EE 445L – Lab 11 (Design Lab)
Justin Nguyen and Trevor Murdock
December 2nd, 2016

1. Overview

1.1. Objectives

Our goal is to design a two player, twin-stick shooter called WIFI DOGFIGHT. Our goal is to make the game have unnoticeable lag, implement peer-to-peer WIFI, have a few modes of gameplay (e.g. dogfight, death match, etc.), have sound effects, nice animations, and add a missile that will be controlled by tilting the controller (whose motion will be detected by the accelerometer). We will make decisions based on the issues of power, timing, data transfer/processing, clock speed, and time to write code.

1.2. Reality

In the end, we didn't get the ESP8266 to work. We got the access point and server running, but were unable to get the connection through. Even after talking to Dr. Valvano, looking at the ESP code from the group that did the walky-talky project (legally of course), and reaching out to those students and Steven Prickett, we were unable to make any further progress.

We did manage to optimize our software design so that there was practically zero lag even with 40-50 objects flying around on the screen. We also got the rest of our hardware working so that our final design had sound effects, nice animations, a missile controlled by the accelerometer, movement controlled by the left thumbstick, and firing direction controlled by the right thumbstick

In terms of power, our system ended up drawing around 130 mA (with the ESP running) and our batteries lasted (roughly) 14 hours. We didn't really have problems with our battery since it was easy to take out and recharge.

1.3. Roles and Responsibilities

- **Justin**
  - Placed the components on the PCB and started initial traces
  - Initialization code (Thumbsticks, ESP, Accelerometer)
  - Testing software
  - ESP Server/Client Code
    - Handshake protocol
  - Game Logic
    - Game clock (time steps)
    - Sound effects/music
  - Enclosure/Parts layout
- **Trevor**
  - Initialization code (DAC/Speaker/Amp, Buttons, LCB)
  - ESP Server/Client Code
    - Data packaging
  - Game Logic
    - Graphics (ships, asteroids, lasers, missiles, etc.)
    - Collision detection (Quadtree)
  - Enclosure/Parts layout

1.4. Interactions with Existing Systems

This project consists of two identical embedded systems communicating indirectly with each other through UART. Players will also interact with our game through the accelerometer, joysticks, buttons, and sound.

2. Function Description

2.1. Performance

Our system was able to successfully do the following: 1) our regulator worked and our board was able to be flashed. 2) we have two independent systems that function individually. 3) we were able to get sound effects to trigger upon firing bullets and missiles. 4) our quadtree had little, if any, memory leaks and was able to be used to detect collisions. 5) our C++ code and overall software design allowed us to make the game design more modular and easily changeable

2.2. Usability

The user interface consists of two analog sticks, an ESP8266 WIFI module, an accelerometer, one game interface button, an ST7735 LCD display, speaker, and a reset button.

3. Deliverables

3.1. Reports

The report for Lab 11 will be written and submitted.

3.2. Outcomes:

We will submit a circuit diagram and PCB layout of our system. We will provide an overview of our software design, current consumption, and cost of the system. We will demonstrate a hardware prototype of the system to Dylan. Additionally, we will include test code and analysis of the system along with discussion as specified in the Lab document.