EE 445L Lab 11 Report: Picocopter

Reece Stevens and Alex Gerome
rgs835, apg744

December 2, 2016

1 Objectives

Below is the requirements document for picocopter, a very tiny quadcopter.

2 Overview

2.1 Objectives

Why are we doing this project? What is the purpose?

The purpose of this project is to design an embedded system from the ground up. This will give students experience on every part of the design cycle. It also gives students an opportunity to design and print their own PCB, and design their own software system. Our specific project is planned to be a self hovering/stablizing quadcopter.

2.2 Roles and Responsibilities

Who will do what? Who are the clients?

The engineers are Reece Stevens and Alex Gerome, and the TA is the client. The group has modified the provided requirements document to clarify exactly what we plan to build. Our current plans are to work on each part of the lab in tandem.

2.3 Interactions with Existing Systems

How will it fit in?

The system will use the TM4C123 chip, and a specially printed PCB to handle our needs.

3 Function Description

3.1 Functionality

What will the system do precisely?

The system will be a quadcopter. It will be designed to hover and stablize itself in a 3D space. Once turned on, it will levitate to a certain height, and then maintain that position until turned off.

3.2 Performance

Define the measures and describe how they will be determined.

All software must be clear and concise. The PCB and hardware must also be laid out in a manner that makes sense, and minimizes PCB area; this is to reduce the footprint of the system. The system's success will be measured by the stability of the system while hovering, and it's ability to regain stability when disrupted.

3.3 Usability

Describe the interfaces. Be quantitative if possible.

There will be an on/off switch for the system. Once activated, the system will turn off, and the accelerometer will send data to the TM4C chip. The accelerometer will be used to stablize the quadcopter.

3.4 Safety

Explain any safety requirements and how they will be measured.

Since we are dealing with a flying object, the system will not be run in close quarters, and necessary precaution will be taken when people are nearby.

4 Deliverables

4.1 Reports

How will the system be described?

Lab reports for labs 7 and 11 will be written, and necessary source files will be submitted.

4.2 Outcomes

What are the deliverables? How do we know when it is done?

There are three deliverables: preparation, demonstration, and report. The following will be included with Lab 7:

A) Objectives: 1-page requirements document

B) Hardware Design: Regular circuit diagram (SCH file), PCB layout and three printouts (top, bottom and combined)

C) Software Design: Include the requirements document (Preparation a)

D) Measurement Data: Give the estimated current (Procedure d), Give the estimated cost (Procedure e)

E) Analysis and Discussion: (none)

And the following are the contents of the lab 11 report:

A) Objectives: 2-page requirements document

B) Hardware Design: Detailed circuit diagram of the system (from Lab 7)

C) Software Design: (no software printout in the report) Briefly explain how your software works (1/2 page maximum)

D) Measurement Data: Include data as appropriate for your system. Explain how the data was collected.

E) Analysis and Discussion: (none). The YouTube video is required

5 Hardware Design

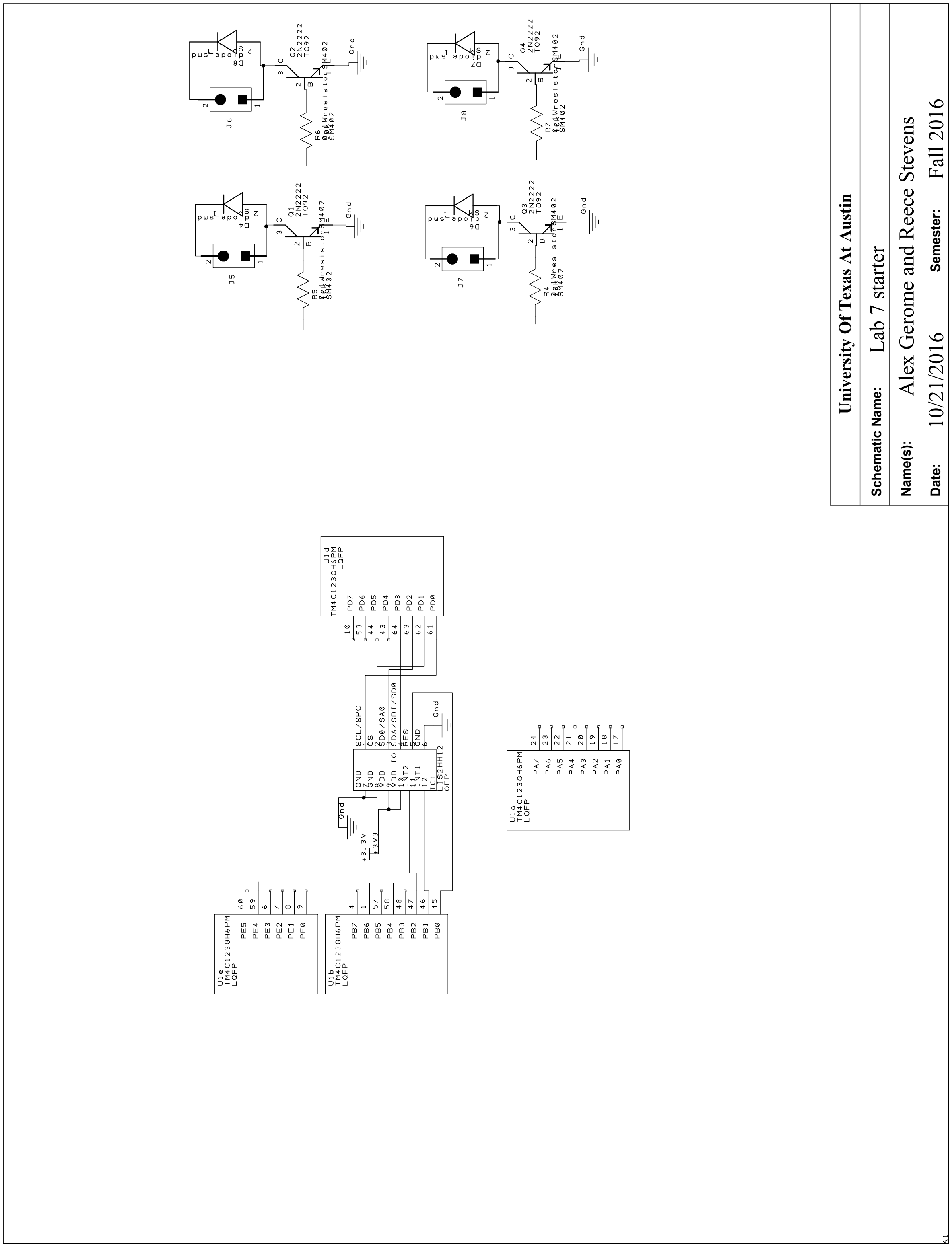

Figure 1: Regular circuit diagram.

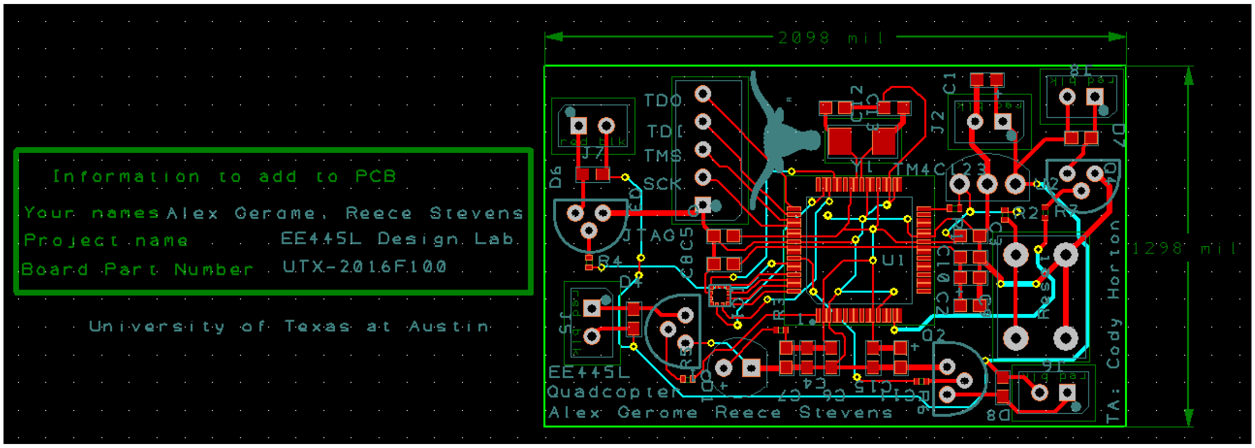

Figure 2: Combined PCB Layout

6 Software Design

The software for picocopter is comprised of an accelerometer module, four motor PWM modules, and P (no ID) controller. At a high level, it is a feedback loop system with the input being motor speeds and feedback information being the accelerometer data. The goal of picocopter to stay afloat means minimizing deviations from 0 on all axes of the accelerometer.

The accelerometer used is an ST LIS2HH12TR, a MEMS 3-axis accelerometer. The low-level registers and basic communication protocols are implemented in `accelerometer.c`. Data collection is triggered by a `DATA_READY` pin on the accelerometer; each read stores the data in an unsafe struct. When reading the data, the main loop will disable the edge triggered interrupt and move the contents of this unsafe struct to a safe global, then re-enable the data ready interrupt. This way, the main function receives only the most up-to-date information from the accelerometer; older samples that are no longer relevant are dropped.

For each of the four motors, a PWM channel is enabled. Each motor is addressable by its `motor_t` identifier, so that the main loop can independently set the duty cycle for each motor using the `set_motor` function.

Finally, the accelerometer information is used to calculate the new motor speeds in the main loop via the P controller. The output of the accelerometer is mapped to the range of the duty cycle settings for each motor, and this value is used to update the motor speeds (see `main.c:p_controller_update` for more implementation details).

7 Measurement Data

The maximum possible estimated current draw from our system is 3.33A, but that assumes all BJTs are open and the motors are at maximum speed; hoowever, this was not fully tested becasue we were never able to flash the board. Since we couldn't flash, we never had all parts soldered on at once to test this. No further perforamce data was collected due to issues in flashing to our board.

The only other data we would need to collect before flying are the calibration constants for our P(ID) controller.

8 Analysis and Discussion

None for this lab. The TA has waived the video requirement for our group due to unexpected errors when constructing the board (we were unable to flash the board due to a crystal error).