Homework 5: Theory of Operation and Hardware Design Narrative

Team Code Name: Every1 DJ **Group No.** 2

Team Member Completing This Homework: Sophie Pouliquen

E-mail Address of Team Member: spouliqu@ **purdue.edu**

Evaluation:

SEC	DESCRIPTION	MAX	SCORE
1.0	Introduction	5	
2.0	Theory of Operation	20	
3.0	Hardware Design Narrative	20	
4.0	Summary	5	
5.0	List of References	10	
App A	System Block Diagram	10	
App B	Schematic	30	
	TOTAL	100	

Comments:

1.0 Introduction

The proposed design is an interactive DJ party system with an accompanied web application for users to add and vote on songs to be played. The physical design consists of two main hardware elements: a Raspberry Pi and a microcontroller. First, the Raspberry Pi will capture songs wirelessly from an online music API. The Raspberry Pi will send song information through Serial Peripheral Interfacing (SPI) to the microcontroller. The microcontroller controls the LED's, motor activated disco ball, LCD to display song information, push buttons for user song interaction and also has an auxiliary output to speakers. Some of the major circuit design constraints include power regulation, frequency control and the filtering of the audio output.

Updated PSSC's:

1. An ability for a user-created webapplication to communicate a selected song to the Raspberry Pi through an http request (API poll).
2. An ability for a computing PC (Raspberry Pi) to stream music through the Grooveshark API and communicate the song information with the microcontroller.
3. An ability to create a 'lights' show corresponding the music beat, which will be sampled from the auxiliary out of the Raspberry Pi via the microcontroller ADC.
4. An ability to control the direction and rotation speed of the LED package through a motor controlled via the micro PWM peripheral.
5. An ability to utilize an LCD to display user-selected (via a control device) metadata (e.g. audio options, light settings, and track information)

2.0 Theory of Operation

The Every1 DJ is powered through a 12V wall power supply. The 12V supply will be used directly to power the stepper motor. A low pass filter with a heat sink will be used to bring the supply voltage down to 5V to power the Raspberry Pi. A low-dropout regulator will be used to drop the voltage from 5V to 3.3V, which helps minimize heat dissipation. The 3.3V will supply power to the microcontroller, the LED driver, the LCD, and the audio filtering circuitry. With these three necessary voltages, all components in the system are powered.

Once the system is powered on, the Raspberry Pi will boot and run “turn-key” code to initiate communication with the web application. Interaction between these two will take place in the form of an API-like interface via HTTP GET and POST requests.

The Raspberry Pi will continually relay audio sampling and song information to the microcontroller via SPI [1]. The microcontroller will then drive the LCD via SPI to display song information to the user [2]. The microcontroller will also communicate with the LED driver via I2C protocol [3]. Table 1 summarizes all the communication protocols for the Every1 DJ system.

Component	**Protocol**
Raspberry Pi	SPI
LCD	SPI
LED Driver	I2C
Web Application	HTTP

Figure 1: Digital Communication Protocols

An external crystal will not be necessary for this design because the on-chip oscillator provides a fast enough clock speed to handle the requirements for the project. The operating frequency will be 4MHz as this is the default frequency of the internal oscillator and will be scaled down for clocking each of the peripherals [4]. The SPI communication with the Raspberry Pi will be clocked at 250 KHz using a prescaler of 16. The maximum frequency of the chosen LCD was not listed on the data sheet but a larger prescaler of 64 will be used, making the frequency 62.5KHz.

The LED driver is a timer-interrupt driven device that is optimized for voltage switch dimming on the attached triple output high power RGB LED’s. The driver offers frequencies up to 1 MHz [3]. The LED driver will receive audio output frequencies, which will determine the power and color of each LED.

The LCD is a passive device that will display song information via SPI from the microcontroller. The serial interface includes SCLK, SDIN, D/C and /CS. The data from the microcontroller is shifted in through an on chip 8-bit shift register on each rising clock edge [2]. Changes on the LCD will appear based on the song and push button selection information from the microcontroller.

The user operated control panel will consist of four push buttons to control play/pause, next and LCD cycling. The push buttons will be interrupt driven. For the buttons associated with song interaction, the microcontroller will need to relay this information, via SPI, back to the Raspberry Pi. This way, new information can be fetched and relayed back to the microcontroller.

3.0 Hardware Design Narrative

Due to the variety of functions required from the microcontroller five ports will be used: B, C, D, E, F. In some cases there may have been an option to use the same ports for multiple functions but this was avoided when possible to reduce the distance and complexity of the PCB routing. SPI is necessary for communication with the LCD and Raspberry Pi and will utilize Port C. The SPI slave select pin will determine which of those two devices the micro will be communicating with. The I2C communication with the LED driver will take place on Port D. External interrupt pins on Port B will be connected to the push buttons. Port F has the capability of analog communication and will therefore be used to sample the audio input from the audio out of the Raspberry Pi. Finally, the PWM for the motor control will be connected at Port E. Each component, port and signal types are summarized in Table 2 [4].

Component	**Port**	**Signal Type**
Raspberry Pi	C	Digital
LCD	C	Digital
LED Driver	D	Digital
Push Buttons	B	Digital
Audio Sampling	F	Analog
PWM for Motor	E	Digital

Figure 2: Microcontroller Port Assignments

The use of decoupling capacitors is required for each of the five VDD/VSS and AVDD/AVSS pairs. The values of these capacitors were chosen based on the recommendations of the manufacturer and will therefore each be 0.1uF [4]. A 0.1uF capacitor will also be used to increase the application's resistance to unwanted resets caused by voltage sags on the Master Clear Pin. The on-chip voltage regulator pin will connected to the 3.3V supply voltage and will require a 10uF capacitor to stabilize the output voltage. Each of the capacitors will be placed as close as possible to the pins ensuring the trace does not exceed a length of 0.25 inch, as shown in the Appendix B schematic. The digital and analog grounds will be separate throughout the design and connect at a single point on the PCB.

4.0 Summary

The Every1DJ system includes the following major subsections of hardware: microcontroller, Raspberry Pi, LCD, LED driver, push buttons, audio sampling and motor control. The decision to power the device with a 12V power supply was due to the power constraints of the stepper motor. From here, reduction in power through low-pass filters and low-dropout regulators will supply power to the rest of the components. The necessary circuitry and supportive components including the audio filtering, voltage regulation and connectivity have been organized in Appendix B. These components, in addition to the web application, comprise the block diagram in Appendix A.

5.0 List of References

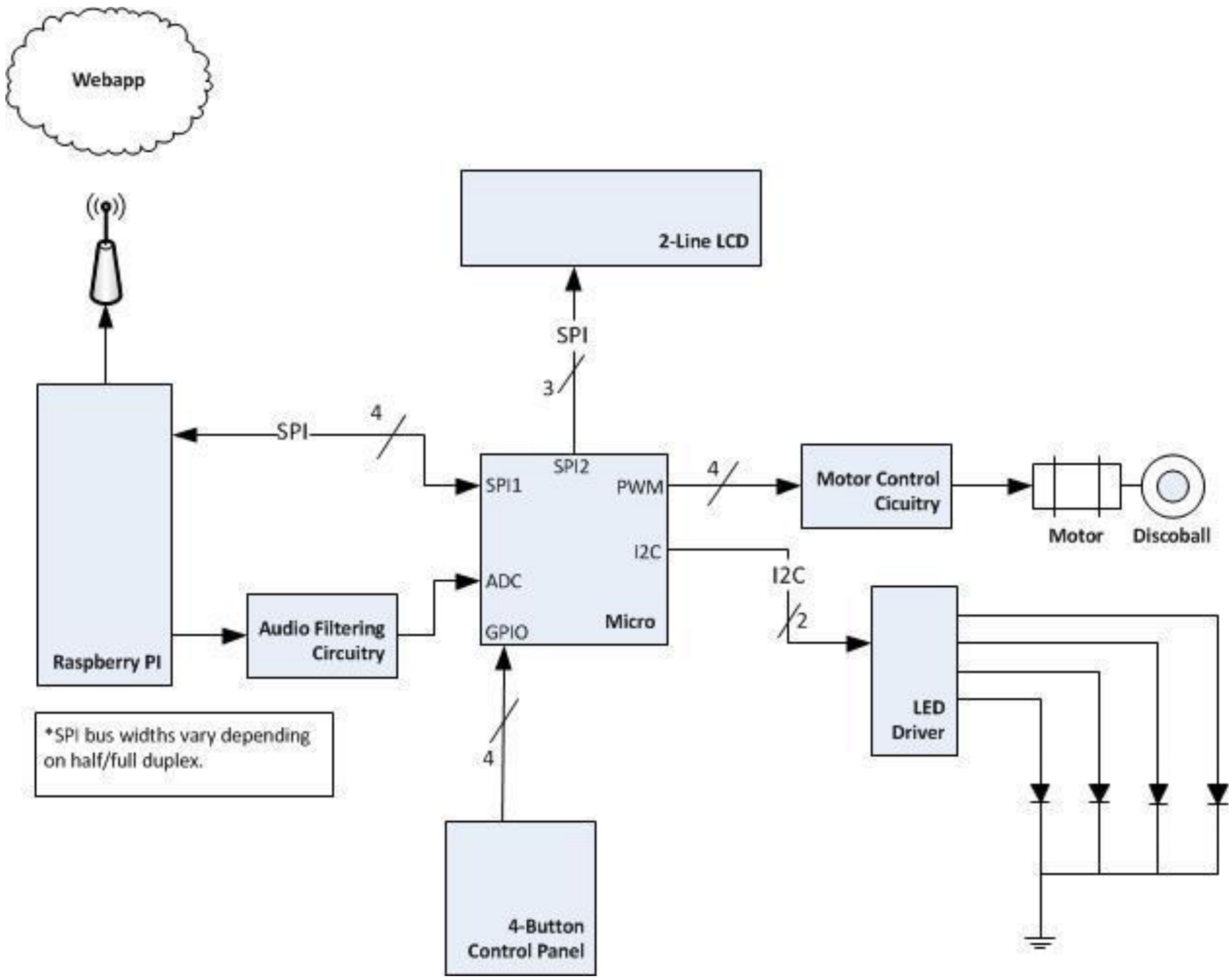
[1] "Raspberry Pi" Internet: http://downloads.element14.com/raspberryPi1.html, [Feb, 6 2014].

[2] Newhaven Display, "Graphic OLED Display Module", NHD-2.23-12832UBB3 datasheet.http://www.newhavendisplay.com/specs/NHD-2.23-12832UCB3.pdf

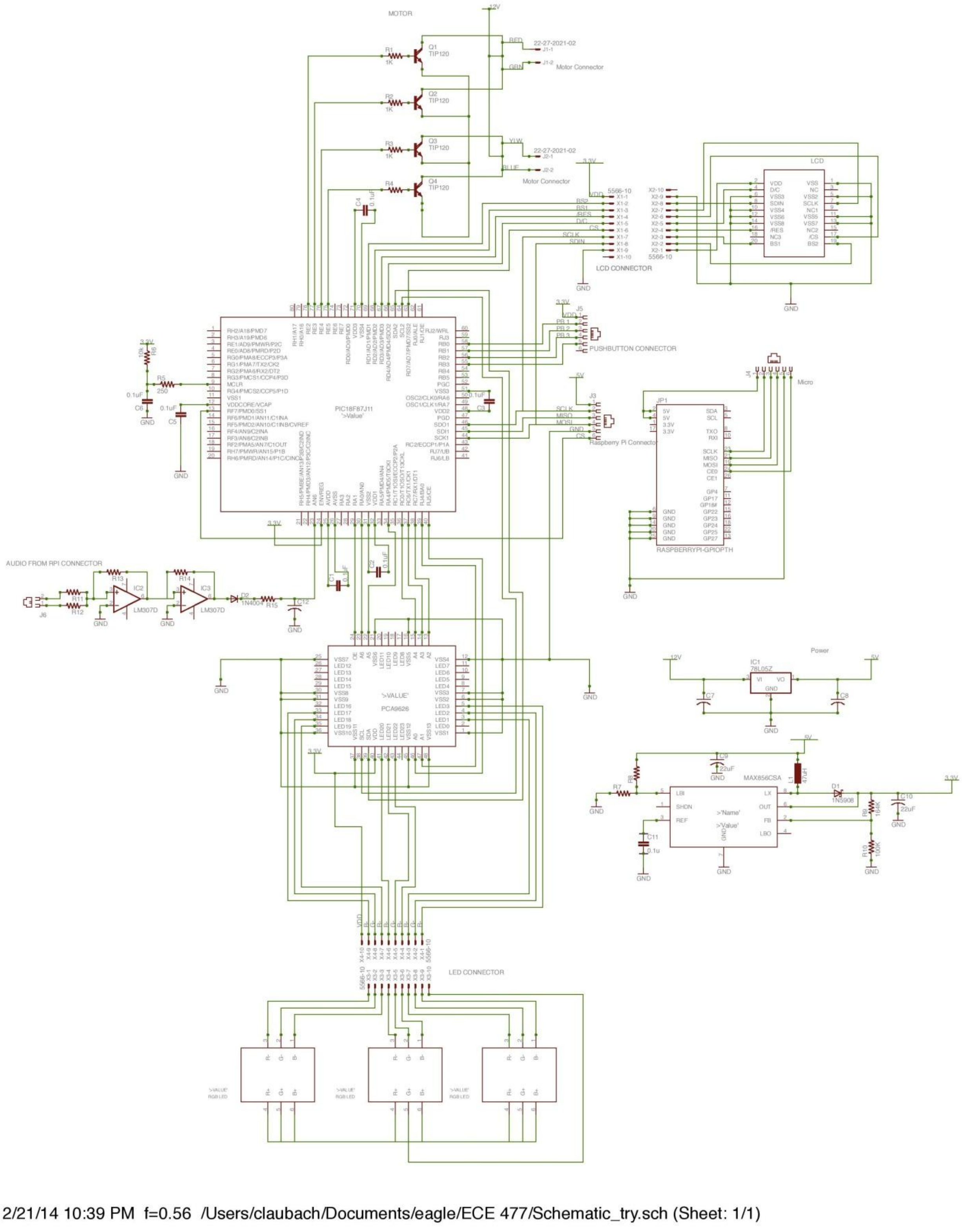
[3] NXP Semiconductors, "24-bit Fm+ I2C-bus 100 mA 40 V LED driver", PCA9626 datasheet, Sept. 2012 http://www.nxp.com/documents/data_sheet/PCA9626.pdf

[4] Microchip, "64/80 in, High Performance, 1-Mbit Flash Microcontrollers", PIC18F87J11 datasheet, Nov. 29, 2011 http://ww1.microchip.com/downloads/en/DeviceDoc/39778e.pdf

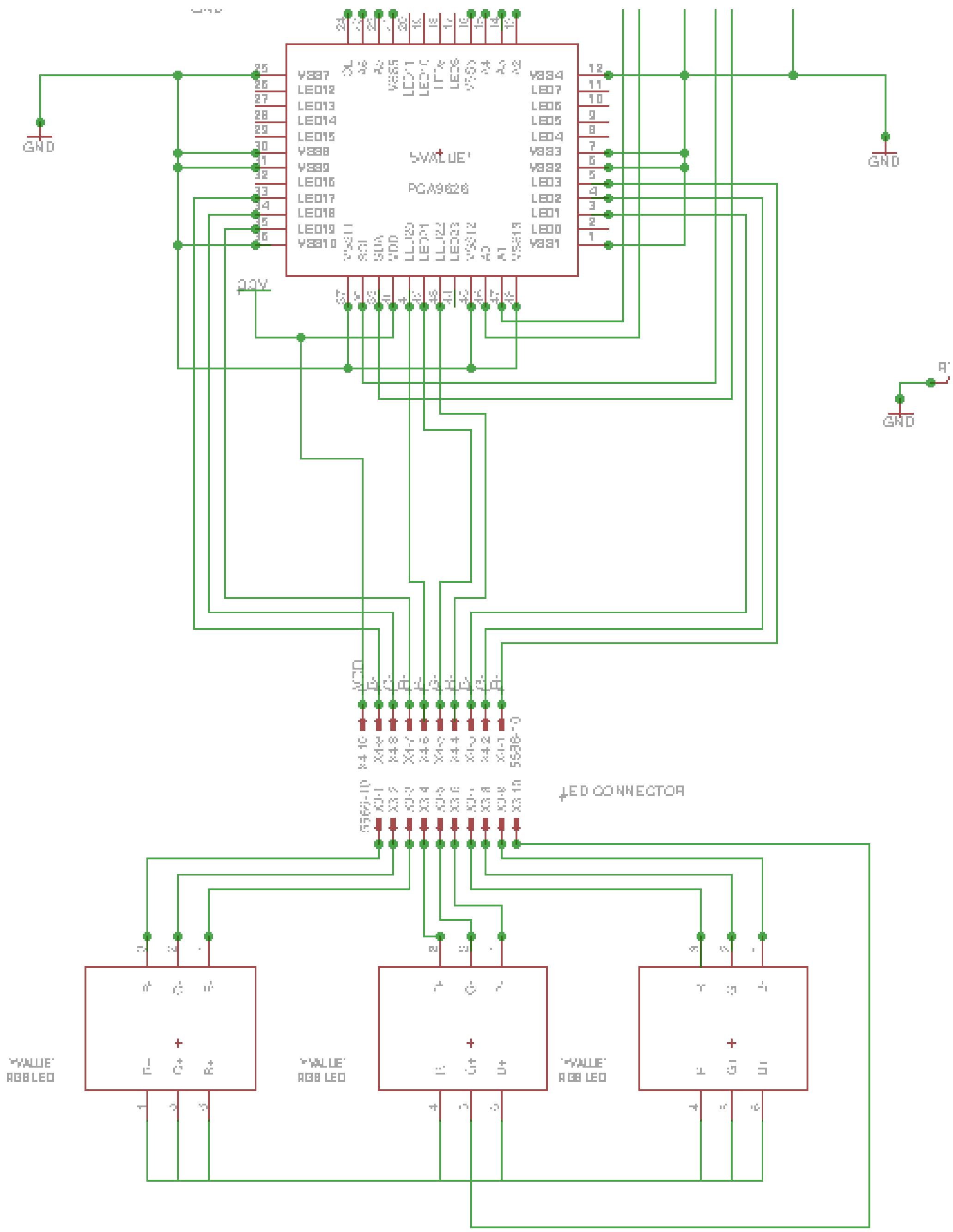
Appendix A: System Block Diagram

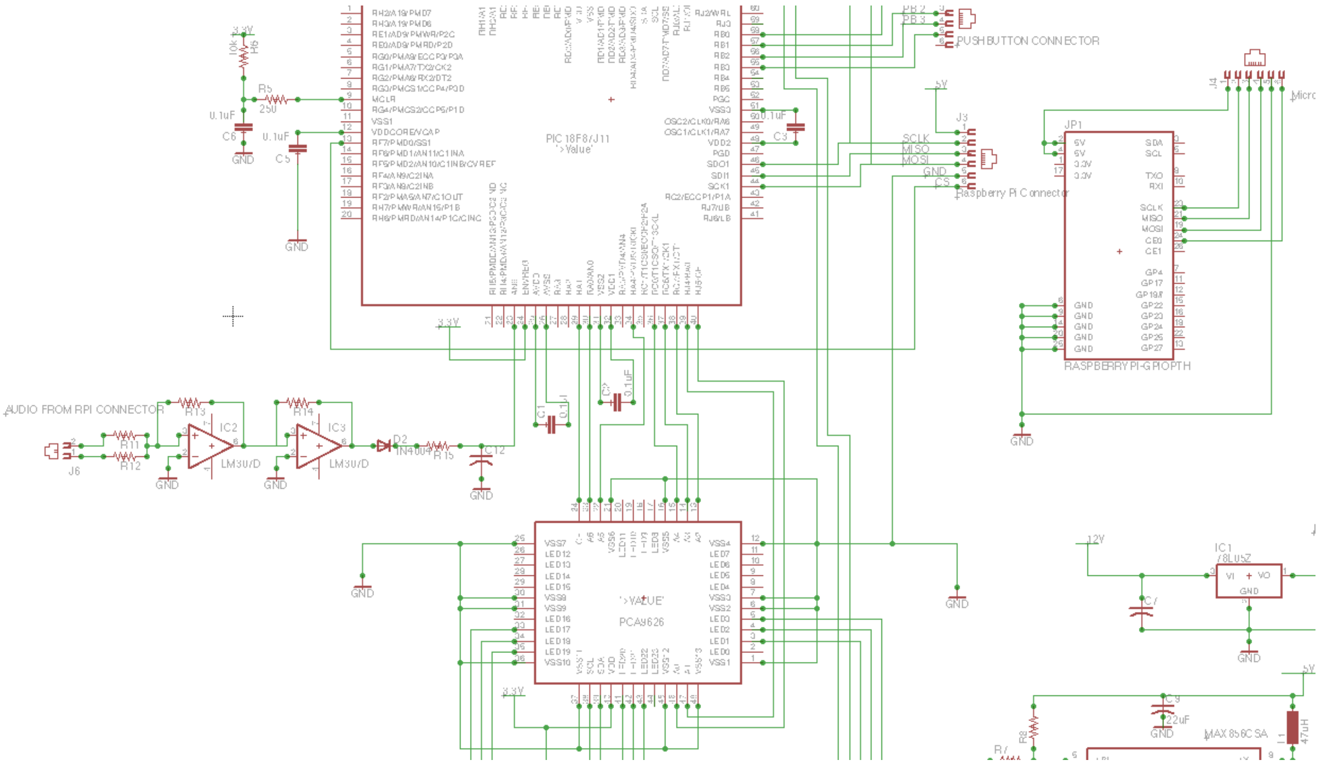


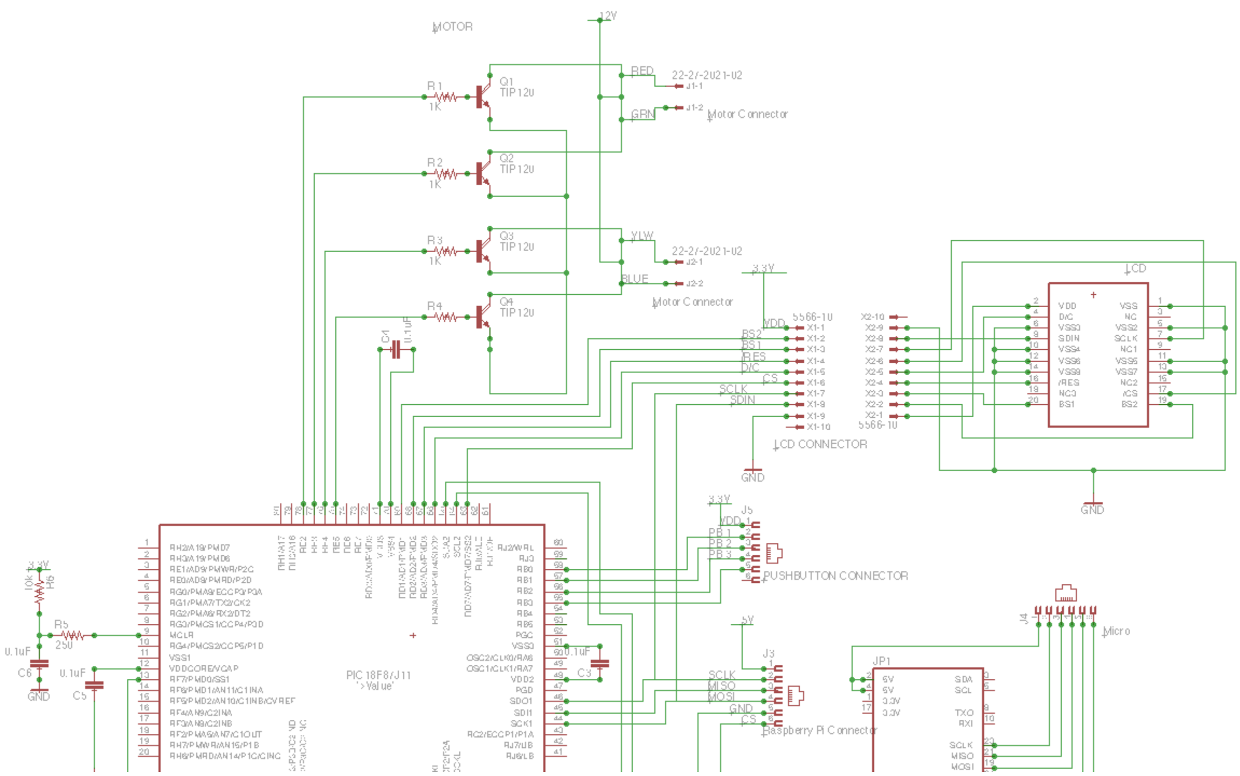
Appendix B: Schematic

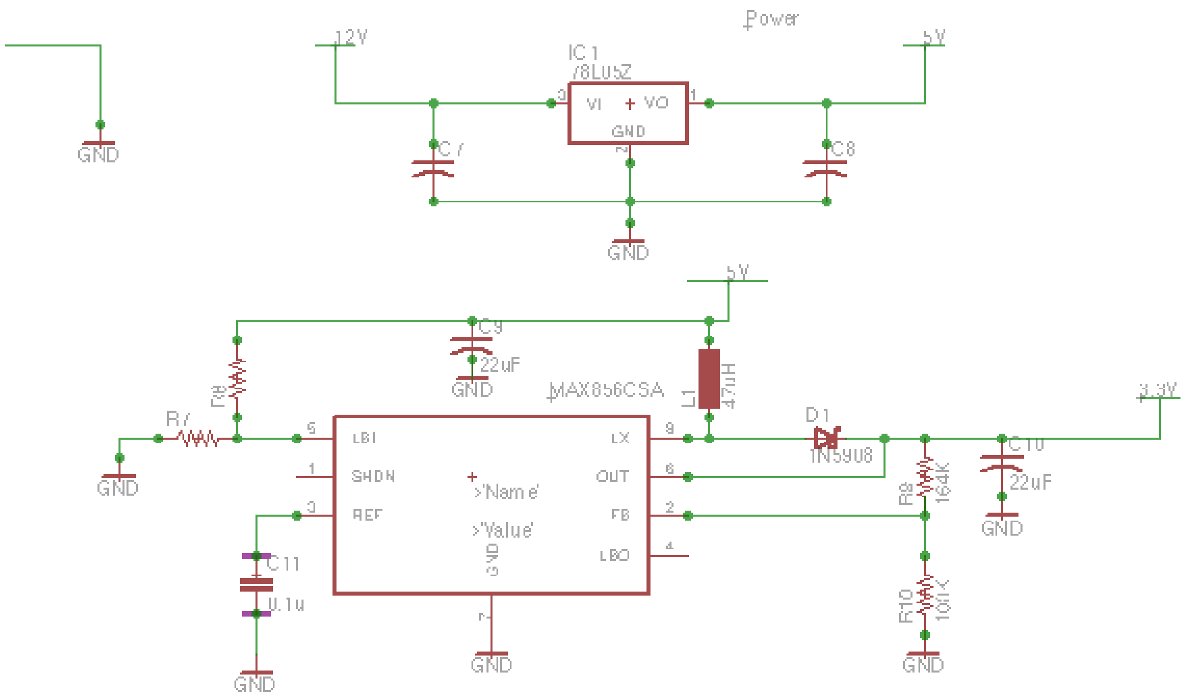


2/21/14 10:39 PM f=0.56 /Users/claubach/Documents/eagle/ECE 477/Schematic_try.sch (Sheet: 1/1)


GND
PCA9626
LED CONNECTOR
3.3V
RGB LED
RGB LED
RGB LED


PUSH BUTTON CONNECTOR
PIC 18F8/J11
Raspberry Pi Connector
JP1
RASPBERRY PI-GPIOPTH
AUDIO FROM RPI CONNECTOR
IC2
IC3
LM307D
1N4004
PCA9626
GND


MOTOR
12V
Q1
TIP120
Q2
TIP120
Q3
TIP120
Q4
TIP120
Motor Connector
LCD
LCD CONNECTOR
PUSHBUTTON CONNECTOR
PIC 18F8/J11
Raspberry Pi Connector
JP1
GND



Power
12V
5V
IC1
78L05Z
VI
VO
GND
C7
C8
GND
5V
C9
22uF
L1
47uH
MAX856CSA
R8
R7
LBI
SHDN
REF
LX
OUT
FB
LBO
>'Name'
>'Value'
D1
1N5908
3.3V
R9
164K
C10
22uF
C11
0.1u
R10
100K
GND