EEL 4924 Electrical Engineering Design (Senior Design)

Final Design Report

April 25, 2012

Project Name: **3D LED Cube**

Team Members:

Name: **Angel Perez**

Name: **Robert Regojo**

Project Abstract:

Our project consists of building a 3 dimensional LED array that will be able to display various graphics through the concept of persistence of vision. The array will also be sensitive to motion in three directions, allowing it to focus certain graphics to a targeted audience through motion detection. There will be several options for display including non-directional animations and direction focused graphics. We will be using infrared sensors to design and build a motion detection system that will be fed into our processor. The processor will, through several inputs, decide what graphic to present and will feed it to an FPGA. The FPGA will then process the necessary data and output to the 512 LEDs to be used in the 3D array.

Table of Contents

Table of Figures

Project Features/Objectives:

The goal of this design is to be able to output and modify the LED array fast enough to see a persistent image:

- The first issue that must be dealt with is the physical construction of the array. The array will be 8x8x8 LEDs, accounting for a total of 512 devices. Due to lack of accessibility we will have to make certain that each LED is functional and stays so throughout the construction.
- A sturdy base and casing will also have to be provided for the array, as the construction doesn't allow for a large amount of structural integrity. A wooden base and a Plexiglas case is proposed to deal with this issue and to protect the LED array from general jostling and movement.
- Due to the very large number of LEDs that need to be used at once, current considerations will have to be taken into account, verifying that we have enough power to supply a good level of luminescence so that we may not only turn on all LEDs but also modify them through pulse width modulation.
- The microprocessor will be in charge of user inputs, motion detection and general code development for the graphics. It will process all inputs and verify what set of parameters need to be outputted to the FPGA. It will also control the pulse width modulation that will be used to modify the dimness of the LEDs
- The FPGA will process the various inputted signals and implement the digital hardware necessary to output the +64 signals required to functionally modify the LED array. Here

we are looking to make the code as fast as possible, so as not to create a bottle neck in our refresh rate.

- Since each LED needs to be controlled individually, memory issues will have to be considered when adding more graphic options. Otherwise, we will have to find ways to streamline our code to allow for more variety without a significant increase in the memory needed.
- The motion detection system will be built from scratch using infrared detection. It will be able to detect motion and focus an image to wherever the motion is detected.

Concept/Technology Selection:

We have chosen the implementation of this project based on our teams experience and the simplest methods by which we see to complete our goals. When constructing the actual LED array we have chosen to construct the array in layers, verifying that all LEDs function after every step. Due to close proximity soldering there is a high chance that some of them may burn out and we would like to catch this early on. Once we have all layers completed we will stack them and solder the layers on by one till they are fully assembled. We shall also place several strong strands of wire to support the structure and increase its integrity.

We have chosen to do the main processing in C through our MSP430. Since both team members have significant experience in coding this device and language it will help develop more intelligent and succinct code. The digital hardware that will be the basis for the LED driver will come from an FPGA. This will allow us the benefit of speed to update outputs as fast as needed as well as reducing our design footprint. Though the PCB design may be more difficult the result would be much more beneficial than using several ICs, especially due to the large number of outputs we require.

There are two ways in which we can fashion the code to control the LEDs. The first would be to preprogram individual bits to be retrieved and outputted sequentially. This is the brute force method but it may prove easier than the others. The main issue however is memory allocation and size, for multiple graphics or the addition of new ones we will probably need to add external memory to process it. The second method which is the one we will attempt is to make code as intelligently as possible so that the designs can be created and output directly from the microprocessor, without other hardware required. This method may be slower but it would be more eloquent and require a physically smaller design. The best solution may be a combination of the two systems to achieve a maximum number of graphics possible.

Finally, our implementation of our motion sensor shall be designed from scratch. While there are several easy to purchase motion detection devices in the market we will have more flexibility with the design, placement and response by creating it ourselves. We have chosen to have a minimum of three motion detectors to account for three intended viewing angles of the cube, front, left and right. This will allow us full awareness of detection surrounding the cube and allow us the possibility to change graphics as needed.

Flowcharts & Diagrams:

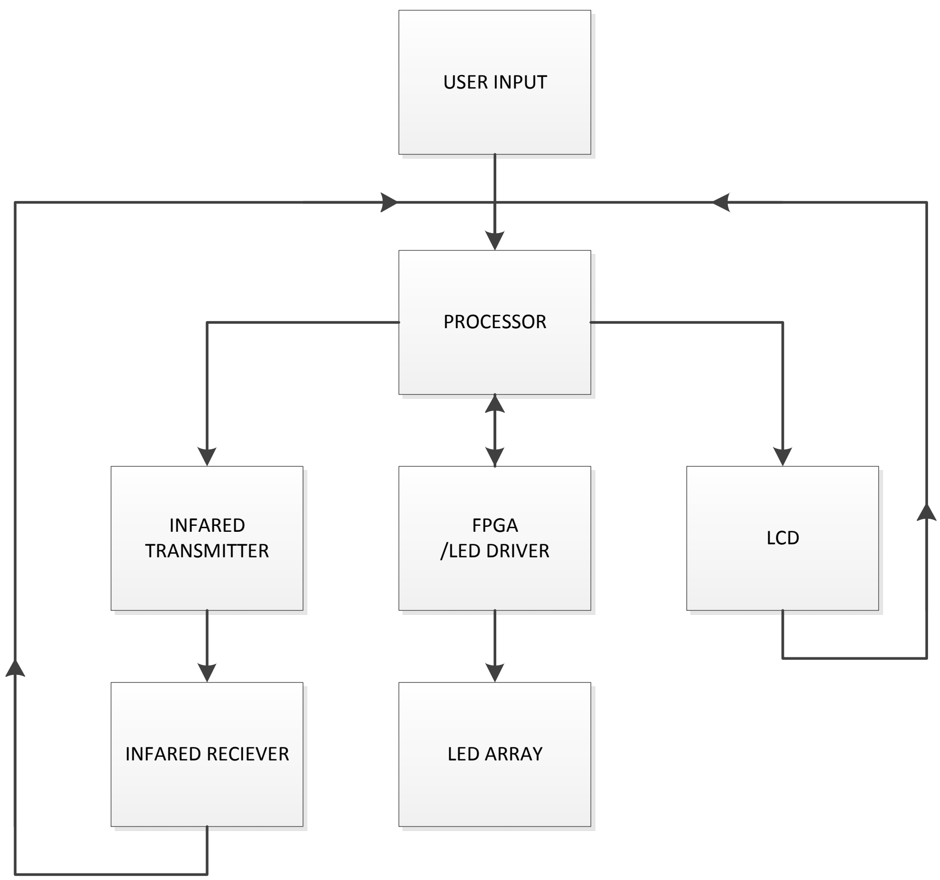

Figure 1: *Hardware Flowchart*

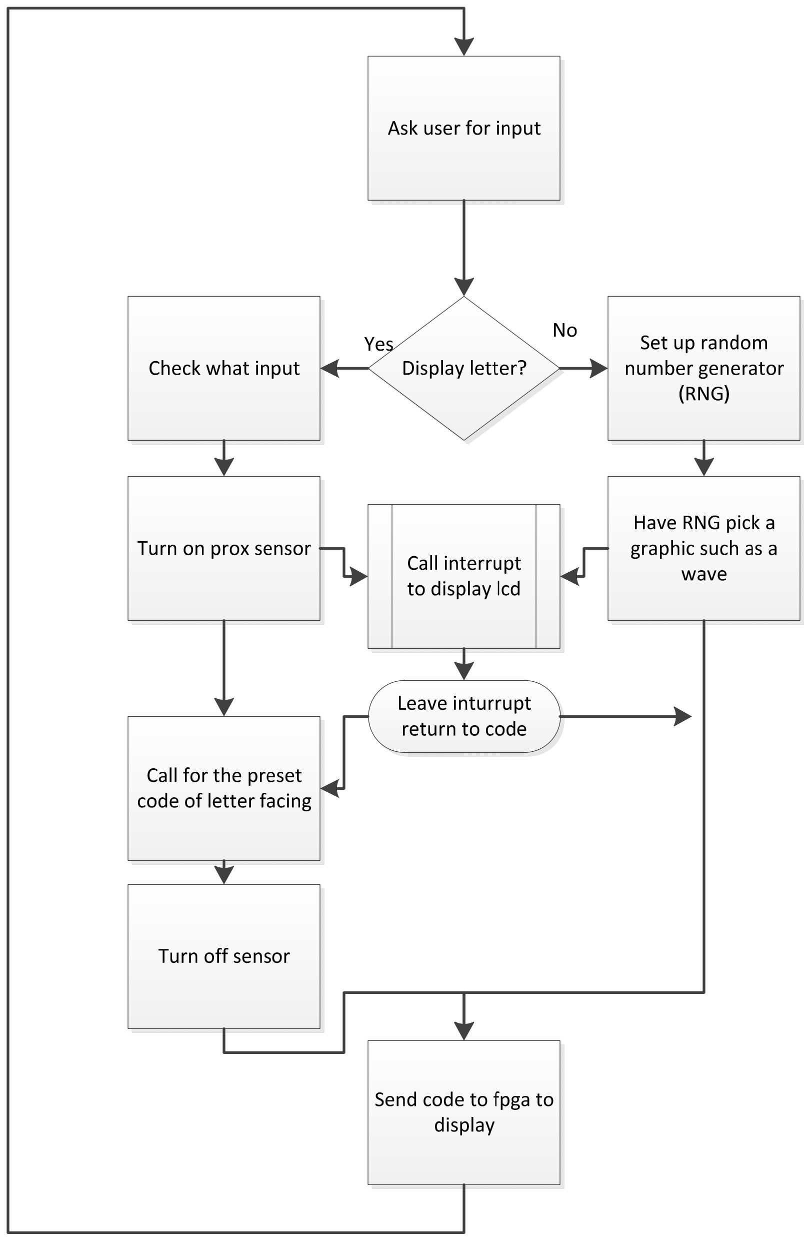

Figure 2: *Software Flowchart*

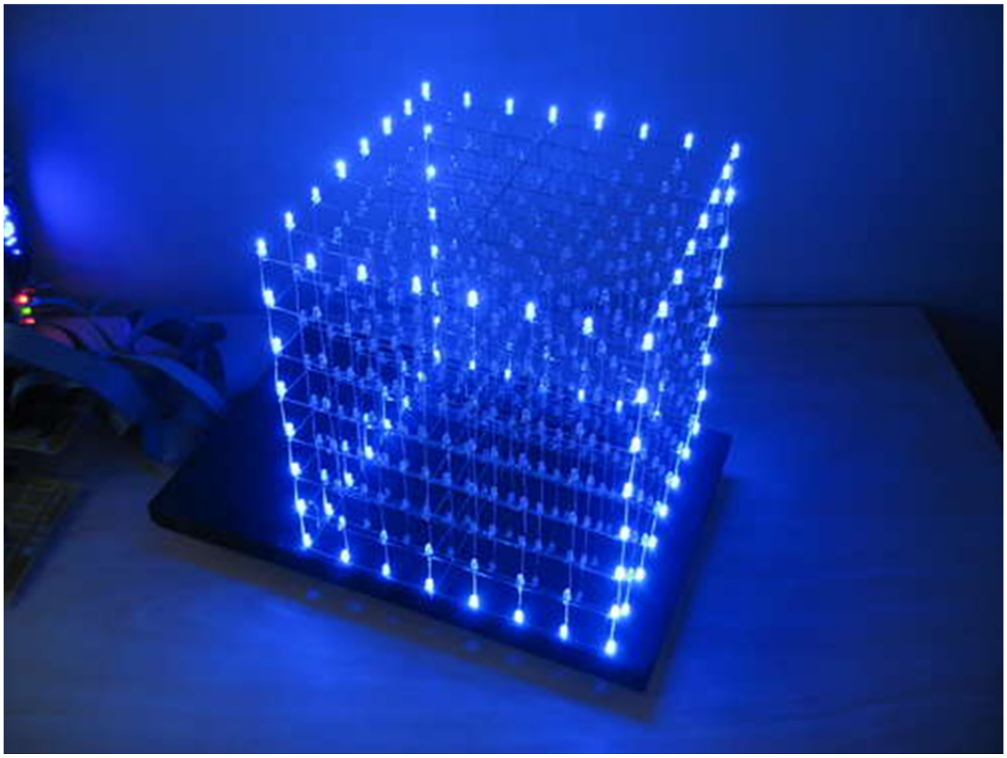

Figure 3: *Finished LED Cube Array*

Design Report: 3D LED Cube

Schematics

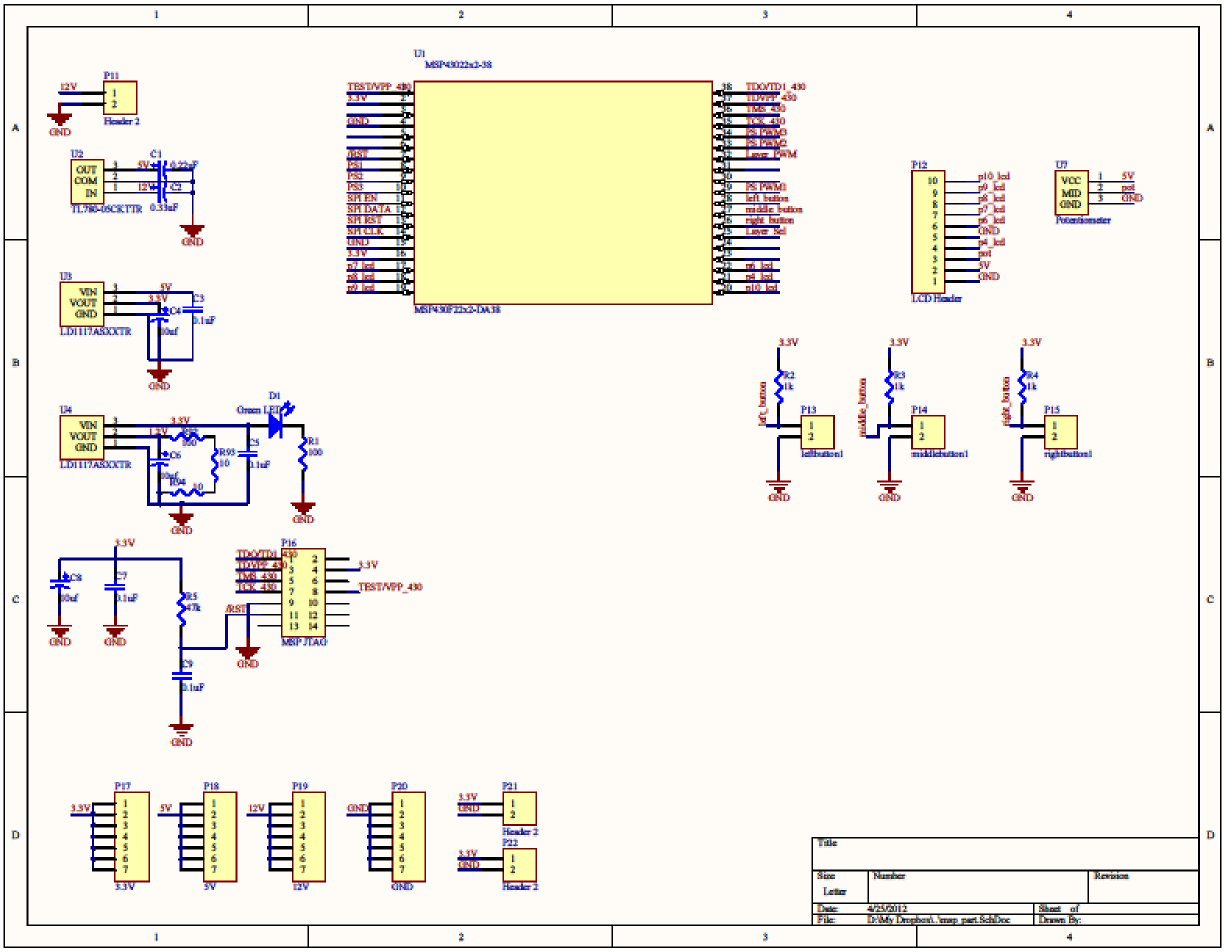

Figure 4: MSP430 and External Devices Schematic

Design Report: 3D LED Cube

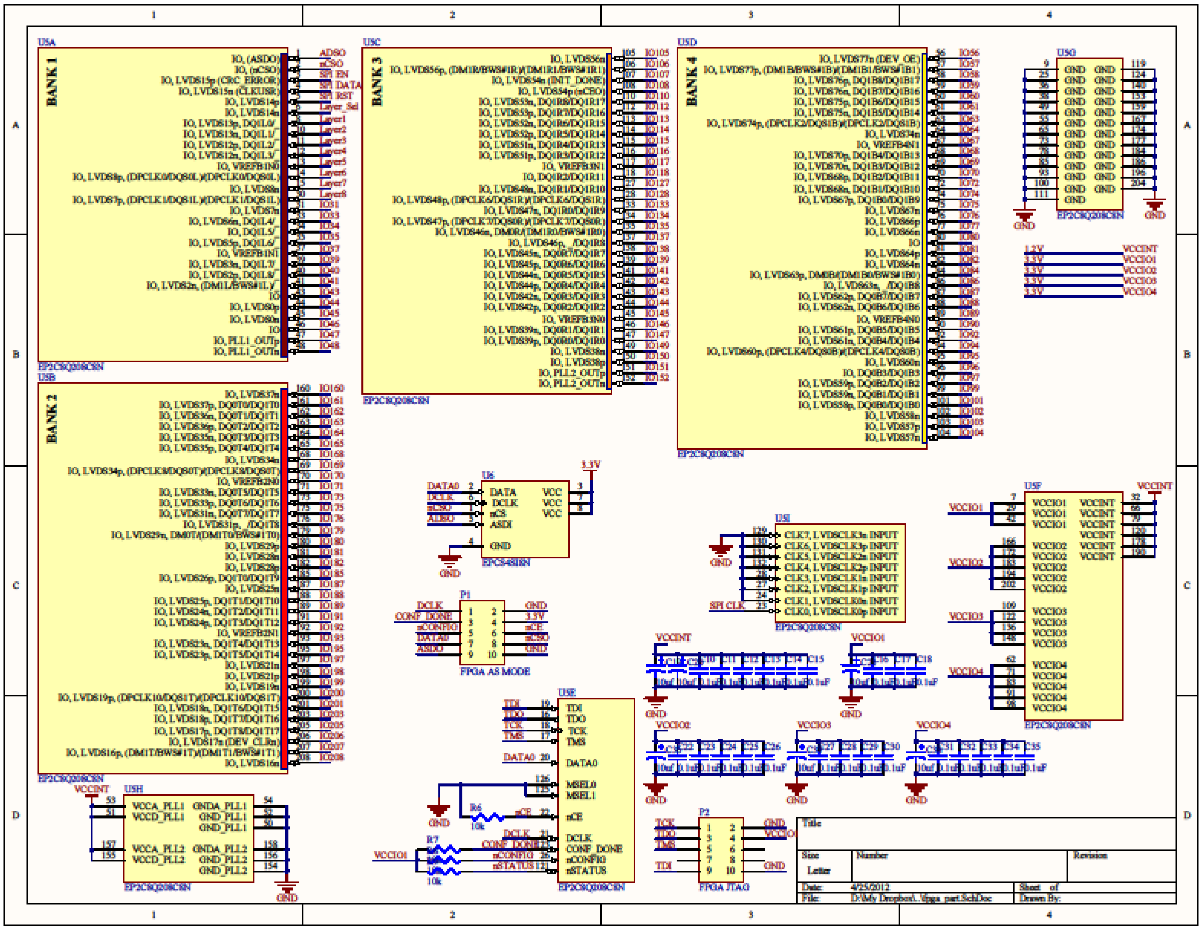

Figure 5: FPGA Schematic

Design Report: 3D LED Cube

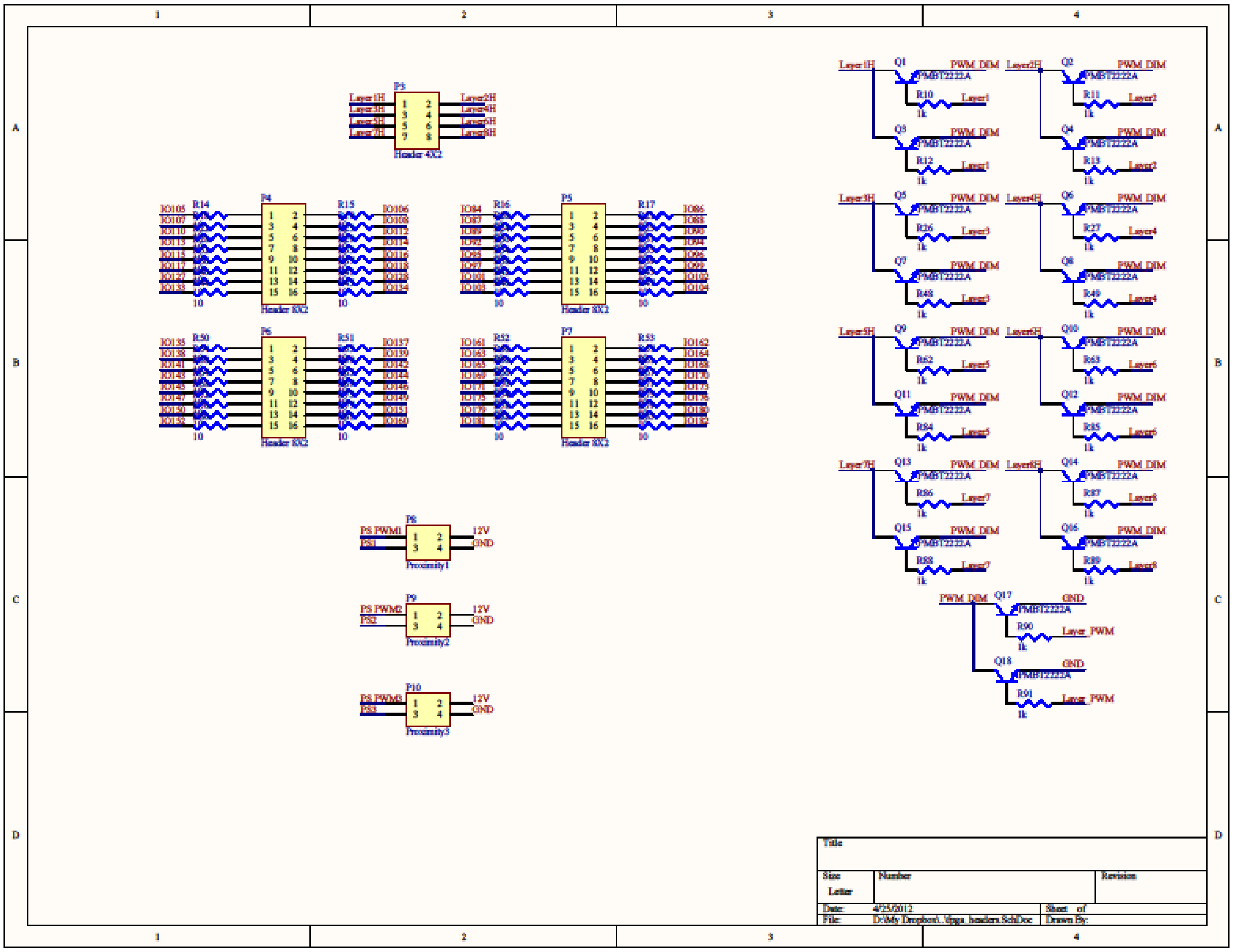

Figure 6: FPGA Communication Schematic

Design Report: 3D LED Cube

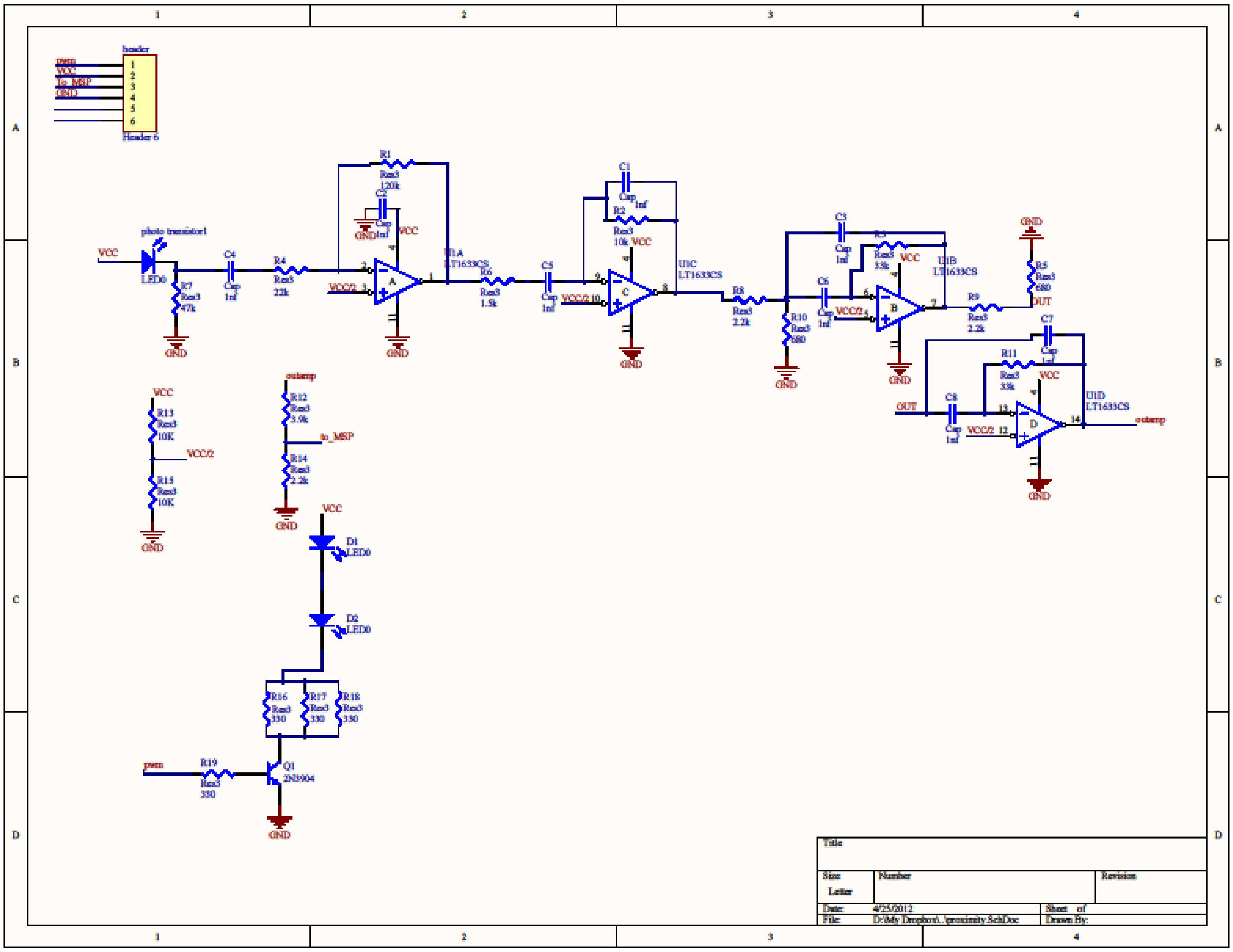

Figure 7: Proximity Sensor Schematic

Separation of Work:

The work was separated in such a way that both of us would be working on different parts of the project at the same time until spring break was over. This of course was done after the led array was built. We decided to split the workload in the following ways. Robert would first handle designing the proximity sensors. This will include designing, simulating, and building the sensors using IR detection. While Rob works on this, Angel will be working on the design of the led controller. This will consist of designing the PCB board including the FPGA. The controller will link the outputs of the registers to the LED's and receive the inputs from the main board, which will hold the micro controller. Once this is done the focus will then be geared into coding the FPGA which will be done by Robert and begin the coding process of the main board which will be done by Angel.

The FPGA coding will consist of implanting both shift registers and multiplexers. The shift registers will be the output to the each led pin while the multiplexer will control which layer is activated at a time. The coding of the main board will consist of implementing the proximity sensors with the user interface buttons, LCD coding and setting up the functions in order to add the graphics later to the main code. Once this task is done the focus will then be turned into designing the main board which will hold the processor, sensors connectors, transistors, user button connector, power supply, and LCD connector. This task will be done mostly by Robert. While this is done Angel will be working on programming the first set of graphics for the Array. Once the main board is designed both of us will focus on implementing the hardware and make sure that the first graphic can be demonstrated. After this is done the graphic programming will then be split between the both of us.

The work load from there on out will be done mutually. This will include building the housing for the array, the housing for the electronics, and also will include testing and debugging the whole project. The schedule of how the work is separated can be seen on the Gantt chart below.

Gantt chart:

	Start Date	Subtask 1	Subtask 2	Subtask 3
Problem determined	7-Jan-12	3	3	3
Research project	9-Jan-12	14	0	0
Design LED Array	2-Feb-12	20	3	0
Design Proximity Sensors	17-Feb-12	20	3	0
Design LED Controller	17-Feb-12	20	3	0
Spring Break	3-Mar-12	12	0	0
Mainboard Design	26-Mar-12	5	0	0
FPGA Coding	12-Mar-12	7	7	0
Graphic Design Coding	29-Mar-12	28	0	0
Processing Programing	12-Mar-12	7	7	2
Hardware Implementation	9-Apr-12	5	5	0
Case Design	13-Apr-12	14	0	0
Final Testing	19-Apr-12	14	0	0
Presentation	25-Apr-12	5	5	0

Design Report: 3D LED Cube