Milestone 1: Communicating with Will Byers

Nicholas Kluzynski
and Nicholas Ambrose
Rowan University

October 17, 2017

1 Overview

It all started with a game of Dungeons & Dragons and the disappearance of a young boy named Will Byers. After countless efforts to solve the mystery of his disappearance, the boy's mother, Joyce Byers, discovers the ability to speak to her son through lights much like a Ouija board. This concept can be recreated using a series of MSP430 microprocessors. By connecting a microprocessor to an LED, then connecting multiple microprocessors together in a daisy-chain, the Ouija Board from the "Stranger Things" can be recreated.

2 Specifications

2.1 Board selection

Each microprocessor has its own special functions and hardware, however only one, the MSP430FR5994, is ideal for this task. While there are other boards that can accomplish the same task, the FR5994 performs the best and is the simplest. While it is not the most cost efficient of options, it is the one that can most effectively accomplish the task. The more inexpensive boards, notably the G2553 and the F5529, are lack the ability to complete task. The G2553 simply does not have enough performance to executes the program, and depending on the UART source, the F5529 requires different ports and specifications. After those two were eliminated from the pool, the price was comparable with the FR2311. However, after looking at the FR2311's data-sheet,it was determined that a lot more labor was needed for the board to reach the expectations of the project, which makes the FR5994 suitable for this task.

2.2 Limitations

Like any device, there are many limitations to performing this on an MSP430 board. The first limitation is the price. Each board costs an average of $12, and each board only controls 1 LED on the Ouija board. The LED has negligible price compared to the board, so it essentially costs $12 for each LED on the Ouija board. For a standard Roman alphabet, this brings the cost of the Ouija board to $312. This is expensive for a Ouija Board, however the cost can be lowered, since the FR5994 has enough timer modules for 2 LEDs, cutting the price in half. Even then, $156 for a Ouija board is too expensive.

There are many other limitations, but not nearly as dominant as the price. Some of these include: the voltage supplied by the processor, storage space for the packet, and inconsistency between the LEDs.

3 Individual Node Design

3.1 LEDs

Each node consists of one MSP430 microprocessor, one RGB LED, and a 1kΩ resistor. As seen in Figure: 1 the RGB LED consists of one red, one green, and one blue LED. Each LED is assigned its own pin on the MSP430 that controls the LED's brightness.

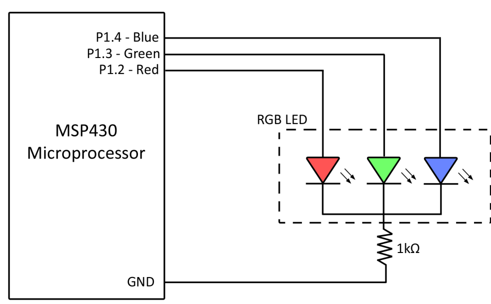

Figure 1: Ouija Board Node Schematic

In order to control change the overall color of the LED, the brightness of the each Red, Green, Blue LEDs need to be altered. The brightness is changed by using the MSP430 Capture and Compare Registers to Set and Reset the state of the LEDs: an example can be seen in Figure: 2. The longer that the LED is on, the more prominent the color will be in the overall output.

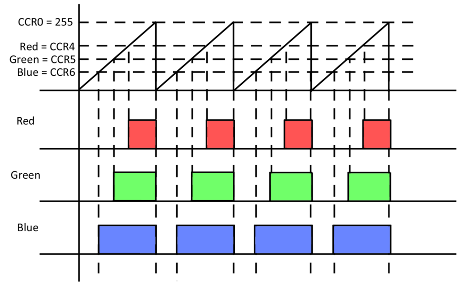

Figure 2: Pulse Width Modulation RGB LED Diagram

3.2 UART

In order to control the node from an outside device, a UART connection sends data to the MSP430, which gets processed and sent to the next board. The UART connection is made by directly connecting to the pins on the development board, or by using a COM terminal through the USB port on a computer. Once a connection is made, data packets can be sent to the node. The data packet follows the format shown in table 1.

Table 1: Table describing the UART packet format

Byte Number	Contents	Example
Byte 0	Number of bytes (N), including this byte	0x50 (80 bytes)
Bytes 1-(N-2)	RGB colors for each node	0xFF (red) 0x00 (green) 0x88 (blue) ...
Byte N-1	End of Message Character	0x0D (carriage return)

Once received, the CCRx registers will take the values of bytes 2, 3, and 4 from the UART package, remove the taken bytes, and resend a new packet to the next node.

4 Usage

In order to properly use this setup, a package of at least five bytes needs to be transferred in the board using UART. Each byte has its own specific meaning, as described above. To repeat, the first byte represents the amount of bytes in the overall string, the middle bytes represented the values of the LEDs, while the very last byte is the end of message check byte.

When the package is sent to the first device, it uses the second, third, and fourth bytes to change the LEDs. Then alters the first byte, and sends the rest to the next board. This is repeated until the package is empty. Packages can be sent via a master node, which can be another MSP430, a Raspberry Pi, a UART device on a computer, or a COM terminal on a computer. The nodes will receive the data at a 9600 baud rate, and the master node must match this for proper transfer.

To check for correctness, the last packet should be sent back to the master node, with only a return value. If it is executed correctly, then the following sequence should follow: as the data is sent, the LEDs will slowly changed one by one, as the data is transmitted. Then, once the data has looped through all nodes, the master node receives only a ”number of bytes” byte, and a return byte.

5 Applications

5.1 Ouija Board

By stringing 26 MSP430's together, each one acting as a node, one could recreate a Ouija board for communication. Each node represents a single letter, and by lighting up one at a time, words and sentences can be spelled out.

In this configuration, the data starting from the master node would specify which LED would turn on by filling in every node with 0s, then giving RGB data to a single LED. The time delay between LEDs would not be the same each time. The UART data takes a while to process, so it may take a few seconds for each one to light up. Then a clear signal needs to be sent, and once that finishes, then the next signal can be sent.

This can be sped up by taking advantage of the multiple colors of the LEDs. By assigning each color to an order, up to 6 letters can be displayed at once. The noticeable colors are: red, green, blue, purple, yellow, and cyan. For this to happen, both the recipient and the sender must agree on the order of each color. If this does occur, this could eliminate up to 10 signals, 5 to light up the each color, and 5 to clear the board again. It will not always eliminate 10 each time, since not all words contain 6 or more letters.

5.2 Other uses

Stringing together these nodes can be used to display packets, where each bit is base 4. Each bit would represent the red, green, and blue parts on the LED. There is no limit to the amount of nodes that can be strung together into a packet. Base 4 is not commonly used, since it is normally overshadowed by binary (base 2), hexadecimal (base 16), and octal (base 4). This is also not simple to display over the LEDs, since each bit has to be controlled by hexadecimal values, which then gets converted to base 4 on the LEDs.

5.2.1 ASCII

The packets can be decoded, as base 4, into ASCII values. Base 4 would require 4 bits for each ASCII character. Using the same amount of nodes as the Ouija board, up to 6 characters can be displayed at a time, same amount as the Ouija board. Although it would not need to be cleared first, unlike the Ouija board. This also is not limited to 26 nodes; the Ouija board cannot go above 6 characters, since it is limited by LEDs, whereas the ASCII nodes are only limited by the amount of nodes. This allows for greater efficiency as the number of nodes grow.

5.2.2 Morse Code

If not ASCII values, it can also be decoded as Morse code. Different colors can be interpreted as dots or dashes, and 'off' represents a space. This is the least productive method, as the amount of LEDs it takes for a single character is much larger than ASCII decoding. However, given enough nodes, it can still be more efficient than the Ouija board.