ADAPTIVE NIGHT LIGHT

Preliminary Research Paper

October 22, 2020

GROUP JAMLER

Authors

Trevor Byler | tdbyler0@frostburg.edu

James Ritchie | jpritchie0@frostburg.edu

COSC 365-001

Professor

Dr. Joy Zheng | Frostburg State University

TABLE OF CONTENTS

I. INTRODUCTION

The Raspberry Pi is a portable computer that is inexpensive, compact, and customizable. It can function as a normal computer. It can be loaded with a full operating system with a GUI and downloadable programs. It is compatible with standard peripherals, such as keyboard, mouse, and computer monitors. The new model of the Raspberry Pi: Raspberry Pi 4, has up to 8 GB of SDRAM, a quad-core processor, ethernet, Wi-Fi, and Bluetooth functionality, and a variety of other ports. Raspberry Pi can be hooked up to a plethora of sensors as well. Sensors can read basic user input, environmental properties, incoming wireless signals, and biometrics. These sensors, the Python programming support built into Raspberry, its low cost, and small form-factor make it a great tool for developing and testing sensor networks and embedded systems [1].

We will be making a small project containing one Raspberry Pi (3B+), one sensor (Adafruit TSL2591 High Dynamic Range Digital Light Sensor), and an output device (LED Charlieplexed Matrix - 9x16 LEDs in conjunction with a IS31FL3731 16x9 Charlieplexed PWM LED Driver). The goal is to use a sensor that accurately measures the level of light in the environment, and then turns its own light on when the ambient light dips below a certain threshold.

II. LITERATURE REVIEW

A. General Sensor Network

Much research has been done in the application of Raspberry Pi's, especially when connected in a network of devices. Researchers at the University of North Texas created a sensor network built with Raspberry Pi and Arduino. They designed and built a wireless network consisting of 3 sensor nodes, 3 router nodes, and a base station. Nodes communicated with an XBee radio receiver. The Raspberry Pi and Arduino acted as microcontrollers that managed the temperature and humidity sensor connected to the node. The engineers had to write applications for the sensor nodes, a database using MySQL, and a web server application for the base station [2].

B. Double Light Sensor

In researching other projects utilizing the same light sensor we are, we found three. The first study being from Openly Published Environmental Sensing (OPEnS) Lab at Oregon State University just briefly touched on the limitations they ran into with using the sensor we chose (TSL2591). Their team was doing environmental data gathering and for the previous design the TSL2591 was sufficient. However, when they created a new design, they wanted to add another light sensor pointing downward to capture albedo from the ground. Because this required two separate light sensors, they had to opt for a different model due to the addressing constraints of the TSL2591. The TSL2591 only has one address which made it impossible wiring two sensors with two different inputs to a single board for their project. [3] As our project does not require two sensors, we should not run into any addressing issues like this group did.

C. Light Sensor in Agriculture

The second research paper we found was from the Department of Electronic Engineering, University of Nigeria, Nsukka. They were performing and experiment on automation of data collection related to crops drying and needed to use the TSL2591 to log the data regarding light. They chose the TSL2591 because it consists of two photodiodes, one optimized to sense radiation in the visible region and the other radiation in the infrared region. They were able to get their sensor to communicate with their microcontroller using the I2C protocol which we touch on in our communication section. And like we hope to, they were able to accomplish their measuring with libraries provided by the manufacturers. They also compared their data to that of their universities weather station (graph below). [4]

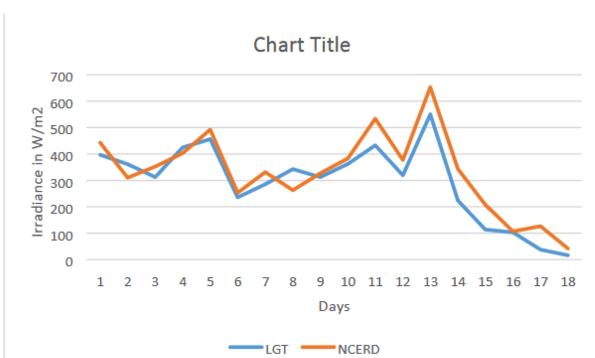
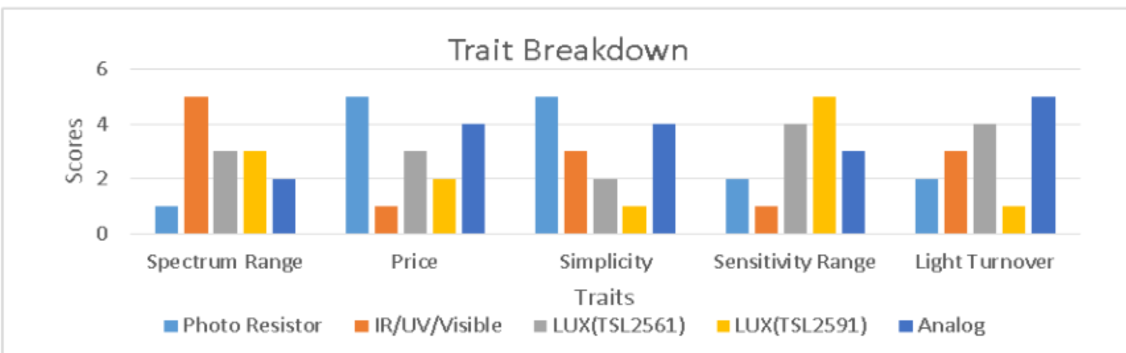


Figure 8: Graph of solarimeter reading and weather station reading [4]

D. Comparison of Light Sensors

Lastly, we read a report from Winona State University in Winona, MN comparing different light sensors to see how they compared to one another. Their group examined 5 different sensors: The Photo resistor, UV/IR/Visible Light (SI1145), High Dynamic Range LUX(TSL2591), Digital Luminosity LUX(TSL2561), and Analog Light (GA1A1S202WP). Their metrics used for determining the quality of each sensor was: spectrum range, cheapness, sensitivity range, light turnover, command read accuracy, and simplicity of sensor implementation. Below is a trait break down graph of their teams finding based on those criteria:

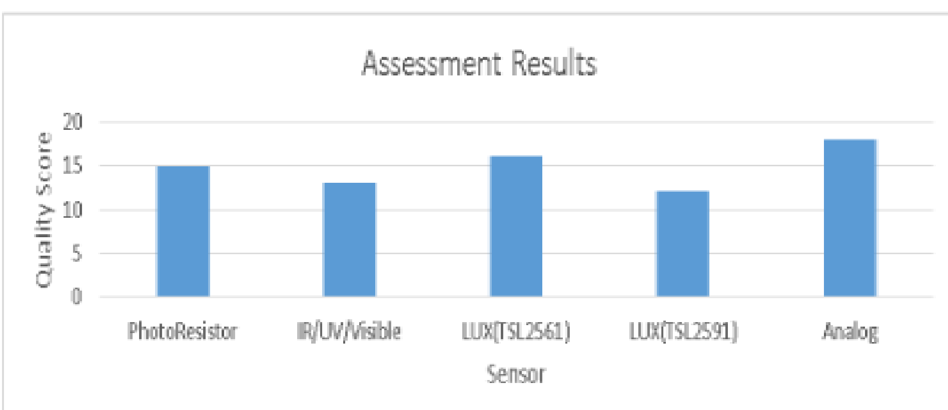


[5]

According to their testing, our sensors only advantage was the sensitivity range that it has.

"The LUX(TSL2591) sensor has the highest sensitivity of the sensors, recorded values easily outpaced the other sensors, and the stated LUX value is the highest." [5]

The group also had another more generalized graph, and way to score the quality of each sensor below:



[5]

Raspberry Pi’s are capable of numerous other applications—from smart home automation to monitoring systems and beyond. We will be using a Raspberry Pi for a very basic project to familiarize ourselves with the Raspberry Pi platform and workflow.

III. MEMORY

The Raspberry Pi has a chip on the board which is the Pi’s random access memory which is where data is held that you are currently working on while the Pi is connected to power [6]. This is volatile memory due to it requiring power, and everything will be lost if power is lost. Because of this a microSD card is required for you to safely write your data to so it can be saved even when power is lost. The RAM and microSD make up the Raspberry Pi’s volatile and non-volatile memories [6]. The Raspberry Pi 3 B+ we will be working with specifically has 1GB of LPDDR2 SDRAM [7]. According to sources we found, the recommend SD card size is ranging anywhere between 4GB-16GB. For example, “for installation of Raspberry Pi OS with desktop and recommended software (Full) via NOOBS the minimum card size is 16GB. For the image installation of Raspberry Pi OS with desktop and recommended software, the minimum card size is 8GB” [7]. We chose to go with the 16GB SD card just to be on the safe side and to ensure we have enough available memory.

V. OPERATING SYSTEM

A. Raspbian

Raspberry Pi has support for many Linux-based operating systems. Each flavor of Linux has different strengths and applications. The standard OS officially recommended by Raspberry Pi is Raspberry Pi OS, or Raspbian for short. Raspbian is based on Debian Linux, hence its name. It offers a standard OS experience with a GUI and development tools. It is open source and widely supported among users of Raspberry Pi [7].

B. Ubuntu

Ubuntu can also be installed on a Raspberry Pi. Ubuntu is one of the most popular Linux distributions. It has many powerful features and lots of support from the community. Ubuntu is the OS of choice for servers. Its features could be used to make a Raspberry Pi some sort of server, such as an FTP server [8]. However, support for Ubuntu on Raspberry Pi specifically is not as strong as support for Raspbian.

C. Kali & Others

Kali is a specialized Linux distribution. Kali offers many powerful tools for penetration testing and hacking. Loading a Raspberry Pi with Kali enables it to be a network sniffer or a vulnerability scanner [9]. Kali is great at what it does, but it only serves a very specific purpose. There are other specialized Linux flavors for Raspberry Pi as well. Some are for media purposes, allowing a Pi to be used as an embedded media player or media server. Other distributions allow retro games to be loaded onto a Raspberry Pi and played with a controller.

We will be using Raspbian, since it is the recommended and most widely supported. It will be easy for beginners to install and use. It has Python support so it will be easy to configure our sensor and light matrix. We don’t need any of the specialized features of other Linux distributions, so Raspbian will fit our needs just fine.

D. Installation

Raspberry Pi makes installing Raspbian simple. A small program called Imager is available on the Raspberry Pi website. It operates similar to the creation of a Windows image on a USB drive, except it requires a formatted SD card. Download and run the Imager. Next, select “Choose OS” from the main menu. Raspbian should be the first option. Next, select “Choose SD Card” and select the formatted SD card from the list. Finally, select “Write.” Once the process finishes, the SD card should be ready to insert into a Raspberry Pi. The SD card is the default boot option, so the OS will run once the Raspberry Pi is turned on [7].

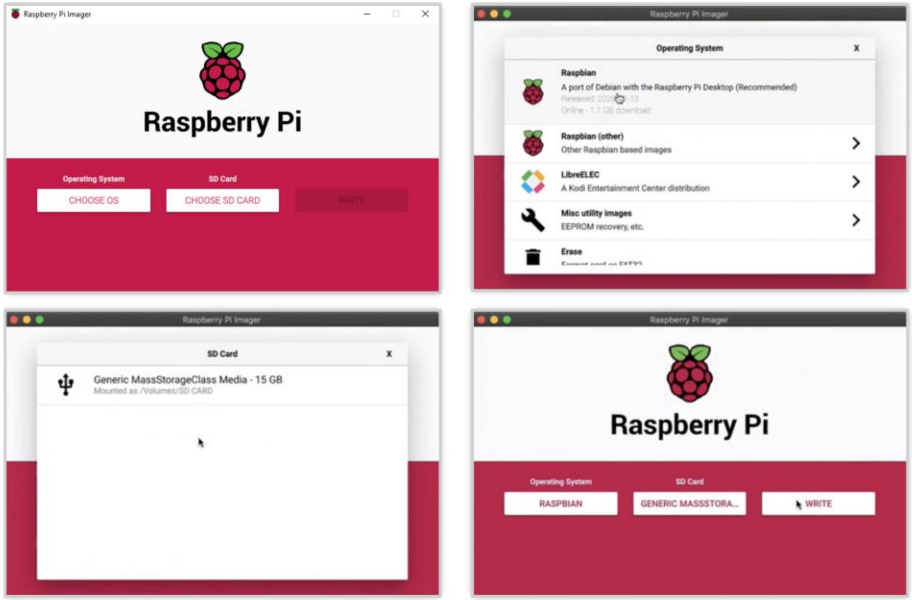


Image Source: https://www.raspberrypi.org/blog/raspberry-pi-imager-imaging-utility/

IV. Communication

In terms of network connectivity, the Raspberry Pi 3 B+ has a few options for communicating on a network or with peripheral devices. Wirelessly the Raspberry Pi offers duel band 2.4 GHz and 5 GHz IEEE 802.11.b/g/n/ac for connection to a local area network. The 3 B+ model also has Bluetooth 4.2/BLE capabilities that can you be used for connecting peripherals to the system [7]. As for wired connections this model of Raspberry Pi includes gigabit Ethernet, though it has a throughput of 300Mbps due to its USB 2.0 connection, of which is has four, to the SoC [7].

The sensor we will be using is the Adafruit TSL2591 High Dynamic Range Digital Light Sensor. In addition, we will also be using output we receive from this sensor to then control a LED Charlieplexed Matrix - 9x16 LED display with the help of a IS31FL3731 16x9 Charlieplexed PWM LED Driver, both also offered by Adafruit. Both the light sensor and the driver for the LED matrix communicate using I2C (I-squared-c) and are programmable via Python with the use of additional libraries [10]. For the light sensor we need to use *CircuitPython* along with the *Adafruit CircuitPython TSL2591 module* [10]. As for the LED driver we will need to install the *Adafruit CircuitPython IS31FL3731* library on to our board [10]. After necessary setup is complete all the libraries needed provide comprehensive ways to attain the data from our sensor in a way that we can then manipulate our LED grid.

VI. CONCLUSION

Raspberry Pi’s are very adaptable devices that are revolutionizing the tech industry. More compact and scalable networks of microcomputers are being prototyped and tested due to these devices. The low cost, range of hardware tiers, sensor capability, and software support make Raspberry Pi a versatile tool. Our night-light project is designed to be an entry-level project for someone who is not familiar working with a Raspberry Pi.

REFERENCES

[1] C. W. Zhao, J. Jegatheesan and S. C. Loon, "Exploring IOT Application Using Raspberry Pi," *International Journal of Computer Networks and Applications*, vol. 2, no. 1, pp. 27-34, 2015.

[2] S. Ferdoush and X. Li, "Wireless Sensor Network System Design Using Raspberry Pi and Arduino for Environmental Monitoring Applications," *Procedia Computer Science*, vol. 34, pp. 103-110, 2014.

[3] M. N. D. C. U. D. J. S. Thomes DeBell, "Evaporometer Upgrades: Improved Weatherproofing and Accuracy in Teal-Time Enviromental Data Gathering," Corvallis, 2018.

[4] E. S. U. S. I. F. Edward Anoliefo, "Automation Of Procedures For Experiments On Hybrid Crop Drying," Nsukka, 2019.

[5] J. Carpenter, "Light Sensor Performance Comparisons," Winona, 2015.

[6] G. Halfacree, in *The Official Raspberry Pi Beginner's Guide : How to use your new computer*, Cambridge, Raspberry Pi Trading Ltd, 2018, pp. 9-15.

[7] "Raspberry Pi," [Online]. Available: https://www.raspberrypi.org. [Accessed 18 10 2020].

[8] "Ubuntu," [Online]. Available: https://ubuntu.com/download/raspberry-pi. [Accessed 18 10 2020].

[9] "Kali," [Online]. Available: https://www.kali.org/docs/arm/kali-linux-raspberry-pi/. [Accessed 18 10 2020].

[10] "Adafruit," [Online]. Available: https://www.adafruit.com/. [Accessed 17 10 2020].