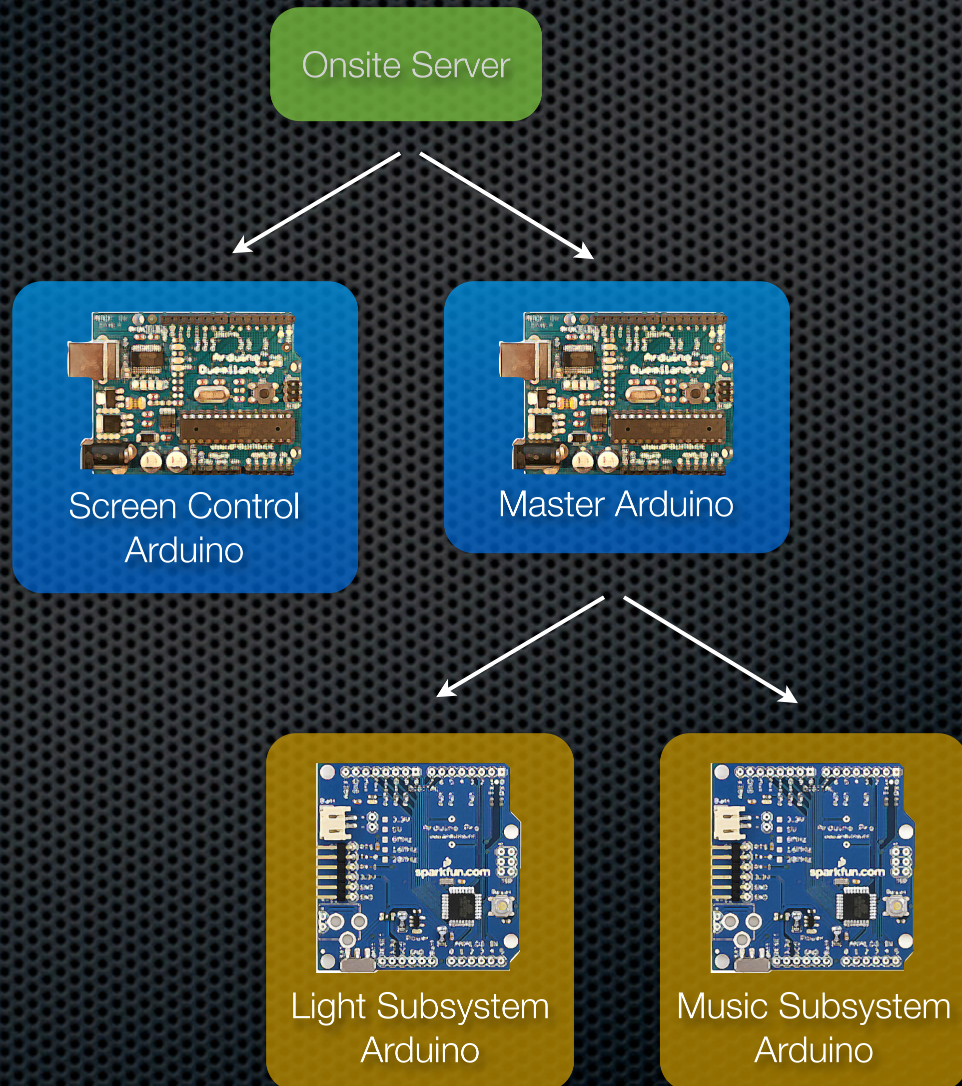

High-end Dataflow
RSS Relocator
External Data Sources
- Gathering of data on the IPA machine
- Central Java application collects, manages, and analyzes all data
- Manages high-level state information
- Processed data is output to files periodically
Server
IPA Machine
Master Arduino
Subsystems


High-end Dataflow
RSS Relocator
External Data Sources
- Processing reads the processed data
- Manages low-level state information
- Communicates with the master Arduino
Server
IPA Machine
Master Arduino
0010101110010
Subsystems

Data Representation

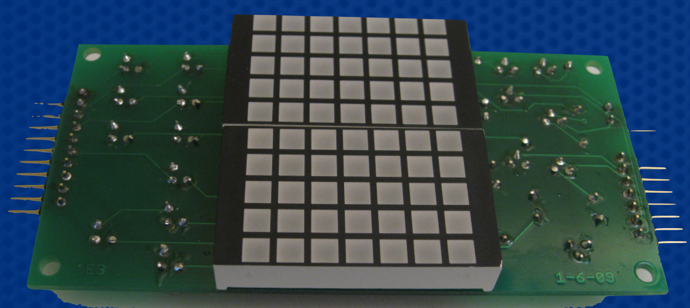
- Utilized LED Screens, LED Lighting, and Sound Reproduction to represent the collected data
- This Required:
  - 700+ feet of individual cables
  - 250+ transistors
  - 24 RJ-45 jacks
  - 5 custom circuit board designs (12 boards total)
  - 4 microcontrollers
  - 2 computers
  - 0 ECE Majors

Data Flow

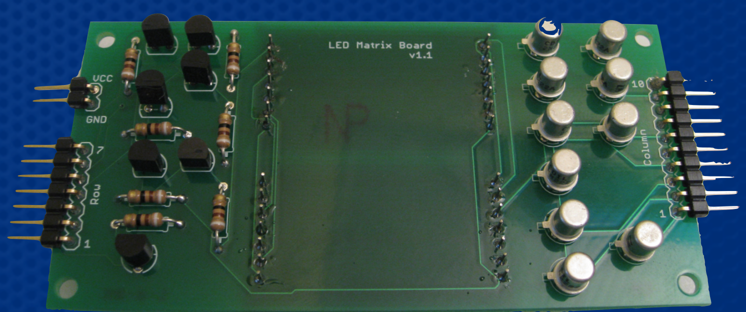

Onsite Server
Screen Control Arduino
Master Arduino
Light Subsystem Arduino
Music Subsystem Arduino

LED Screens

- Controlled directly from server by Processing
- Uses POV to scroll text or images
- Scrolls up to 256 bytes
  - Letters average 4-5 bytes ≈ 50 letters

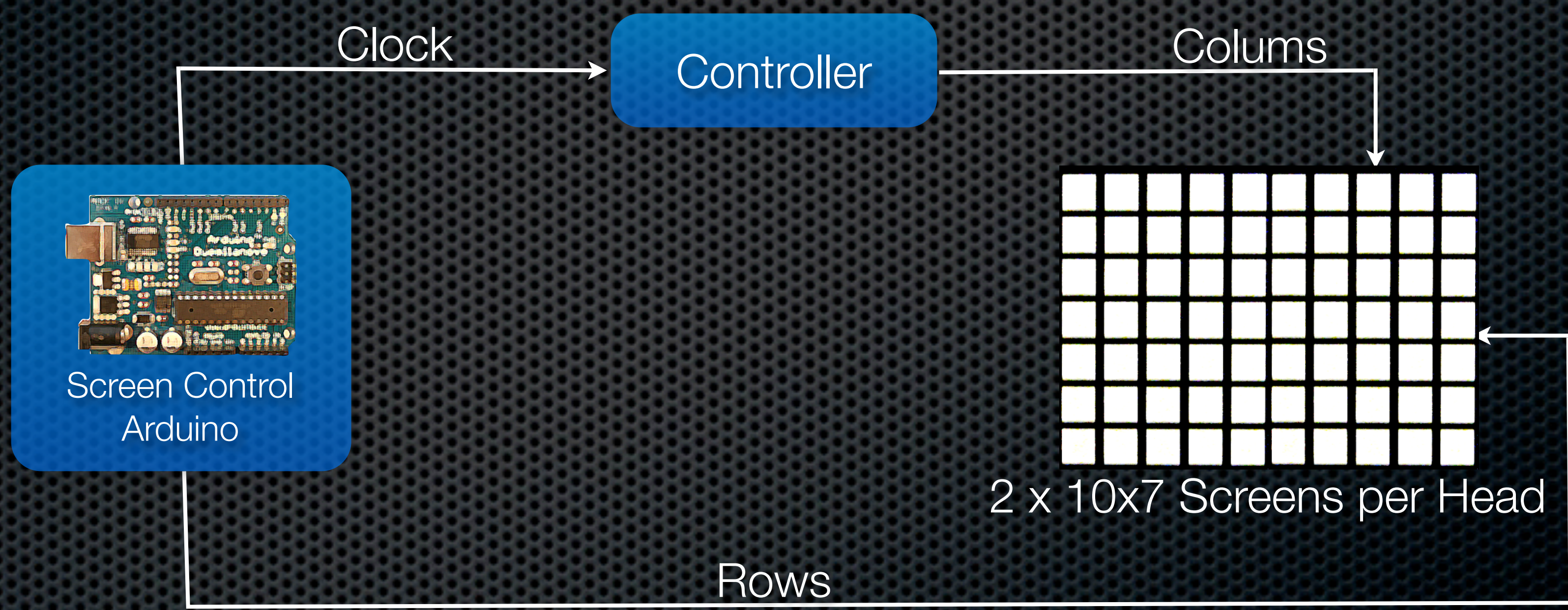


Front of LED Screen



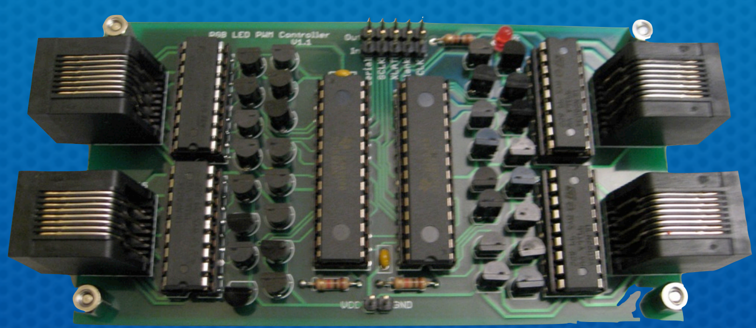
Back of LED Screen

LED Screens (Cont.)

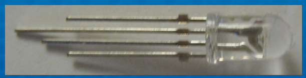


RGB LED Lighting

- Controlled by Master Arduino
- 36 RGB LEDs
  - 6 rows * 3 groups/row * 2 LEDs/group
- 54 PWM channels
  - 6 rows * 3 groups/row * 3 ch/group
  - 64 PWM channels max
- Plays various pre-defined patterns and colors

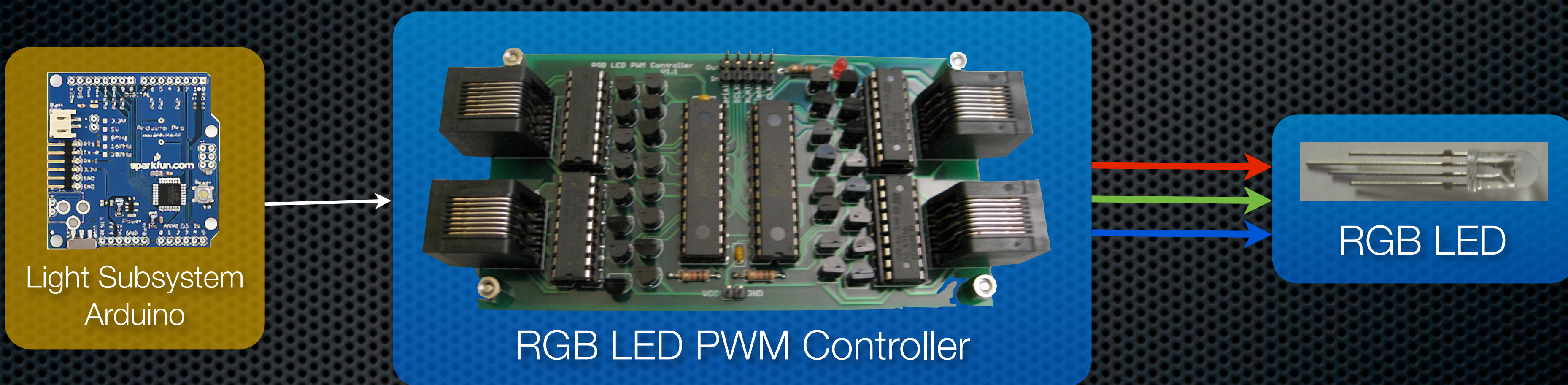


RGB LED PWM Controller



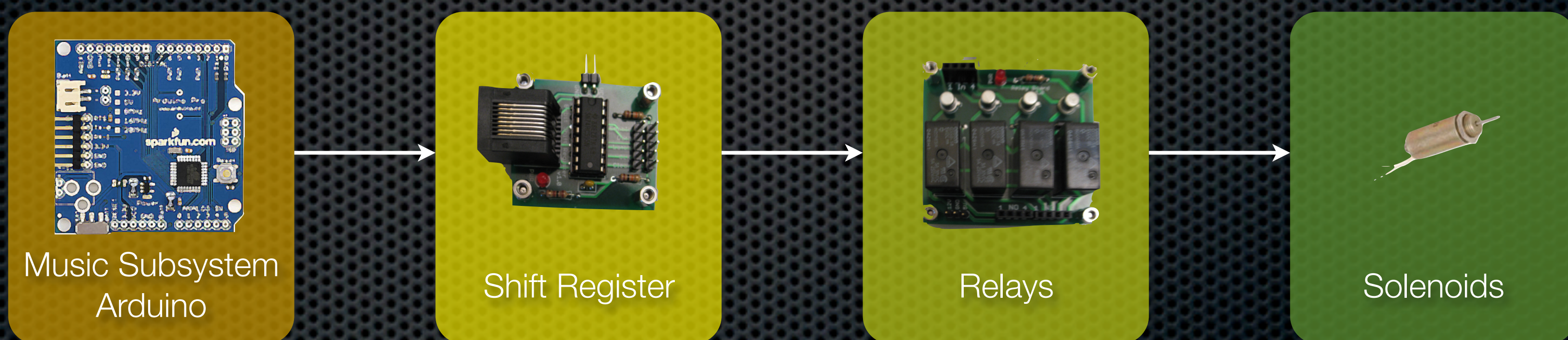
RGB LED

RGB LED Lighting (Cont.)



Sound Reproduction

- Controlled by the Master Arduino
- Plays pre-programmed songs/tones
  - 8 bits and a delay
- Simple control chain



Sound Reproduction

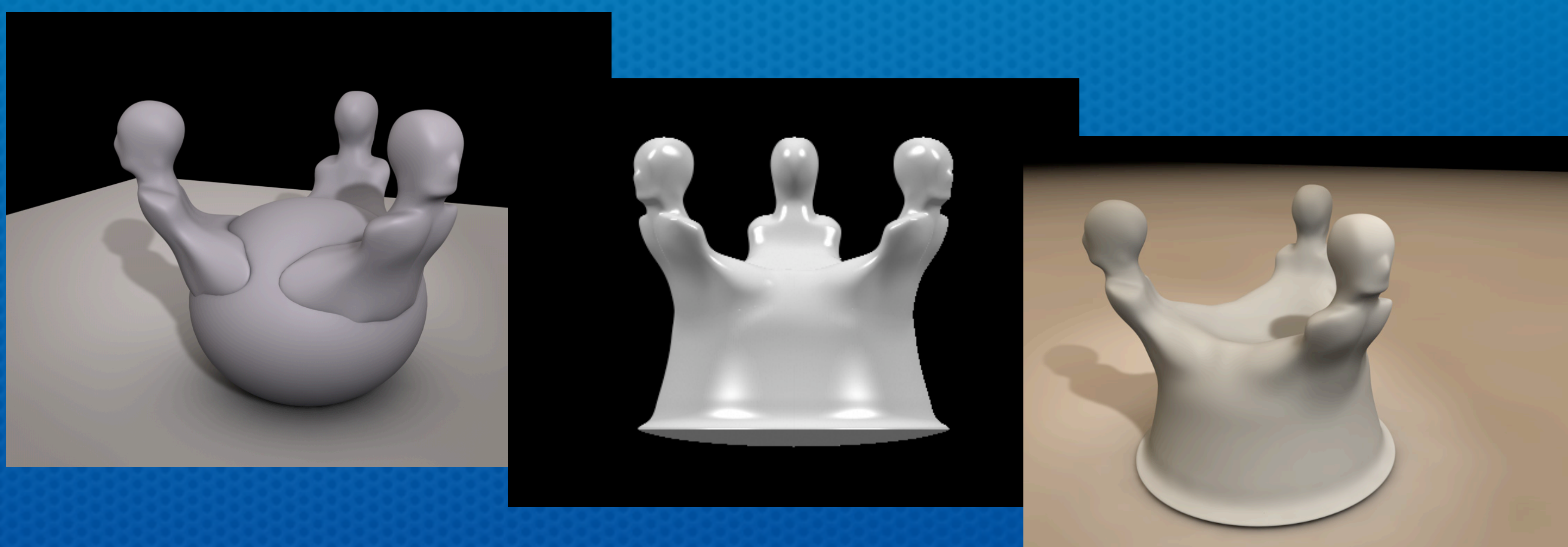
- Utilized solenoids to strike keys on a modified xylophone
- A total of 8 notes
  - 1 Octave + 1

Modified Xylophone

Plunger Striking Key

Physical Form

- Amorphous figure with three torsos and heads emerging from it.
- Tightly stretched cloth skin to allow for free flowing curves
- Skeleton made of tiered MDF ribs allows for cloth to be lofted in order to create the form



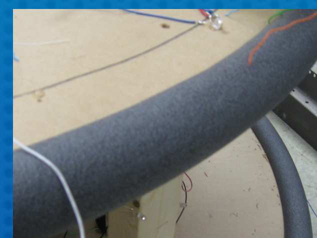
Various Renders of Sculpture’s Form

Skeleton

- Ribs allowed for flexibility of shape
- MDF was chosen as the rib material
  - Easy to work with shape and drill
  - Easy to mount things to
  - Somewhat heavy but quite durable
- Foam was added to MDF ribs to soften hard edges
- Pine 2x4's were used to connect ribs and subsystems
- Casters with elastic wheels and braking were added to base for ease of transportation
- Preformed foam heads

Skeleton (Cont.)

Complete Skeleton



Foam to Soften Edges

Braking Casters

Foam Head

Cloth Skin

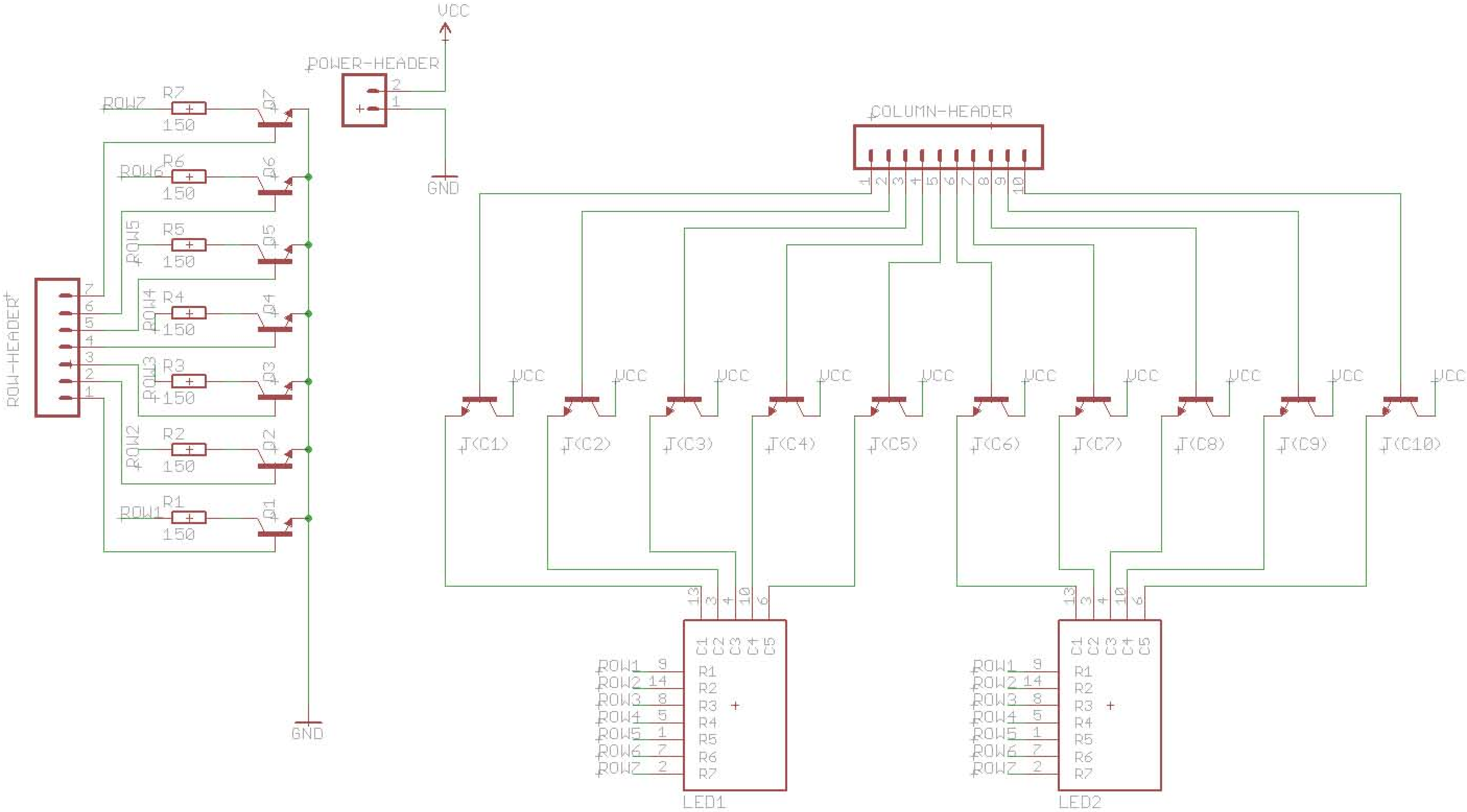
- The use of cloth allows for a sweeping and free flowing form
  - Unity of the three forms emerging from the base
- Spandex cloth was chosen
- The particular choice of fabric allows for:
  - LED light to be displayed in a diffuse manner
  - Neutral color doesn’t lead to predisposed emotions
  - Easy to clean and wash
  - Removable

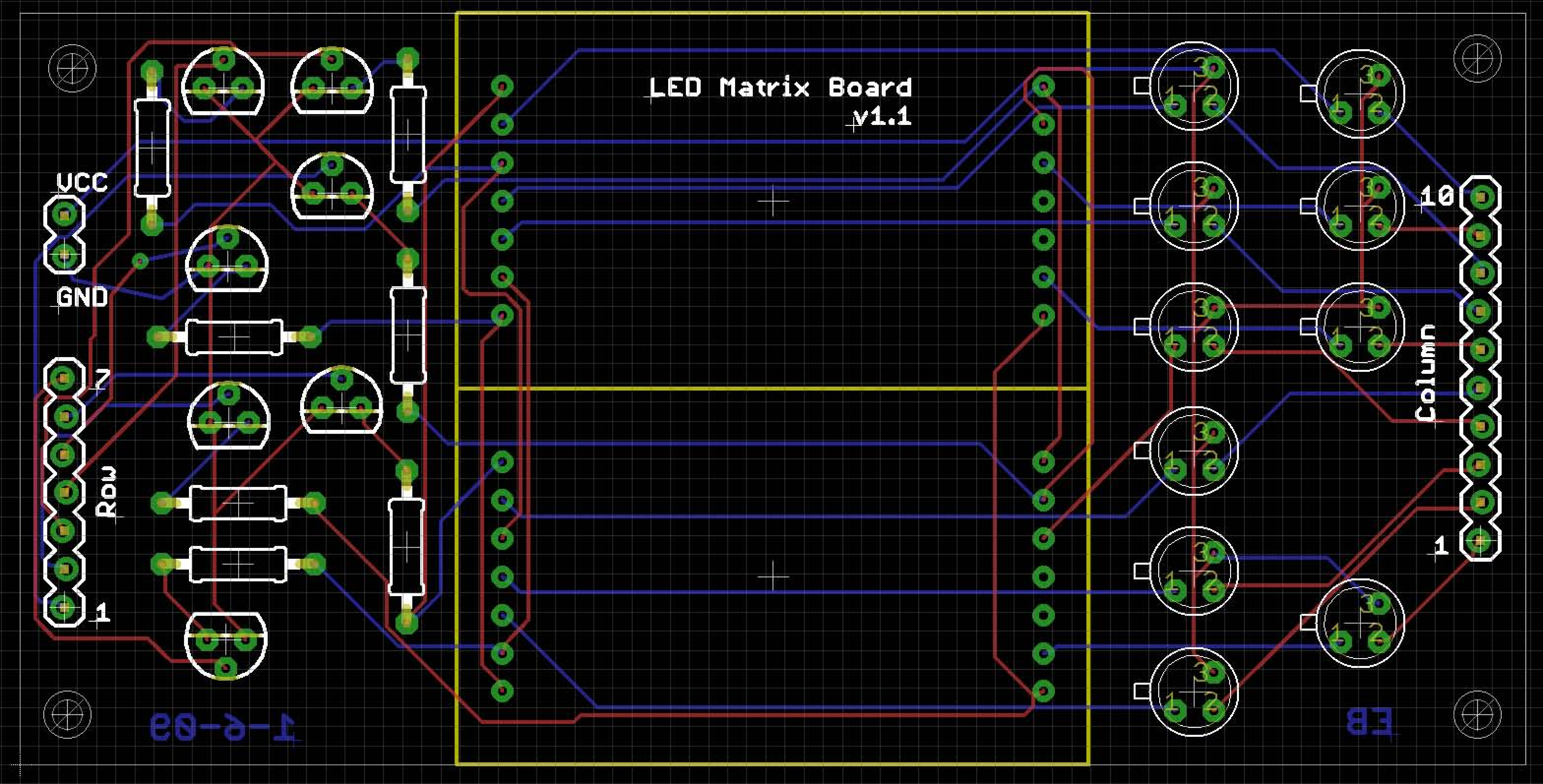
Finished Form

Special Thanks

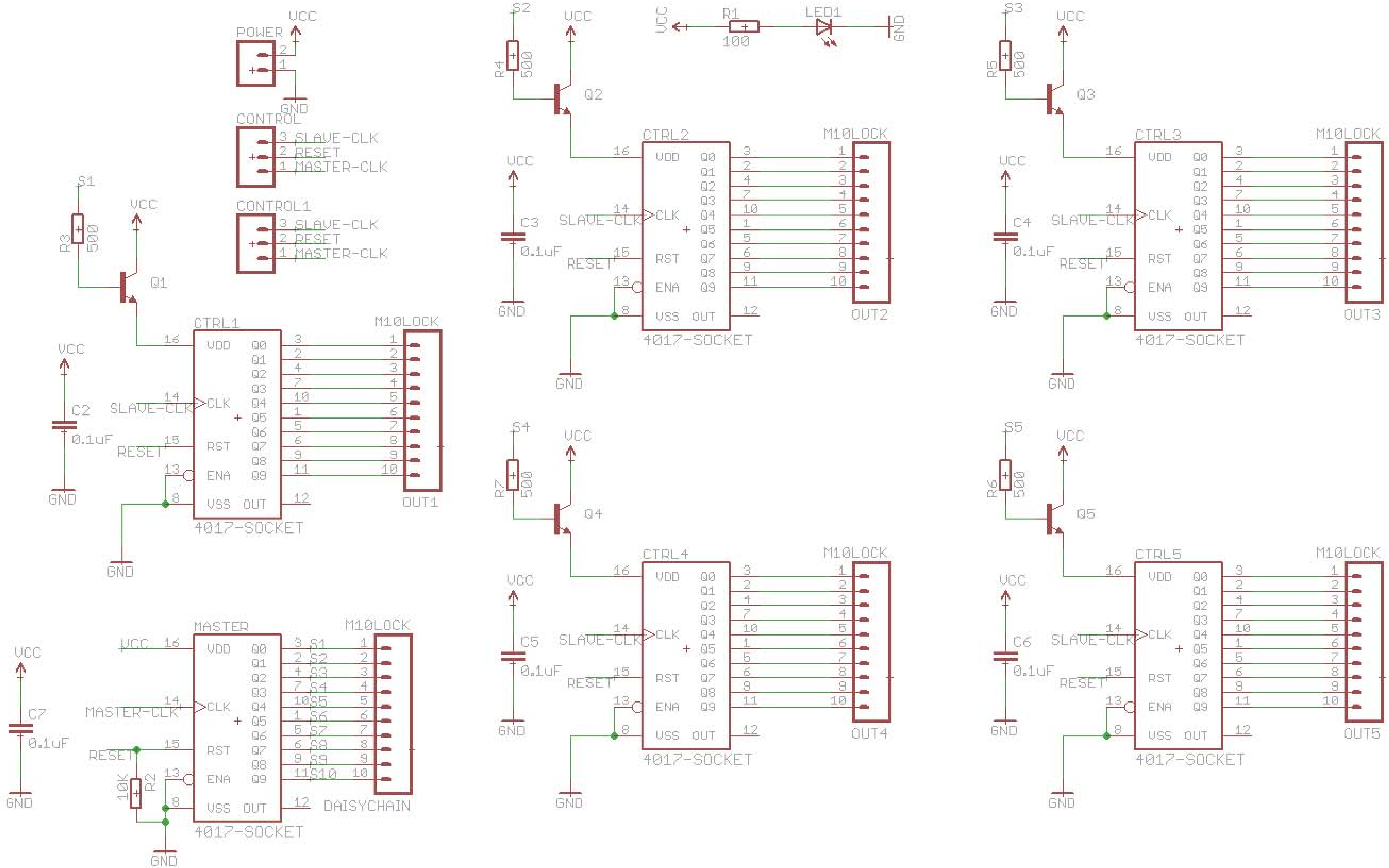
- Professor Rosenstock
- WPI PC Shop
- WPI Network Operations
- DelSignore Electric
- Humanities and Arts Department
- Arduino Team

Appendix B: Electronics Diagrams

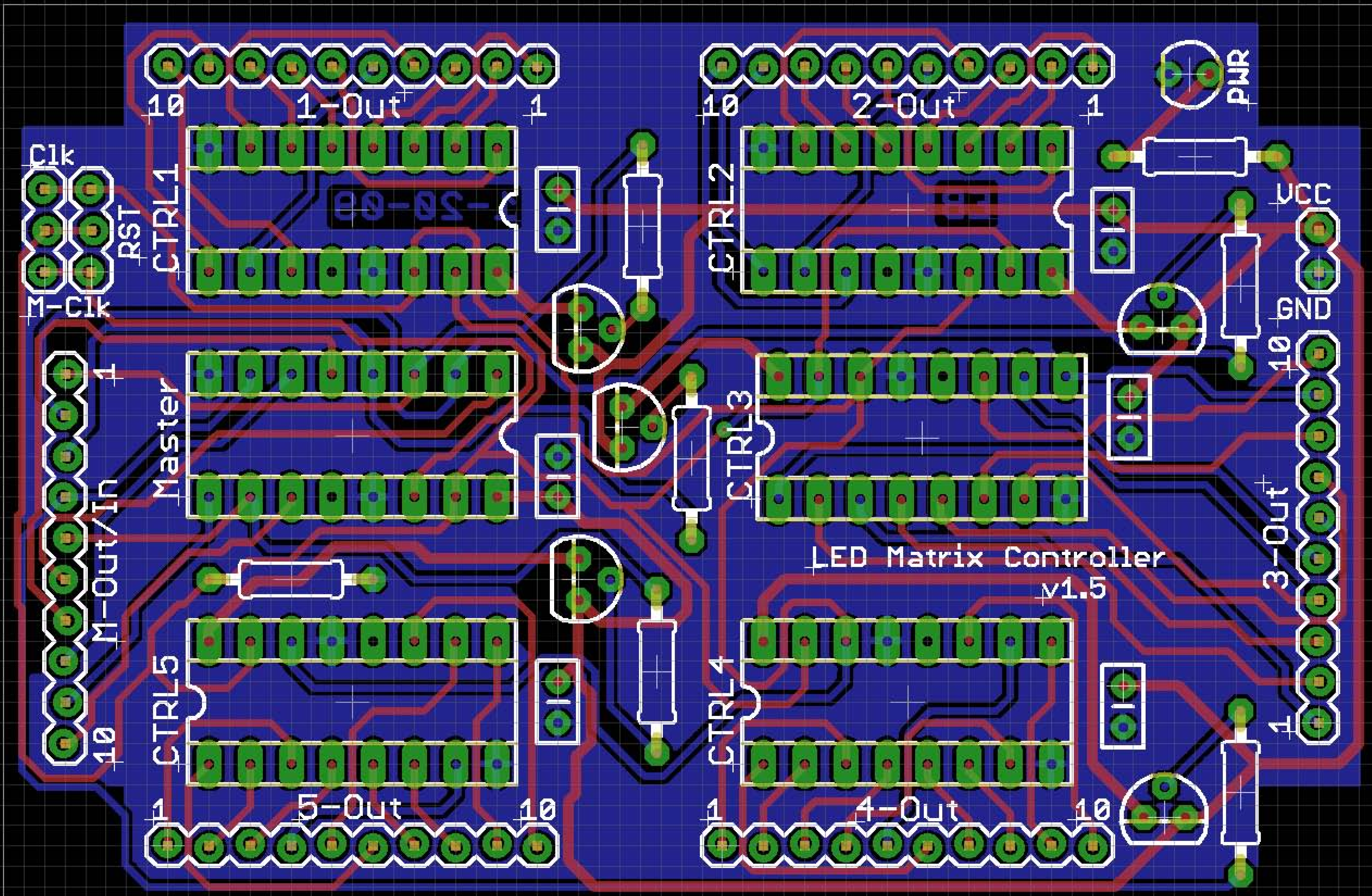

VCC
POWER-HEADER
2
1
GND
ROW7
R7
150
Q7
ROW6
R6
150
Q6
ROW5
R5
150
Q5
ROW4
R4
150
Q4
ROW3
R3
150
Q3
ROW2
R2
150
Q2
ROW1
R1
150
Q1
ROW-HEADER
7
6
5
4
3
2
1
GND
COLUMN-HEADER
1 2 3 4 5 6 7 8 9 10
VCC
T(C1)
T(C2)
T(C3)
T(C4)
T(C5)
T(C6)
T(C7)
T(C8)
T(C9)
T(C10)
13 3 4 10 6
C1 C2 C3 C4 C5
ROW1 9 R1
ROW2 14 R2
ROW3 8 R3
ROW4 5 R4
ROW5 1 R5
ROW6 7 R6
ROW7 2 R7
LED1
LED2



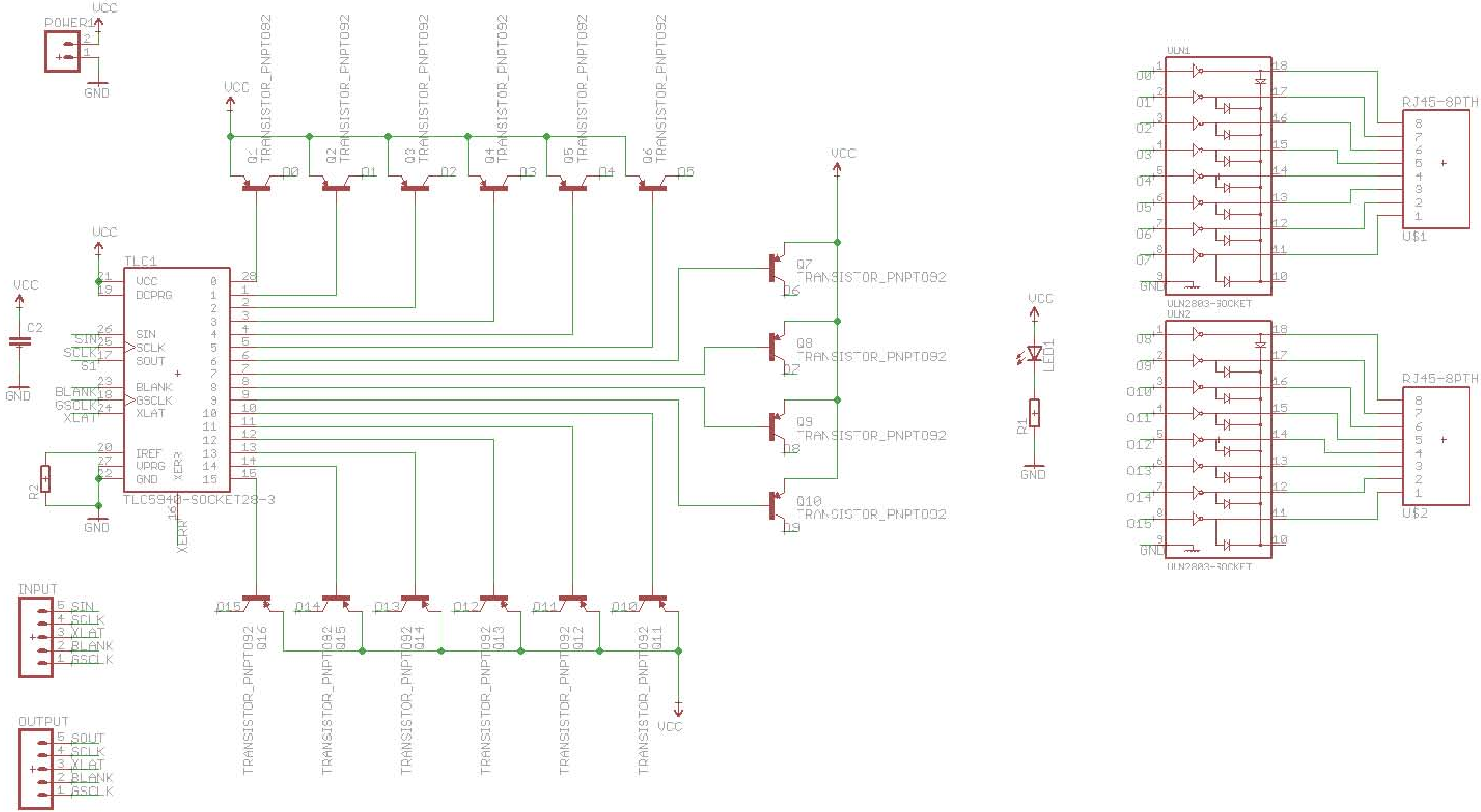
LED Matrix Board
v1.1
VCC
GND
Row
7
1
Column
10
1
1-6-09
EB

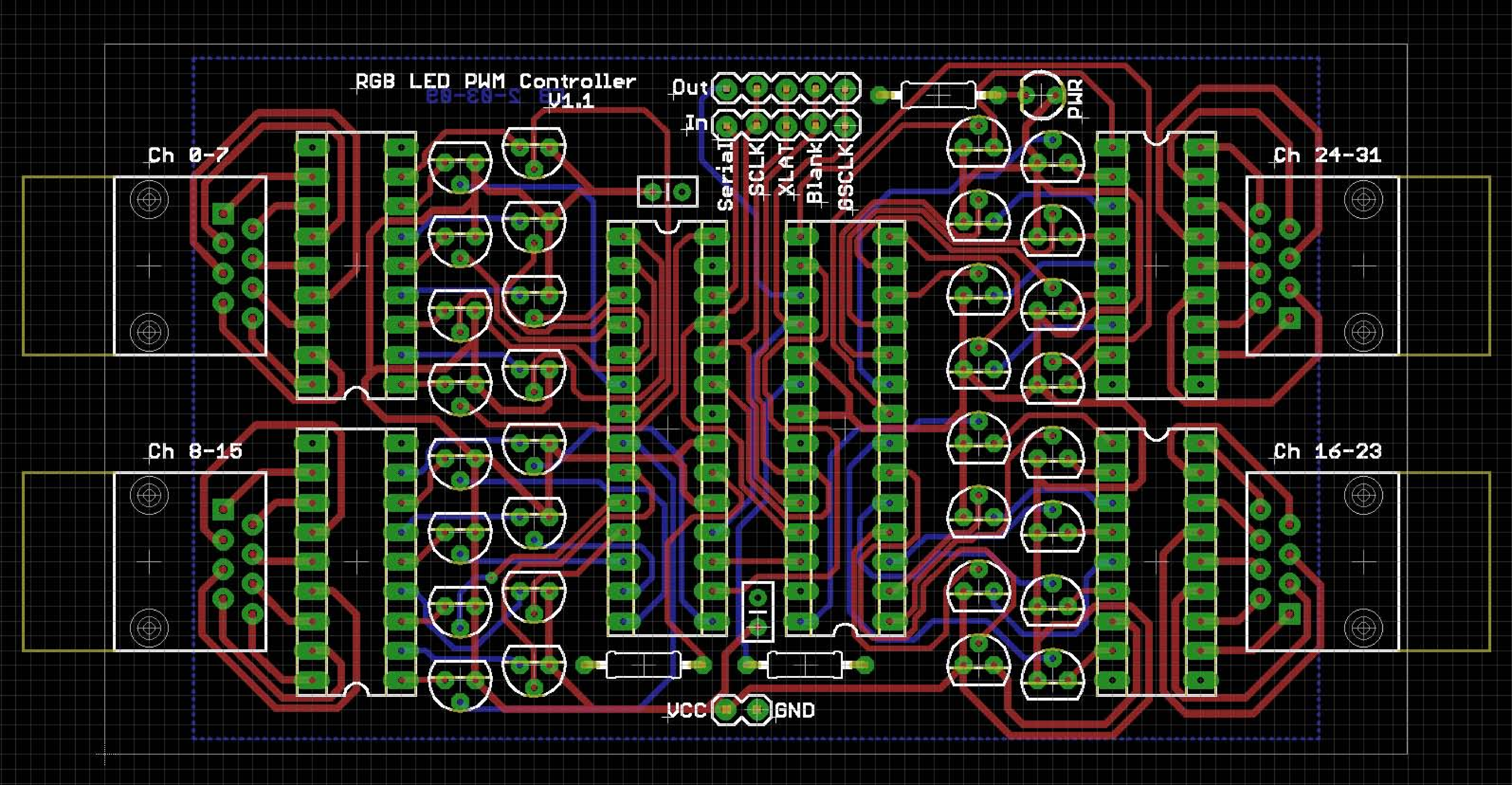


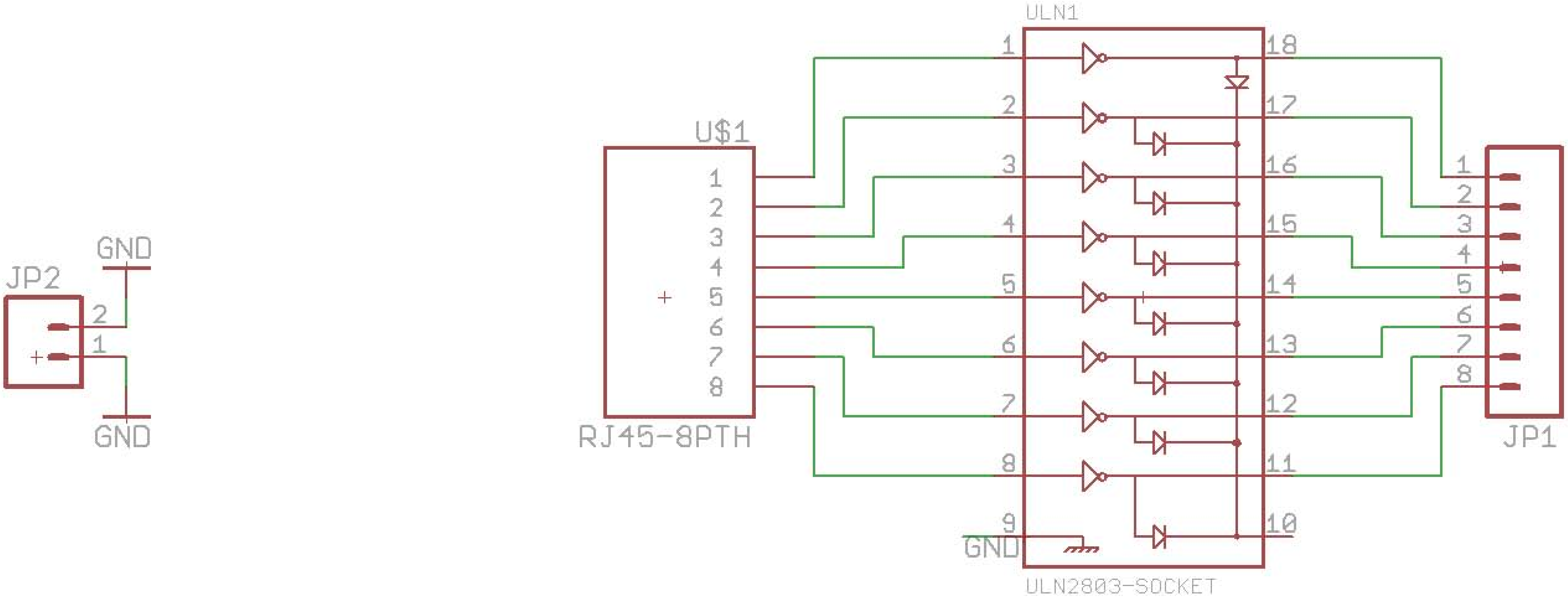
POWER
VCC
GND
CONTROL
3 SLAVE-CLK
2 RESET
1 MASTER-CLK
CONTROL1
3 SLAVE-CLK
2 RESET
1 MASTER-CLK
S1
R3
500
VCC
Q1
C2
0.1uF
CTRL1
VDD
CLK
RST
ENA
VSS
OUT
Q0
Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
Q9
SLAVE-CLK
RESET
4017-SOCKET
M10LOCK
OUT1
MASTER
MASTER-CLK
C7
R2
10K
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
DAISYCHAIN
S2
R4
Q2
C3
CTRL2
OUT2
R1
100
LED1
S3
R5
Q3
C4
CTRL3
OUT3
S4
R7
Q4
C5
CTRL4
OUT4
S5
R6
Q5
C6
CTRL5
OUT5

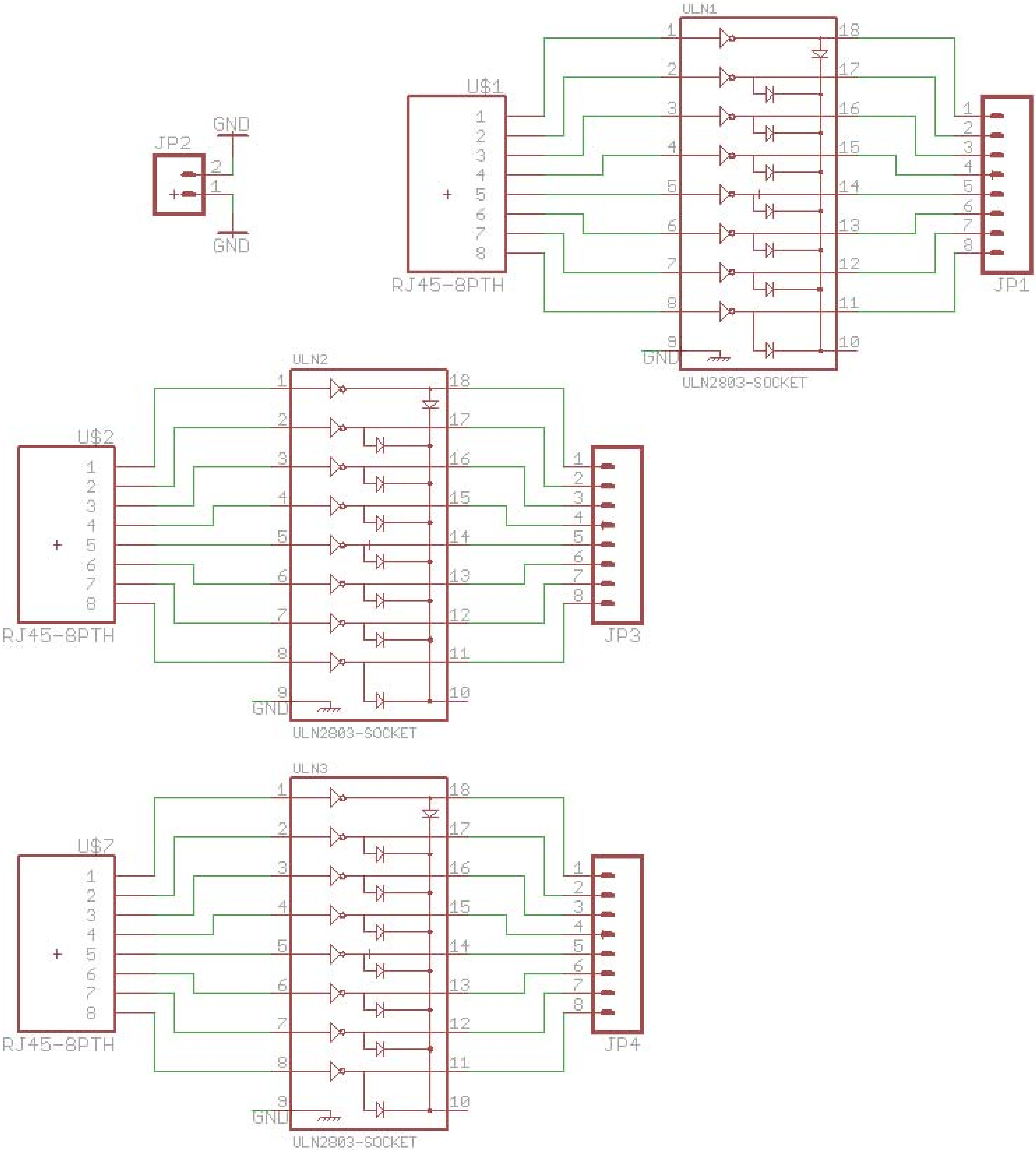


LED Matrix Controller
v1.5
1-Out
2-Out
3-Out
4-Out
5-Out
PWR
VCC
GND
Clk
RST
M-Clk
M-Out/In
CTRL1
CTRL2
CTRL3
CTRL4
CTRL5
Master

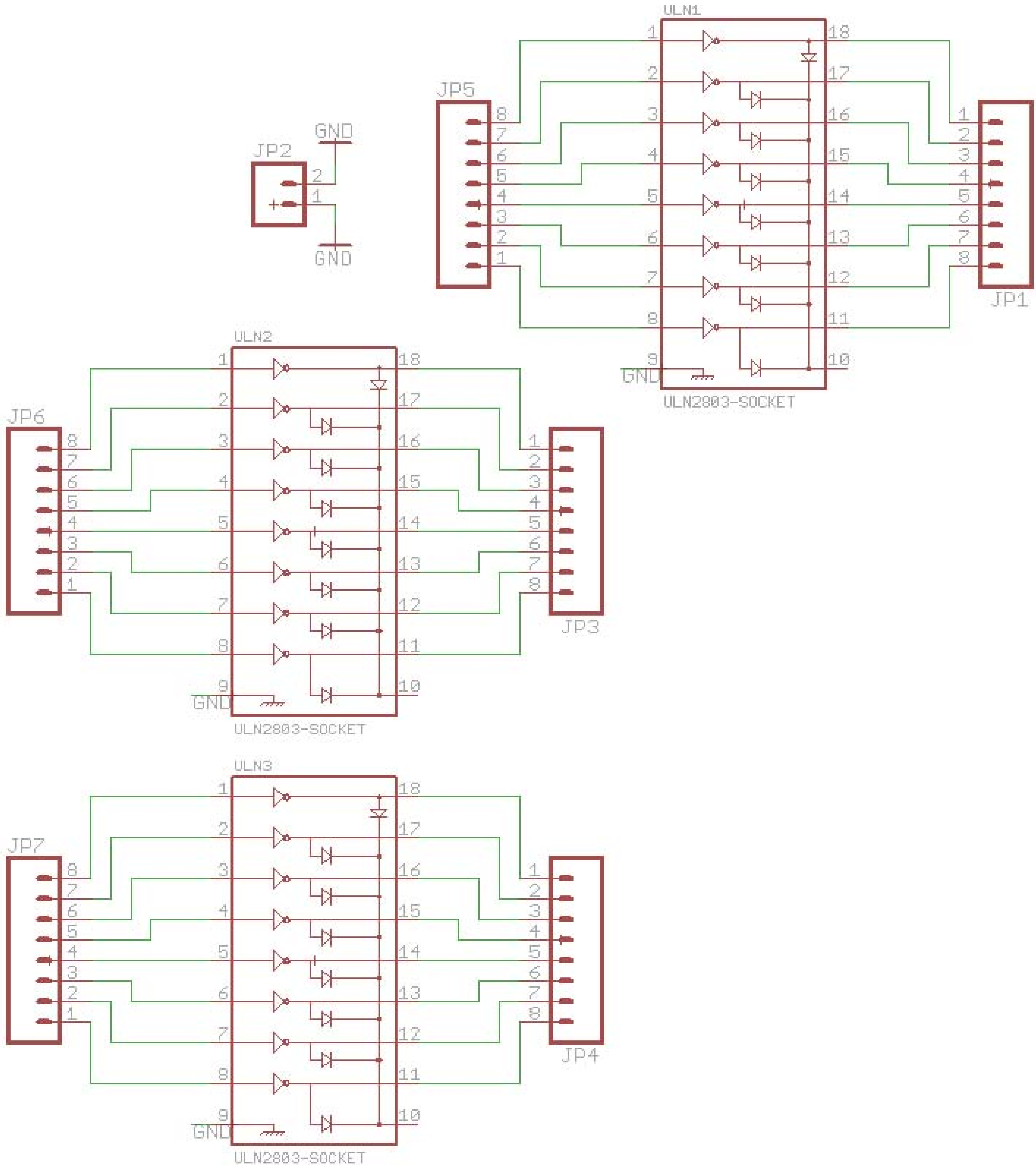

VCC
POWER1
GND
TRANSISTOR_PNPTO92
Q1 Q2 Q3 Q4 Q5 Q6
O0 O1 O2 O3 O4 O5
TLC1
VCC
DCPRG
SIN
SCLK
SOUT
BLANK
GSCLK
XLAT
IREF
VPRG
GND
XERR
TLC5940-SOCKET28-3
C2
R2
SIN
SCLK
S1
BLANK
GSCLK
XLAT
Q7 TRANSISTOR_PNPTO92
Q8 TRANSISTOR_PNPTO92
Q9 TRANSISTOR_PNPTO92
Q10 TRANSISTOR_PNPTO92
O6 O7 O8 O9
Q16 Q15 Q14 Q13 Q12 Q11
O15 O14 O13 O12 O11 O10
INPUT
5 SIN
4 SCLK
3 XLAT
2 BLANK
1 GSCLK
OUTPUT
5 SOUT
4 SCLK
3 XLAT
2 BLANK
1 GSCLK
LED1
R1
ULN1
ULN2803-SOCKET
ULN2
ULN2803-SOCKET
RJ45-8PTH
U$1
U$2

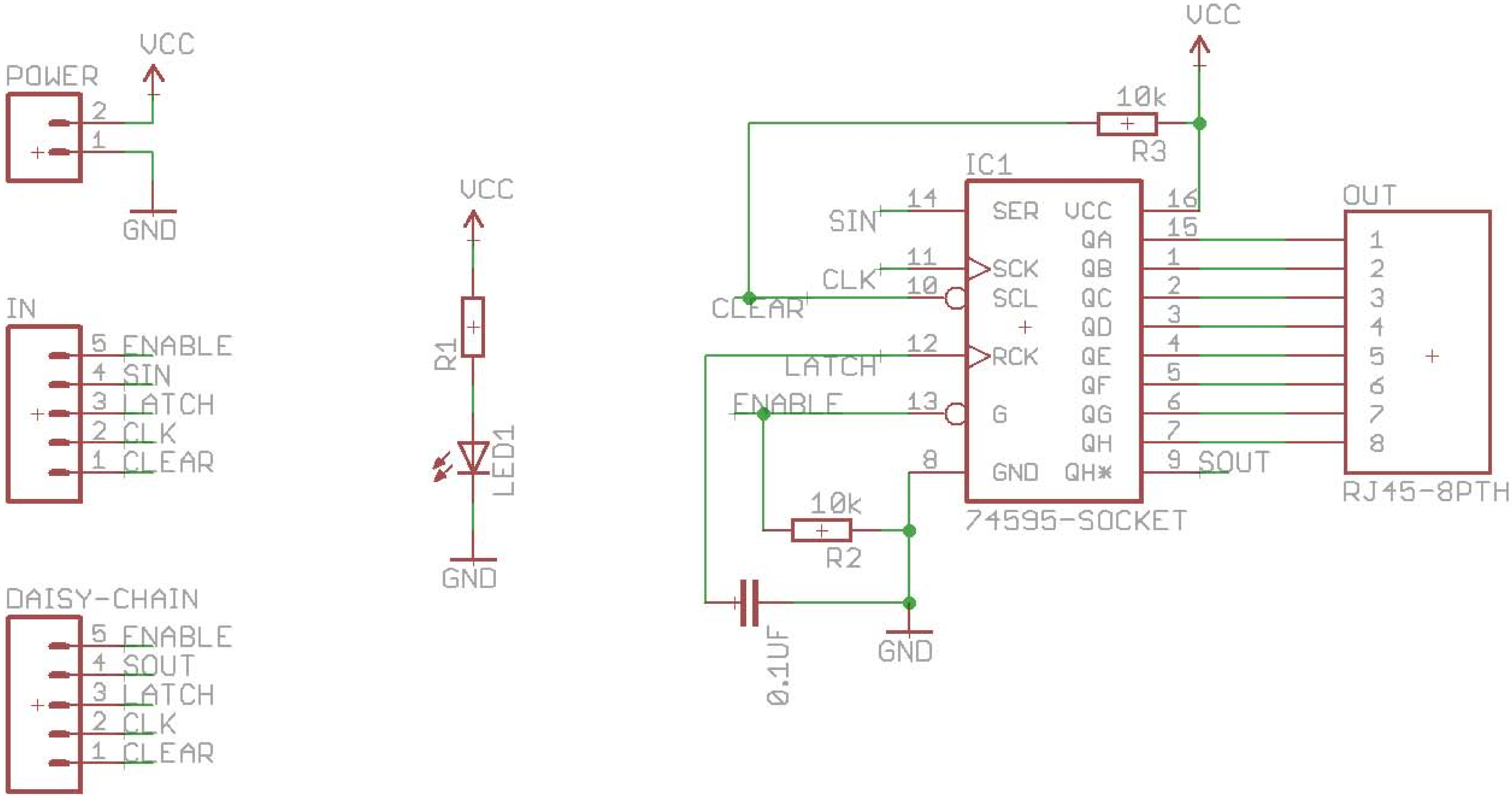

RGB LED PWM Controller
V1.1
Out
In
Serial
SCLK
XLAT
Blank
GSCLK
PWR
Ch 0-7
Ch 24-31
Ch 8-15
Ch 16-23
VCC
GND


JP2
GND
2
1
GND
U$1
1
2
3
4
5
6
7
8
RJ45-8PTH
ULN1
1
2
3
4
5
6
7
8
9
GND
18
17
16
15
14
13
12
11
10
ULN2803-SOCKET
JP1
1
2
3
4
5
6
7
8

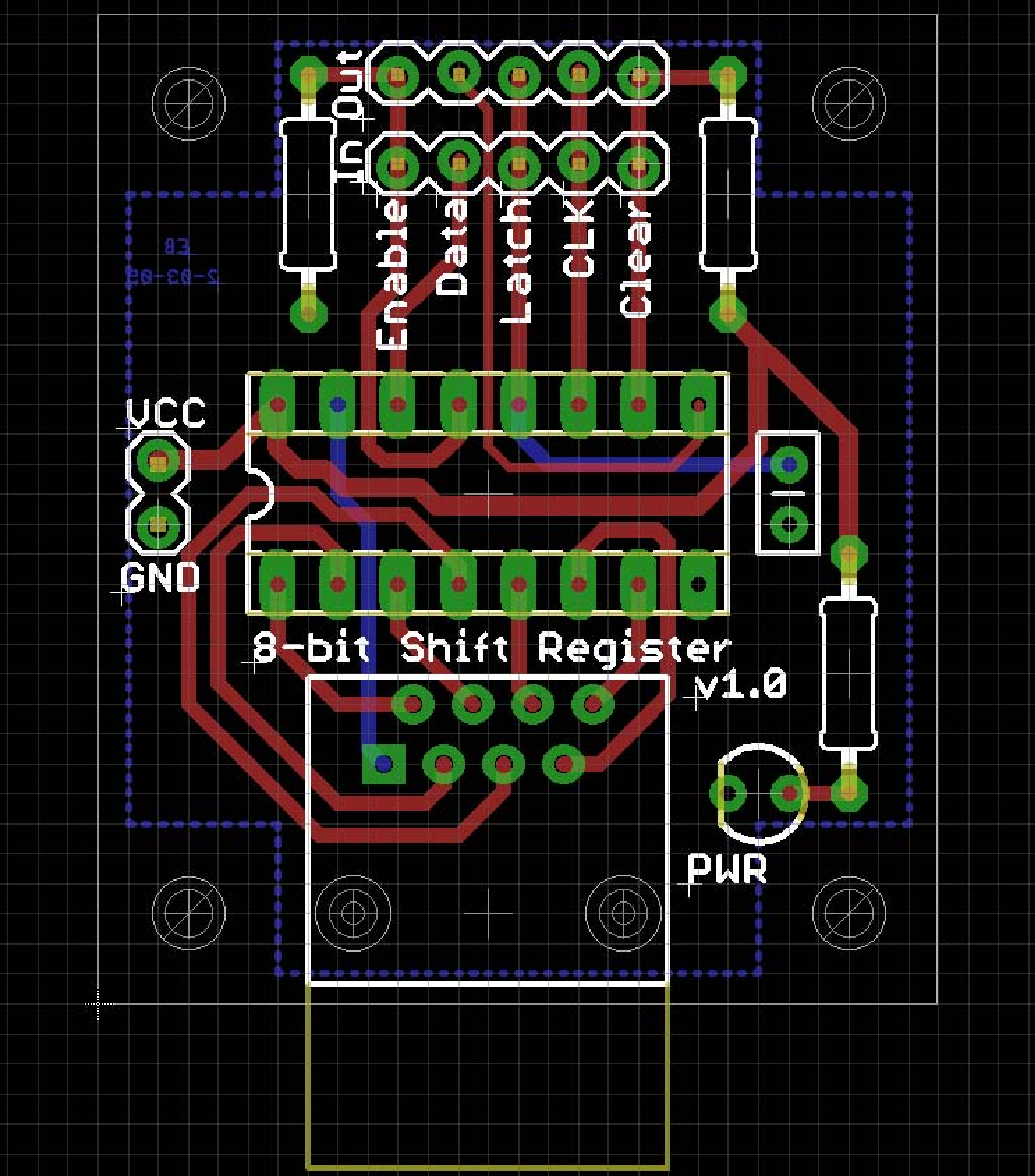


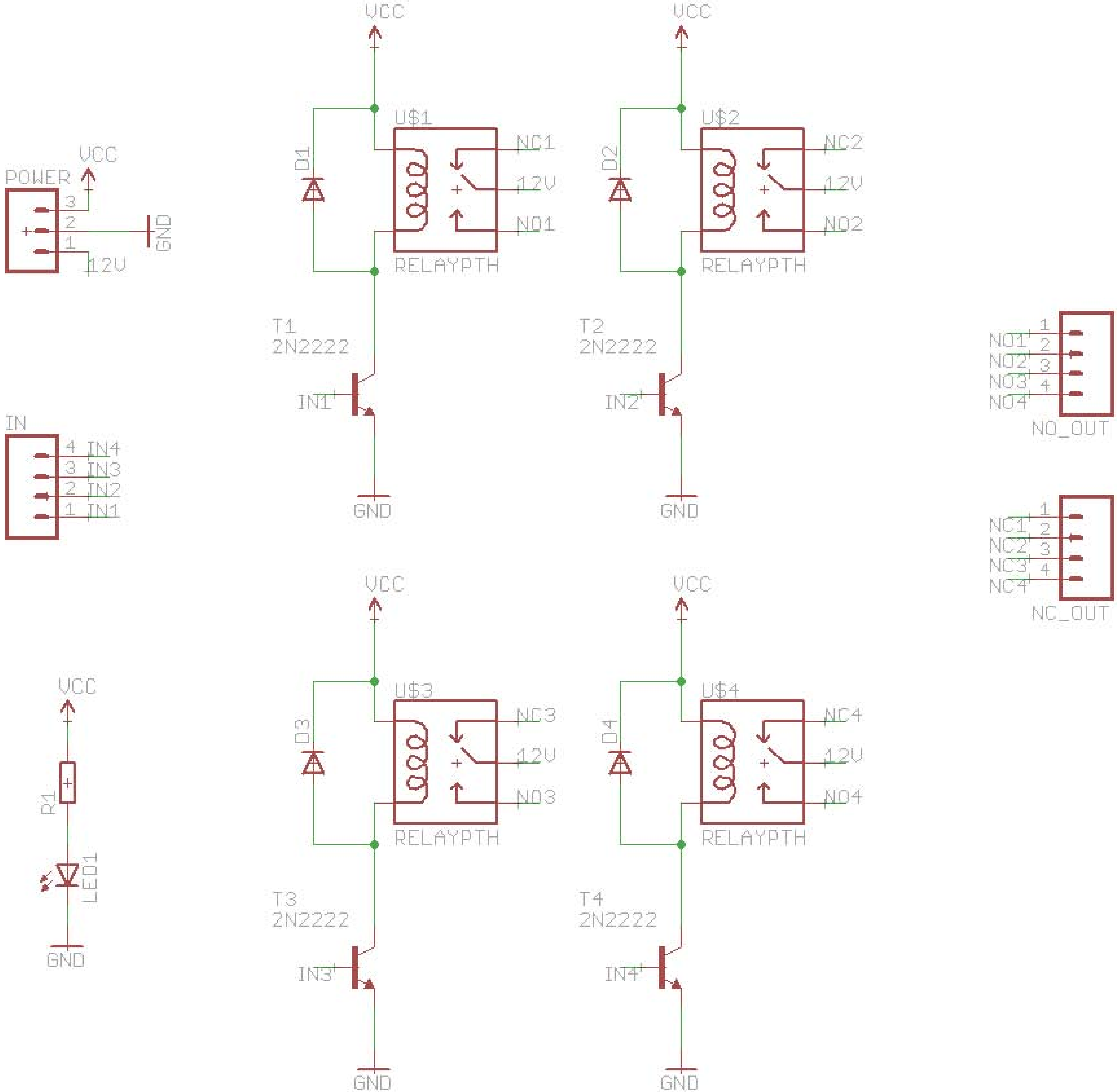
JP2
GND
GND
U$1
RJ45-8PTH
ULN1
ULN2803-SOCKET
JP1
U$2
RJ45-8PTH
ULN2
ULN2803-SOCKET
JP3
U$7
RJ45-8PTH
ULN3
ULN2803-SOCKET
JP4


ULN1
JP5
JP2
GND
GND
JP1
GND
ULN2803-SOCKET
ULN2
JP6
JP3
GND
ULN2803-SOCKET
ULN3
JP7
JP4
GND
ULN2803-SOCKET

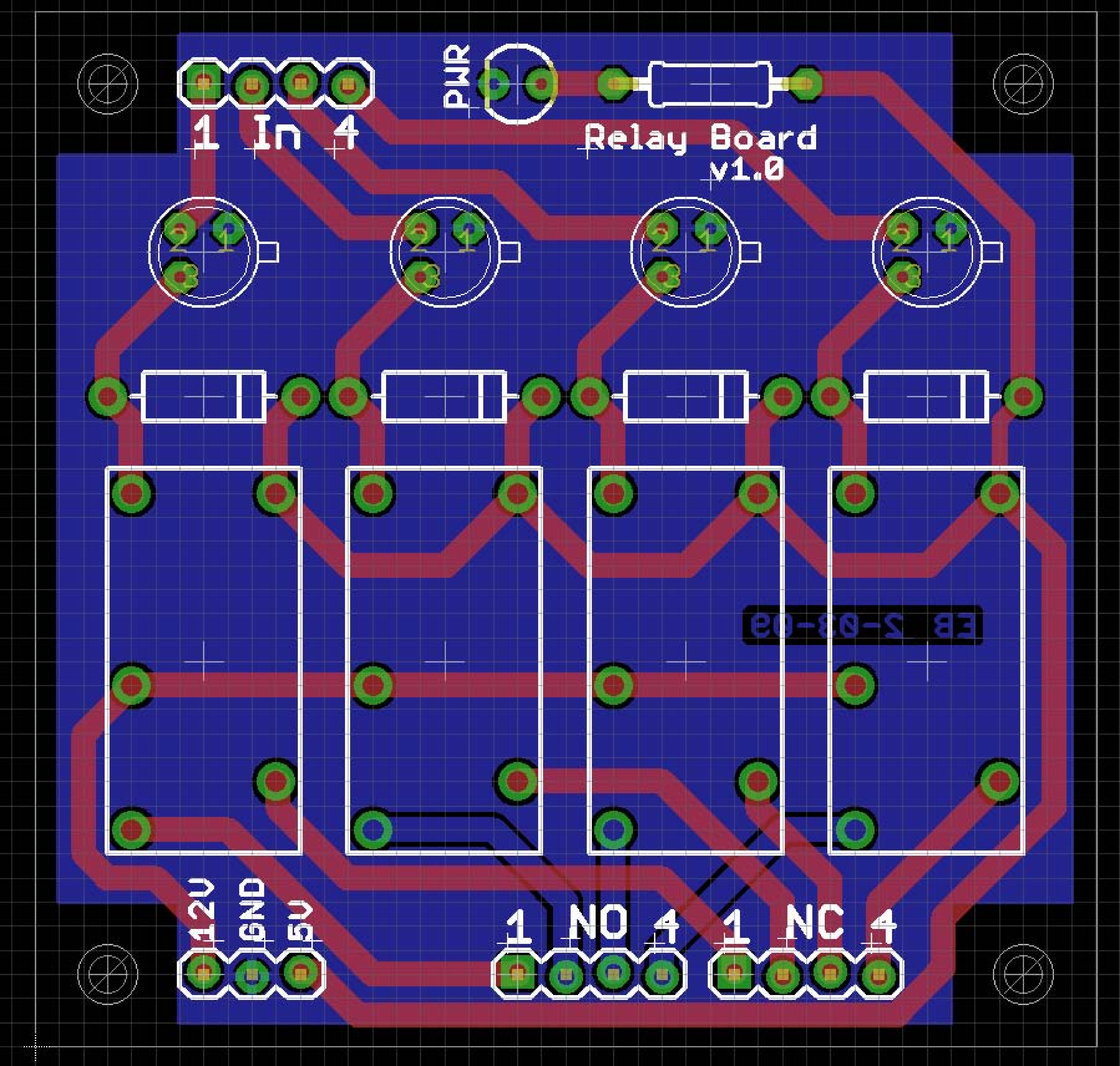


POWER
VCC
GND
IN
5 ENABLE
4 SIN
3 LATCH
2 CLK
1 CLEAR
DAISY-CHAIN
5 ENABLE
4 SOUT
3 LATCH
2 CLK
1 CLEAR
VCC
R1
LED1
GND
VCC
10k
R3
IC1
SER
VCC
SCK
SCL
RCK
G
GND
QA
QB
QC
QD
QE
QF
QG
QH
QH*
SIN
CLK
CLEAR
LATCH
ENABLE
SOUT
74595-SOCKET
10k
R2
0.1UF
GND
OUT
RJ45-8PTH


In
Out
Enable
Data
Latch
CLK
Clear
VCC
GND
8-bit Shift Register
v1.0
PWR



VCC
POWER
3
2
1
12V
GND
IN
4 IN4
3 IN3
2 IN2
1 IN1
R1
LED1
U$1
U$2
U$3
U$4
D1
D2
D3
D4
NC1
NC2
NC3
NC4
12V
NO1
NO2
NO3
NO4
RELAYPTH
T1
T2
T3
T4
2N2222
IN1
IN2
IN3
IN4
NO_OUT
NC_OUT



1 In 4
PWR
Relay Board
v1.0
12V
GND
5V
1 NO 4
1 NC 4

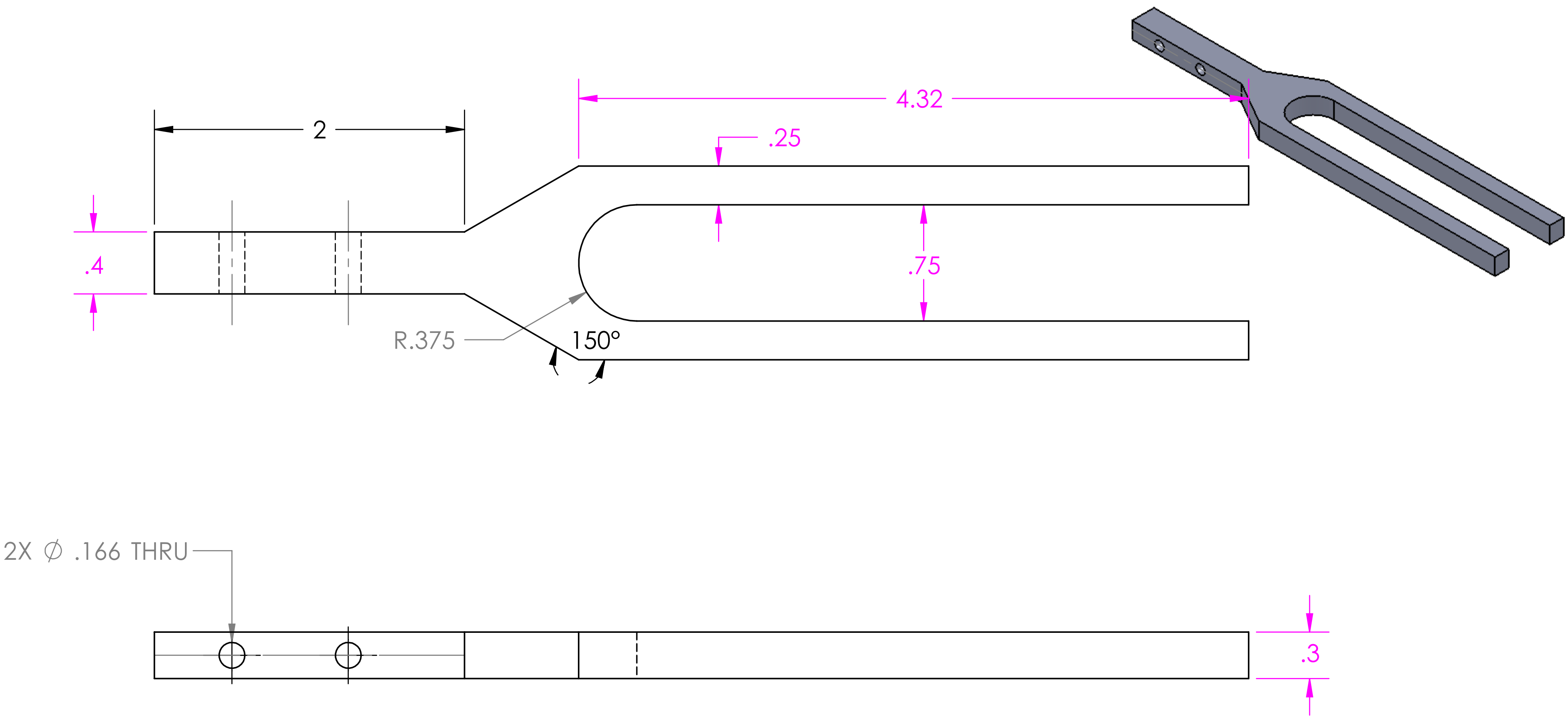
Appendix C: Mechanical Drawings

Figure C-1
Tuning Fork Drawing

Figure C-2
Tuning Fork Mount Assembly Drawing

Figure C-3
Tuning Fork Mount Assembly Drawing 2

Figure C-4
Solenoid Holster Drawing



WPI

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

NEXT ASSY	USED ON
APPLICATION	

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
TOLERANCES:
ANGULAR: MACH ± 0° 30'
ONE PLACE DECIMAL ±0.5
TWO PLACE DECIMAL ±0.15

INTERPRET GEOMETRIC TOLERANCING PER: ASME Y14.5

MATERIAL

FINISH

DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	CCW	4-30-2009
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

COMMENTS:

Worcester Polytechnic Institute

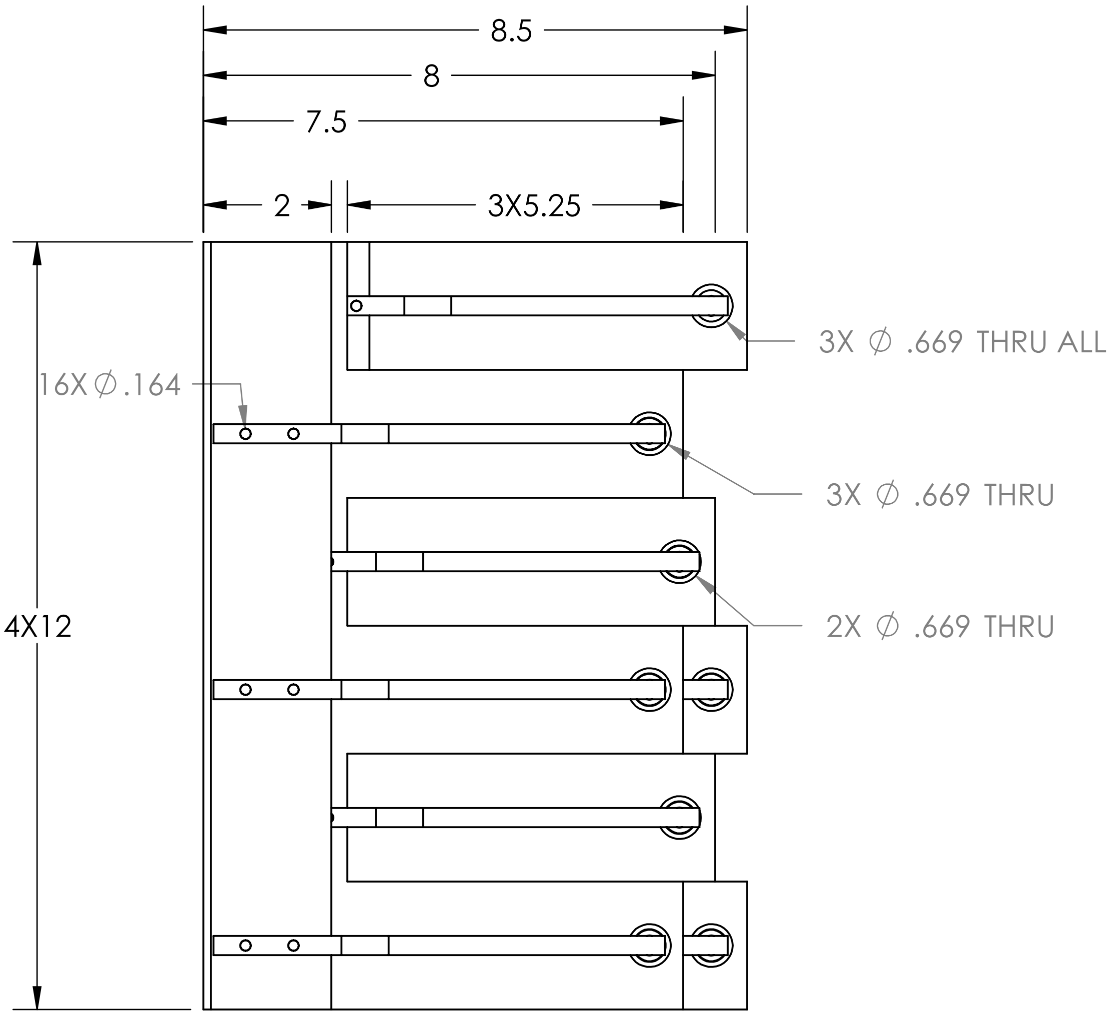
TITLE:

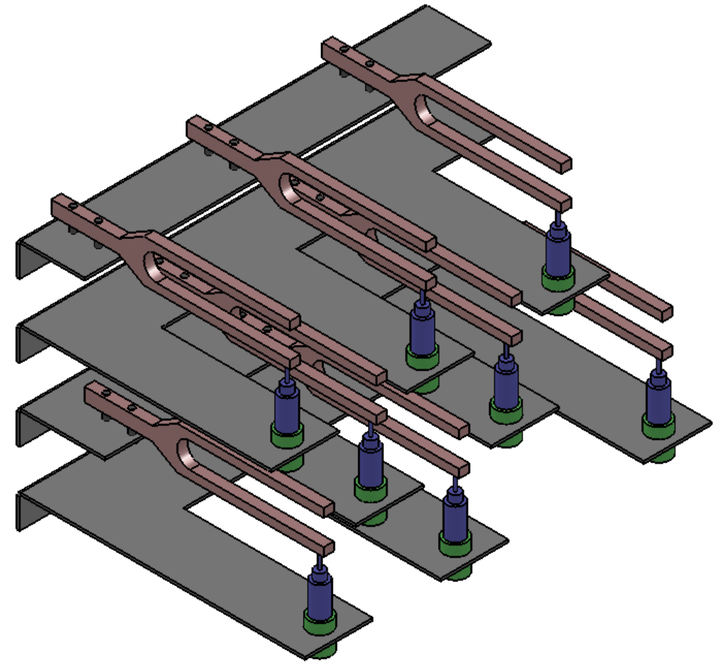
Tuning Fork

SIZE	DWG. NO.	REV
A	Fork Draw	

SCALE: 1:1 | WEIGHT: | SHEET 1 OF 1

5 4 3 2 1





		UNLESS OTHERWISE SPECIFIED:		NAME	DATE
		DIMENSIONS ARE IN INCHES TOLERANCES: ANGULAR: MACH ± 0° 30' ONE PLACE DECIMAL ±0.5 TWO PLACE DECIMAL ±0.15	DRAWN	CCW	4-30-2009
			CHECKED		
			ENG APPR.		
			MFG APPR.		
		INTERPRET GEOMETRIC TOLERANCING PER: ASME Y14.5	Q.A.		
		MATERIAL	COMMENTS:		
NEXT ASSY	USED ON	FINISH			
APPLICATION		DO NOT SCALE DRAWING			

PROPRIETARY AND CONFIDENTIAL

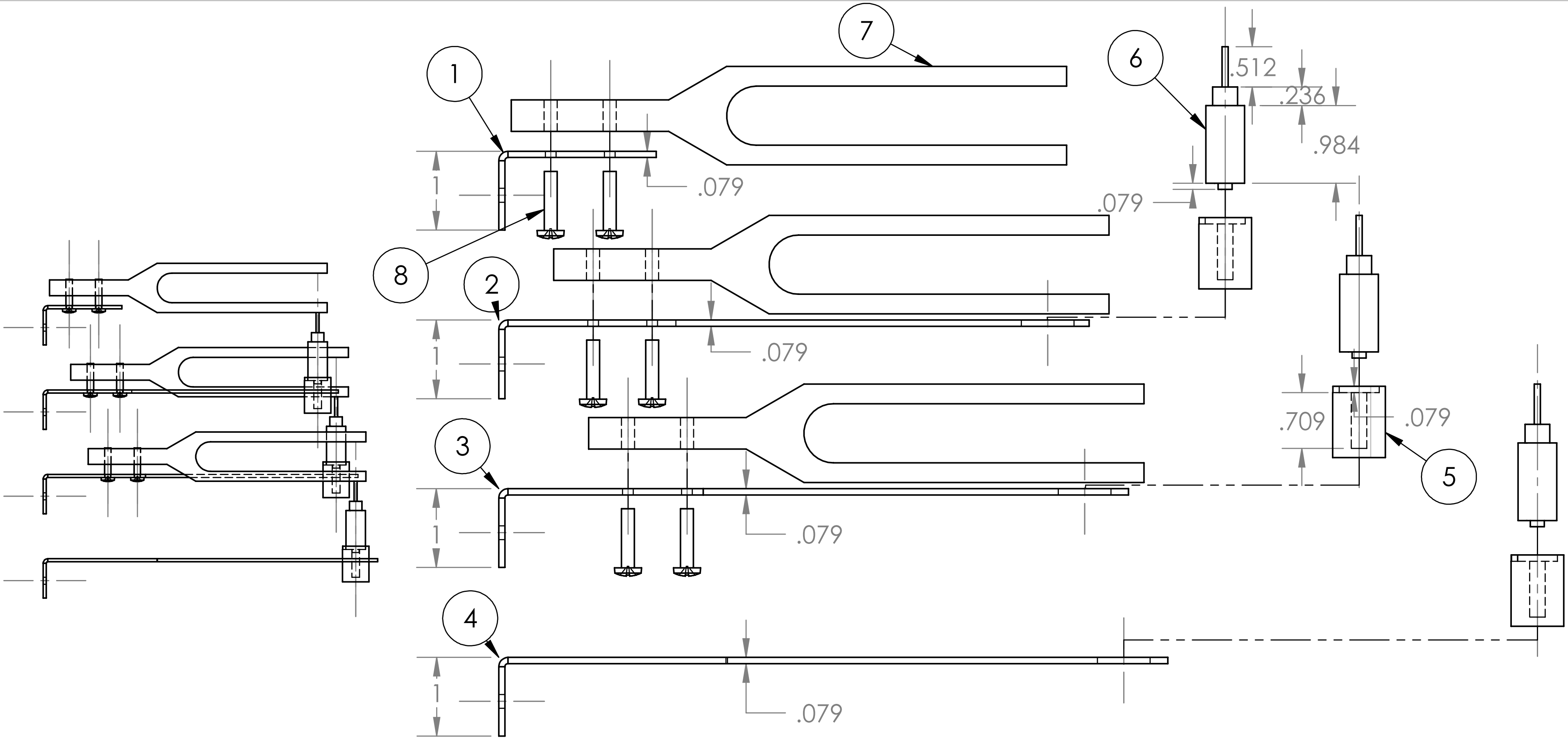
Worcester Polytechnic Institute

TITLE:

Fork Mounts

SIZE	DWG. NO.	REV
A	Mount Assembly	
SCALE: 1:1	WEIGHT:	SHEET 1 OF 3

5 4 3 2 1



ITEM NO.	PART NUMBER	QTY.
1	SheetMetalFlange Position 1	1
2	SheetMetalFlange Position 2	1
3	SheetMetalFlange Position 3	1
4	SheetMetalFlange Position 4	1
5	Holster for Solenoid	8
6	Solenoid	8
7	A	8
8	CR-BHMS 0.164-32x0.75x0.75-N	16

Exploded View

NEXT ASSY | USED ON

APPLICATION

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±

INTERPRET GEOMETRIC TOLERANCING PER:

MATERIAL

FINISH

DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	CCW	4-30-2009
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

COMMENTS:

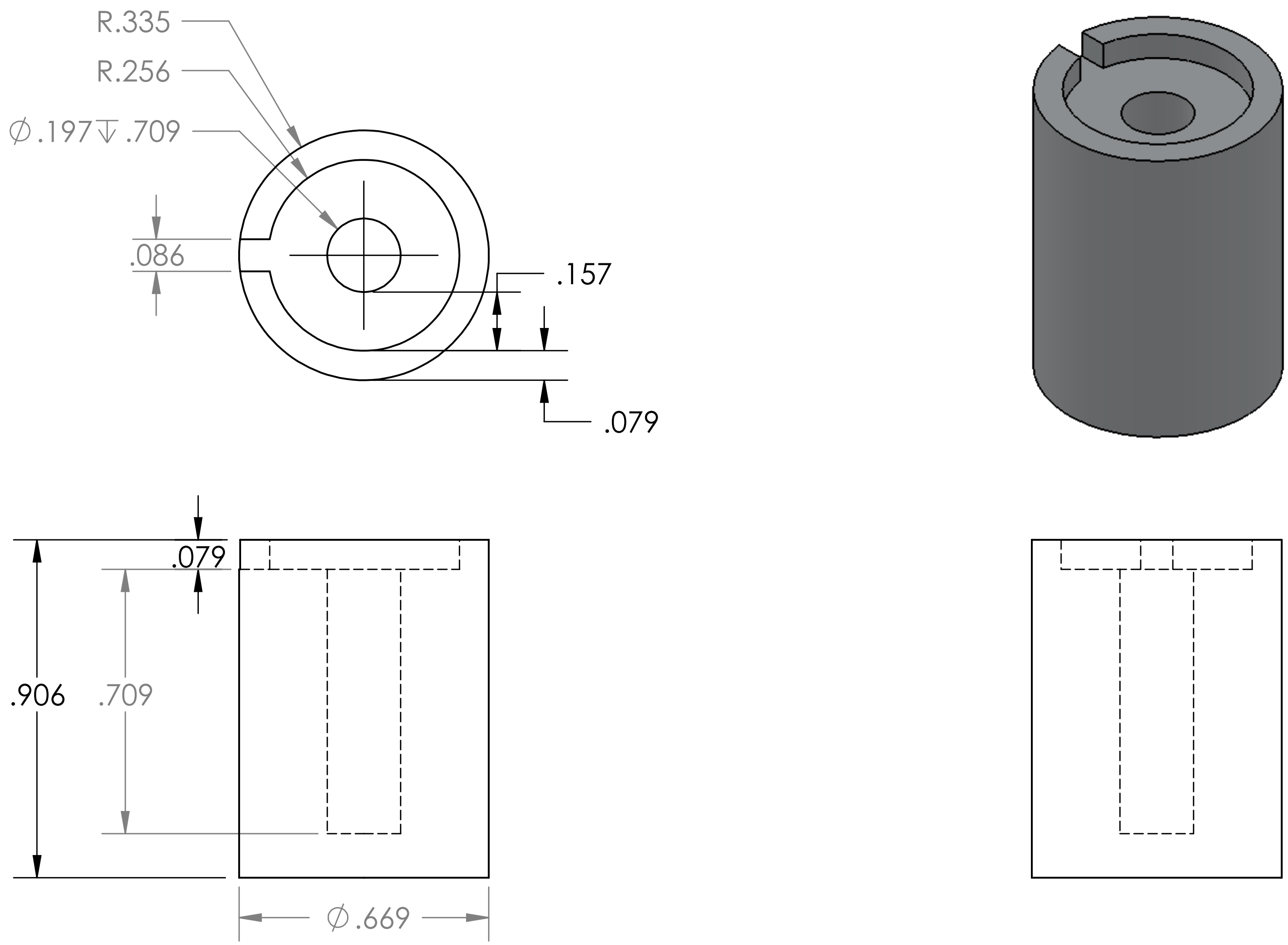
TITLE:

Fork Mounts

SIZE **A** DWG. NO. Mount Assembly REV

SCALE: 1:8 WEIGHT: SHEET 2 OF 3

5 4 3 2 1



UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±

INTERPRET GEOMETRIC TOLERANCING PER:

MATERIAL
6061 Alloy

FINISH

DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	CCW	4-30-2009
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

COMMENTS:

NEXT ASSY | USED ON

APPLICATION

TITLE:

Solenoid Holster

SIZE **A** DWG. NO. Mount Assembly REV

SCALE: 2:1 WEIGHT: SHEET 3 OF 3

5 4 3 2 1