EE445L – Lab 6: Introduction to PCB Design

Michael Park and Jack Zhao

03/11/16

1.0 OBJECTIVE

Requirements document

1. Overview

1.1. Objectives: Why are we doing this project? What is the purpose?

The objectives of this project are to design, build and test an alarm clock. Educationally, we are learning how to design and test modular software and how to perform switch input in the background.

1.2. Process: How will the project be developed?

The project will be developed using the TM4C123 board. The system will be built on a solderless breadboard and run on the usual USB power. The system will use four external switches (B3F-1050), and a speaker. There will be at least four hardware/software modules: switch/keypad input, time management, LCD graphics, and sound output. The process will be to design and test each module independently from the other modules. After each module is tested, the system will be built and tested.

1.3. Roles and Responsibilities: Who will do what? Who are the clients?

Mahesh, our TA is the client. Michael will draw the schematic for the hardware and implement it. Michael will also write a software module for "set time", “set alarm”. Jack will write software modules for "turn on/off alarm" and "display". Michael and Jack will design the framework for the entire system, integrate modules, and test the system together.

1.4. Interactions with Existing Systems: How will it fit in?

The system will use the TM4C123 board, a ST7735 color LCD, a solderless breadboard, and be powered using the USB cable.

1.5. Terminology: Define terms used in the document.

Power budget – using power budget, one can estimate the operation time of a battery-powered embedded system by dividing the energy storage by the average current required to run the system.

Device driver – A collection of software routines that perform I/O functions.

Critical section – Locations within a software module, which if an inerrrupt were to occur at one of these locations, then an error could occur.

Latency -A response time of the computer to external events.

Time jitter – Deviation from the true time of data.

Modular programming – A style of software development that divides the software problem into distinct and independent modules.

1.6. Security: How will intellectual property be managed?

The software written for our projects will be kept in our cloud and hard drive.

2. Function Description

2.1. Functionality: What will the system do precisely?

The clock must be able to perform five functions. 1) It will display hours and minutes in both graphical and numeric forms on the LCD. The graphical output will include the 12 numbers around a circle, the hour hand, and the minute hand. The numerical output will be easy to read. 2) It will allow the operator to set the current time using switches. 3) It will allow the operator to set the alarm time including enabling/disabling alarms. 4) It will make a sound at the alarm time. 5) It will allow the operator to stop the sound. An LED heartbeat will show when the system is running.

2.2. Scope: List the phases and what will be delivered in each phase.

Phase 1 is the preparation; phase 2 is the demonstration; and phase 3 is the lab report. Details can be found in the lab manual.

2.3. Prototypes: How will intermediate progress be demonstrated?

A prototype system running on the TM4C123 board, ST7735 color LCD, and solderless breadboard will be demonstrated. Progress will be judged by the preparation, demonstration and lab report.

2.4. Performance: Define the measures and describe how they will be determined.

The system will be judged by three qualitative measures. First, the software modules must be easy to understand and well-organized. Second, the clock display should be beautiful and effective in telling time. Third, the operation of setting the time and alarm should be simple and intuitive. The system should not have critical sections. All shared global variables must be identified with documentation that a critical section does not exist. Backward jumps in the ISR should be avoided if possible. The interrupt service routine used to maintain time must complete in as short a time as possible. This means all LCD I/O occurs in the main program. The average current on the +5V power will be measured with and without the alarm sounding.

2.5. Usability: Describe the interfaces. Be quantitative if possible.

There will be four switch inputs. In the main menu, the switches can be used to activate 1) set time; 2) set alarm; 3) turn on/off alarm; and 4) display mode. In set time and alarm modes, two switches add and subtract hours and the other two add and subtract minutes. After 10 seconds of inactivity the system reverts to the main menu. The display mode switch toggles between graphical and numeric displays. The switches will be debounced, so only one action occurs when the operator touches a switch once.

The LCD display shows the time using graphical display typical of a standard on the wall clock. The 12 numbers, the second hand, the minute hand, and the hour hand are large and easy to see. The clock can also display the time in numeric mode using numbers.

The alarm is a simple 440hz square wave. The sound amplitude is just loud enough for the TA to hear when within 3 feet.

2.6. Safety: Explain any safety requirements and how they will be measured.

The alarm sound will be VERY quiet in order to respect other people in the room during testing. Connecting or disconnecting wires on the protoboard while power is applied may damage the board.

3. Deliverables

3.1. Reports: How will the system be described?

A lab report described below is due by 2/19/16. This report includes the final requirements document.

3.2. Audits: How will the clients evaluate progress?

The preparation is due at the beginning of the lab period on 02/11/16.

3.3. Outcomes: What are the deliverables? How do we know when it is done?

There are three deliverables: preparation, demonstration, and report.

2.0 HARDWARE DESIGN

1) One page description of the battery (printout from the web) (Preparation 2)

Figure 1: Battery

4.8V 3000 mAh NiCD battery pack made of 4 x C high quality NiCd cells.

- Rapid charging at 3A Max.
- Max. Discharging rate: 6 Amp
- Dimension: 4" (L) x 1"(w) x 1.82" (h)
- Weight: 10.5 Oz (300 g)
- 4" length 18 AWG prewired
- Please choose our Universal smart charger (2.4V - 7.2V) to recharge the battery pack. The charging time is about 4 hours
- Ideal for DIY a external battery pack for Fuji digital camera, camera flashlight and any portable device with 5V DC Plug in

2) One page description of the box (like Figure 6.2, Preparation 3)

Dimensions

Length	Width	Height
9.500 in 241.300 mm	6.340 in 161.036 mm	2.500 in 63.500 mm

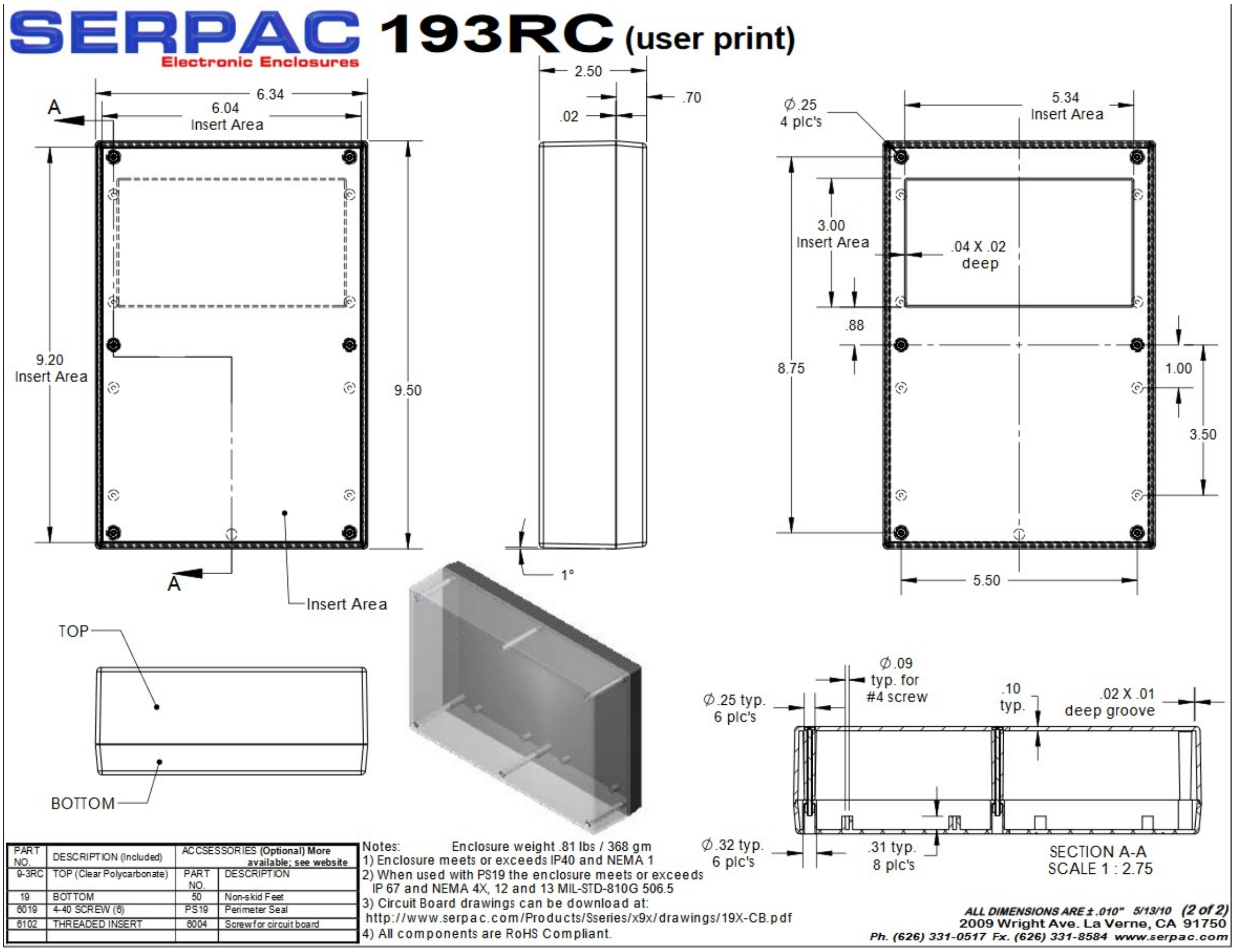

Figure 2: Box Description

3) Three pages showing the new component you created (Procedure 3), and an example PCB using it

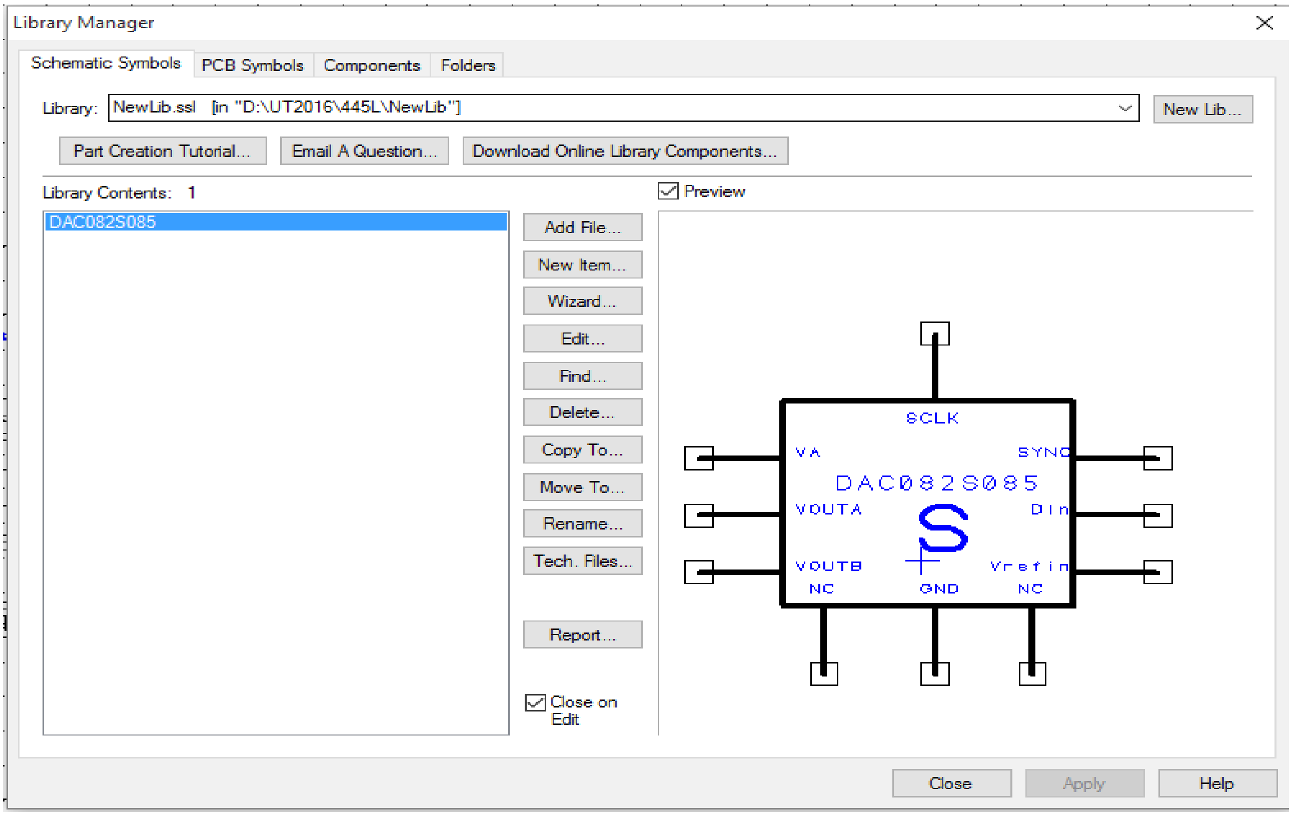

Figure 3: Component Created 1

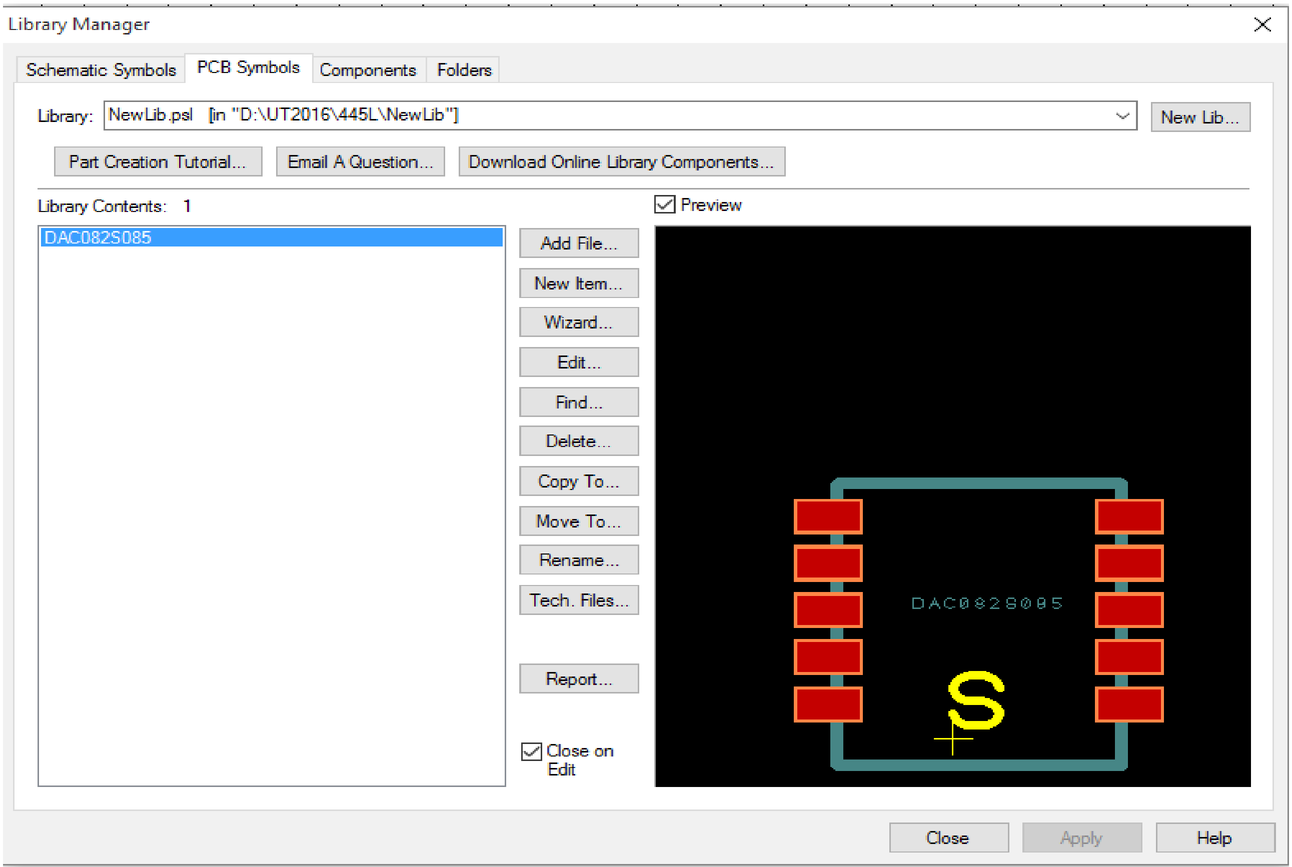

Figure 4: Component Created 2

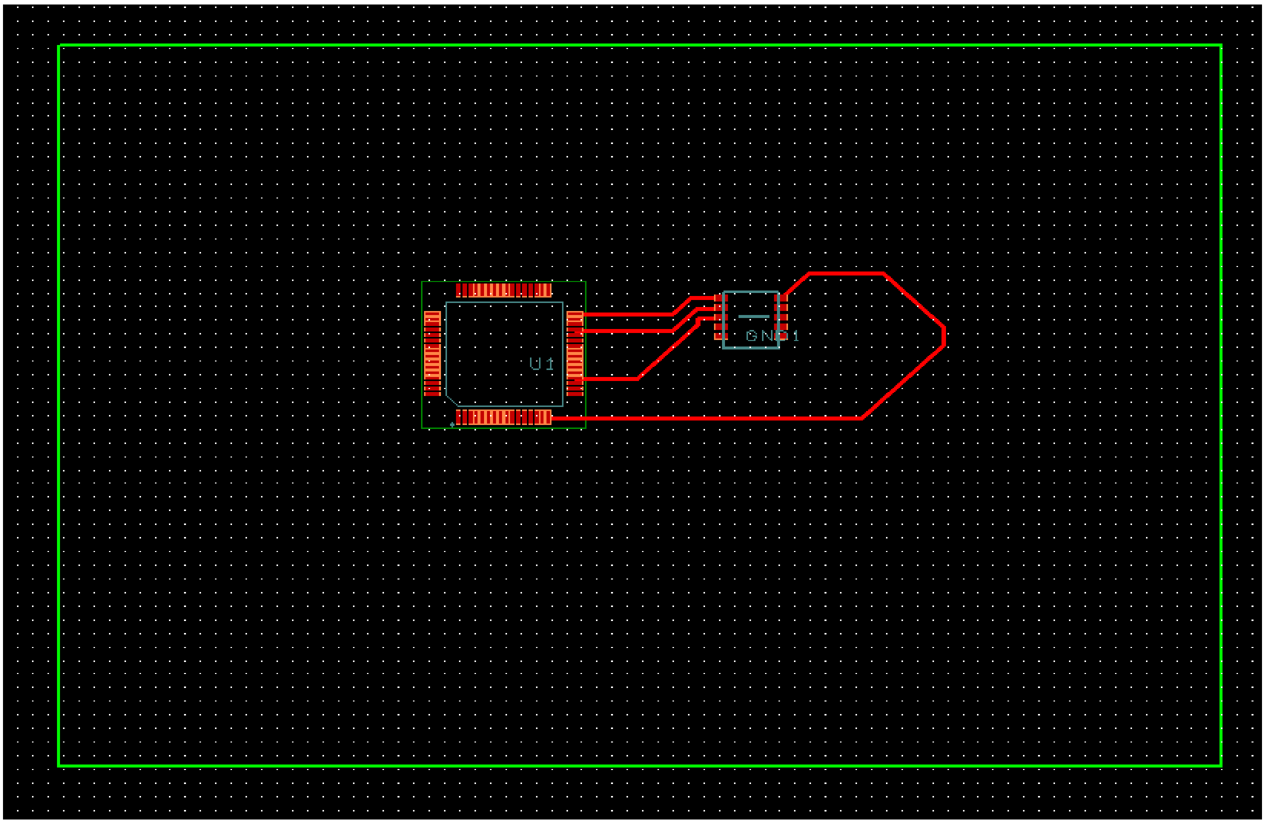

Figure 5: Component Created Example

4) Two mechanical drawings (Procedure 9)

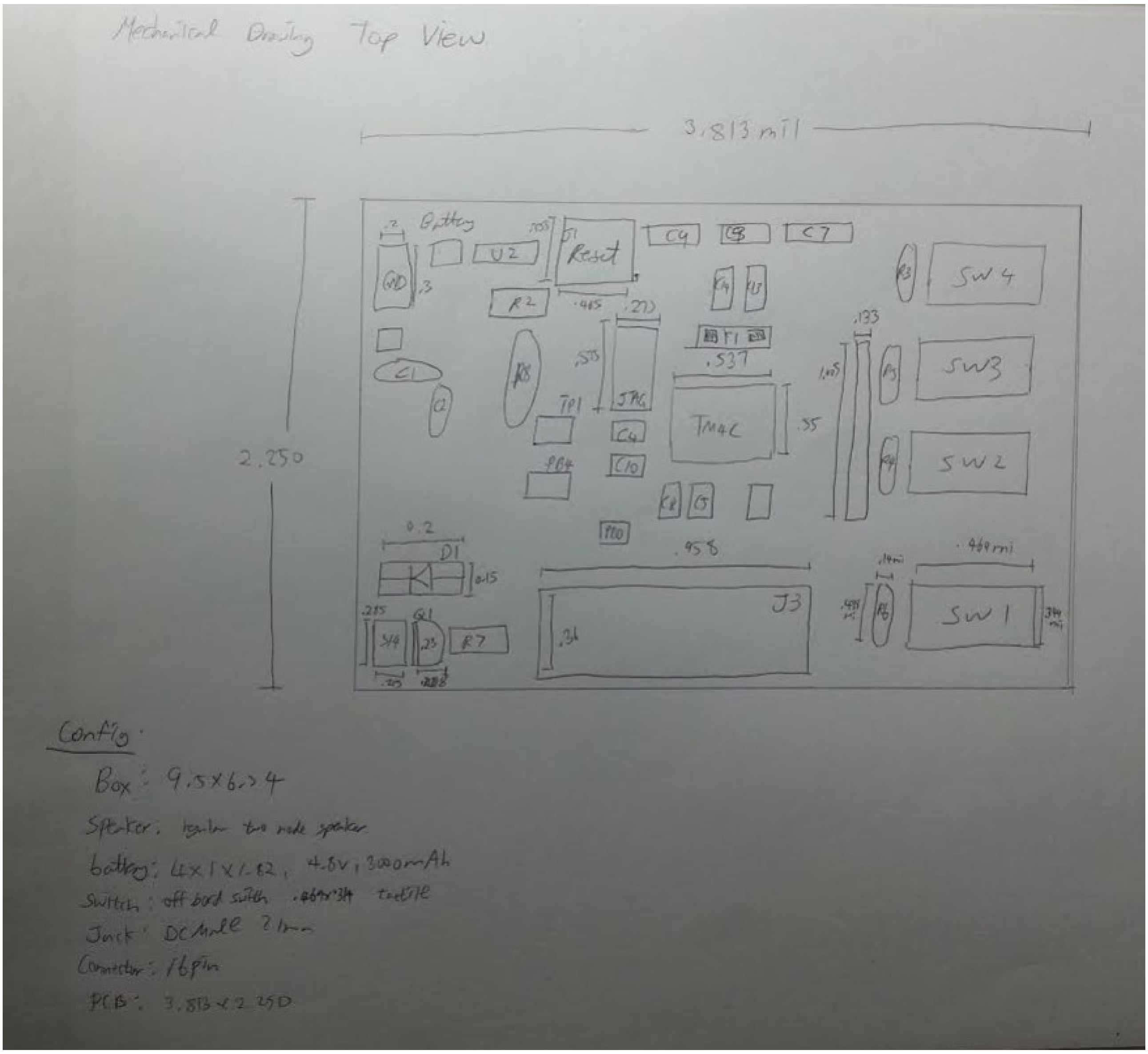

Figure 6: Mechanical Drawing Top View

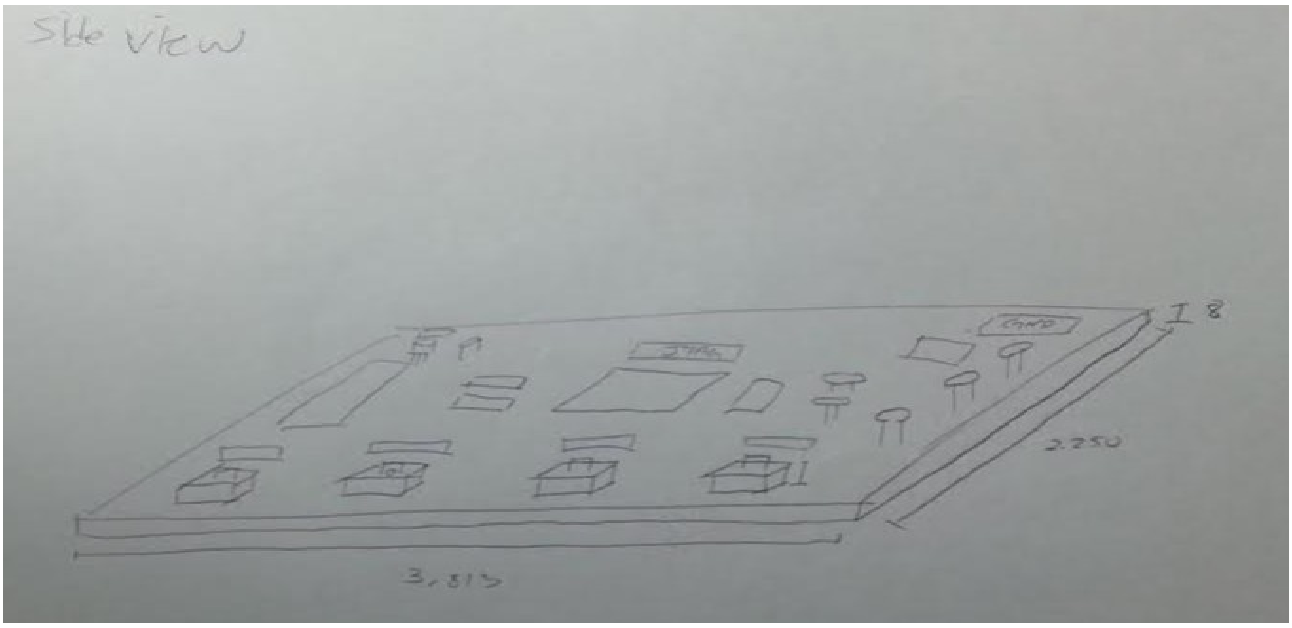

Figure 7: Mechanical Drawing Side View

5) Final circuit diagram of the embedded system, SCH file

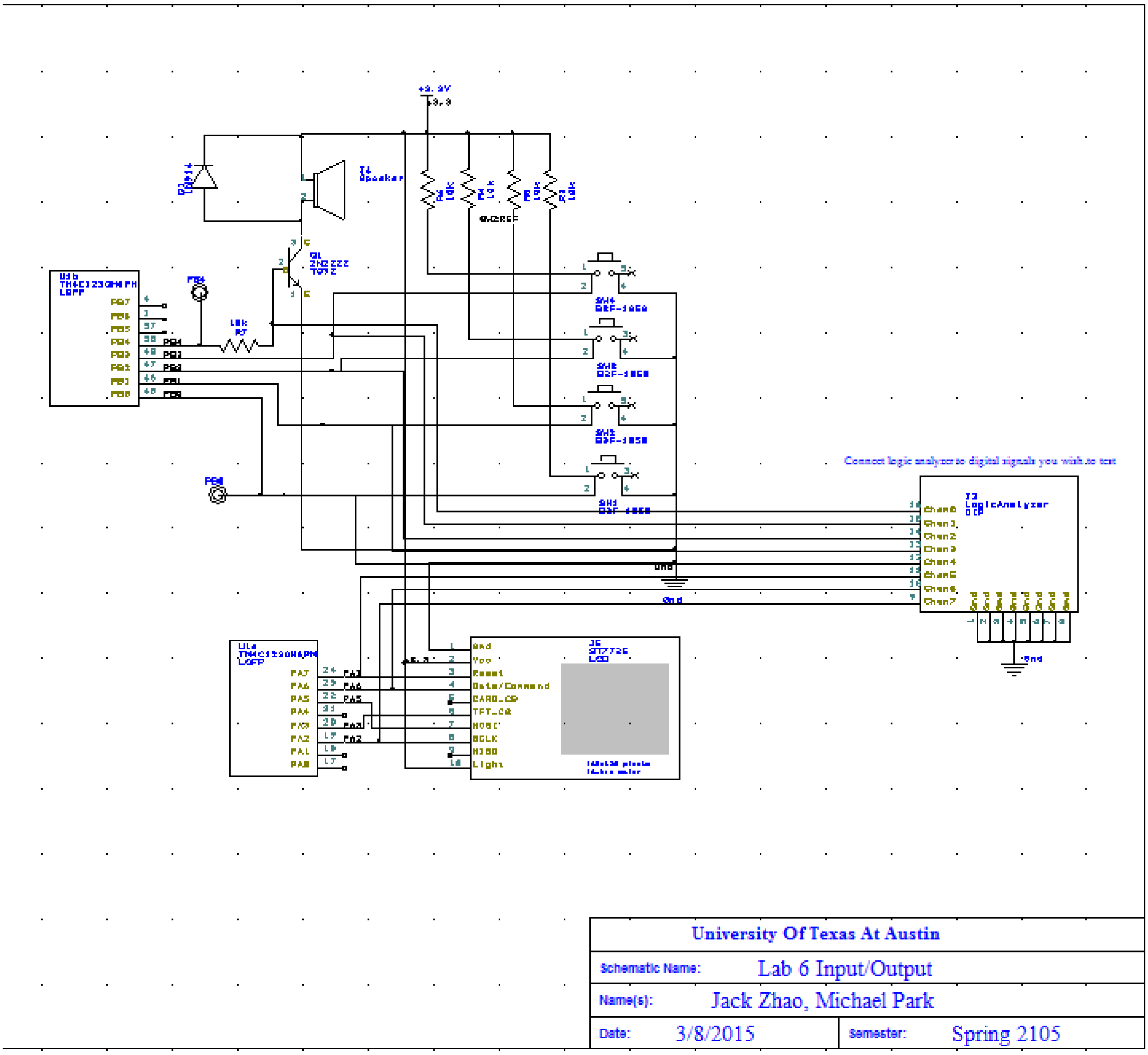

Figure 8: Circuit Diagram

6) Cardboard mockup of the PCB layout (with top copper/silk on top and bottom copper/silk on bottom)

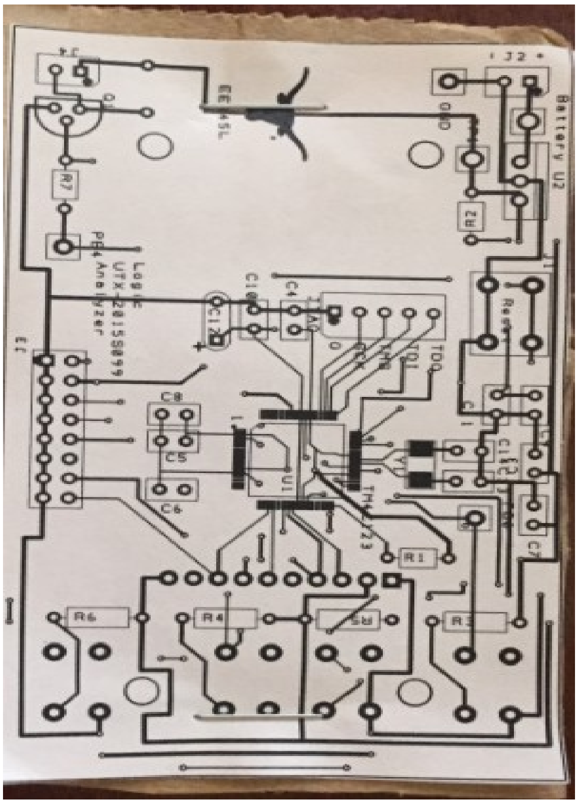

Figure 9: CardBoard Front

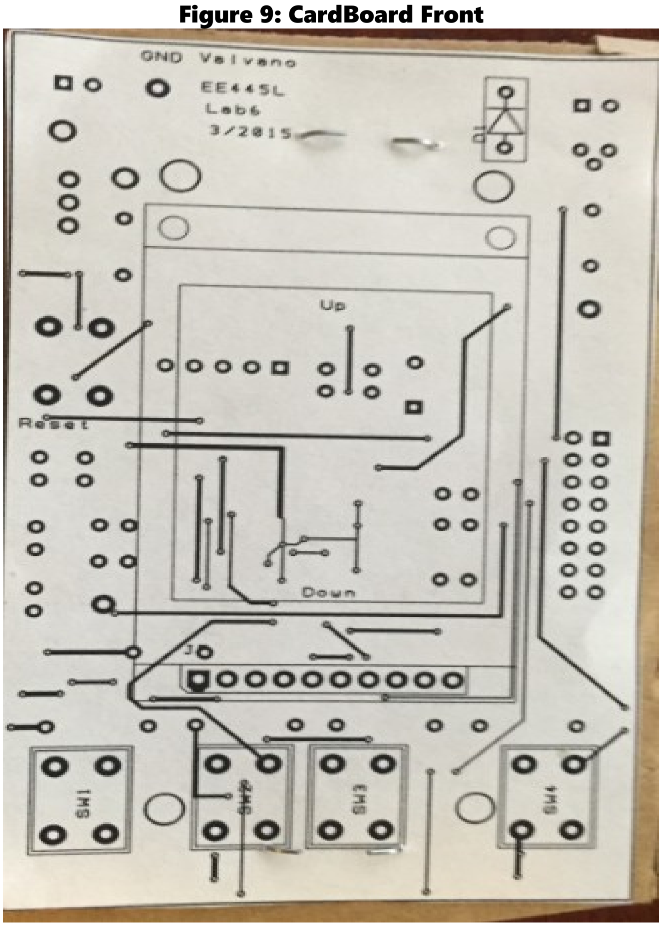

Figure 10: CardBoard Back

3.0 MEASUREMENT DATA

1) Bill of Materials (quantity, package type, cost, and supply current) (Procedure 2)
separately uploaded (battery and box information at the bottom of BOM)

2) Explain how you chose the battery (Preparation 2)
We chose our battery so that it can power our system for at least 24 hours. Our system draws 125mA and the voltage range for our voltage regulator LM2937 is 4.75 – 26V. We wanted a rechargable battery that could run our system for 24 hours. 125*24 = 3000. Therefore we needed a battery of at least 4.75V and 3000mAh. Our 4.8V battery with 3000mAh meets this requriement.

4.0 ANALYSIS AND DISCUSSION (1/2 page maximum)

1) Explain the testing procedure you would suggest for the system (Procedure 1)

1. Connect a scope to test point PB4.
2. Download the software on the microcontroller using JTAG.
3. Run TM4C123 software that generates square wave of 440hz.
4. Read the scope and verify the signal.