I²C Five Digit 7-Segment Display

Ben Holleran March 13, 2014

The following document details how the 7-segment display for the *Gordonator* bridge tester was designed, and works.

# Table of Contents

Contents

[Table of Contents 1](#_Toc382522469)

[Project Overview 2](#_Toc382522470)

[System Overview 3](#_Toc382522471)

[System Diagram 3](#_Toc382522472)

[System Description 3](#_Toc382522473)

[Design Process 4](#_Toc382522474)

[Inception 4](#_Toc382522475)

[Research 5](#_Toc382522476)

[Brainstorming 6](#_Toc382522477)

[Analysis 6](#_Toc382522478)

[Design 6](#_Toc382522479)

[Testing 7](#_Toc382522480)

[Documentation 7](#_Toc382522481)

[Project Management 7](#_Toc382522482)

[Conclusion 8](#_Toc382522483)

# Project Overview

The *Gordonator* bridge tester is an electro-mechanical senior project at Vermont Technical College. The goal of the project is to build a machine that can measure the force needed to break structures designed by the civil department, Specifically bridges constructed with popsicle sticks with a one meter span. The tester will be capable of replicating the Troitsky Bridge Building Competition at the Concordia University in Canada.  
  
The Load Display subsystem will be displaying the force exerted on the bridge by the piston. Since the expected load is in excess of 9,999 LB, and less than 99,999 LB the display will have 5 digits. The display will also have 4” letters so the winning number can be seen from the back of the room.

# System Overview

## System Diagram



## System Description

As seen in the above system diagram, the display will be broken into 5 identical systems, each with a different IIC address. Each cell will have a single 7-segment display and the circuit needed to drive it. This will allow for easy replacement, and ease of design.

## Design Process

## Inception

The first step in the design process for the Load Display was its inception. There needed to be some form of a display to show the crowds how much force each bridge could hold. Some competitions do not need a load display, specifically high school competitions where weight of some kind is added incrementally until the bridge breaks. The weight can then be added up empirically by measuring the amount of water, or counting up how many weights were added. Since this tester will be using a hydraulic cylinder to provide force, allowing much higher loads than practical by the added weight method, there needs to be a display.

Research

The current tester at the Troitsky Bridge Building Competition was researched, and found to have a display only 1” high. Numerous people who have attended the competition have expressed that the display is too small, and is not visible from the seating area.

The problem was discussed with fellow students, professors, Industry professionals, and all of that was held together with queries to Google.  
  
While researching current existing technology a few interesting discoveries were made:

* Many 2” and 3” clocks exist premade, but as clocks only have 4 digits.
* Many of the chips designed to run 7 segment displays are designed to run clocks.
* There are large numerical displays that are not clocks, but they are very expensive. The cheapest being around $500
* DigiKey sells individual 7 segments of all sizes.
* Texas instruments makes a nice IIC shift register suitable for running a single digit.

While researching communications it was determined that there were a few different technologies that were suitable.

* Serial
  + Single receiver, would require a MUX to run the separate digits
  + Very easy to use
* IIC
  + Multiple receivers
  + More difficult to use
* Direct IO
  + Easy to use
  + Needs a complicated circuit
  + Not enough pins

Brainstorming

After the available technologies were researched, the design was brainstormed to determine the best way to build the load display. It was determined that there were not enough pins on the Pi to do Direct IO and that a MUX circuit would be harder to make than the other available solutions. Serial was disqualified since it can have only one receiver, and that would make that receiver need to have enough pins to run all the displays, bringing back the Direct IO problem. There is multi receiver serial, but if going that route, IIC makes for a much better solution since it is in essence multi receiver serial, with the protocols already in place.

At one point it was considered to rate each system based on a number of criteria and pick the best one numerically, but those charts tend to be bullshit, so we just discussed the systems until it was clear which ones were less favorable, and then made an educated gut decision.

In the case of displays, it was originally thought that the best way to make a display would be to make the 7-segments from scratch using 5mm LEDs and large circuit boards. This was dismissed as too complicated and time consuming. Hacking a Walmart display was considered to be a bad idea in general, and all in all, the best solution seemed to be to buy premade 7-segments.

## Analysis

Once the 7-segments were chosen, the forward voltages of the LED arrays were measured and plotted on a Gaussian bell curve. This was then used to get a statistical voltage for the displays, allowing a more accurate current limiting resistor to be added.

The Datasheet said that the forward voltage on the 7-segents might be as low as 8V, meaning the resistor would have 4V from the 12V line across it. The calculated value to regulate to 60mA meant that if the forward voltage of the LEDs was 10V, the upper limit in the datasheet, there would only be 30mA going through the LEDs. This would significantly cut down on visibility. By using the voltage determined from the Gaussian bell curve, a resistor value was found that would put 70% of the LEDs at 50mA while being 90% sure that no LED would have over 60mA

Design

It was determined that the best way to construct a Load display would be to purchase 4” 7-segment displays from DigiKey, buy an IIC 7-segment driver chip from TI, create a board to connect everything, and then use a Python program running on the Pi to tell it what number to display over IIC.

## Testing

Once the Raspberry Pi was up and running, testing began on the chips and 7 segment displays. At first there were many unknown errors. Many of them proved to be problems with understanding of the IIC bus and driver issues on the Pi, but those were quickly sorted out. One of the more difficult errors to track down was a bad solder joint on the surface mount driver chips. After this discovery, all pins on all the chips were resoldered by hand and individually checked for continuity and shorting.

## Documentation

At all stages of the process, notes were taken in a project notebook. Some documentation for code was added to the GIT repository as commits. These comments were clear so that when looked at later, they would be easily understood.

## Project Management

Most of the project management: Task assignment, issue tracking, timeline, and file management, was all done by GitHub, a free online project management website. This tool will continue to be used throughout the project and will allow other people interesting in the design of the project to see what decisions were made.

# Conclusion

This subsystem is on track to being finished for its integration with the main project. The goal is to have a rock solid subsystem, so that any issues will be from its integration to the rest of the system, rather than issues with the subsystem itself.