Milestone 1: Stranger Things Light Wall

Tim Hollabaugh & Tim Gayed
Rowan University
October 21, 2018

1 Design Overview

This goal of the project was to create a byte addressable RGB LED that could receive packet of information and pass the packet onto the next LED in the chain. Using an MSP430G2553, the task was decipher these packets of hex code and to turn the RGB LED on of different values for each color using UART (universal asynchronous receiver-transmitter) and Pulse-Width modulation (PWM) on each of the LEDs. After the controller generated the correct color for the LED based on its position in the chain, it would transmit the packet to the next controller in the chain.

1.1 Design Features

- Custom Milled PCB featuring tinned copper plating for increased durability
- Customizable Brightness Adjustment Settings
- Seamless integration with other addressable LEDs in the chain
- 16,777,215 Possible color combinations

1.2 Design Resources

Github repository link: https://github.com/RU09342-F18/milestone-1-tim
2N7000 Datasheet: https://www.onsemi.com/pub/Collateral/2N7000-D.PDF

1.3 Board Image

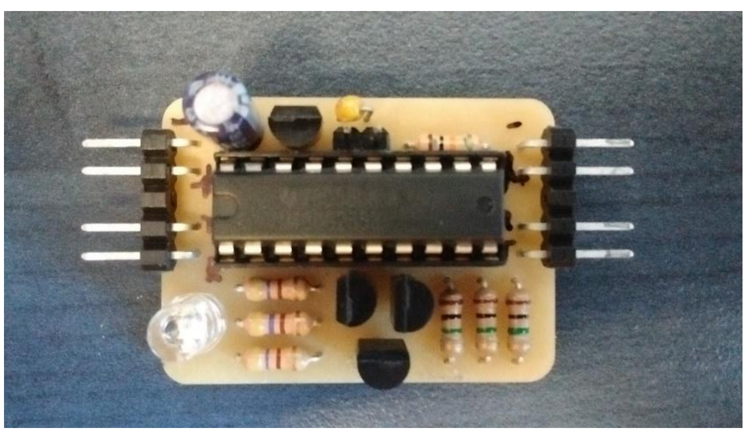

Figure 1 Addressable RGB LED Device

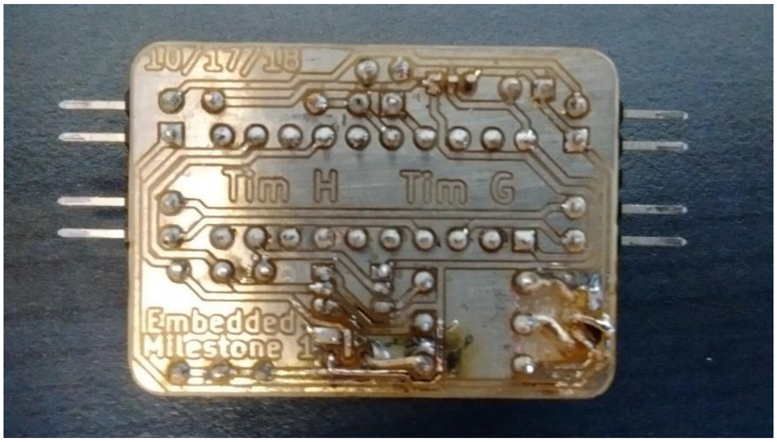

Figure 2 Bottom View of Addressable RGB LED Device

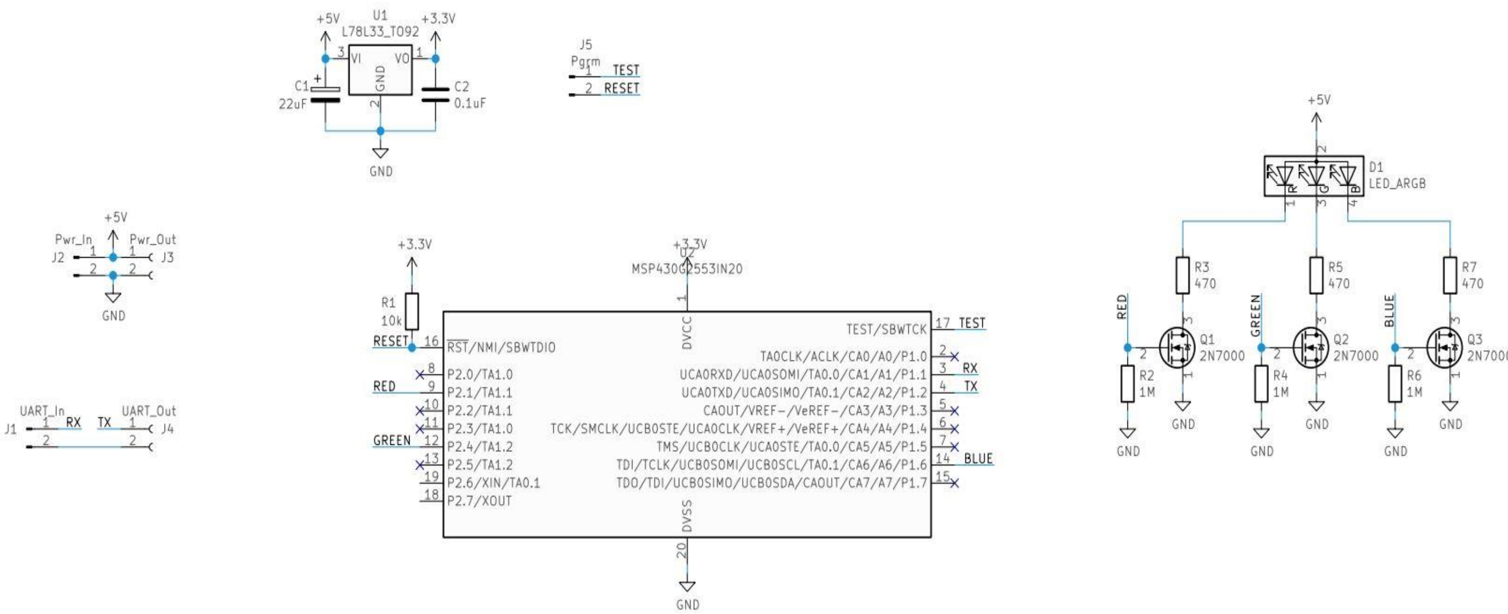

Figure 3 Design Schematic

Figures 1 and 2 show the final design of the Addressable RGB LED Device. The design was fabricated into a PCB using the PCB mill and treated with a tinning solution to improve trace integrity and device durability.

Figure 3 shows a schematic of the how the individual LEDs were connected in the RGB LED. The NMOS functions as a low side switch which amplifies the voltage going into the LED. The switch turns on when the micro-controller pin is in the on state. This switch will be used to pulse the LED.

System Specifications

Parameter	Specifications	Notes
SMCLK	1 MHz	Frequency Speed of the timer used for PWM
Baud Rate	9600	Modulation Rate in symbols per second
TA0CCR0	P1.6	Determines the duty cycle for Blue
TA0CCR1	P2.1	Determines the duty cycle for Red
TA0CCR2	P2.4	Determines the duty cycle for Green

LED Color	Forward Operating Voltage	Description
Red	2.2V	Minimum voltage for LED
Green	2.4V	Minimum voltage for LED
Blue	2.5V	Minimum voltage for LED

2 System Description

The objective of this system was to be able to send and receive packets using UART, and then decipher the data into an addressable RGB LED using PWM. The MSP430G2553 is a low-power, low-cost microcontroller and is the heart of this system. Each of the pins of the RGB LED were assigned to a pin on the MSP430G2553. Using a 2N7000 NPN MOSFET on each of the LEDs allowed for the LEDs to shine brighter. After processing the packets over the RX Pin, the MSP430G25533 then addressed the color values and brightness of the RGB LED.

2.1 Featured Devices

- **MSP430G2553**
  - Processes the designated package, designates the data to the corresponding pins and determines the color and brightness of the LED
  - This TI micro-controller used in the system processes the communication program and the inputted packet. Using this component’s two timers and three capture compare registers for each timer the desired effect was possible. The clocks used for each of the two timers was the Sub Main clock which is a 1 MHz clock.
- **RGB LED**
  - The LED is the output of the instructions processed by the MSP430G2553. The LEDs share a common ground in a common cathode configuration. Each of the input pins are connected to pins on the MSP430. The Red LED was assigned to pin 2.1, Green was

assigned to pin 2.4, and Blue was assigned to pin 1.6 on the MSP430.

2.2 Bill of Materials

Quantity	Device
3	470 ohm resistor
3	1 M ohm resistors
3	2N7000 MOSFETs
1	MSP430G2553
1	Copper PCB Mill Sheet
1	RGB LED
1	Breadboard
1	22uF Capacitor
1	.1uf Capacitor
1	L78L33 Voltage Regulator

3 SYSTEM DESIGN THEORY

3.1 Requirement 1

One of the design requirements is to pulse the red, green, and blue part of the RGB LED individually with the MSP430G2553. This requirement was achieved using SMCLK and three capture compare registers. Each capture compare register will determine the duty cycle of a specific pin. In this case, CCR1 corresponds to P2.1, CCR2 corresponds to P2.4, and CCR0 corresponds to P1.6. The timer is set using SMCLK in UPMode and output mode in reset/set. When the counter reaches CCR1, the output on P2.1 will be reset or 0. The same applies to the other CCRs. When the counter reaches CCR0, the output on the pins will be set to 1 and the counter is reset. By changing the values stored in CCR0-2, the user can control the PWM of the different RGB LEDs, effectively controlling their brightness.

3.2 Requirement 2

The addition of the N-MOSFETs for the RGB LED were optional improvements to the design requirements. The low side switches configuration provides the RGB LED with more voltage and current than can be supplied with the microcontroller. Realistically, the microcontroller would need more voltage to control LED strips with more than one RGB LED. If this were the case, the microcontroller could not output enough voltage and current to drive all the LEDs in the strip without additional hardware amplifying the signal.

4 Using the Hardware

After uploading the Addressable LED program to the device, verify the hardware has been wired properly. Once the correct pins have been connected to the MOS devices, connect the MSP 430 to your computer through USB. Run the software Realterm or Cutecom and set up the port and baud rate properly to 9600. After setup is complete, bytes can be sent through the software to the MSP430 using UART communication. The first byte the MSP will receive is the length bit, then the next three bytes will determine the red, green, and blue lights respectively. In order to link the Addressable RGB LED device to other boards, the Rx pins of one board need to be connected to the Tx pin of the next board in the chain. This will transfer packets of data received by the first board, , then sent to the next board in the line after the contents of the packet has been modified.

4.1 **Device Chain Hierarchy**

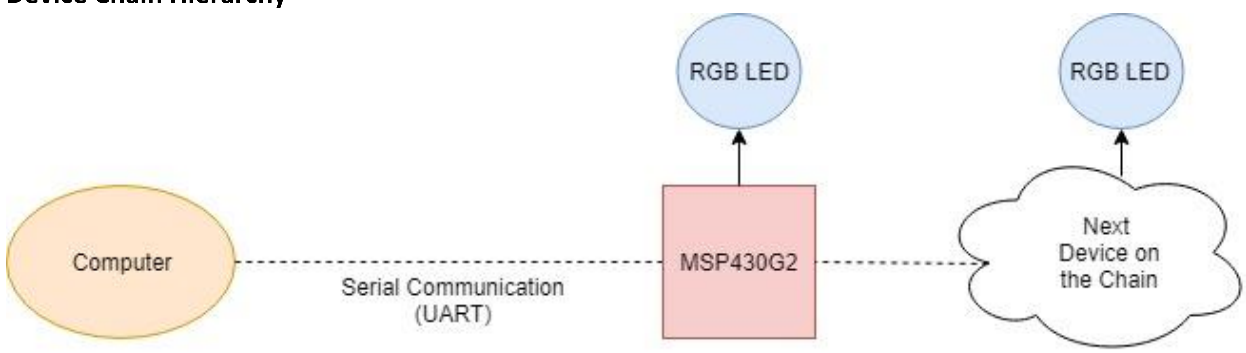

Figure 4 Device Chain Hierarchy

4.2

Figure 4 shows the connections of MSP430 boards when they are daisy chained. The Addressable RGB LED device has been shown to work with any MSP430 Device running compatible software. The connections enter in one board's Rx pin and exit through the boards Tx pin. This allows the board to continually transmit packages of data through the chain. The computer sends the initial package of data.

4.3 Communicating With the Device

The MSP430G2553 can communicate with other devices using the RX and TX pins. By default, these pins are jumped so that the user can communicate with the microprocessor through USB. The user can use any serial communication program like RealTerm or Cutecom to send bytes to the microprocessor. Ensure the BAUD rate and the ports match up in set up to ensure the two can communicate with one another. If the user wants to link multiple processors together, they would need to remove the jumper from the RX and TX pin. The user would then need to connect their RX to the TX pin of another board and the TX pin of their board to the RX pin of another board. Ensure all the microprocessors share a common ground.

5 Test Setup

In theory an infinite number of devices can be linked together, however only up to 4 devices have been tested together. The sequence of bytes of variable length is sent from the computer and is passed on through the chain. Figure 6 below shows the prototyping setup that was used before replicating the design on a PCB mill.

6 Appendix

6.1 Milled Board Design

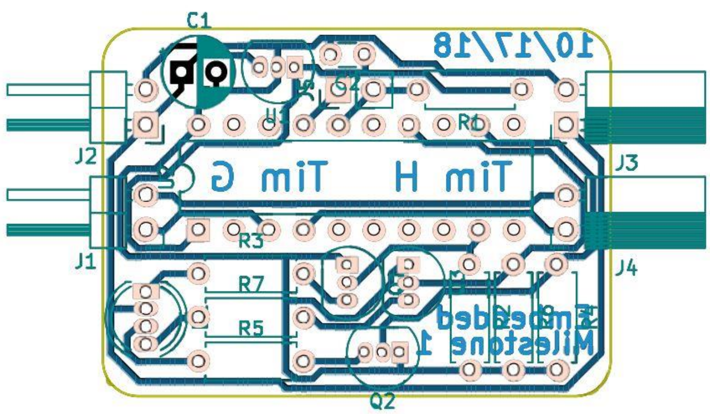

Figure 5 Mirrored View of PCB Design

6.2 Rapid Prototyping Setup

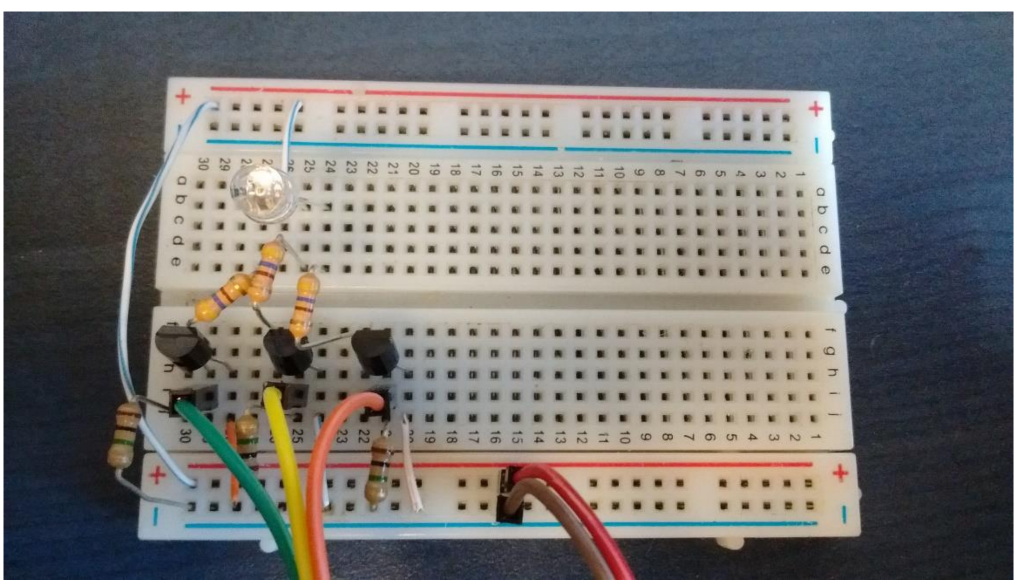

Figure 6 Rapid Prototyping Setup

6.3 Serial Communication

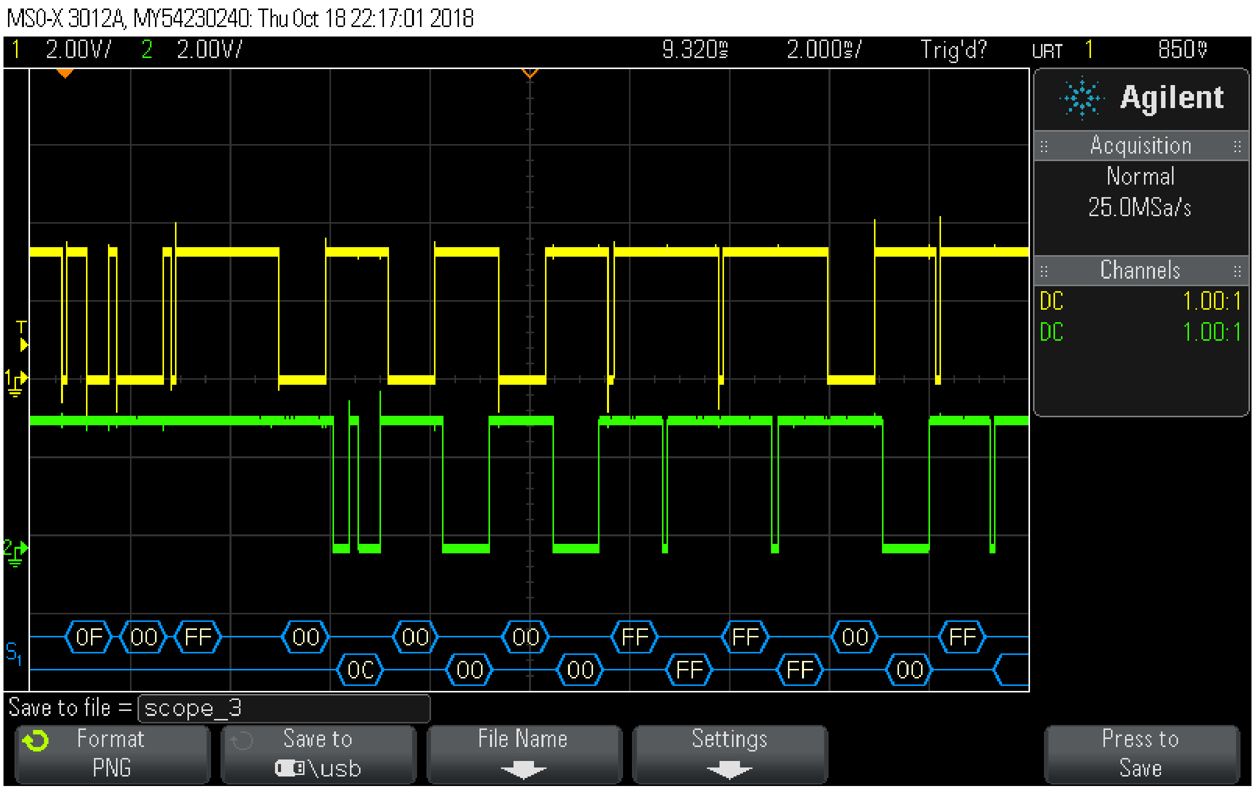

Figure 7 An Oscilloscope Measurement of Serial Data Being Transmitted and Received over UART

7 References

Family User Guide

http://www.ti.com/lit/ug/slau144j/slau144j.pdf

MSP430G2553 Datasheet

http://www.ti.com/lit/ds/symlink/msp430g2553.pdf

RGB LED Datasheet

https://www.sparkfun.com/datasheets/Components/YSL-R596CR3G4B5C-C10. pdf

TI Resource Explorer UART 9600 Echo Example Code

http://dev.ti.com/tirex/#/Device/MSP430G2553/?link=Software%2FMSP430Ware%2FDevices%2FMSP430G2553%2FPeripheral%20Examples%2FRegister%20Level%2Fmsp430g2xx3_uscia0_uart_01_9600.c

Example UART Configuaration Code

https://www.embeddedrelated.com/showarticle/420.php