Drew Maatman – Kevin Etta
Caroline Gilger – Logan Wedel
Tuoxuan Ren
Dr. Cris Ababei

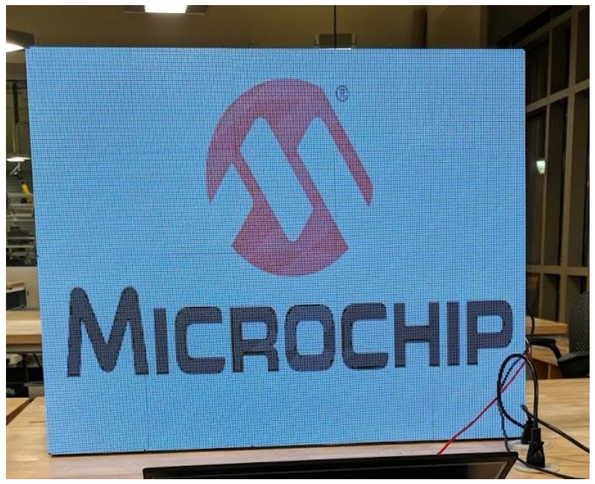
LED Display System

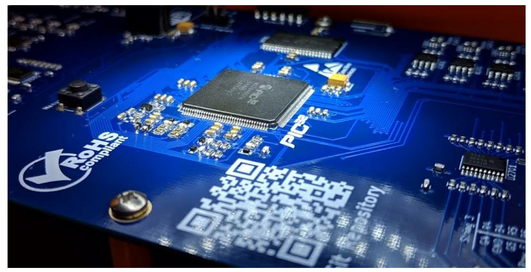
OPUS College of Engineering
MARQUETTE UNIVERSITY

Abstract

This Senior Design project implements the hardware and software necessary to realize a LED display system. Main features of this system include:

- **LED Array, 320x256 Pixel Resolution**
- **PIC32MZ-Based Custom Logic Board**
- **250 W Custom Power Supply**
- **Custom Android Application**





Conclusion

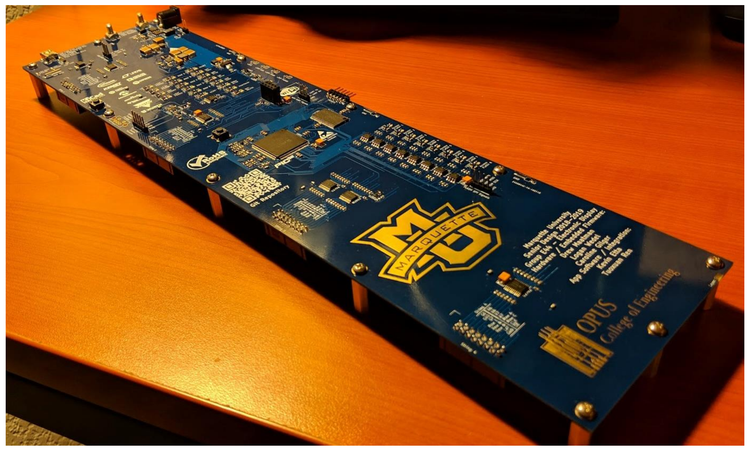
Project achieved:

- Successful hardware implementation of LED Display, Logic Control Board and Power Supply Board.
- Android app controls state of screen and uploads images
- MCU Firmware controls LED panels and cycles through images without requiring user control

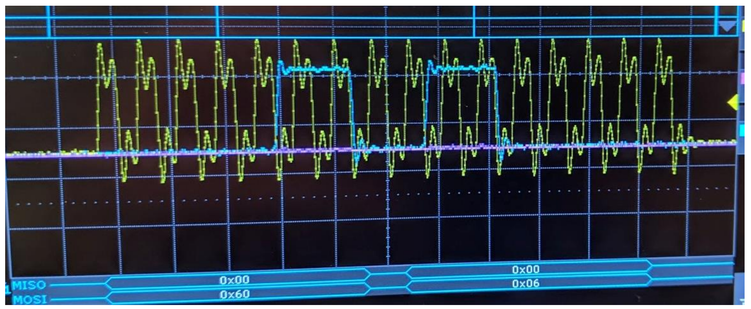
Custom Logic Board

The LED array is controlled by a custom logic board built around a Microchip PIC32MZ EF microcontroller. Image data is received using a ESP8266 WiFi module and each image is stored in one of 8 serial Flash memory chips for long-term storage. One image is stored per Flash chip. Images to be displayed are first moved to an off-chip SRAM IC, then moved into local memory. Image data is stored in a novel scheme that requires no decoding to be performed by the microcontroller. The logic board can be operated by a computer through a serial connection, or can be run independently in ”autopilot” mode.

Populated Logic Board:



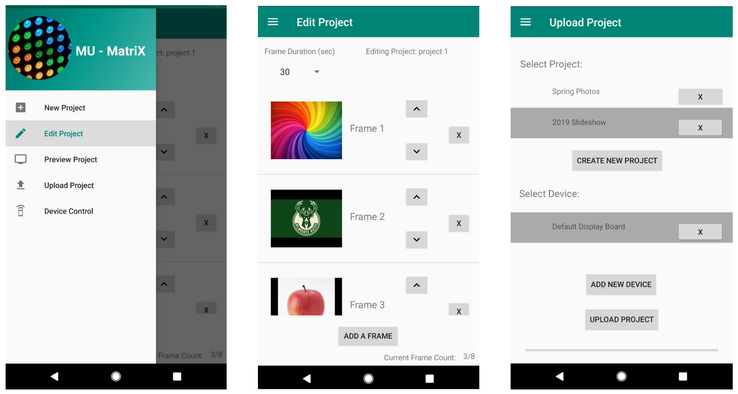
Sample SPI communication between Flash and external SRAM:



Custom Android App

The multi-screen android app serves two primary purposes: saving custom project configurations which contain up to 8 images with variable display times, and communicating with the ESP8266 WiFi module on the logic board. The app allows for images to be scaled to fit the full dimensions of the display or maintain their aspect ratio, which limits the stretching. Once images are added to the project structure, they can be reordered or removed due to the ease of use the app provides. Multiple projects can be constructed, each with custom names, image configurations, and display times. The app also supports adding additional devices to connect to, keeping it extremely flexible for scaling the scope of the system.

Main menu (left), project editing page (middle) and project upload page (right):



Device communication is simplified to three important functions: turning the LED Display ON or OFF, adjusting the brightness of the display, and uploading a project full of images to the device. A project upload sequence consists of converting the selected images into the specific format they will be stored in while being held in the flash chips on the logic board and then opening a TCP connection with the ESP8266 on a separate thread. While packets of the image are sent over the TCP connection on the background thread, a progress bar displays the percentage of the project that has been uploaded on the UI thread.

Custom Power Supply

To power the LED array, we developed a quad-phase, voltage mode control buck converter. It is based around a Linear Tech LTC7851 regulator. The supply provides 5 VDC at 45 A from 12 VDC. Power supply features include overcurrent protection, overvoltage protection, and overtemperature protection. Operation of the power supply is controlled by the logic board.

Power supply (left), section of PCB layout (right):

LED Array

20 LED panels, each in a 64 by 64 pixel arrangement, were arranged to create the massive 320 x 256 pixel display. Data is shifted through the HUB75-E serial interface into the display at around 6 MHz. By refreshing the display eight times for a single image, 4,096 colors can be created for each pixel. Using this multiplexing technique, the screen is refreshed at around 60 Hz. Refresh is invisible to the naked eye.

Single LED panel (left), rear of fully assembled LED array (right):