A yellow and brown logo

Description automatically generated

Motor Thrust Stand  
User and Service Manual

Contents

[Hardware 2](#_Toc185251116)

[Processing 2](#_Toc185251117)

[Sensors 2](#_Toc185251118)

[Setup 2](#_Toc185251119)

[Installation 2](#_Toc185251120)

[Un-installing 2](#_Toc185251121)

[User Interface 3](#_Toc185251122)

[Launcher 3](#_Toc185251123)

[Lab Mode 4](#_Toc185251124)

[Dyno Mode 5](#_Toc185251125)

[Configuration 6](#_Toc185251126)

[Serial 6](#_Toc185251127)

[Logging 7](#_Toc185251128)

[Load Cell 8](#_Toc185251129)

[Airspeed Sensors 9](#_Toc185251130)

[Electrical 10](#_Toc185251131)

[Usage 11](#_Toc185251132)

[Startup 11](#_Toc185251133)

[Shutdown 11](#_Toc185251134)

[Calibration 12](#_Toc185251135)

[Requirements 12](#_Toc185251136)

[Voltage Divider 12](#_Toc185251137)

[Current Sensor 14](#_Toc185251138)

[Appendix 16](#_Toc185251139)

[Errata 16](#_Toc185251140)

[Datasheets 16](#_Toc185251141)

# Hardware

Datasheets for each component are included in the Appendix

## Processing

Raspberry Pi 5

Teensy 4.0

## Sensors

Voltage Divider – A simple variable voltage divider nominally calibrated to 10:1.

Differential Pressure Sensors – Two MP3V5010DP differential pressure sensors connected to a pair of pitot-static tubes placed in the incoming and wake airflow.

Load Cell ADC – HX711 24-bit load cell amplifier breakout board from Spark Fun.

Current Sensor – One ACS758ECB-200U current sensor measuring up to 200 amperes of continuous current and capable of handling 1200 amperes of current in bursts of up to 1 second.

# Setup

## Installation

The software is provided with an installation script which will install all necessary packages and libraries. It will compile the relevant class files, download the WMU logos found on <https://wmich.edu/brand/visualidentity/logo> and create a desktop shortcut for the thrust stand software.

1. Download the repository from GitHub: <https://github.com/EvEdSchob/MotorThrustStand>
   1. Clone the repository using a git client or download it as a zip file
   2. If it was downloaded as a zip extract the file to a local folder on the Raspberry Pi
2. Navigate to the directory: MotorThrustStand/PiCode/ThrustStand in a terminal window
3. Run the command below to make the script executable  
   sudo chmod +x thruststand-install.sh
4. Enter the command below to start the installation process  
   sudo ./thruststand-install.sh
5. If the installer runs successfully the application can be executed in three different ways:
   1. Run the command thruststand in the terminal window. This will provide debugging output to the terminal.
   2. Click the shortcut icon on the desktop
   3. In the Raspberry Pi OS menu navigate to Accessories > ThrustStand

## Un-installing

To uninstall the software simply run: sudo ./thruststand-install.sh --uninstall

# User Interface

## Launcher

* This is the first screen shown to the user when the software launches

A screenshot of a computer

Description automatically generated

Figure : Launcher Screen

* Lab Mode – Launches the Lab Mode Screen
* Dyno Mode – Launches the Dyno Mode Screen
* Configure – Launches the Configuration Screen
* Exit – Closes the program

## Lab Mode

A screenshot of a computer

Description automatically generated

Figure : Lab Mode Screen

* Thrust – Displays the live thrust value in the selected unit
  1. Tear/Zero – “Zeroes” the thrust measurement value
  2. Hold – Freezes the user display for easy reading
  3. Units – Selects the unit of thrust measurement: Newtons, Pounds, Kilograms
* Current – Displays the live current value in Amperes
* Voltage – Displays the live voltage value in Volts
* Airspeed (Before) –
* Airspeed (After) –
* Set RPM
  1. Blades – Sets the number of blades on the propeller
  2. RPM – Sets the RPM setpoint for the PID controller on the Teensy
* Motor Start/Stop – Enables and Disables commands to the speed controller. Sets throttle to zero when disabled.
* Logger Start/Stop – Activates and Deactivates data logging. Creates a new file on disk each time it is enabled and finalizes said file when disabled.

## Dyno Mode

A screenshot of a computer

Description automatically generated

Figure : Dyno Mode Screen

* Most controls are identical to those found in Lab Mode. Refer to the Lab Mode section for details of each shared control.
* Throttle – The slider directly controls the throttle percentage between 0% and 100%
* RPM – Displays the live RPM measurement

## Configuration

* Save Calibration – Stores calibration constants for all sensors in a local file
* Load Calibration – Retrieves calibration constants from the data file
* Reset to Defaults – Loads default calibration constants from the hard-coded values
* Back – Exits the calibration screen

### Serial

A screenshot of a computer

Description automatically generated

Figure : Configuration Screen - Serial Tab

* The configuration window automatically opens in the serial tab
* Ports Dropdown – Lists the available serial devices
* Refresh – Refreshes the list of serial devices
* Connect – Attempts to establish connection with the selected serial device

### Logging

A screenshot of a computer

Description automatically generated

Figure : Configuration Screen - Logging Tab

* Save Location – Shows the path to the location where logging files will be saved
* Browse – Launches a directory chooser window to select where the log file will be saved
* Current log file: - Displays the name of the log file to be saved

### Load Cell

A screenshot of a computer

Description automatically generated

* Known Weight – Enter the current weight for calibration purposes
* Unit Dropdown – Selects the unit of the weight being used for calibration
* Calibrate – Calculates the new calibration constant for the load cell
* Current Calibration – Displays the current calibration constant

### Airspeed Sensors

A screenshot of a computer

Description automatically generated

* Unit Dropdown – Selects the unit for airspeed calibration
* Known Airspeed – Textboxes for entering the externally measured air speed
* Calibrate – Buttons initiate the calibration constant calculation

### Electrical

A screenshot of a computer

Description automatically generated

* Known Current – Textbox for entering the externally measured current value
* Known Voltage - Textbox for entering the externally measured voltage value
* Calibrate – Buttons initiate the calibration constant calculation

# Usage

## Startup

1. Turn on the power strip under the wind tunnel to power the device
2. Wait for the system to start up and come to the desktop
3. Launch the software:
   1. Run the command thruststand in the terminal window. This will provide debugging output to the terminal.
   2. Click the shortcut icon on the desktop
   3. In the Raspberry Pi OS menu navigate to Accessories > ThrustStand
4. Press the **Configure** button on the main launcher window.
5. In the **Serial** tab, use the drop-down menu to select the correct serial device then press the **Connect** button.
   1. If no device is found ensure that the USB cable is properly connected between the Teensy and the user interface then press the **Refresh** button.
6. (Optional) Press the **Load Calibration** button to load the previously saved calibration values. Skipping this step will cause the software to use the default calibration values from the software.
   1. If the system needs to be recalibrated, please see the calibration instructions on page 6.
7. (Optional) Switch to the **Logging** tab and browse the filesystem for a location to save the logging file. If no alternate location is selected, files will be saved to the user’s home folder.  
   NOTE: A log file will only be created if the **Logger Start/Stop** button is pressed within either Lab or Dyno mode.
8. Press the **Back** button to return to the launcher.
9. Press the **Lab Mode** or **Dyno Mode** button to enter the desired mode.

## Shutdown

1. From either mode or the Configuration screen press the **Exit** button to return to the Launcher.
2. Press the **Exit** button on the launcher to close the ThrustStand software.
3. In the Raspberry Pi OS Menu select Shutdown… > Shutdown to halt the Raspberry Pi.
4. Turn off the power strip under the thrust stand to fully power the system down.

# Calibration

## Requirements

Before beginning calibration ensure that all of the hardware listed below is available and in working order.

* Mouse
* Keyboard
* Multimeter with DC Voltage, Resistance, and Current modes
* Current Calibration Harness
* Anemometer
* Tachometer

## Voltage Divider

1. Disconnect the USB Cable from the Teensy microcontroller in the sensor package.
2. Open sensor package case
3. Remove Teensy microcontroller
   1. This prevents damage if the input voltage is above 3.3v and the trim potentiometer is adjusted incorrectly.
4. Attach a battery or other power source to the input of the sensor package
5. Measure the input voltage by placing the positive probe of a multimeter on one leg of the current sensor and the negative probe on the ground terminal as shown in Figure 6. Record this voltage.



Figure : Input Voltage Measurement

1. Measure the output from the voltage divider by placing the positive probe of the multimeter on TP3 on the circuit board and the negative probe in the same location described in step 4. Record this voltage.

A hand holding a screwdriver to a green circuit board

Description automatically generated

Figure : Voltage Divider Measurement

1. For a nominal configuration, (10:1 divider ratio) adjust the blue trim potentiometer so that the output voltage is one tenth of the input voltage.
   1. E.g. An input voltage of 12.19 volts results in a TP3 voltage of approximately 1.219 volts as shown below  
      A diagram of a circuit

      Description automatically generated
   2. This will allow voltages of up to 33v to be safely measured by the device.
   3. Higher voltages may be measured as necessary by re-calibrating the potentiometer for a higher divider ratio.
2. Once the trim potentiometer has been correctly adjusted, place the Teensy back into its slot with the USB port facing out.
3. Reconnect the USB cable to the micro USB port on the Teensy and one of the ports on the Pi
4. Start up the user interface and enter the **Configuration** menu.
5. In the **Serial** tab select the correct serial device and press the **Connect** button.
6. Switch to the **Electrical** tab.
7. Enter the measured battery voltage into the **Known Voltage** field and press the **Calibrate** button. This will cause the code to calculate the new divider ratio.   
   **NOTE: Please verify that the calculated value matches the expected value before continuing.**
8. Press the **Save Calibration** button to save to the local file.

## Current Sensor

1. Start the User Interface, start the ThrustStand software and enter the Configuration mode.
2. In the **Serial** tab select the correct serial device from the dropdown and press the **Connect** button.
3. Use a multimeter in voltage mode to record the output voltage of the power source.
4. Measure the resistance of the Current Calibration Harness by connecting it to a multimeter in resistance mode with the shorting plug inserted.  
   **NOTE: It should be close to 10 ohms.**

A digital multimeter with wires

Description automatically generated

Figure : Current Harness Testing

1. Remove the shorting plug from the calibration harness and set it aside.
2. Connect the current harness to the output wire from the sensor package.

A yellow electronic device with wires

Description automatically generated

Figure : Current Calibration

1. Connect the power source to the input wire on the sensor package.   
   **NOTE: Do not leave the system connected in this configuration for extended periods of time or the resistor will become dangerously hot!**
2. Enter the current value shown on the multimeter into the **Known Current:** field in the user interface and press the **Calibrate** button.
3. Press the **Save Calibration** button to save to the local file.
4. Disconnect the power source or the current calibration harness.

# Appendix

## Errata

1. External power is not needed for the sensor package board. The Teensy supplies sufficient voltage out through the Vin when receiving USB power to supply both regulators such that all onboard components are supplied correctly. Powering the board from the external source will cause both the 5v regulator and the Teensy to overheat.
2. There is a wire soldered to the bottom of the PCB which provides a signal ground for the voltage measurement. If this wire breaks or is removed voltage measurement will be erratic.
3. The MP3V5010DP sensor range is too great for the sensitivity of the Teensy’s ADC. These sensors have a maximum pressure differential of 1.45psi. The calculated resolution of the system is as follows:

* 2.21 meters per second
* 4.96 miles per hour
* 7.25 feet per second
* 7.95 kilometers per hour

This is the theoretical limit for the smallest measurements that can be made by the pitot system. More precise measurements would require a sensor with a narrower maximum range and/or an ADC with a higher bit-depth.

1. At present, no method exists to calibrate the RPM measurement from within the user interface.
2. An established method for calibrating the load cell has not been developed.
3. The Tear/Zero button in both Lab and Dyno mode temporarily resets the display back to the value “0” but does not “zero” the value of the load cell measurement.

## Datasheets

Datasheets for all sensors will be provided on the subsequent pages.