Milestone 1- Stranger Things

Nicholas Klein
Rowan University

October 22, 2018

1 Design Overview

To pull together and apply information learned in the first half of an Embedded Systems Course, a nodal LED color processor is created. When added in a long chain, many LEDs can be flashed in different colors and intensity to a similar affect as the string of Christmas lights in Netflix's Stranger Things. The topics used include timers, interrupts, pulse width modulation. Using a serial UART connection, a packet is sent to the LED node chain which specifies the intensity of the RGB components in the LEDs. While the experiment itself is only to make a single node, as many nodes as desired can be connected to make the chain each with its own colors and intensities.

1.1 Design Features

- Modular nodes to form LED chains as long as the user likes.
- LED color can be customized to any color the user desires, other than black.

1.2 Featured Applications

- Customizable chain of RGB LEDs.

1.3 Design Resources

All code, resources, and instructions are hosted here:
`https://github.com/RU09342-F18/milestone-1-the-artist-formerly-known-as-a-team/tree/master/Milestone_StrangerThings`

1.4 Block Diagram

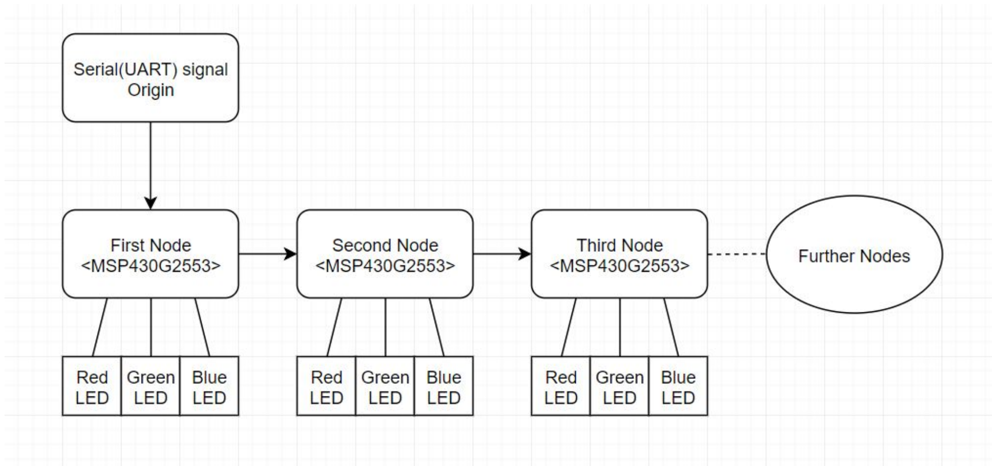

Figure 1: Block Diagram

1.5 Board Image

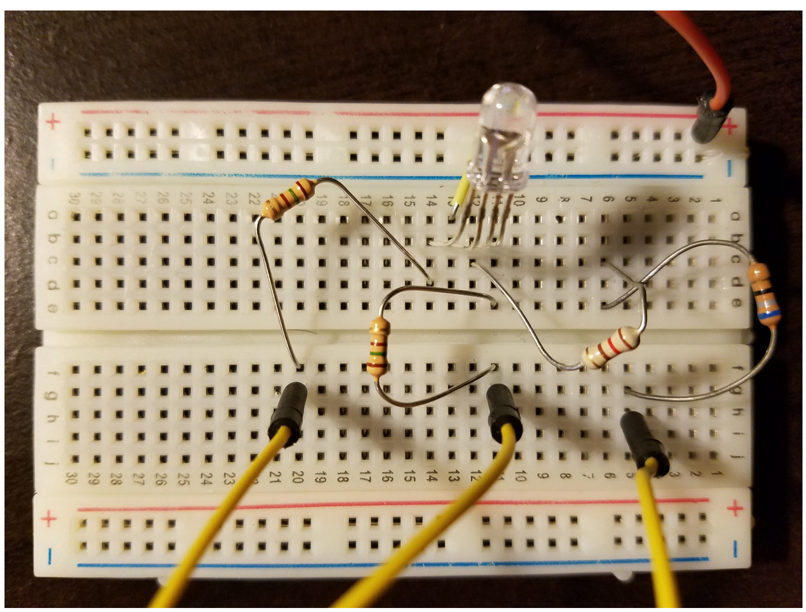

Figure 2: Breadboard

2 Key System Specifications

PARAMETER	SPECIFICATIONS	DETAILS
Red LED Voltage	1.8V	Voltage drop across the LED.
Green LED Voltage	2.3V	Voltage drop across the LED.
Blue LED Voltage	2.5V	Voltage drop across the LED.
VCC	3.3V	Supplied voltage to the common anode of the RGB LED.
Packet Size	2+(Nodes*3) Bytes	One byte begins the packet giving the total length of the packet in bytes. Three bytes follow giving PWM info for each LED. Three more bytes per extra node give PWM info to each other node.

3 System Description

The Stranger Things Milestone primarily deals with using skills learned in class and using them in an application. To do this the microcontroller is programmed to use timers for capture and compare for the pulse width modulation of the LEDs and interrupts are used when a serial byte is sent to the node. It also makes use of basic electronics skills making the circuitry for the LED. Lastly UART communications to and from a microcontroller are explored by recieving and transmitting the packet.

3.1 Detailed Block Diagram

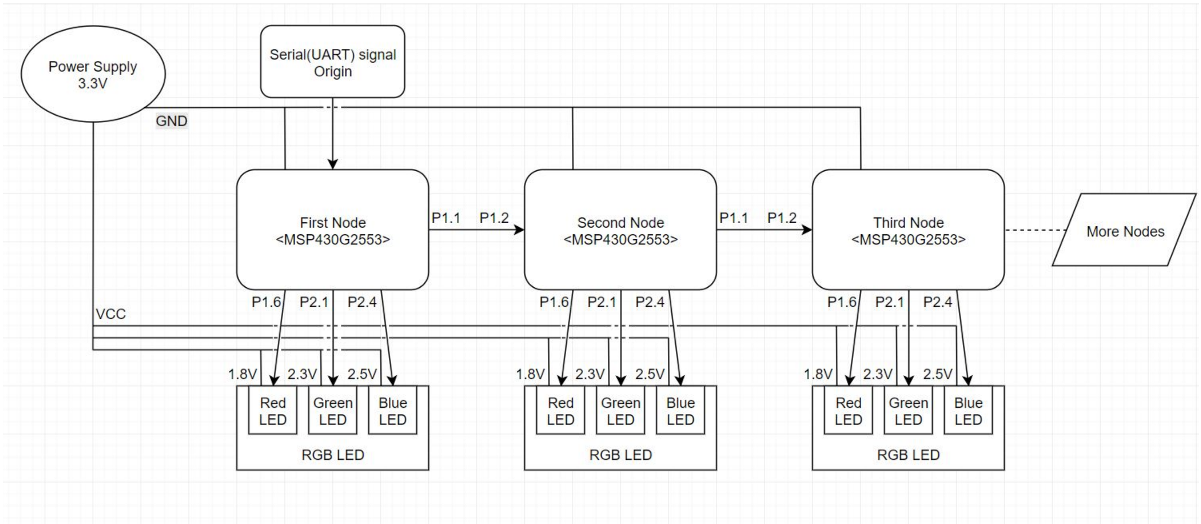

Figure 3: Detailed Block Diagram

3.2 Highlighted Devices

- MSP430G2553 - This Microcontroller serves as the processing unit for a node. It takes in the UART packet, controls the PWM signals to the LED, and sends a new packet to the next node.
- RGB LED - The LED is connected to a circuit that allows the PWM signals to light the red, green, and blue components to certain intensities which together can create a light of any color other than black.

3.3 MSP430G2553

The MSP430G2553 is a microprocessor from TI that serves as the processing unit of a node. The processor receives an information packet which contains the duty cycles for each of the color components for the RGB LED. as the bytes pass, the microprocessor determines how long the packet length is in bytes and transmits the length of the current packet minus three bytes to the next node in the chain. The second byte contains a number *x* which creates a duty cycle of $x/255\%$ for the RED LED which is sent out through Pin 1.6. For this setup the on signal is the low voltage as the pin serves as the ground to the 3.3V power supply. The third and fourth byte function just as the seconds byte does except that they control the Green LED with Pin 2.1 and the Blue LED with Pin 2.4 respectively. Each byte following the fourth is transmitted to the next node in the chain without further processing from this node.

3.4 RGB LED Circuit

The circuitry for the RBG led is represented in Figure 4. The RBG led itself is common anode so it is tied to VCC and each color pin is then attached to resistors which are connected to the control pin of their respective color on the microprocessor. When the pin is low (GND), current flows through the LED. The current sent through each colored LED is 12mA. The Voltage Drop across each is 1.8V across Red, 2.3V across Green, and 2.5V across Blue. When the pin is high (3.3V) there is no voltage difference and no current flows.

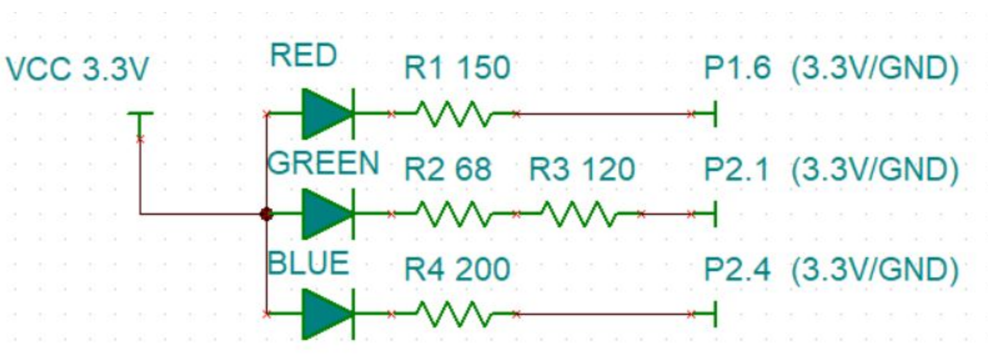

Figure 4: Circuit Diagram

4 SYSTEM DESIGN THEORY

This system is designed to control the intensity and color of an LED using Pulse Width Modulation controlled by an MSP430G2553 microcontroller. The instructions that dictate these parameters are sent via a UART (Universal Asynchronous Receiver/Transmitter) serial connection

4.1 Pulse Width Modulation

The system's pwm is controlled by assigning the received byte dictating that LED's intensity to the TAxCCRx register corresponding to that LED. As the internal 1Mhz clock starts counting at zero, the corresponding Pin that grounds the LED goes low, causing a voltage difference which sends current through the LED. The clock counts to the assigned number in TAxCCRx then shuts off the LED by raising the Pin's voltage to 3.3V. The clock continues to count up to 255 where it rolls over to zero again, lighting the LED. This cycle allows a percentage of the time to be when the LED it lit and the rest it is off. The up time is referred to as the duty cycle.

4.2 UART Serial Connection

UART is a microchip that controls the receiving and transmission of serial data to and from the microcontroller. It translates the bytes a computer sends into bits the processor can understand and upon transmission converts them back into the bytes a computer normally processes. It also processes parity bits which allow the microprocessor to be sure that information is sent and received without error. The UART chip also controls the interrupt used to run our process when a serial byte is received.

5 Getting Started/How to use the device

First, be sure that the RX and TX jumpers on your MSP430G2553 microcontroller are set to the horizontal position, otherwise receive and transmit will not work. Using Code Composer Studio, Eclipse, or any IDE that can flash the MSP430G2553 build the code in the github listed in the design resources section and flash it to the board. Take the LED Breadboard as shown in Figure 2 and connect the VCC jumper to the 3.3V female pin on the back of the board. Take the Red ground jumper (connected to row 20 in the photo) and connect it to the female pin of 1.6 on the back of the board. Do the same with the Green ground jumper (Row 5) to pin 2.1 and the Blue ground jumper (Row 11) to pin 2.4. If this is the first node, connect it to the computer that will send the serial connection. If not, connect Pin 1.1 to the transmit pin of the prior node. If there are nodes after this one, connect a jumper from pin 1.2 to their receive pin.

6 Getting Started Software/Firmware

Using realterm or a prefered serial data terminal program on the computer you intend to send the original packet, set the baud rate to 9600 and select the com port that is connected to the first node then build a packet of bytes to send to the node chain. The first bit should simply be the number of bytes of the entire packet. Each following byte should be a number 0 through 255 describing the intensity desired of the LED it will be sent to. When finished, send it and watch the light show.

6.1 Communicating with the Device

Communications to and from the microprocessors are sent via UART. The origional packet will be sent from a computer in the manner described above. In the case of the first node, it will receive a packet from the computer sending the original packet and will send the new packet to the next node via the transmit pin, Pin 1.2. Following nodes will connect the previous node's transmit pin to their receive pin, Pin 1.1, and will connect following pins in the same way as the first.

7 Test Setup

Set up two nodes as described above. Packets described here are written as byte values separated by spaces. Test the single node capability by sending a packet of [4 255 0 0] to set the first node's red LED to full. Then send [4 0 255 0] to light up the green LED to full and shut off red, then [4 0 0 255] to set the blue LED to full and shut off green. Next test the LEDs together. Send [4 255 255 255] to set all LEDs to full, showing white. Test the pwm intensity by sending [4 127 127 127] so half white will display from the RGB LED. Last test all the nodes by sending two nodes worth of bytes. Send [7 0 255 255 255 0 0] so the first node will have green and blue lit and the second node will have red lit.

7.1 Test Data

Packets to send in order:

- 4 255 0 0
- 4 0 255 0
- 4 0 0 255
- 4 255 255 255
- 4 127 127 127
- 7 0 255 255 255 0 0

8 Appendix

8.1 Bill of Materials

- 1 TI-MSP430G2553
- 1 USB A to micro USB B cable for UART
- 1 bread board
- 1 RGB LED (common anode)
- 1 200Ω resistor
- 1 150Ω resistor
- 1 120Ω resistor
- 1 68Ω resistor
- 5 jumper wires per node

8.2 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 USCI_A0, 9600 UART Echo ISR, DCO SMCLK Example Code
  http://dev.ti.com/tirex//Device/MSP430G2553/?link=Software%2FMSP430Ware%2FDevices%2F