Daynight Panel

Andrew Haas, Ornella Ngalamulume, Ricardo Rivera-Matos, and Jordan Larsen

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

Abstract **— Providing a sense of outside daylight can "ground" a human's circadian rhythm in instances where access to daylight is sparse, infrequent, or unavailable. The Daynight Panel is designed to improve the internal ambient conditions of a room by illuminating it with artificial daylight or moonlight. The panel is designed to provide a relief for the eyes and provide the inhabitants of a room with the warmth and relief of being connected to the outdoors while they are inside. The Daynight Panel will be paired with a mobile application to allow custom and scheduled updates to allow users to enjoy a unique experience. There are many devices that provide this service, but the aim of this project is to make the panel affordable, configurable, scalable, and relatively easy to install.**

Index Terms **— Bluetooth, circadian rhythm, lighting control, LED lamps, microcontrollers, mobile applications.**

I. INTRODUCTION

The Daynight Panel is a system that simulates outdoor lighting by using a grid of LEDs behind a light diffuser. The LEDs provide a source of light by emitting exact, calculated measurements of color, composed by a mix of color intensity and color temperature. The light diffuser will mask the individual LEDs to create the blurred or frosted effect of looking through a window. The diffuser and the LED's are intended to create the illusion of looking out into a window or of light coming in from a window. By pairing the Daynight Panel to the Daynight app for iOS, users will be given the ability to take full advantage of the customizability with the LEDs. Users will be able to mix colors that mimic outdoor lighting effects, tap in to RGB LEDs to draw their own creative vistas, and save favorite configurations to use anytime.

All these features will be performed and controlled by a microcontroller, which will be at the heart of the panel controlling the LEDs, sending information to or receiving instructions from the Daynight app.

The Daynight Panel will be optimized to provide the best outdoor atmosphere. It relieves homeowners, architects and designers of the burden of comfortably illuminating living spaces and it creates opportunities for saving time and money in efforts to achieve this effect otherwise. For example, this expands the opportunity to accommodate more indoor-locked rooms or basements while providing a pleasant atmosphere. A daylight simulation system may also be used to benefit people living in gloomy climates, people that suffer from isolated work environments, closed off or walled-in classrooms or housing quarters, those who experience jet lag, or humans in space.

II. DESIGN MOTIVATIONS AND MARKETING

This section is focused on elaborating project motivations, goals and objectives as it pertains to marketing, the customer, and how these requirements are quantified and met through engineering specifications. These motivations are based on the following marketing requirements.

- The panel and accompanying software applications should have a modern appeal and design.
- The panel needs to be 'smart' by acting on data gathered through sensors or an iOS app to reproduce daylight.
- The panel needs to be safe and easy to install.
- The panel should have a sustainable lifespan.
- The panel should be affordable when compared to similar light panels currently on the market.

The modern appeal is necessary to compete with what is currently on the market. There are currently many panels and fixtures on the market that mimic daylight or daylight color temperature. These systems usually come in one of three forms: an individual panel or set of fixtures which the user controls by an application, a panel that can be switched or dialed into different settings, or an extravagant statement piece.

The panels set and controlled by an application are usually the most affordable and configurable. One of the current market leaders in this product is Phillips with their HUE lightbulbs. These bulbs are designed with a variety of settings to operate without the constant need for Wi-Fi. They provide a collection of smarts lights loaded with features dealing with different modes of control, improved home automation, entertainment and notifications [2]. Their product comes as a collection which can be easily configured for any location. This form is the closest parallel to what the panel will achieve. The main difference boils down to the end shape of each device. The panels rectangular shape is designed to make it seem like a window or portal designed to improve the ambiance of a room.

The panels with dials or switches come in all shapes and sizes. Some of these resemble frosted or diffused windows reflecting a warm or cold outdoor scene depending on the setting. These panels, beautiful as they are, seem outdated because their controls are usually hardwired inside the panel. This leaves the panel to be inconveniently controlled by a switch at the entrance of a room. The Daynight Panel brings ease of operation from anywhere within a home or a room and an infinite amount of display options.

The third form is the extravagant statement piece. These are usually architectural projects designed for a specific location and because it is so specialized it becomes unaffordable for most. These panels cater to a specialized audience and require the rooms to be customized for the panel rather than the other way around. The panel on one hand can be a simple fixture but on the other it can be a dynamic piece of wall art that can be reconfigured and hung in just about any location.

By bringing custom controls, good hardware, sleek housing and ingenious software design together, the Daynight Panel refreshes a stale and stagnant market for sunlight-mimicking light panels. These new ideas help fill the gap and have served as the motivation and inspiration for this project.

III. RESEARCH AND PARTS SPECIFICATION

This section details the research done for the various hardware and software components chosen in the Daynight Panel. The research will be discussed in subsections correlating to the subsystems of the Daynight Panel

A. Lighting

The Daynight panel utilizes three different types of SMD LEDs for daylight simulation. It uses a 6500 Kelvin (K) 'cool', a 2700K 'warm', and a RGBW LED. The cool LED creates light similar to afternoon conditions, the warm LED creating light similar to dawn and dusk conditions, while the RGBW paints a color spectrum onto the panel.

The parameters that dictated what LEDs to use were current draw, lumen output, and price. The most discriminating factor, current draw, limited the panel to LEDs that had a normal draw of 100 mA to match the LED drivers.

Color mixing between LEDs is used to

obtain the full daylight spectrum. This is done by mixing and matching varying outputs of the cool and warm LEDs so that temperatures between 2700K and 6500K can be obtained. This is the same process that RGB lights use to achieve a variety of colors.

A diffuser is used to aid in the color mixing, with the additional utility of reducing glare and homogenizing the multiple LEDs into a single source. Diffuser film that attaches to a piece of acrylic was chosen to do this.

The film was chosen with a middling thickness of 0.005 inches, with higher thicknesses being more diffusive. The middle option was chosen because if one film wasn't enough, a second could be placed on the other side of the acrylic. The Daynight panel ended up needed two films to properly diffuse the LEDs.

B. LED Driver

Rather than using several of the individual general-purpose input/output (GPIO) pins on the microcontroller to drive multiple LEDs, an LED Driver is used. The basic requirements for the panel's LED Driver are to be able to drive our higher lumen output LEDs while maintaining board space and power efficiency, support serial communication, drive each LED individually, support high resolution color mixing, and to be easily scalable.

The TLC59116 is a 16-channel, current sink, LED driver capable of driving 16 LEDs using a constant current source rated for an absolute maximum current of 120 mA per channel. To achieve brightness control, the TLC59116 employs analog dimming using an external current resistor and digital brightness control by use of integrated pulse width modulators. The brightness resolution is 8 bits or 256 steps for the digital brightness control. The individual output channels are controlled by sending instructions from the microcontroller to the TLC59116 over an I^2C bus. The bus is composed of a clock, data, and reset line pulled up to VCC coming from the microcontroller to establish a bus. The drivers would each connect to this bus and receive instructions from the controller on how to drive the LEDs.

C. Power

The Daynight panel will use a 120W AC-DC power supply. This number comes from the rounding up of the highest voltage needed in the circuitry. The greatest voltage needed on the Daynight panel is the 9.4 V forward voltage of the warm white LED, so the output voltage of the power supply is rounded to 12 V. The input current is determined by finding the sum of the major sources of current draw. The major sources of current draw are the LEDs and the maximum current draw of the LEDs is 7.44 A. To ensure safety, the total current is rounded up to 10 A.

Buying an off-the-shelf power supply unit has the advantages of speed and reliability. Quick online research reveals countless 120W power supplies with an output of 12 VDC capable of 10 A current draws. These power supplies typically come with a warranty and marks of certification ensuring safety and emissions performance. Ultimately the Aiposen S-120-12 120W power supply was selected for the Daynight Panel to supply the printed circuit boards with 12 VDC at up to 10 A.

Beyond the AC to DC conversion, four voltage rails for each of the LEDs and ICs must be produced. The current draw from each of the power rails calculated and recorded in Table 1.

Table 1. Current Draw

	Voltage [V]	$Current_{max}$ [A]
VCC	5.0	<1
RGBW	3.3	4.8
COOL	6.5	1.2
WARM	9.4	2.88

Based on these requirements, four switching regulators were selected. These specific regulators were chosen not only because they can supply enough current and voltage, but also because they integrate as many passive components as possible thereby reducing the BOM.

Table 2. Switching Regulators

	Voltage [V]	$Current_{max}$ [A]
LMZM23601	5.0	1
LMZ23605	3.3	5
LMZ34202	6.5	2
LMZM33603	9.4	3

D. *Microcontroller*

The choice for the appropriate microcontroller is centered around the following features. The controller must support the Bluetooth LE Module, have hardware serial, I^2C, SPI support and USB to Serial programming capability. These are the basic features which govern the requirements for discrete components directly connected to the microcontroller.

The ATmega328P is the controller for the Daynight Panel. It provides 32KB of storage, 23 I/O pins, and can operate between 1.8V and 5.5V. The microcontroller's operating voltage determines the definition for a logic high and low on the I^2C bus. The Daylight panels control system will consist of several peripheral devices talking to the controller and since the mode of communication will be I^2C all devices need to be able to connect to the Data and clock bus. Therefore, all devices connected on the bus should be able to communicate at the bus's voltage with a decent margin of safety to ensure there are no floating values or misconstrued signals to corrupt the data being transmitted.

The ATmega328P is the controller that has been chosen for our application. It is just the right size and does not limit the panel's capabilities while still meeting budget constraints.

E. *Communication Protocol*

Wireless networking plays a crucial role in our project in order to facilitate communication between the Daynight app and the Daynight Panel. Here, different kinds of wireless networking and communication were considered and discussed to find the best method of transmitting information across the Daynight Panel's peripheral devices.

Since our project consists of an integral line of communication between the user's device and a nearby Daynight Panel, personal area network (PAN) protocols like Bluetooth are employed for communication.

Bluetooth has two classifications of technology: Bluetooth Classic and Bluetooth Low Energy. Bluetooth Classic was designed with the initial intention of creating a constant stream of data within a personal working area. This constant stream of data brings up concerns about long-term energy consumption, which is important to consider when building software for mobile devices with limited battery capability [1].

Bluetooth Low Energy (also sometimes called Bluetooth Smart, or Bluetooth LE) was introduced alongside Bluetooth 4.0 in 2011 [2]. Bluetooth LE aimed to provide similar functionality to smaller devices that didn't require a continuous stream or exchange of information. A clear benefit to devices that desired this kind of support is that these operations aren't energy-intensive. These devices would only require infrequent and small packages on information. An example of this behavior in relation to the Daynight app and the Daynight Panel is sending a one-time control over Bluetooth LE. During the rest of the day, the

Daynight app and Daynight Panel do not require any more interaction since there is no information ready to exchange.

Though there is appeal with a common technology like Wi-Fi, online servers or databases would make the software communication process easier between the Daynight app and the Daynight Panel. Additionally, the operation of the panel would directly rely on availability of Wi-Fi networking in the home and would not be available for users that do not have in-home network. In this case, Bluetooth provides for a more reliable communication and has easy-to-use protocols for small transmissions of data.

F. *Daynight App for iOS*

Mobile applications can be written in a range of programming languages. In iOS application development, two programming languages are most common: Objective-C and Swift.

Objective-C is a superset of the C programming language that introduces object-oriented concepts. Objective-C was developed in the 1980s by NeXT Computer, which was eventually acquired by Apple in 1997. Apple used Objective-C to create iOS and macOS [3].

Swift was announced by Apple at the Worldwide Developers Conference (WWDC) in 2014. It has become one of the fastest growing programming languages in history [4]. Swift is held to high standards by the following three pillars:

- Safe
- Fast
- Expressive

Swift competes with C programming languages like C, C++, and Objective-C with the speediness for performance tasks (hence the descriptive name for the language)! Swift is the language of choice for this project because of its safety, speed, and ease of use.

Core Bluetooth is a framework in Swift and Objective-C that provides extra functionality in code to allow apps to communicate with Bluetooth LE devices. In the scope of our project, the software app will be equipped to communicate with the Bluetooth LE receiver to transmit and receive information that the microcontroller will use to control the Daynight Panel.

IV. PROJECT DESIGN AND INTEGRATION

With hardware and software components chosen, the next step would be to design the overall project to meet these desired specifications by integrating the selected components together. Two printed circuit boards are designed to integrate the hardware components. On the software side there are two programs that are written. One is for the microcontroller and the other is for the smartphone.

A. *LED Module Printed Circuit Board*

The LED Drivers and the LEDs will be configured on the LED Module PCB. The LED Driver circuit is configured by connecting the cathode of the LEDs to the output channels of the LED Driver and the anode to the appropriate forward voltage rail. On the driver, a unique I^2C address must be hardcoded by using VCC and GND on the address pins. The constant current source for all the channels is set to 93.75 mA using an external resistor according to (1). Finally, the LED Driver is connected to the SDA, SCL, and RESET lines. This configuration is repeated six times on the board and the power and communication lines are connected to the power and logic PCB.

$$I_{OUT} = \left(\frac{1.25V}{R_{ext}}\right) * 15 \quad (1)$$

B. *Power and Logic Printed Circuit Board*

The four switching regulators are configured on a smaller PCB that contains the power components, the microcontroller, and a soft reset push button switch. The switching regulators are placed on a half of the printed circuit board that contains a copper ground plane. This cooper ground plane along with thermal vias help pull heat away from the regulators. The microcontroller is configured on the other half of the PCB. The ATMega328P only requires two capacitors and a 16 MHz crystal to operate. In the I^2C configuration, pull up resistors are needed on the communication lines. A reset push button is also added to enable a soft reset in the case of a glitch. The microcontroller is also using SPI to communicate to the Bluefruit Bluetooth LE daughter board soldered to the PCB. A ground plane was not added to this half of the PCB to reduce the capacitance on the communication traces.

C. *Microcontroller Software*

The panel requires the integration of a Bluetooth module, a micro-controller and the LED drivers. All scenarios require commands to be issued from the app to elicit a response in the drivers. The controller takes the commands, processes them and sends them to the drivers. The controller receives an array of characters from Bluetooth module which provides a key and a value for the red, blue, green, cold and warm LED's. The key is a code for detailing the type of scene that the panel displays. The LED values range from 0-255 and are each 3 characters long. The complete packet is a 20-character array. The packet size is limited to improve communications. Currently, there are packets for 3 cases. The first case paints the entire panel a single color, the second case paints each cell an individual color, the third case automatically runs the panel through a daylight sequence mimicking light for the current time of day.

All this processing is done with several functions in Arduino. In summary, all commands are entered into a function that reads the packet and calls the appropriate function to decipher data or assign the appropriate value to the appropriate driver.

D. Daynight App for iOS

The Daynight app for iOS will follow the common software architectural pattern Model-View-Controller (MVC). This architectural pattern describes the organization and flow of data when dealing with user interfaces. Although this pattern is used in a majority of mobile software applications, it can be extended for use in web applications and desktop clients, too.

The *model* encompasses all data types and data structures used throughout the app. The key here is to have the *model* abstracted from the state of the UI. That is, the storage of data is not dependent on—and is unaware of—processes handling the UI. The *view* represents all output to the UI. Any information that is present including data, symbols, and other view elements like tables, tabs, or bars are included in the *view*. Data in the *model* may be presented in the *view*, but the *view* does not manipulate data in the *model*. The *controller* bridges the abstraction between the *model* and the *view*. Here, any interactions to the UI that will result in modification of the data in the *model* will be handled. Similarly, if the data in the *model* is updated and needs to be displayed, the *controller* updates the *view*.

V. OPERATION AND USE

Using the Daynight Panel is as easy as using the Daynight app for iOS. On the home view, the user is shown the current state of the Daynight Panel with options for custom control below it. The collection of "swatches" is populated with a user's saved vistas. Changing the vista on the Daynight Panel can be done by tapping on any of the swatches. To create a new vista, the user can tap on the "+" icon. This will provide the user with the option of choosing which kind of vista they'd like to create:

A Daylight vista allows the user to select any color mix in the range of daylight color temperatures. This is achieved by mixing our strong cold white and warm white LEDs.

Dynamic vistas bring scheduling to the Daynight Panel and allow the user to enjoy a self-rotating set of 6 pre-mixed vistas to mimic actual sunlight. This option replaces customizability for simplicity, and lets the user kick back, relax, and enjoy the sun.

Colorful vistas tap in to our RGB LEDs to bring customization back in to play. Sliders allow the user to make custom combinations of colors to paint the whole Daynight Panel for entertainment or enjoyment.

Finally, the Creative vista option is where customization runs wild; users can select custom colors for each LED to allow for an endless number of unique scenes to be created, right from the user's fingertips; we think this is the most entertaining and attractive feature of the panel.

The control is made up of several functions whose essential purpose is to process and send commands to the app or to the driver. First a command is received as an array of characters which are translated to 5 integers holding the RGBCW values. These values are written to the drivers LED registers in a special order depending on the scene that is being portrayed. Then the drivers are told how to turn on the LED's.

The steps above describe typical controller behavior, but additional features are included to improve interaction with the controller and the app. One luxury feature is the controller storing commands on the chip so the app can request that data.

The app and the controller software are designed to be versatile so that new features can easily be integrated. As app capability expands the controller software can easily follow.

VI. POTENTIAL ENHANCEMENTS

Due to the time constrained nature of this project there are several enhancements and features that are not currently available. If given more time and resources the changes described below would be further researched and potentially implemented in a future revision.

A. LED Module Printed Circuit Board

The least researched components in the development of the LED Module PCB are the various connectors that interface with the Power and Logic PCB. Initially for ease of setup on the bench, banana jacks were selected for the power connections. Unfortunately, this connector implementation is too bulky for a finished product, so the board was converted for use screw terminal blocks and bare wire. Although this power connector implementation is an improvement, it is not the ultimate solution. With another PCB revision, male Molex connectors would most likely serve as an ideal connector that is compact, professional, and easy to connect and disconnect. Similarly, the data connections between the two PCBs were implemented using header pins and jumper cables. Ideally, in a future revision the header pin connections would be replaced with a clampable ribbon cable.

Increased LED resolution will require some research because of the increased power that high LED density requires. The resolution is directly linked to the LED density. The more LEDs there are in an area of the panel, the brighter it will be and the more distinct a scene can become. The improved resolution will also help to enable other modes on the panel for future designers.

B. Power and Logic Printed Circuit Board

The Power and Logic PCB contains an error where the output capacitor for the LMZM33603 was omitted. This was remedied by hand soldering a modification to the PCB. In a future revision, this capacitor will be included properly in the board layout.

Additionally, a surface mount variation of the ATMega328P would be a more relevant choice rather than the through hole SPDIP packaging. To achieve this surface mount configuration hardware to program the microcontroller on the PCB would be necessary.

To further streamline costs and PCB real estate, the Bluetooth daughter board can be eliminated from the Daynight Panel and replaced with Bluetooth circuitry integrated into the board design.

An additional feature for the Daynight Panel would be a power status push button. This would employ a typical soft latch power switch circuit to enable and disable the 12 V power input. The status of the soft latch power switch could be displayed with a status LED.

Finally, another revision of the Power and Logic PCB could be to merge the Power and Logic PCB and the LED Module PCB into one single board design. This design could potentially have components on both sides of the board.

C. Microcontroller Software

The software is very barebones it could definitely use some fine tuning with features like transitions between scenes. The other features are related to allowing increased customization of features like transitions which would smooth out the changing of scenes. Startup scenes for when the panel has been switched off. Notifications about error detection. Sensor integration so the panel can show in real time the color temperature outside. Notification on the panel signaling information from other apps. Conditions for what data to hold when the panel is reset. There is a lot of potential for future features because the software is written to be versatile and responsive

D. Daynight app for iOS

Good software practice allows code to be modular and scalable. The development of the Daynight app for iOS was executed in stages, and there is always room for more. Given a timeline free of hard deadlines, small but meaningful changes could be introduced.

Dynamic vistas set up a schedule of six scene updates over the span of 24 hours. To make things easy for the user, the scenes are already hand-picked and pre-populated. Currently, the scenes are unable to be customized. Customizability could be introduced to Dynamic vistas to bring more power to the user. Though, we believe that the current state of customization will be exciting and fresh for everyone.

Currently, all user data is persisted on a per-device basis. This means there are no unnecessary login screens, no privacy risks, and no clunky “saving” UI. This also means that on a new or restored device, data won’t be persisted. Thankfully, device replacements like this don’t happen often. Adding the ability to sync to iCloud would be a welcome addition to the way data is persisted with the Daynight app and would prevent users from losing their apps data in long-term use cases.

VII. CONCLUSION

The Daynight Panel brings innovation to light-mimicking technology. We feel that our project stands apart from competitors because it has great control, powerful LEDs, easy-to-use software, and intuitive design in an affordable package. All of these components are aspects of quality products we see across all markets, and feel are important to emulate in the Daynight Panel.

Having intuitive and well-designed control is important because it is what will make the simulation seem realistic and intuitive.

There are many daylight mimicking lighting devices on the market but what we have designed is affordable for mass consumption. The star of the Daylight Panel is our choice in powerful LED. It is bright enough to light up a room, like the sun would, despite being behind a diffuser. The panel is sleek yet the cheap and powerful. We use surface mounted LEDs that are powerful and capable of providing any ambient condition in a room. The choice for price point considered safety, compatibility, and appearance. The resulting panel is as sleek as can be manufactured by hand and can easily become more modern and sleeker with professional aid.

It is very easy to produce a bright light, but these kinds of lights are often cold and unsettling. Since they can also be painful to the eye, the chosen LED's are able to mix light color to create the warmth that is missing in a lot of fluorescent lights popularly used today.

By bringing custom controls and good hardware and software design together, the Daynight Panel refreshes a stale and stagnant market for sunlight-mimicking light panels. This new use for LED's helps to fill the gap in residential and commercial LED applications and has served as the motivation and inspiration for this project.

VIII. References

[1] B. Ray, "Bluetooth Vs. Bluetooth Low Energy: What's The Difference?," Link Labs, November 2015. [Online]. Available: https://www.link-labs.com/blog/bluetooth-vs-bluetooth-low-energy. [Accessed October 2018].

[2] Bluetooth SIG, "Radio Versions," [Online]. Available: https://www.bluetooth.com/bluetooth-technology/radio-versions. [Accessed October 2018].

[3] A. Singh, "What is Mac OS X?," OS X Book, 2014. [Online]. Available: http://osxbook.com/book/bonus/ancient/whatismacosx/history.html. [Accessed October 2018].

[4] Apple Inc., "Welcome to Swift," Apple Inc., [Online]. Available: https://swift.org. [Accessed October 2018].

[5] Texas Instruments, "TLC59116 16-Channel FM+ I2C-Bus Constant-Current LED Sink Driver," February 2008. [Online]. Available: http://www.ti.com/lit/ds/symlink/tlc59116.pdf. [Accessed 31 October 2018].

[6] Phillips, "Meet Hue," 11 November 2018. [Online]. Available: https://www2.meethue.com/en-us.