TWILIGHT

Reconfigurable Self Organizing Office Lighting

Rauhul Varma
Naren Sivagnanadasan

Undergraduate Design Project

Dept. of Electrical and Computer Engineering

University of Illinois
Urbana-Champaign

2017

Contents

1 Introduction

1.1 Objective

With the development of low cost wireless chips and microcontroller the prevalence of internet connected devices is ever increasing. This provides a great deal of potential as we can start to exercise more nuanced control of previously simple tasks (e.g. lighting) which can allow new interesting interaction models. However, current "smart" devices instead of enabling these interaction models in a natural way, force extra complexity onto the user[1] in order to access the functionality. For instance, almost all connected device platforms require an app on your phone to configure the actual "smart" functionality, which is a user interaction that is really unnatural to most people. More intelligent behaviors typically require even more apps on your phone. Installation is a long process of pairing with a hub of some sort or connecting to wifi networks. The system also does not know how to run itself either and requires the (typically novice) end-user to program it. And if there system needs to be moved or a bulb needs to be changed out, then the system has to be reprogrammed. It is clear that the interaction models currently implemented in smart lighting systems are not user friendly and do not enable functionality that would improve outcomes for users. We can see from examples like Nest that when the user experience is considered as a key component of the development of a product and if the system does not rely on a user to enable the intelligent behavior, the benefits of the product are more likely to be utilized.

We present a platform called Twilight that seeks to demonstrate a similar level of consideration to the user experience using connected devices, allowing users to enjoy the benefits of a lighting system that thinks for them, trying to optimized the environment for productivity and comfort. Lighting is one of the subtle factors that affect our mood and productivity and building this system will bring us closer to the grand vision of our environment modifying itself to maximize comfort for its inhabitants at any given time and so it makes sense to develop systems that optimize lighting conditions[3]. However, instead of the home, Twilight targets the workplace, a space where people will spend 50% of their lives but as of yet does not have the same level of intelligent systems developed for it due to differing requirements such as scale, reconfigurability, and management requirements. By using ideas inspired by biology in self organizing systems, we seek to demonstrate a reconfigurable lighting system for workplaces that is easy to control in an intuitive fashion, easy to maintain both from a technical development standpoint and a repairs standpoint and also be a good platform for people to experiment with controlling distributed peer to peer systems. All of this can be done for significantly less than the current lighting systems being deployed in office buildings today.

1.2 Background

The environment people live and work in has a deep impact on many things including mood, psychological health and productivity[2]. Simple changes like having the correct color temperature at different times of the day or or having the environment handle simple background tasks to reduce cognitive load may help people live happier and healthier lives[3]. However, the main blockers to having these intelligent environments widely deployed include cost and rigidity of the system (i.e. the system cannot be torn down and rebuilt easily or parts are hard to replace). Twilight aims to rectify some of these problems.

Typically there are a couple issues that can cause systems to be rigid and hard to use. Dependence on user interface flows that are not intuitive or do not leverage the habits people are used to (e.g. using a light switch to control the lights) means that whatever baked in smart functionality doesn't get used[1]. These systems when not used in the design user interaction scheme often fall back to some sort of default functionality that is not much better than the standard light fixture (e.g. turns on as a pure white light by default). These systems are rigid as well because they typically have some sort of external reliance like a hub or cloud service in order to coordinate the system. The hub is a not so great solution as it is a bottleneck, typically there is are limit to how many nodes a hub can support (usually far lower than the number of bulbs needed to light a house). For cloud services there is a direct dependence on an internet connection to have a functional intelligent lighting system.

A potential solution to the rigidity problem of traditional systems is to use ideas from biology, namely the idea of collective behavior. By developing a federated system to govern the control of the entire of lighting system, we remove the central controller bottleneck. It also introduces flexibility into the system, as nodes can now enter and leave the system without the system collapsing. The system can partition itself and continue to operate as well. This sort of approach has been demonstrated before [4]. Farrow et. al. present a platform built upon this core idea and show that intelligent collective behavior can be displayed without a central controller.

1.3 High-Level Requirements

1.3.1 Simple Usage

Almost all existing smart lighting solutions require an app on a user's phone to configure their *smart* functionality. This is an unexpected and unnatural user interaction model for most people; the expected interaction is through a light switch.

Even more intelligent behaviors typically require even configuration through the companion app. Installation is a long, disingenuous process of pairing with a hub or connecting to a WiFi network.

Twilight will not require any significant external control for its operations, will make programming the system should be easy for application developers and most importantly not required by the end user.

1.3.2 Easy Installation

Installation and maintenance of Twilight should be trivial to the point where a student can put a fixture up with no help. As a result, Twilight should be compatible with standard ceiling tiles and be robust to individual Twilight Stars failing.

Twilight will detect and localize all Stars in the system and automatically reconfigure if connection to one is lost.

Twilight Stars will run off standard AC wall power and will daisy chain power to one another to reduce cabling.

1.3.3 Education Tool

Twilight is sponsored ACM@UIUC, an organization that exists to help students explore the world of computing, mainly through experimentation and project building. This is one of many platforms ACM@UIUC is bringing up to provide the members opportunities to work with complex technologies in a tangible way. Therefore Twilight must be a system that lets students easily express their creativity on this platform.

2 Design

2.1 Block Diagram

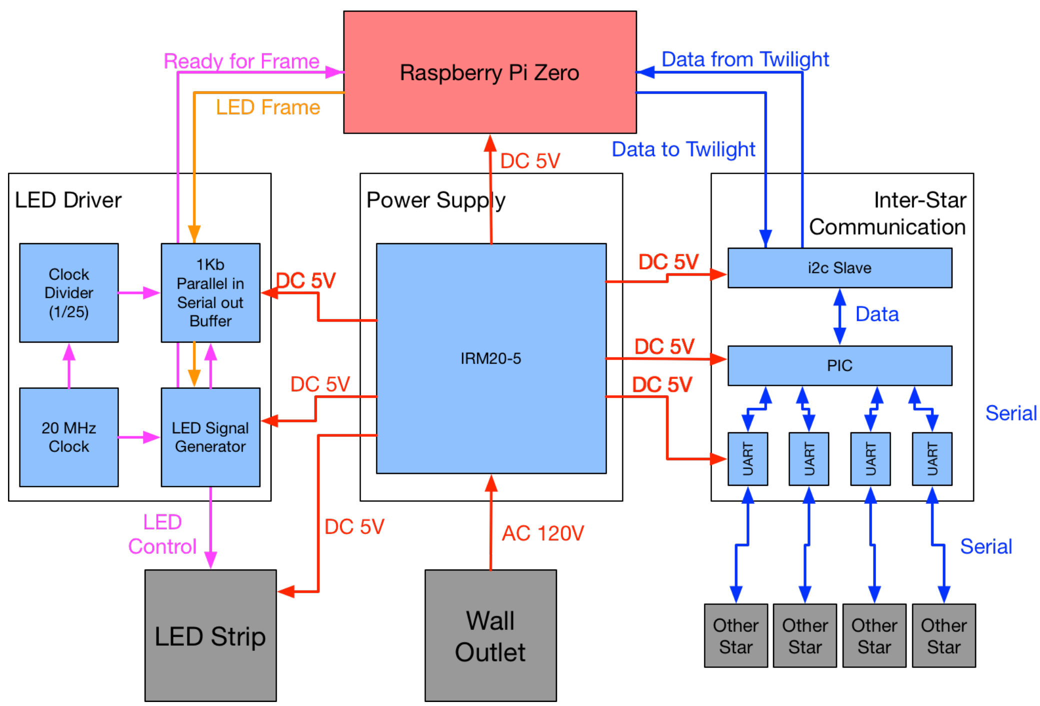

Figure 1: Electrical Block Diagram

The backbone of each light is a Raspberry Pi Zero responsible managing the software execution of the node as well as controlling the LED Driver and reacting to the messages from other nodes. Though the self organizing functionality of the project can be accomplished on a microcontroller, we choose the Pi because it provides a comfortable environment for developers (which is crucial as this is a learning platform).

We will be designing a custom daughter board for the Pi; it will include an LED driver to control an RGB LED strip within each Star. We decided to use an RGB strip because they allow for more complex and dynamic animations and smart effects than a purely warm LED strip (i.e. can change color temperature between a typical range of 2000K and 6300K). The daughter board will also include a power supply for the Pi and LEDs to run off of standard AC power and a I2C to 4-port UART breakout to allow for inter-Star communication. We initially planned on using ethernet between each unit however the decided against it due to the technical complexity of creating/housing the corresponding required networking switch.

2.2 Physical Design

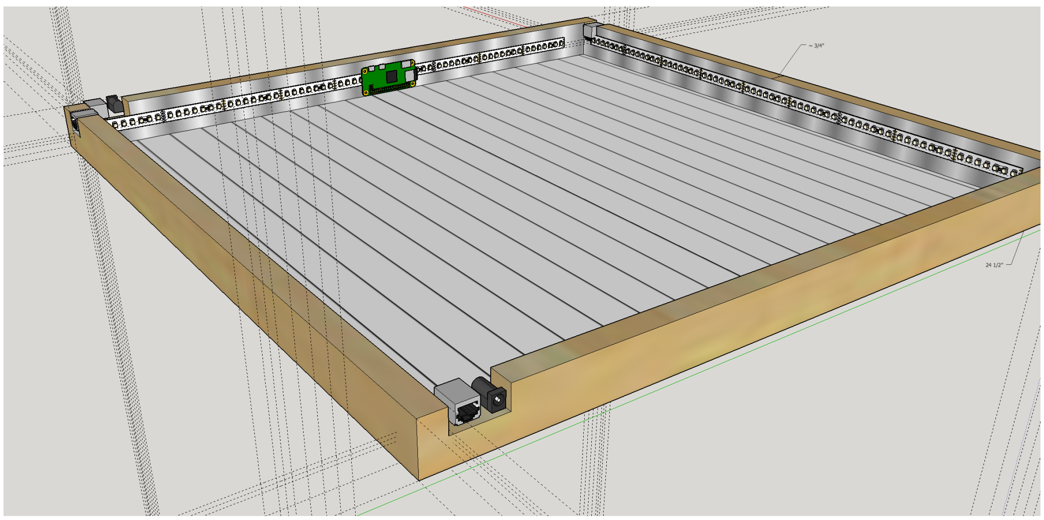

Figure 2: Top view of Star frame

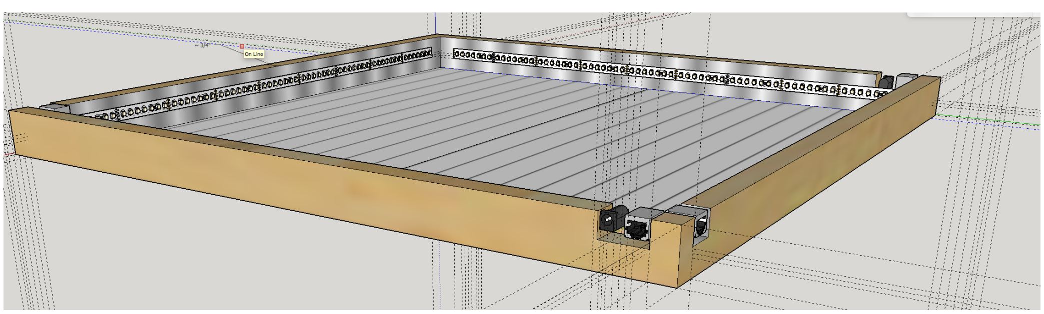

Figure 3: Barrel and RJ45 Jack inlaid into Star frame

The entire system consists of a collection of 12 Stars. Each Star is a magnetically suspended wooden box with the same dimensions as a standard ceiling tile–a square with external width 24.5" and internal width 23". The internal sides are covered with with aluminum tape, improving the total internal light reflection. 140 of LEDs are wrapped along the these inside edges. The bottom face of the frame is covered by a canvas defuser mounted to the inside of the frame. A Raspberry Pi Zero with a custom Pi hat is mounted to the inside of the Star and controls the Star's on board communications, LEDs and power systems. Two barrel jacks to provide and distribute power are inlaid into the two adjacent sides. Lastly, an RJ45 jack is also inlaid into the frame for serial communication between Stars.

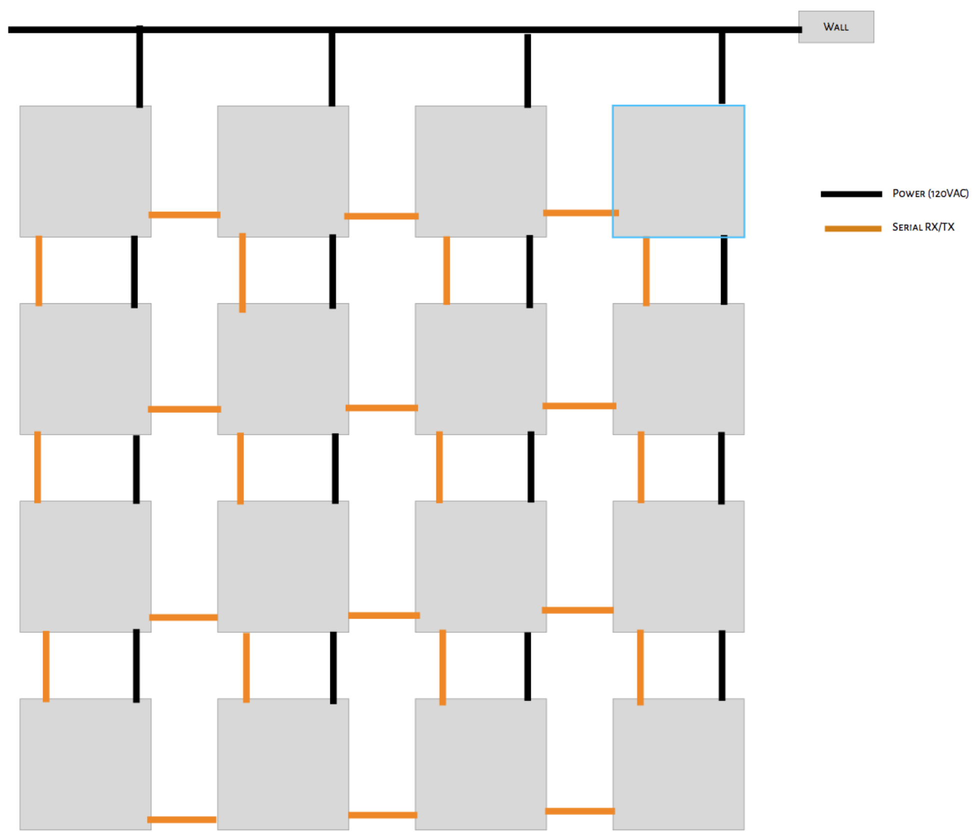

Figure 4: Proposed Default Topology

Figure: 4 is the proposed default topology. Each block is connected to its neighbors via a serial connection and power is distributed via 120VAC rails extending from the wall, into each block and out the other side, which powers the Pi and LEDs. While other configurations of the system will be supported this will be the default implementation as it most closely matches the space we are going to deploy the system in.

2.3 Block Requirements

2.3.1 LED Driver

1. Display 30 frames per second on a Star with a variability between frames of no more than ±3ms.
    - Note: Frame rates lower than 30 fps appear disjoint and choppy, creating significant detriment to a user's experience of the system.
    - Verification: Capture animations on a Star with a high-speed camera, ensure that each frame takes between 30ms and 36ms to display.
2. The current frame is displayed until a new one is available from the Raspberry Pi.
    - Verification: Generate frames at 1 fps (starving the Star) and check for flickering. No flickering should be observed.

2.3.2 Power Supply

1. Convert $120V_{AC}$ from a standard AC wall outlet to $5V_{DC}$.
    - Verification: Measure output on oscilloscope.
2. Supply 3A at $5V_{DC}$ continuous current and up to 4A peak.
    - Verification: Load Power supply with 1.6Ω resistor and protective fuse, check that this fuse never trips.
3. Weigh no more than 1 pound.
    - Verification: Weigh component.
4. No larger than 55mm in any dimension.
    - Verification: Measure component.

2.3.3 Inter-Star Communication

1. Robust to multiple stars attempting to initiate a message transfer.

- Verification: 2 or more nodes cannot be listened to at the same time.

2. Messages from any node looking to initiate a message transfer will eventually be transmitted.
   - Verification: 2 or more nodes messages will all be received eventually.
3. Communication will not dead lock due to loss of connection.
   - Verification: A node can be unplugged mid transfer and the remaining node returns to listening.
4. Message data can be transfered uncorrupted.
   - Verification: Large amounts of data can be transfered between 2 nodes.
5. A localization map must be able to be developed from the connections between nodes.
   - Verification: Any node can query the topology of the network.

2.4 Risk Analysis

The Power Supply block appears to be the most difficult to implement and biggest risk to a successful completion of Twilight. The main reason for this is an overall lack of strong knowledge with power systems (beyond that in ECE 330) between the group members. Moreover, on top of the non-trivial electrical requirements, this block has significant weight restrictions that make the design more difficult. That being said, there is a substantial amount instructional material and guides on the subject online that restore confidence in completing this block.

2.5 Functional Overview

Each block in the diagram is essentially independent of all other blocks other than the fact that they live on the same pcb. The only interdependence is between the Power Supply and the other two as the Power Supply is how the other blocks are able to function.

2.5.1 LED Driver

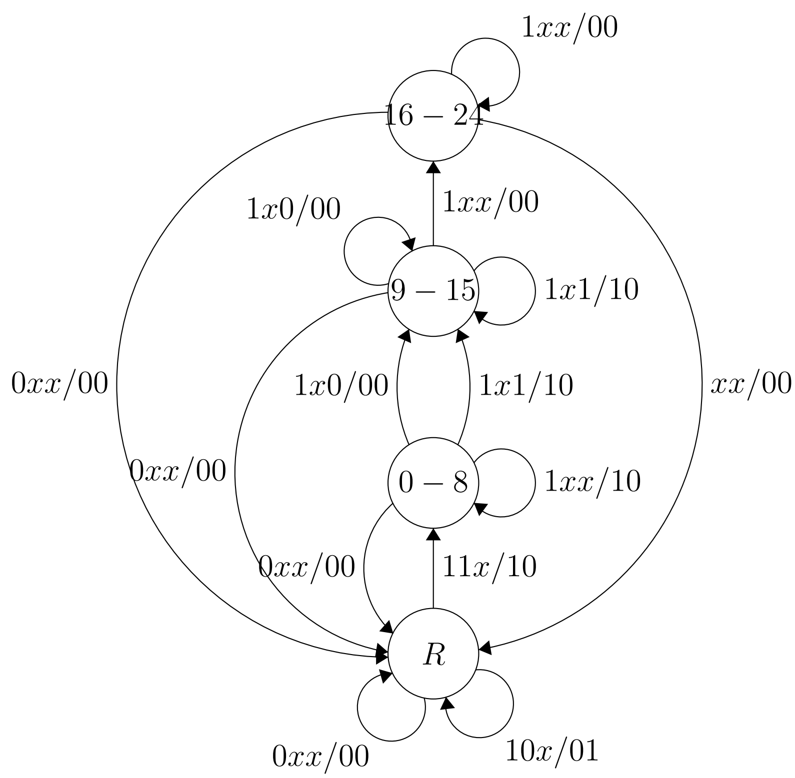

Figure 5: Condensed LED control signal generator FSM

LED Driver is responsible for taking frames (a description of the color of each LED in a Star) from the Raspberry Pi and displaying them.

The Raspberry Pi commits a frame via 16b parallel stores to a 1Kb frame buffer on the daughter board. The Pi then triggers the LED Signal Generator to start pushing the data onto the LED Strip. The Signal Generator Serially reads the data in the 1Kb buffer and creates the digital signals the LED Strip expects.

The Generator uses a 20 MHz clock to meet the times laid out by the LED Control Signal Spec and 1/25 clock divider to control serial reading from the frame buffer. The signal generation works via an FSM shown in Figure 5, an expanded version of this FSM can be found in the appendix in Figure: 8.

2.5.2 Power Supply

For the power supply, Twilight uses an integrated switching power supply that provides 4A at 5v. This approach was used due to a couple of reasons. Working with AC power is dangerous and using a integrate system mitigates that risk. Also the use of a switching power supply allows for a much smaller package than a traditional transformer based power supply. This allows the power supply to achieve its weight and size requirements. However, a switching power supply is more complex than a standard transformer based one and would push the amount of work for this project out of the scope of this class.

2.5.3 Inter-Star Communication

The I2C to UART block manages the inter-Star communication. Raspberry Pi Zero comes with a single dedicated UART handler exposed on GPIO. However each block may have to support up to 4 connections. We therefore use I2C and PIC as a way to multiplex a single connection to a PI to the four serial connections it needs to support. The design is based off of an Atmega 2560, which has 4 hardware UART implements are protocol translating each incoming/outgoing UART connection into a I2C slave. We also developed a protocol to manage communication between 2 nodes such that nodes will not try to talk over each other.

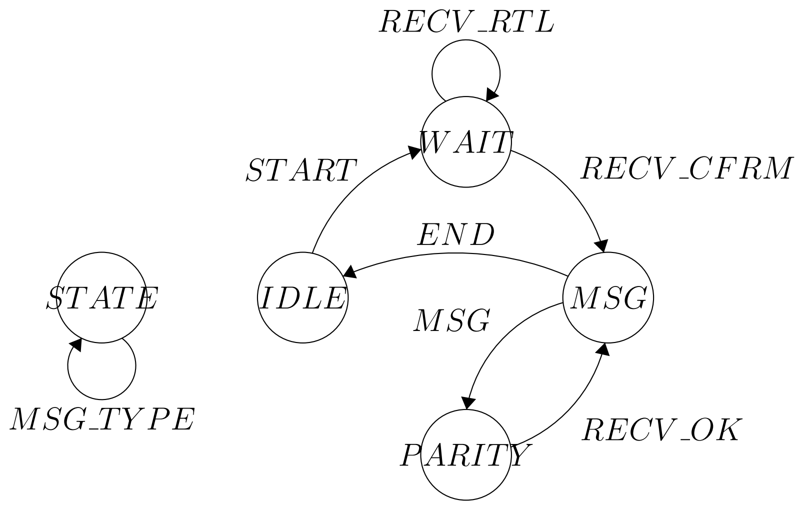

Figure 6: Initiator Protocol

The protocol starts in with the initiating block (referred to as the initiator) as shown in Figure 6. The initiator starts in the idle state, and after deciding to send a message to a node, sends a request to start a connection (termed: "Ready to Listen") and waits for a "Ready to Listen Confirmed Message" from the receiver. Once getting the confirmed message, the initiator begins transmitting the message, sending a packet, then a hash to confirm integrity. Once that is completed, the initiator returns to the idle state.

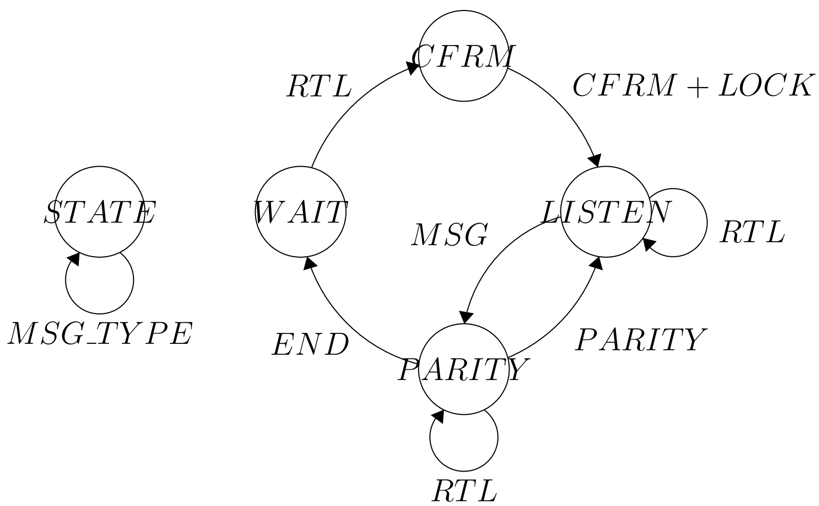

Figure 7: Receiver Protocol

For the receiver, once the "Ready to Listen?" message is received, then it either responds with confirm to start a connection or it ignores it (e.g. if there is another communication occurring). After a confirm is sent, the RX side of the connection on the responder is locked, so it will not listen to anyone else. Then it will wait for a packet, then for a hash to verify integrity. Once the initiator send the END message, the receiver returns to the wait state.

2.5.4 System Integration

Since we are creating a multinode system, it is important to discuss how the Stars should connect together to realize a full Twilight system.

On system initialization (i.e. first installation) the system will begin with a localization procedure where starting from the blocks with only 2 connections, a localization message will propagate through the system assigning each block a location. Then the system will enter the application space and begin to execute its current application (in this case the color temperature application). Periodically clocks will be synced using Network Time Protocol (NTP) and there will be a nodes in the system that will act as a border routers, providing a network connection, so updates can be applied.

3 Schedule

- 10/3/17
  - Initial Inter-Star network MVP
  - Parts ordered (canvas + connectors + other hardware)
- 10/10/17
  - Prototype LED controller (on breadboard)
  - Partial implementation of Power Supply
- 10/17/17
  - Finish Power Supply
  - Finish Star network hardware
- 10/24/17
  - Send daughter board out to fab
  - Finish Inter-Star network stack
- 10/31/17
  - Populate PCBs + verify
  - Start application layer development
- 11/7/17
  - Address potential PCB bugs + send out new revision
  - Begin construction of final Twilight Stars with canvas diffuser
- 11/14/17
  - Populate revised PCB
- 11/21/17
  - Finish API
  - Create frontend + demo app
- 11/28/17

  - Install array
  - Prep demo
- 12/4/17
  - Demo

4 Ethics and Safety

The biggest safety issue we face is working with wall power since it is $120V_{AC}$. Since every unit in the system will be working off wall AC power, verification of the power supply for the Pi is crucial. Additionally there are concerns around issues like epilepsy when dealing with fast animations. This can be addressed by limiting the frequency of the LEDs and by reviewing software deployed on the system.

There are potential environmental concerns that may arise when sourcing LEDs as some have been found to contain lead, arsenic and other dangerous substances. Proper care in sourcing RoHS compliant components must be taken.

Moreover, while this platform is designed to be used to improve peoples' mood (through the use of color temperature modulation), one could the system to decrease peoples' quality of life. For instance, many people have complained about short wave heavy white LEDs used in streetlights preventing people from sleeping.

Extended viewing of LEDs directly could also cause retina damage. The system also could potentially contribute to the growing issue of light pollution (though this is mostly an issue with outdoor lighting).

5 Appendix

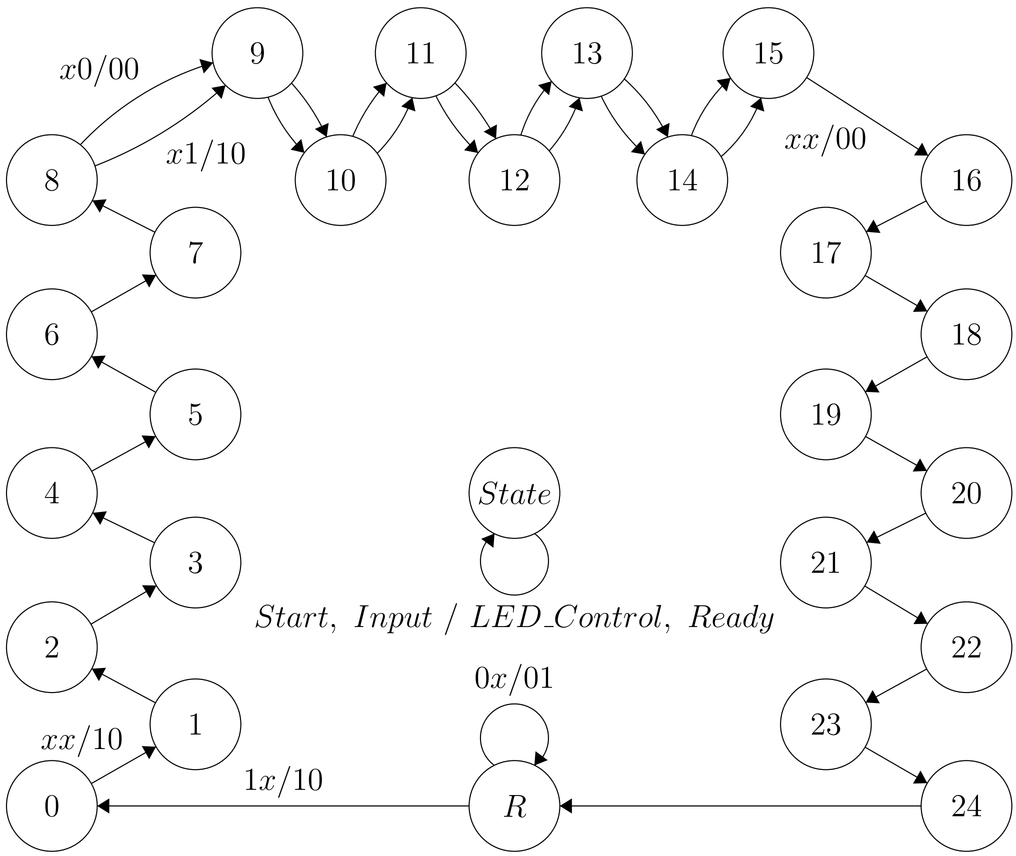

Figure 8: Expanded LED control signal generator FSM

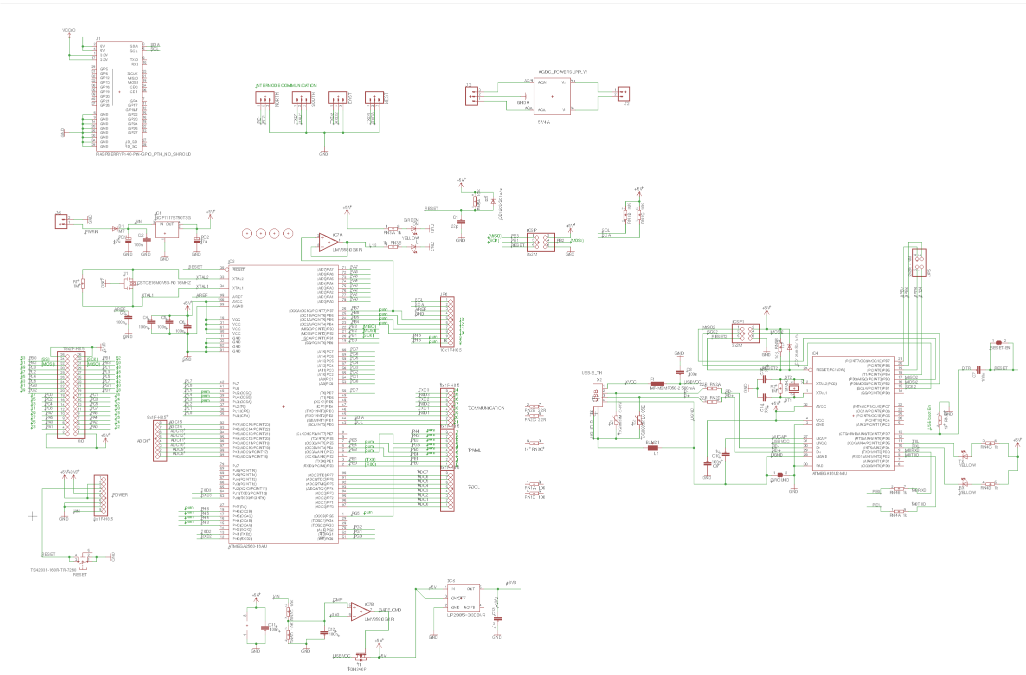

Figure 9: Twilight Star daughter board PCB schematic

References

[1] C. Rowland, "What's different about user experience design for the Internet of Things?", O'Reilly Media, 2017. [Online]. Available: https://www.oreilly.com/learning/whats-different-about-user-experience-design-for-the-internet-of-things. [Accessed: 03- Oct- 2017]

[2] W. van Bommel and G. van den Beld, "Lighting for work: a review of visual and biological effects", Lighting Research & Technology, vol. 36, no. 4, pp. 255-266, 2004.

[3] I. Knez, "Effects of indoor lighting on mood and cognition", Journal of Environmental Psychology, vol. 15, no. 1, pp. 39-51, 1995.

[4] N. Farrow, N. Sivagnanadasan and N. Correll, "Gesture based distributed user interaction system for a reconfigurable self-organizing smart wall", Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction - TEI '14, 2013.

[5] H. Ishii, H. Kanagawa, Y. Shimamura, K. Uchiyama, K. Miyagi, F. Obayashi and H. Shimoda, "Intellectual productivity under task ambient lighting", Lighting Research and Technology, 2016.

[6] R. Küller, S. Ballal, T. Laike, B. Mikellides and G. Tonello, "The impact of light and colour on psychological mood: a cross-cultural study of indoor work environments", Ergonomics, vol. 49, no. 14, pp. 1496-1507, 2006.

[7] J. Kim, J. Ko and M. Cho, "A Study of Integrated Evaluation of System Lighting and User Centered Guideline Development - Focused on the Lighting Design Method for Office Space -", Korean Institute of Interior Design Journal, vol. 23, no. 6, pp. 78-86, 2014.

[8] D. Park, Y. Lee, M. Yun, S. Song, I. Rhiu, S. Kwon and Y. An, "User centered gesture development for smart lighting", HCI Korea 2016, 2016.

[9] F. Tan, "User-in-the-loop smart lighting control system", Masters, Delft University of Technology, 2016.

[10] D. Burmeister, A. Schrader and B. Altakrouri, "Reflective Interaction Capabilities by Use of Ambient Manuals for an Ambient Light-Control",

HCI International 2016 – Posters' Extended Abstracts, pp. 409-415, 2016.

[11] A. Lucero, J. Mason, A. Wiethoff, B. Meerbeek, H. Pihlajaniemi and D. Aliakseyeu, "Rethinking our interactions with light", interactions, vol. 23, no. 6, pp. 54-59, 2016.