**Lab 3: ESOS UI Service**

Stuffing, Soldering, and Partial-Build Testing Procedures

Feb. 18, 2021

Eli Lawrence, Erin Parker, Carol Pollard, Kane Simmons, Drew Stork

**Purpose**

The purpose of this document is to outline the steps the team took to test the code written for this lab.

**Process**

The first order of business to prepare for this lab was to set up a GitHub repository to allow for easier collaboration between teammates. To do this, a project folder was created on GitHub for the lab and all group members were invited to work on the repository. Next, all group members set up their Atom text editor to pull from the repository. Then, the repository was populated with all necessary files for this lab and a file was created for the t3\_app application.

According to the requirements laid out in the lab assignment, the functions for the esos\_f14.c file were written, followed by the header file, and lastly, the t3\_app file. The threshold for the code to report if the encoder was turning slow, medium, and fast were agreed upon by the team in a group discussion.

**Testing**

The testing procedure for the written code began with compiling all files that were modified by the team. Once the compilation was able to complete without any errors, a code review was conducted by the team to catch any errors or missing comments. After this was done, the target board was programmed and the functionality of the code was tested.

The board testing started with verifying the functionality of each of the options in the user interface. After this was completed, the encoder was turned slowly, quickly, and at a medium speed to check whether or not the user interface correctly reported the speed of the rotation.

As a side note, the encoder on the target board is of poor quality and therefore unable to provide the most accurate values to the code. As a result, even though the encoder may be turning a certain speed, the software may not report it as turning at the speed that it actually is.