**RSS: Thruster Testing Apparatus Operation Manual**

This report will detail the necessary steps to operate the thruster testing apparatus designed and built by the 2021/2022 capstone team. Software required to use the testing apparatus include Arduino IDE and Python 3 (preferably with an editor such as VS Code). Due to the use of high-pressure air, it is recommended that the operators where safety glasses while the air is being actuated.

**It is important to note that the laptop connected to the Arduino should not be plugged into an external power source while the user is reading data from the load cell. This reduces noise in the load cell readings.**

**Do not unscrew the nozzle from the load cell while it is mounted to the base plate. This can damage the transducer. Please remove the load cell from the base plate and hold the nozzle mount while switching the nozzle.**

**Apparatus:**

* RSS thruster testing apparatus (see last page for parts list)
* Precision masses
  + 2 grams to 400 grams (if available, see below)
* LiPo battery
* Micro-USB cable

**Procedure:**

**1) Pretesting**

Record initial supply pressure of the air cannister.

|  |  |
| --- | --- |
| **Initial Supply pressure (psi)** |  |

Before testing the pneumatics with the air supply, each transducer should be tested and verified as well as the solenoid valves, according to the procedure below.

**Calibration of the load cell**

The load cell must be calibrated to ensure the transducer is operating accurately and as error-free as possible. To calibrate the load cell, a minimum of two masses must be used. Using more masses will increase the accuracy of the calibration, as linear regression is used to fit the load cell signal to the mass values on a linear calibration curve. Optimal calibration would use masses ranging from 0% to 100% of the full scale, increasing in small increments, and then 100% to 0% of the full scale decreasing in the same increments. Since the thruster assembly has associated mass, and to ensure that the maximum capacity of the load cell is not exceeded, the following calibration masses are recommended for a 500 g load cell (adjust as necessary for a smaller or larger load cell).

Table . Example calibration procedure for a 500 g load cell.

|  |  |
| --- | --- |
| Calibration Trial | Mass (g) |
| 1 | 2 |
| 2 | 5 |
| 3 | 20 |
| 4 | 50 |
| 5 | 100 |
| 6 | 200 |
| 7 | 400 |
| 8 | 200 |
| 9 | 100 |
| 10 | 50 |
| 11 | 20 |
| 12 | 5 |
| 13 | 2 |

The number of masses used can be adjusted as necessary to decrease the time to calibrate or account for what masses the user has available. What is important is that the full range of the load cell is calibrated.

The load cell would ideally be calibrated prior to each use however it is only necessary if the load cell has been removed from its mount, or if it has been over a year since its last use (at a minimum). The following steps outline the procedure for calibrating the load cell.

1. Mount the load cell to the calibration stand on the base plate
2. Ensure the thruster tube is not touching the base plate and thus imparting a force on the load cell
3. Connect the Arduino to the user’s laptop via micro-USB cable
4. Run RSS\_CalibrateAndReadLoadCell.ino
   1. Enter “c” in Arduino serial monitor to begin calibration
   2. Input number of calibration trials. The example in Table 1 above has 13 trials. At a minimum, 2 trials are required with each trial testing a different mass.
   3. Place masses on load cell
   4. Input known mass value
   5. Record data points (a and b) values generated from RSS\_CalibrateAndReadLoadCell.ino
      1. These will be hard coded into RSS\_RunExperiment.py later
5. Unmount the load cell from the calibration stand
6. Mount the load cell to the load cell bracket in testing orientation

Table . Calibration values obtained from calibration procedure.

|  |  |
| --- | --- |
| **Parameter** | **Value** |
| A | 0.000262 |
| B | -0.064279 |

**Verification of solenoid valves**

In addition to testing the transducers, actuation of each solenoid valve should be performed to ensure functionality prior to adding air to the system. The following steps outline the procedure for testing and verifying the solenoid valves.

1. Plug LiPo battery into solenoid valve circuit board
2. Run RSS\_CalibrateAndReadLoadCell.ino
   1. Enter “r” in Arduino serial monitor
   2. Enter a number 1 – 4 to actuate desire solenoid vales
      1. Actuate all 4
3. Ensure all valves are operational by listening for a clicking sound

Table . Verification of the solenoid valves

|  |  |
| --- | --- |
| **Solenoid Valve** | **Operational (Y/N)** |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |

**Verification of the air supply pressure**

The following steps outline the procedure for testing and verifying the pressure of the air.

1. Run RSS\_CalibrateAndReadLoadCell.ino
   1. Enter “r” in Arduino serial monitor
   2. Record the pressure readout with no supply pressure
2. Turn on air supply
   1. While air supply is on, the pressure transducer script will read out the pressure data continuously
   2. Confirm desired pressure is being supplied to system
   3. Adjust second stage regulator as necessary to achieve desired inlet supply pressure
3. Turn off Air supply
4. Adjust pressure reading according to readout when no air was supplied

Table . Comparison of analog pressure gauge and pressure transducer readings.

|  |  |
| --- | --- |
| **Measuring Instrument** | **Readout (psi)** |
| Pressure Transducer (no supply pressure) |  |
| Pressure Transducer (desired value) |  |

**2) Testing**

The following steps are to be performed upon completion of the pretesting requirements.

1. Open RSS\_RunExperiment.py on computer
   1. Hardcode calibration values A and B recorded from load cell calibration into RSS\_RunExperiment.py (initiated as aCal and bCal in Python script)
   2. Update calDate to reflect when aCal and bCal values were entered
2. Load RSS\_ControlTransducers.ino onto Arduino from computer
3. Run RSS\_RunExperiment.py on computer
   1. Input number of solenoids to actuate
   2. Input duration for data collection prior to actuation of solenoids
   3. Input duration for actuation of solenoids
4. Wait for test to complete
   1. While testing is occurring, the raw data from the load cell is being stored in a csv file in RSS\_RunExperiment.py
   2. A second csv file will read the raw data, process it, and generate a plot of force (grams) versus time (seconds)
5. Record observations
6. Repeat testing as necessary
   1. Press reset button on Arduino
   2. Restart RSS\_RunExperiment.py code on computer
7. Close tank valve for air supply

Observations

|  |
| --- |
|  |

**3) Post Testing**

Record final supply pressure of the air cannister.

|  |  |
| --- | --- |
| **Final Supply pressure (psi)** |  |

The following steps are to be completed after all testing is done.

1. Open all solenoid valves to ensure there is no pressure left in the system
   1. Run RSS\_CalibrateAndReadLoadCell.ino and actuate each valve
   2. Unplug LiPo battery after each valve has been actuated
2. Verify there is no pressure by taking a reading from the pressure transducer
   1. Run RSS\_CalibrateAndReadLoadCell.ino and confirm a zero read out
3. Unplug arduino

**Parts List**

All parts will be mounted on the base plate (except for the LiPo battery).

Diagram, engineering drawing

Description automatically generated

|  |  |
| --- | --- |
| **Components​** | |
| 1​ | Air Supply Tank​ |
| 2​ | Second Stage Regulator​ |
| 3​ | Solenoid Valves​ |
| 4​ | Manifold​ |
| 5​ | Measured Thruster Nozzle Coupling​ |
| 6​ | Load Cell (testing orientation)​ |
| 7​ | Load Cell Calibration Stand​ |
| 7a​ | Load Cell (calibration orientation)​ |
| 8​ | Three Thruster Nozzle Coupling​ |
| 9​ | Circuit Board​ |
| 10​ | Base​ |
| 11​ | Pressure Transducer​ |