AEROMOBILE REGENERATIVE SUPERCIRCULATION TEST STAND (ARSTS)

A Thesis Presented to

The Faculty of the

Fritz J. and Dolores H. Russ
College of Engineering and Technology
Ohio University

In Partial Fulfillment
of the Requirement for the Degree

Master of Science

by

Jason J. Fink

November, 2004
# Table of Contents

List of Tables .................................................................................. v

List of Figures .................................................................................. vi

Chapter 1

1 Introduction ....................................................................................... 1
  1.1 Evolving Transportation Needs .................................................. 2
  1.2 The Flying Car ........................................................................... 4
  1.3 ARSTS Specifications ................................................................. 6
  1.4 Literature Review ....................................................................... 7
  1.5 Thesis Final Objective ............................................................... 11

Chapter 2

2 Developing the ARSTS ................................................................. 12
  2.1 ARSTS Design .......................................................................... 13
    2.1.1 Measurement Equipment .................................................. 13
    2.1.2 Conceptual Stand Design ............................................... 20
    2.1.3 Stability Analysis ............................................................ 24
    2.1.4 Deflection Analysis .......................................................... 27
  2.2 ARSTS Fabrication ................................................................... 28
    2.2.1 Preliminary Construction ................................................ 29
    2.2.2 Trouble Shooting ........................................................... 33
    2.2.3 Final ARSTS Fabrication ............................................... 37

Chapter 3

3 Operating the ARSTS ................................................................. 40
  3.1 Stand Computer Codes ............................................................ 41
  3.2 Stand Calibration ...................................................................... 42
  3.3 Bias Investigation ..................................................................... 49
  3.4 Experimentation by the User ................................................... 60
  3.5 Vehicle Testing ........................................................................ 61

Chapter 4

4 Conclusions and Recommendations ........................................... 65
<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bibliography</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>Appendix A</td>
<td></td>
<td>71</td>
</tr>
<tr>
<td>ASU Facility Specifications</td>
<td></td>
<td>72</td>
</tr>
<tr>
<td>Appendix B</td>
<td>ARSTS Design Specifications</td>
<td>73</td>
</tr>
<tr>
<td>Appendix C</td>
<td>ARSTS Matlab Programs And Users Manual</td>
<td>74</td>
</tr>
<tr>
<td>Appendix D</td>
<td>Matlab Script Code for ARSTS Matlab Programs</td>
<td>75</td>
</tr>
<tr>
<td>Appendix E</td>
<td>JR3 PCI Software – how to install Win98</td>
<td>76</td>
</tr>
<tr>
<td>Appendix F</td>
<td>MATJR3PCI for MATLAB 5</td>
<td>83</td>
</tr>
<tr>
<td>Appendix G</td>
<td>Additional Stand Calibration Data and Figures</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>101</td>
</tr>
<tr>
<td></td>
<td></td>
<td>102</td>
</tr>
<tr>
<td></td>
<td></td>
<td>104</td>
</tr>
<tr>
<td></td>
<td></td>
<td>105</td>
</tr>
<tr>
<td></td>
<td></td>
<td>112</td>
</tr>
<tr>
<td></td>
<td></td>
<td>113</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Tables</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2-1</td>
<td>17</td>
</tr>
<tr>
<td>Table 2-2</td>
<td>19</td>
</tr>
<tr>
<td>Table 3-1</td>
<td>46</td>
</tr>
<tr>
<td>Table 3-2</td>
<td>55</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1-1</td>
<td>7</td>
</tr>
<tr>
<td>Figure 1-2</td>
<td>8</td>
</tr>
<tr>
<td>Figure 2-1</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2-2</td>
<td>15</td>
</tr>
<tr>
<td>Figure 2-3</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2-4</td>
<td>20</td>
</tr>
<tr>
<td>Figure 2-5</td>
<td>19</td>
</tr>
<tr>
<td>Figure 2-6</td>
<td>21</td>
</tr>
<tr>
<td>Figure 2-7</td>
<td>23</td>
</tr>
<tr>
<td>Figure 2-8</td>
<td>29</td>
</tr>
<tr>
<td>Figure 2-9</td>
<td>30</td>
</tr>
<tr>
<td>Figure 2-10</td>
<td>31</td>
</tr>
<tr>
<td>Figure 2-11</td>
<td>32</td>
</tr>
<tr>
<td>Figure 2-12</td>
<td>34</td>
</tr>
<tr>
<td>Figure 2-13</td>
<td>35</td>
</tr>
<tr>
<td>Figure 2-14</td>
<td>36</td>
</tr>
<tr>
<td>Figure 2-15</td>
<td>38</td>
</tr>
<tr>
<td>Figure 2-16</td>
<td>39</td>
</tr>
<tr>
<td>Figure 3-1</td>
<td>43</td>
</tr>
<tr>
<td>Figure 3-2</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-3</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-4</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-5</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-6</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-7</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-8</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-9</td>
<td>..................................................</td>
</tr>
<tr>
<td>Figure 3-10</td>
<td>..................................................</td>
</tr>
</tbody>
</table>
1. Introduction

This thesis will address the need to create a test stand that can measure the aerodynamic performance of non-conventional lift-generating vehicles and test subjects. These subjects have been and will continue to be created for the purpose of experimenting with new and radical means of generating lift, the most notable of which is supercirculation. The ultimate goal is to implement the positive results of this research in the Ohio University flying car project, or Aeromobile so that a feasible new means of intermediate distance travel can be made available to the modern commuter.

Chapter one will provide a brief review of the automobile and airplane and how their technology is starting to become obsolete in meeting the needs of the commuter. It will then provide a brief introduction to the flying car, a new technology possessing the ability to meet the needs of the commuter and how the test stand developed within this thesis is critical to developing the technology. A literature review is also presented to guide the development of the stand with respect to desired performance and functionality. The chapter concludes with specific goals and design requirements for the test stand so that the research it enables will benefit the flying car project to the best extent possible.

Chapter two of this thesis is divided into two main sections each containing multiple subsections. The first section covers the conceptual and analytical design process and explains the affects of available materials, electronics, fabrication
processes and the desired performance on the stand's design. The final conceptual design is then supported with a stability and deflection analysis. The second section highlights the initial stand construction and the resolution of unexpected problems. The chapter then concludes with the stand accessories and final assembly in good working order ready for calibration.

The calibration of the stand, proper operation and guidance for the user are provided in Chapter three. The chapter begins with the calibration of the stand and the investigation of overshoot and bias. The results of the investigation are then used to guide the development of the computer codes and stand operation as presented to the user. The chapter concludes with the presentation of the stand in final working order and contains a test run with one of the Aeromobile's quarter scale engines in order to validate the stand.

1.1 Evolving Transportation Needs

Throughout history people have been making improvements to every aspect of everyday life. The need to travel and transport one's property from one place to another is no exception to this trend. Achieving the highest travel speed in a safe and cost effective manner has spawned the invention, development and mass production of all types of transportation vehicles with the most recent being the airplane. Unlike the train, boat or automobile, air transport is free from many of the limitations imposed by the natural landscape of the earth. Although weather and
Mach number sometimes limit the top speed and destination to which one travels, aircraft have become the best means of traveling long distances in the shortest amount of time.

With a diverse market of passengers and specific transport needs, thousands of aircraft in all different shapes and sizes have been developed over the years to facilitate the growth and improvement of life on a global scale. Common examples of such needs for these aircraft are a high top speed, low stall speed, low cost of production, and a high payload capacity, all within a compact size and shape. Unbound by roads and terrain, the airplane has excelled at accommodating the modern traveler over great distances. Yet the automobile is certainly not without merit even by today’s standards. For the medium to short radius daily commute, it is still the fastest and safest means of transportation. There is no luggage to check, no airport or runway required to initiate travel, and it requires very little training to obtain an operator’s license. It can also safely operate under more severe weather conditions than those permitted by aircraft.

While both types of vehicles offer tremendous transport capabilities, both also have limitations with respect to making the most efficient use of time, especially with respect to intermediate travel distances where the fastest travel requires combining these two modes of transportation. One reason for this is perhaps that they are still two completely separate entities requiring entirely different facilities and infrastructure for operation. The merging of these two modes
of transportation into one cohesive vehicle would offer new possibilities for faster and safer transport for the intermediate range traveler. Several vehicle concepts have been either proposed or prototyped in an attempt to accomplish this task and that process has spawned an intensive list of new design challenges for safety and cost effectiveness.

1.2 The Flying Car

One such project undertaking this challenge is the Ohio University Flying Car also known as the Aeromobile. The background for this project can be read in greater depth in the Ohio University periodical Perspectives [1] which credits the contributions made to the project by the faculty and students on the team. The article in the periodical also discusses the challenges faced by the team as they work towards the goal of making the Aeromobile a feasible concept for mass transportation. The airframe of this vehicle will need to have vertical take off and landing capabilities (VTOL) if the final prototype is to be feasible. While the comprehensive list of additional challenges is beyond the scope of this research, enabling efficient and cost effective vertical take off and landing is essential for the success of the Aeromobile. VTOL further implies that the vehicle has no forward velocity relative to the ambient air by which it is surrounded. Other than the helicopter, few commercial aircraft possess such capabilities. The Harrier and Joint Strike Fighter are two examples of modern aircraft capable of such feats as well as
reaching high speeds but neither is fuel efficient or cost effective in terms of the needs of the daily civilian commuter.

These military aircraft employ a downward, concentrated high-speed stream of air in order to generate the lifting force required to ascend and descend in a controlled fashion. Such means are noisy, expensive and would be damaging to natural foliage in the vicinity of the take off area should they be operated outside of a conventional airport facility. It is for these reasons that a new means of generating VTOL capability is desired if they are to be implemented for mass-transportation.

One proposed new means for VTOL is the supercirculation principle. It proposes that sufficient lift could be generated on a lifting body by introducing circulated forced air currents [2] to a conventional wing. While there is great debate to the exact phenomenon taking place when a wing generates lift, it is generally agreed that its lifting performance is directly proportional to the difference in pressure above and below the wing as caused by a difference in the speeds of the airstreams above and below the wing with respect to one another. The higher airspeed of the stream above the wing creates a low pressure region relative to the high pressure region below the wing created by a slower moving air stream. This net pressure differential acting over the two surface areas of the wing results in a net upward force known as the lift. Supercirculation research seeks to find new ways of achieving lift by developing upper surface blowing over wings and other more complex flows.
1.3 Aeromobile Regenerative Supercirculation Test Stand Specifications

In order to research and develop the supercirculation principle, a test stand for testing complex flows is essential. The goal of this project is to design, optimize and fabricate the Aeromobile Regenerative Supercirculation Test Stand (ARSTS) so that it is capable of measuring the important aerodynamic forces acting on a vehicle or wing section, the most dominant being lift and drag. This ARSTS should permit analysis indoors such as in a laboratory setting as well as outside for vehicles using combustion to generate propulsion or lift. The stand should therefore be able to be moved from one test site to another in a reasonable amount of time and with minimal human effort unaided by man-lifts or other machinery. The stand must also be able to test a wide variety of test subjects of various configurations in a safe, timely, and cost effective manner to the researcher, utilizing some form of semi-universal fixture that can be adapted as required to analyze such subjects.

An example of one such test subject is the Aeromobile, Ohio University’s quarter scale flying car (Figure 1-1). The Aeromobile will have a gross takeoff weight of twenty pounds and will be capable of generating a maximum thrust of twenty-two pounds. The vehicle’s fuselage measures thirteen by thirteen by forty-four inches and has a five foot wingspan. This is the maximum size test subject that the stand must be able to accommodate with respect to a test subject’s weight, force generation and physical dimensions.
1.4 Literature Review

To facilitate the collection of meaningful data, the stand must be capable of accurately measuring various parameters of the test subjects using a data acquisition system operating in real time. A review of many different sources of information about derived from facilities possessing these qualities has been conducted. Many journals, textbooks and company product guides have been researched, and the information found therein has been applied to this thesis where relevant.

Figure 1-2 shows the chord profile and free body diagram for a test subject illustrating its weight and generated thrust, lift and drag. The Figure also illustrates the need to ensure a neutral attitude of the test subject with respect to some
horizontal datum and the sensor. This idea is further supported by the findings of Ales Hribernik, Gorazd Bombek, and Ivan Markocic in their research entitled, “Velocity Measurements in a Shotblasting Machine,” which can be found in the journal of *Flow Measurement and Instrumentation* [5] which stresses the importance of sensor location to ensure functionality and accuracy. This permits the researcher to verify that the test subject is level, or that it has a zero angle of attack should it be desirable to apply an external current of air upon the test subject. Another article from the same journal [6] stresses the importance of maintaining high signal-to-noise ratios and minimizing static charge build-up around the sensor, although direct comparison of this example is not applicable.

![Figure 1-2 Defiant canard BL 20 airfoil](image)

**Figure 1-2 Defiant canard BL 20 airfoil**
Many factors contribute to the effectiveness of the supercirculation process in enabling lift through the circulated forced air streams. In order to help properly identify and understand these factors through research, the development of the stand should keep in mind the aerodynamic impacts its physical structure will impose upon the test subjects. The nature by which the stand supports test subjects, as well as its physical dimensions and data collection process should be realized in a manner that will impose as little external aerodynamic influence as possible. For example, the span and wing configuration of the 1/4-scale model will influence the height of the stand through the consideration of ground effect [7] and trailing vortices [2]. In other words, the process by which the proximity of the ground to the test subject’s wing tip interferes with the flow stream near the wing tip as lift is generated could be of concern if the stand height is not sufficiently large. Additionally, the data collection system should organize and store the data in an easily retrievable archive for viewing, analyzing and manipulating.

The information contained in Appendix A is cited from the University of Arizona’s website [3] regarding its new 5.1 million dollar wind tunnel facility containing both supersonic and sub-sonic test sections. Although it’s expected that the majority of supercirculation test subjects will not be immersed in a free stream (although the stand should permit it) since they will generate their own air currents for lift, the information cited in Appendix A [3] suggests a few sampling parameters for the stand since they may possibly be used (as a ratio with
the speed of the information source, Mach 0.1) as the upper limit of test stand’s measurement accuracy. In other words, the 0.1% for air current measurement appears to be a reasonable value to evaluate the currents generated by the test subjects. Additionally, the sampling rate of the supercirculation test stand might (again using air speed ratios) consider the performance of the ASU facility as a starting point, namely: Intel P-III 800 MHz MS Windows based PC, and 333 KHz sampling rate.

The information is also useful as it demonstrates a working acceptable resolution (0.1% full scale) for the force measurement device employed by a facility where precision and accuracy are crucial. Additionally the ratios between the normal and moment forces suggest consideration should be made to the expected performance of the test subjects. The axial force measurement has the lowest full-scale range: only 2 lb, which suggests the center tests subjects with low drag characteristics. On the other hand, the need to test either large forces or long moment arms about this axis is clearly indicated by a full-scale capacity of 300 in-lb about the roll axis, the largest full-scale capacity moment which can be tested by the stand. This is possibly the result of the need to research the performance of a wide range of aileron applications.

This setup in use by the ASU facility [3] employing a 6-axis force/moment sensor and data acquisition system for load measurement, is further supported as being comprehensive with respect to the forces measured by the findings of Gab-
Soon Kim and Hun-Doo Lee in their article entitled, “Development of a six-axis force/moment sensor and its control system for an intelligent robot’s gripper,” in the journal of *Measurement Science and Technology* [4]. The article also supports the use of analog to digital signal conversion and a PC card for data transfer and processing.

**1.5 Thesis Final Objective**

The end result of this thesis will encompass a versatile test stand capable of permitting the research necessary to obtain a more thorough working knowledge base of supercirculation and other lift generating processes. In addition, a thorough calibration of the stand will be conducted which includes a series of separate tests performed to ensure accuracy in measuring all forces and moments that may be relevant to supercirculation research. It will also be demonstrated that the stand is capable of verifying the static weight of the Aeromobile, static thrust produced by the Aeromobile, and the lift and drag measurements of an airfoil separately tested in a wind tunnel. These design criteria along with others to which the stand has been designed and constructed are summarized in a list of specifications in Appendix B. The remaining chapters and sections within this thesis will address each of the design specifications and the manner in which they have been realized.
2. Developing the ARSTS

The development of a test stand used to measure aerodynamic forces generated by a wide variety of test subjects is a complex process. Many factors need to be thoroughly addressed if the end result is to provide accuracy and versatility in a useable fashion. The operational characteristics of the test subjects tested by the stand, influence the stand’s geometry and electronic instrumentation. Evaluating these characteristics will also affect other aspects of the design, selected materials, and fabrication processes. The following sections and subsections in this chapter document this process for the ARSTS.

Chapter two is broken down into two main sections which both contain several subsections. In section 1 the associations between the required testing abilities of the stand, available electronics packages, cost constraints, and available hardware are shown and the conceptual solution determined. In section 2 the physical fabrication of the stand, discovered problems and implemented solutions are discussed. The end of section 2 also illustrates the accessories used to complement the final assembly in order to meet the functional requirements of the ARSTS.
2.1 ARSTS Design

Subsection 2.1.1 reviews the electronic load cells available from industry for the ARSTS application as well as the costs and benefits associated with each type of cell. Subsection 2.1.2 illustrates how the conceptual design enables the most accurate and effective use of the selected electronics package. The last two subsections use conventional mechanics to validate the conceptual design prior to fabrication.

2.1.1 Measurement Equipment

Approximately $5,500 was budgeted for the test stand’s electrical components, but most facilities capable of the performance objectives desired for this project cost $50,000 or more. Initial searches for instrumentation were made to locate load cells that were both affordable and suitable since they are the most accurate measuring devices for this application. Complex mountings employing spring scales would not be sufficiently accurate and would obstruct the flow path around the test subject. Figure 2-1 shows a single degree of freedom load cell made by Cooper Instruments. This device measures tension and compression along one axis by using an internal strain gauge bound to the metallic cell housing. The electrical characteristics of the gauge change as forces are applied to the housing causing it to deform. This correlation can be mapped with an equation and enables the accurate measurement of the applied forces. In three dimensional space, there
are three axes through which axial forces can be applied in tension or compression, or about which moments can be applied in the clockwise or counterclockwise directions.

![Figure 2-1 Single degree of freedom load cell](image)

For the purposes of this application, it’s most desirable to use a six degree of freedom cell since all possible forces and moments can then be measured: three axial forces in tension or compression, and three moments in either the clockwise or counterclockwise directions.

For each additional degree of freedom that a load cell is capable of measuring there are additional electronics that must be integrated with the cell and therefore the cost of the cell increases. For this reason initial investigation was made to achieve a stand/cell configuration that would permit 6 DOF measurement at low cost using a combination of 1 DOF Sensors. By employing a four-post attachment to a test subject and vertically stacking two 1 DOF cells at each post, all
six degrees of freedom can be measured using a decoupling matrix in a computer algorithm. Figure 2-2 illustrates this concept.

![Figure 2-2 Four point stand concept](image)

Although the four-point stand concept is affordable, it is less than desirable as it disrupts the flow field around the test subject at multiple locations, and could therefore reduce its effectiveness as a tool for experimenting with supercirculation.

Figure 2-3 illustrates another concept employing a multiple cell arrangement to measure lift while keeping cell costs low. This arrangement has only two, in-line connection points to the test subject and would cause far less interference with the flow field, but would only be capable of measuring two or three degrees of freedom. The most effective arrangement will utilize only one connection to the test subject while providing the ability to measure all six degrees of freedom with a multi-axis cell.
Keeping Forces Independent

- Track restricts loads on 1 DOF Cells to tension or compression only
- Roller bearings minimize friction forces resulting from thrust created by test subject

Table 2-1 shows several load cells of varying degrees of freedom and the costs associated with them. The relationship between measured degrees of freedom and cost is nearly linear and increases by approximately $1000.00 per additional degree of freedom.

Figure 2-3 Two point stand concept
Table 2-1 Load cell configurations

<table>
<thead>
<tr>
<th>Quantity(s) Measured</th>
<th>DOF</th>
<th>Calibration</th>
<th>Accuracy</th>
<th>Sample Rt</th>
<th>Cost</th>
<th>Manufacturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fx</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>$130</td>
<td>SMD</td>
</tr>
<tr>
<td>Mz</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>$895</td>
<td>Honeywell Sensors</td>
</tr>
<tr>
<td>Fx</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>$1,350</td>
<td>Cooper Instruments</td>
</tr>
<tr>
<td>Fy, Mz</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>$1,795</td>
<td>Cooper Instruments</td>
</tr>
<tr>
<td>Fx, Fy, Fz</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>$3,000</td>
<td>Kistler Instrument</td>
</tr>
<tr>
<td>Fx, Fy, Fz, Mz</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>$9,000</td>
<td>Kistler Instrument</td>
</tr>
<tr>
<td>Fx, Fy, Fz, Mx, My, Mz</td>
<td>6</td>
<td>120 lb</td>
<td>0.75%FS</td>
<td>28 kHz</td>
<td>$6,275</td>
<td>ATI Daq</td>
</tr>
<tr>
<td>Fx, Fy, Fz, Mx, My, Mz</td>
<td>6</td>
<td>150 lb</td>
<td>0.1% FS</td>
<td>1 MHz</td>
<td>$45,000</td>
<td>M. M. &amp; Tool</td>
</tr>
<tr>
<td>Fx, Fy, Fz, Mx, My, Mz</td>
<td>6</td>
<td>100 lb</td>
<td>1% FS</td>
<td>8 kHz</td>
<td>$4,036</td>
<td>JR3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Electronics Package</th>
<th>DOF</th>
<th>Cost</th>
<th>Manufacturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amp, Cable &amp; A/D</td>
<td>6</td>
<td>$1,000</td>
<td>N/A</td>
</tr>
<tr>
<td>Board</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The table also shows a cost allotment for an electronics package which provides a link between the cell and the computer. This is accomplished using a cable and receiver card which plugs into the PCI bus slot on the computer and typically costs around $1000 per cell as listed in the table. The last and third to last rows of the table contain information for the two cells providing the best performance at the lowest cost. Both cells can be calibrated to measure the required range of forces at a reasonable sampling rate. They are the JR3 6DOF load cell, model 67M25A-I40 which samples at 8 kHz and the cell made by ATI which
samples a 28 kHz. The costs of these cells are considerably less than for the one manufactured by Modern Machine & Tool which samples at more than 1 MHz and is accurate to several decimal places. For the purpose of this project, the sampling rate of 8 kHz and full scale range of 130 lbs normal force of the JR3 cell meet the desired performance requirements. JR3 provides a 10% price discount when products are purchased by an academic institution placing the cost of their cell ($4,036 after discount) within the allotted budget. The cost of the cable and receiver card to operate the cell from JR3 is $882 with the academic discount, keeping the total cost of the electronics package ($4,919) within the project budget.

Table 2-2 displays the complete physical load cell information and calibration numbers for the selected JR3 load cell. The performance characteristics of the cell and electronics package meet the necessary requirements for supercirculation experimentation in terms of the sampling rate (8 kHz) and full-scale measurement capacity.

The cell must be capable of measuring forces greater than those expected by the ¼-scale Aeromobile since they are current estimates and may change as the model develops and new hardware replaces old. The model will weigh around twenty pounds when completed. A 60-degree banked turn could produce a 2g wing loading creating 40 lbs of lift force in the positive z-direction. The model is also capable of producing twenty lbs thrust in the x-direction. It is evident from a
comparison between these values and those listed in the table that the cell will provide an acceptable range of measurement. The range can also compensate for

Table 2-2 Selected JR3 load cell

<table>
<thead>
<tr>
<th>JR3 Sensor Model 67M25A-I40 50L130</th>
<th>Serial No. 2756</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominal Diameter: 67 mm</td>
<td>Nom. Height: 25 mm</td>
</tr>
<tr>
<td>Electrical Load Settings</td>
<td>Sensor Load Ratings</td>
</tr>
<tr>
<td>Fx</td>
<td>50 LBS</td>
</tr>
<tr>
<td>Fy</td>
<td>50 LBS</td>
</tr>
<tr>
<td>Fz</td>
<td>100 LBS</td>
</tr>
<tr>
<td>Mx</td>
<td>130 in-LBS</td>
</tr>
<tr>
<td>My</td>
<td>130 in-LBS</td>
</tr>
<tr>
<td>Mz</td>
<td>130 in-LBS</td>
</tr>
</tbody>
</table>

inaccurate performance predictions or an increase in the forces produced as the model progresses to larger scales. One unfortunate feature of the cell is that its transformation matrix decouples the forces and moments to the left-hand rather than right-hand sign convention. This convention is displayed in the subsequent figures of this document from this point forward.
2.1.2 Conceptual Stand Design

The stand was designed around the selected JR3 cell to enable utility. As Figures 2-2 and 2-3 illustrated, having more than one connection point between the cell and the test subject is less than desirable and now unnecessary since the JR3 cell internally measures and decouples the forces and moments associated with 6 DOF measurement. Therefore a single point connection or stem design is most appropriate and beneficial as illustrated in Figure 2-4.

![Fin stem stand](image)

**Figure 2-4 Fin stem stand**

The single point connection between the stand, cell and test subject shown in Figure 2-4 provides the least resistance to the flows generated by supercirculation
research. The hinged fin (purple) provides the ability to level the test subject with respect to the ground or horizontal datum as well as introduce an angle of incidence. However, the drawback is that this fin is very thin in the y-direction and has a low moment of inertia about the x-axis. It provides little resistance to bending about the x-axis from side loads imposed along the y-axis. For this reason, a hollow tube was used instead to achieve the stem design and this configuration can be seen in Figure 2-5. The hollow tube places the stem’s mass further from the center of the x and y axes than the fin of Figure 2-4 providing a greater moment about them (parallel axis theorem) and therefore greater resistance to bending.

Figure 2-5 Tubular stem stand design
A search was made to locate a spherical joint to permit leveling the cell and test subjects. Companies specializing in fixtures had several models available, but none were found which could both be easily attached to the base structure and cell, as well as provide a rigid joint resistant to engine vibration from test subjects. Therefore the spherical joint shown in Figure 2-6 was developed using the CAD program Solid Edge available from Unigraphics. The ball (2” diameter) and rod at the center of the joint were trailer-hitch components purchased from an automotive parts store. The rest of the joint, as well as the stand’s components (not including bolts) were machined from 4130 steel. Figure 2-6 shows a detailed illustration of the joint including the as-built assembly (Figure 2-6, b). Fine pitch metric bolts were used to fasten the components of the assembly so that they would not loosen from engine vibration transmitted from test subjects.
Figure 2-6 Spherical joint assembly

Figure 2-6,a shows the fastening order of the 6.0 x 1.0 metric bolts. Coupler “one” is fastened to the main post extending upward from the base. Couplers “two” and “three” are then bolted in sequence with the bolt arrangements offset by 90 degrees as illustrated by the vertical black lines. The dimensions of couplers “two” and “three” are driven by the ball since its diameter is fixed, coupler “one” is then driven by coupler “two” and the main post.

The operation of the joint to level the cell is easily explained using Figure 2-6, b. The four 6.0x1.0 mm metric head screws with washers are tightened in star formation using a small crescent wrench to the point where the washers just become snug. The cell is checked to make certain it is level and then each screw is tightened
in star formation an additional ¼-turn. This is all that is necessary to withstand 22 lbs of thrust as generated by the ¼-scale model.

2.1.3 Stability Analysis

The stand will sustain a multitude of forces from testing. It is therefore important to design its limitations beyond the range of forces and moments generated by acceptable test subjects. The ¼-scale model dictates the upper limits of these forces, the most substantial of which are the combined loadings it can generate such as a combination of lift, thrust and roll. The forces create moments about the base of the stand, so weight must be added to counteract them and hold the stand in place so that accurate measurements can be made. It is important to check that the main post will not deform, yield or deflect from these forces, and also that the stand will not tip about its base.

Since the goal is to keep the stand rigid, the problem can be solved with a statics and mechanics of materials analysis. Figure 2-7 shows a free body diagram of the stand for the simple load case of the ¼-scale Aeromobile engine thrust. The purpose is to check if the selected materials appear suitable before continuing with a more detailed analysis. The thrust acts through the centerline of the vehicle approximately eight inches above where the load cell would be placed on the top of the stand tube. For modeling purposes, the stand is shown as a continuous tube.
Figure 2-7 shows the forces applied to the stand in the y-z plane. The max thrust the stand can sustain without moving is limited by the summation of moments about point A and the coefficient of static friction between the base and the ground. The variable $N$ in equation 2-1 represents the minimum sum of stand and base weight required to prevent the stand from sliding when the maximum thrust, 22 lbs and coefficient of friction, 0.6 (steel on concrete) [8] are substituted for $F_s$ and $u_s$ respectively.

$$F_s = Nu_s$$ (2-1)
Solving equation 2-1 yields a minimum gross stand and base weight requirement of 36.7 lbs to prevent it from sliding. The final weight of the stand and base fabrication was 119 lbs therefore no additional weight is required to be added. Equation 2-2 shows a summation of moments (CC positive) about point “A” in Figure 2-7 to be sure the stand will not tip. The friction force, \( F_s \) passes through this point and does not contribute to the moment, \( M_A \). The normal force, \( N \) will be equal to the weight of the stand and base combined, \( W_T \). Therefore, both \( F_s \) and \( N \) are neglected in the tipping calculation of equation 2-2.

\[
\Sigma M_A = W_T(18") - F_T(56")
\]  

(2-2)

Setting the moment about point A, \( M_A \) equal to zero in equation 2-2 for a static analysis and substituting the \( \frac{1}{4} \)-scale thrust of 22 lbs for \( F_T \), the minimum stand and base weight combination required to prevent tipping is 61.1 lbs. Since the final weight was 119 lbs no additional weight is required to prevent the stand from tipping during testing of the \( \frac{1}{4} \)-scale model. The stand will then be in static equilibrium and will not tip or move. The weight of the model was not included in determining the normal force between the stand and the concrete since it is expected to be negated by the lift generated by the model. Experimentation with setting the stand on the linoleum floor in the lab approximated the static coefficient between the two surfaces to be around 0.2. The addition of two 10 lb weights to the base of the stand is therefore sufficient to prevent it from sliding in the lab.
2.1.4 Deflection Analysis

Since the stand is static, the deflection of the top of the stand with respect to the base can now be checked. Equation 2-3 is the singularity function for a cantilevered beam with a concentrated loading and shows the deflection of the stand post as a function of the applied forces, post length, and material properties [9].

\[ y_{\text{max}} = \frac{F}{6EI} \left[ l^3 - 3al^2 - (l-a)^3 \right] \]  
\( y_{\text{max}} \) = maximum deflection of beam at its free end  
\( F \) = applied force  
\( E \) = Young’s modulus of elasticity  
\( I \) = 2\textsuperscript{nd} area moment of inertia  
\( l \) = beam length  
\( a \) = distance to load location from boundary condition

Equation 2-3,b [9] shows the reduced form of equation 2-3,a with the variables defined by their nominal values according to Figure 2-7. Since a practical test run of the \( \frac{1}{4} \)-scale model will involve some type of combined loading, the value of the applied force, \( F \) has been doubled in an attempt to provide a conservative analysis.

\[ y_{\text{max}} = \frac{Fa^2}{6EI} (a - 3l) \]  
\( F \) = 44 lbs  
\( E \) = 30 Mpsi  
\( I \) = \( 0.25(\pi)(2.5)^4 - 0.25(\pi)(2.26)^4 \) in.\(^4\)
\[ l = 56 \text{ in.} \]
\[ a = 56 \text{ in.} \]

The primary purpose of the stand is to experiment with supercirculation or lift generating devices. It is thought at this point that deflection values of a few hundredths of an inch or less will be acceptable since most testing is expected to involve steady-state scenarios rather than highly dynamic situations. Solving equation 2-3,b the maximum deflection at the end of the post, or load cell location, is \( y_{\text{max}} = 0.0077 \text{ in.} \). Note the length, "l" and location of the application of the load, "a" as both being 56 in. These values were used since the final stand height is 48 inches and there is a rigid connection from that point to the center of the thrust-line on the ¼-scale model, essentially creating a longer moment arm about the base. The increase in the height of the stand to 48 inches was to provide more ground clearance for air flows. Fabrication of the stand was initiated since the deflection of the stand was well within the acceptable range.

### 2.2 ARSTS Fabrication

Section 1 in this chapter demonstrated the calculations made to determine the proper geometry, dimensions and connections for the ARSTS given the materials and fabrication processes available. Section 2 details the fabrication and final assembly of the ARSTS based on those calculations and the modifications
necessary to achieve the end result of a working reliable stand which can meet the requirements set forth in Chapter 1. The following three subsections highlight the problems discovered, solutions determined and the impact of their implementation on the ARSTS.

2.2.1 Preliminary Construction

Figure 2-8 shows the initial stand fabrication with the load cell mounted on top and the cable link to the receiver board installed. Some spare weights in the form of grout bags and an eight-inch wide-flanged beam were readily available for anchoring to permit testing.

![Initial stand fabrication](image)

**Figure 2-8 Initial stand fabrication**
The design employed a 24x36x1/8" 4130 steel base plate as the bottommost piece of the stand upon which weight could be added for anchoring as seen in Figure 2-9. A dozen 6.0x1.0 mm metric screws were used to bolt the base plate to the ¼ in plate and upon which the main post coupler was attached using four through bolts spaced radially at 90 degrees. The coupler and main post were also attached at the base in the same manner that the main post is attached to the spherical joint (Figure 2-6) using the (8) metric bolts spaced at 90 degrees and staggered from the pattern of the through bolts.

![Figure 2-9 Initial stand fabrication](image)

The factory coordinate settings for the cell are shown in Figure 2-10. Notice it is a left hand coordinate system and is different from the conventional right hand aircraft axis assignment. The convention is arbitrary for the purpose of
collecting data, but may become important in preventing confusion when data is shared with other members of the aerodynamic community, and the stand user is cautioned to pay attention to this change in sign convention.

![Factory coordinate system](image)

**Figure 2-10 Factory coordinate system**

The data link between the load cell and the computer is through an analog to digital signal conversion board as shown in Figure 2-11. The board is inserted into the computer main frame through the PCI bus slot and requires 5 volts power supply for operation. The slot powers the board enabling it to sample the load cell readings at 8 kHz (8000 times per second).

The first attempt to investigate the stand’s rigidity proved that it lacked the desired stability for successful operation. The displacement of the top of the stand
(where the load cell is mounted) with respect to the base in either the x or y direction was approximately 0.5 in when a load of about 20 lb was applied. This displacement is unacceptable if accurate test results are to be obtained since most systems of interest will involve oscillating forces.

A close inspection determined that the displacement was most likely the result of two compounding aspects of the initial construction. The base plate (Figure 2-9) was warped from the cut-off process used by the distributor in
cutting it to the specified dimensions. This resulted in a natural tendency of the plate to deform further when a moment was imparted upon it by the main post. The second aspect of the initial construction lending to the lack of rigidity was the bolted connections between the base members using the 6.0x1.0 mm metric screws. The minimal surface contact and tolerance afforded by the screws created a more flexible joint than could be provided with a welded joint. This was due to an initial desire for the user to be able to disassemble the stand as much as possible should future adaptation to other fixtures become necessary. While this was desired, it was obvious achieving the desired end would need to begin with changing these features.

2.2.2 Trouble Shooting

With the source of the problem known, analysis was resumed to calculate the required modifications to the current construction. The previous deflection analysis of section 2.1.4 was made under the assumption of a fully clamped boundary condition between the base of the stand and the supporting surface. Assuming the base would be rigid also never addressed the possibility of warping. An attempt to better model the stand by addressing the relationship between applied forces and the response of the base plate and connecting members was made as shown in Figure 2-12.
Although this model does not account for the warping of the base plate (which further contributes to the problem), the resulting stress distribution of Figure 2-12 shows the maximum stress of 6.6 ksi occurring at the base of the post as well as within the base plate. This confirms the suspicion that the base plate is part of the problem which is further visually depicted by the color coded displacement plots of Figure 2-13.

The displacements along the z axis in the base plate are illustrated in Figure 2-13,a. While low in numerical value, they equate to much larger displacements at the end of the main post where the load cell is mounted as indicated in Figure 2-13,b. This suggests that replacing the base plate altogether with a more rigid member will greatly reduce the deflection and result in an assembly more closely modeled by the original deflection analysis of section 2.2.1.
The max displacement of the load cell should not exceed 0.15 in. according to the ALGOR finite element analysis results shown in Figure 2-13,b. The exact material (4130 steel) properties were used in the calculations made by the model. A more detailed model would have to be created using an assembly of separate parts to confirm if the remainder of the displacement (0.35 in.) is the result of the bolted connections. This type of analysis is tedious and costly with respect to time.

Since the model shows the problem is in the region of the base, the assumption was made that welding the components would solve the problem without spending the time to create a new model with separate parts that could discredit the initial design. A minor modification was made to model a more rigid
base plate member and simulate the welded connection. Figure 2-14,a,b,c shows the results of the analysis.

Although the boundary constraint of Figure 2-14,a shows a continuous ground contact, the rigidity of the base results from the large moment of inertia created by the wide-flanged beam. The results show the stress and deflection have
both been reduced by a factor of about three. With the assumption of the
improvements from welding, the stand deflection from the applied loads was now
again predicted at an acceptable level.

2.2.3 Final ARSTS Fabrication

The final base plate construction for the stand consisted of a welded “cross”
using 3x8 in wide-flanged beam as depicted in Figure 2-15. The center span of the
beam was tapped for 5/16x24 English hex bolts used to anchor the base and coupler
of the stand’s main post. The larger diameter bolts around the perimeter as shown
in Figure 2-16 deflect less under the given stress and simulate a welded connection.

Aside from the new base plate and connecting hex bolts, the only other
aspect of the stand that was changed from the initial design was the height of the
main post. The final length locates the load cell 48 inches above ground permitting
the necessary ground clearance for ¼-vehicle testing of complex flows. The
remainder of the stand was left unaltered.
To complete the required functionality of the stand, a moveable platform was built upon which the stand can easily be placed by two average researchers. A mount for engine testing as well to which other devices such as the ¼-scale model could be attached was machined from 6061 T-6 aluminum and later used in the calibration process. Additionally another flat plate was fabricated to which small thin test subjects could be mounted. These sub-components necessary for the stand’s operation are illustrated in Figure 2-16.
Figure 2-16 Stand sub-components
3. Operating the ARSTS

A well conceived, designed and fabricated test stand containing high quality electronics is not very useful if its operation, capabilities, and limits are not thoroughly understood by the user. This test stand must be calibrated to ensure its accurate and precise operation. This process needs to be documented in an easily understandable format that allows the user to begin testing with accuracy shortly after an introduction to the equipment. The following sections and subsections of this chapter cover this process for the Aeromobile regenerative supercirculation test stand.

Section 3.1 briefly introduces the computer codes used to calibrate and operate the stand. Section 3.2 explains the physical and computer processes used to calibrate the stand and how the computer codes mentioned in Section 3.1 are used. The calibration process begins with an investigation of the cell’s response to the application of known loads. The cell’s characteristics with respect to overshoot and bias, as well as the accuracy of the jr2.m code’s data acquisition with respect to time are evaluated. The results of this initial calibration are presented in Section 3.2 and they have been used to structure the application of the jr1.m calibration code.

Section 3.3 further elaborates on the jr1.m calibration code and the calibration process by discussing the investigation of cell bias for the x-axis in tension in the 0 to 5lb range. The findings of this section combined with those of
Section 3.2 are later used to support the conclusions drawn in Chapter 4: Conclusions and Recommendations.

Sections 3.4 and 3.5 highlight the process of using the stand once it has been properly calibrated as discussed in Section 3.2. Section 3.4 discusses the physical hardware and computer codes used to operate the stand. An actual test subject (1/4-scale Aeromobile engine) is then tested in Section 3.5 and the results are confirmed and discussed. Instruction manuals for operating both the jr1.m calibration code and the jr2.m operational code are listed in the ARSTS User’s Manual in Appendix C of this document on page 76.

3.1 Stand Operation with Matlab

Appendix D on page 83 contains the Matlab script codes in *m-file* format that operate the load cell using the Mathworks program Matlab. The file names are: *jr1.m, jr2.m, ViewData.m, and UnusedCode.m*. The *jr1.m* file contains the script code used to calibrate the test stand by taking axial force measurements in increments of $\frac{1}{2}$ lb and moment readings in increments of 4 in-lb. Section 3.3 of this report highlights the use of this calibration code to investigate bias in x-axis and the results discovered. The files *jr2.m, ViewData.m, and UnusedCode.m* can be used to operate the stand during experimentation once the stand has been calibrated.
3.2 Stand Calibration

The JR3 Sensor Model **67M25A-I40 50L130** has the full scale ratings listed in Table 2-5. These are the maximum loadings the cell can accurately measure to within 1% of full-scale as listed by the manufacturer. The manufacturer provides a means of changing the full-scales using the cell command, “change_full_scales” listed in Appendix E and written into the script code in the “UnusedCode.m” file at the end of Appendix D. Testing showed this did not have an impact on the accuracy of the cell but was provided to increase the resolution of the manufacturer’s real-time display module which does not store data from tests.

Figure 3-1,a shows a set of weights which were checked using a very accurate digital scale that read to the hundred thousandths place in pounds (although 0.01 lb is accurate enough based on cell capabilities) and used to calibrate the cell. The actual values for each of the weights were recorded and combinations were made to range from ¼ lb up to 20 lb. These weights were then used to calibrate an Accu-Weigh model T-20 spring scale (Figure 3-1,b) which was also used in the stand calibration. The scale has an uncertainty +/- 0.125 lb.
The calibration process began with testing each axis in tension and compression as well as moments about them in the range from 0 lb up to the forces imposed by the ¼-scale model (20.0 lb). An initial overshoot resulting from the application of each load was observed and clearly followed a time constant inherent in the cell’s circuitry. Error rather than bias was observed throughout the 20 lb range of applied loads with exception given to the x and y axial forces under 5 lb.

A consistent cell bias in measuring x and y axial forces under 5 lb in both the positive and negative direction was found. For instance in Figure 3-2, actual loads of 1.0 lb and -1.0 lb axial force were applied, but the cell reported loads of +/- 1.15 lb respectively. However for an actual load of 4.0 lb axial force the cell reading would be approximately 4.05 lb as shown in Figure 3-3. This suggested that the
bias was not constant and further investigation would be required. The results of this additional investigation are displayed later in this section.

![Graph showing Fy overshoot at low scales](image)

**Figure 3-2 Fy overshoot at low scales**

With a working test process established, the calibration continued by checking the accuracy of changing loads with respect to time. Table 3-1 shows the intended force-time load arrangement for one of the test runs of this type. An independent clock was monitored using the www.time.gov website to track the loads with time as they were imparted on the cell.
While it is difficult to quantify the error associated with this type of testing, the results of the cell readings for test run two (Figure 3-4) display accuracy and precision. Notice the relationship between the two diagrams. Since the axial forces are imparted almost directly on the cell, the moment readings associated with them are low. However, since the test mount was only 8 inches to either side, large forces were needed to create a moment and they show up clearly in the axial force plot. In
addition to the forces and moments measured, the plots indicate that the program is measuring consistently and accurately with time.

Table 3-1 Test run to verify multiple loadings with time

<table>
<thead>
<tr>
<th>time (s)</th>
<th>Fx (lb)</th>
<th>Fz (lb)</th>
<th>Mx (in-lb)</th>
<th>Mz (in-lb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10-20</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20-30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30-40</td>
<td>-10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40-50</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50-60</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60-70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70-80</td>
<td>-11.022</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80-90</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90-100</td>
<td>80</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100-110</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>110-120</td>
<td></td>
<td>-80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>120-130</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>130-140</td>
<td></td>
<td></td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>140-150</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>150-160</td>
<td></td>
<td></td>
<td>-80</td>
<td></td>
</tr>
<tr>
<td>160-170</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>170-180</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Additional testing was done in this manner in an attempt to check the cell’s accuracy for multiple combined loadings. The data tables with the intended force-time information and resulting plots are presented in Appendix G along with the other plots from additional test runs. Although the peak cell moment readings in Figure 3-4,b for Mx and Mz do not match the intent indicated in Table 3-1, it can be
seen from the figure that this error stems mostly from the inability of the researcher to impart these forces simultaneously with the accuracy desired. For instance, the peak moments of 65 in-lb should be 80, but the difference between these two values is attributed to the combined loading situation which resulted from a vector force application.

In addition to the error stemming from misdirected forces, a small portion of the error is attributed to uncertainty in the spring scale and the error in the cell. Figure 3-4,b can be used to quantify the error in the cell. The nominal moments for \( M_x \), \( M_y \), and \( M_z \) for the first load are 65, 5 and 5 in-lbs. Using the uncertainty in
the spring scale of +/- 1/2 division (0.25 lb), 2 of the remaining 5 in-lbs could be attributed to the spring scale. This leaves 3 in-lbs (3.75% error) unaccounted for by the cell.

Additional inspection of the data suggests that the moment readings about the x and y axis are taken from the geometric center of the cell rather than the “tool side” surface to which test subjects will be mounted. This should be kept in mind when comparing moment and axial forces to determine properties such as engine thrust, center of gravity, resultant lift, and all control moments.
3.3 Bias Investigation

Having established a properly working stand with the exception of the observed bias, a more thorough calibration was made to ensure future users the most reliable stand performance by removing the bias. This was done using the jr1.m code which can be found in Appendix D. The code was used to further investigate the bias of the low scale x and y axial forces in the range from zero to five pounds as well as sweep through the remaining 2 forces and 6 moments (+/- for all three axis). A set of 6 repeated data sweeps was made with the code for both the positive x-axis and the positive moment, Mx about the x axis. A single sweep was made for the remaining 5 forces and 5 moments. The “for” loop in the jr1.m code beginning with:

\[
\text{for } i=1:32; \\
\]

(3-1)

designates the number of measurements to be taken when the code is executed. The expression above shows that 32 independent measurements were taken in 4 in-lb increments to encompass the range of values from 0 to 128 in-lb for all six moments. This was done by applying ½ lb increments at a constant moment arm of 8 inches to either side of the center of the cell as seen in Figure 3-5 which shows a moment measured about the z-axis.
Figure 3-5 also shows the use of a right angle ruler to minimize inconsistency inherent in the application of the moment. The moment is applied by anchoring an adjustable tie-rod with two hardware clamps to a cabinet (any rigid body to which the clamps can be applied is acceptable). The tie-rod and clamps are brand independent and were purchased at the local hardware store. The force imparted by the T-20 spring scale was controlled using the adjustable tie-rod by closing (shortening) the rod to increase the tension in the scale.

The only change in the jr1.m script when measuring axial forces instead of moments is that the number 32 is replaced with 40 since ½ lb measurements were made from 0 to 20 lb. Ultimately the number is arbitrary and can be set at the user’s
discretion. The user should keep in mind that very small increments will require much more precise load measuring equipment.

The results of the repeated 6 data sweeps for the measurement of Fx are presented in Figure 3-6 which shows the range of measurements under 5 lb. The average, maximum and minimum values are color-coded accordingly in the figure to illustrate the results of the cell's measurements. The plot clearly shows the presence of a bias at this lower measurement range which can be accounted for in the jr2.m program in order to improve the accuracy of the cell. The bias is seen in the 0 to 3.7 lb range and has three distinct regions of change within that range. The first region is the range from 0.5 to 1.0 lb. The second distinct region is from 1.0 to 3.0 lb, and the third distinct region is from 3.0 to 3.7 lb (actual weight being measured). The modifications made to the jr2.m code to accommodate the bias present in these regions contain only a few lines of code. This is the result of the bias within each region being relatively linear in nature.
From 0.5 to 1 lb the bias has an initial value (y-offset) of 0.1829 lb at 0.5 lb actual load and a final bias of 0.1 lb (0.83 slope). In other words for a reading of 0.6829 lb the user would subtract 0.1829 to get the actual value, and for a reading of 1.1 lb the user would subtract 0.1 lb. This yields the equation:
where: $y =$ reported value (containing bias) 
$x =$ actual value imparted on the cell.

The actual values being measured by the cell follow the 45-degree line:

$$ y = x $$

This information can be used to develop an equation that will account for the bias within the entire 0.5 to 1.0 lb range which can be used in the jr2.m program to account for the bias and report the correct value to the user. By expressing the actual equation (3-2) in terms of the desired equation (3-3) and solving for $x$ (actual value) in terms of $y$ (value reported by cell containing bias), the jr2.m code can be made to account for the bias.

Expressing equation 3-1 with respect to 3-2 yields the following:

$$ y = x + (-0.1658(x-0.5) + 0.0829) $$

where: $y =$ reported value by cell 
$x =$ actual value being measured 
-0.1658 = slope of eq. 3-2 minus 1.0 
0.5 = x-axis offset 
0.0829 = y intercept offset of eq. 3-2 w.r.t. eq. 3-3
To use equation 3-4 in the jr2.m routine to account for the bias it must first be solved for the actual value \( x \) in terms of the measured value \( y \). Solving equation 3-4 in terms of the desired actual value \( x \) yields:

\[
x = 1.198753y - 0.198753
\]

(3-5)

Since arriving at equation 3-5 began by shifting the actual equation (3-2) down by a constant value of \( y = 0.1 \) lb, the final form must be modified to account for this change in terms of the actual value as follows:

\[
x = 1.198753(y - 0.1) - 0.198753
\]

(3-6)

where: \( x \) = actual value being measured

\( y \) = value reported by cell containing bias

Equation 3-6 has been substituted into the jr2.m code to account for the bias in the 0.5 to 1.0 lb range.

The average and standard deviations resulting from the 6 data sweeps for the 5 points of interest in the 1.0 to 3.0 lb range are shown in Table 3-2 below. The bias is relatively constant at an average value of 0.102 lb above the actual load applied to the cell. This allows for simple changes to be made to the jr2.m code to account for the bias in the 1.0 to 3.0 lb region. The code reports to the user the actual load measured by subtracting the average bias of 0.102 lb from each value reported by the cell.
The last of the three distinct regions demonstrating a bias is the 3.0 to 3.7 lb range. Similar to the 0.5 to 1.0 lb range, this range has relatively linear relationship and can be treated as such for the purpose of changing to the jr2.m code to account for the bias. The equation for the bias in the 3.0 to 3.7 lb range is:

$$y = 0.8564x + 0.1077$$  \hspace{1cm} (3-7)$$

where: $y =$ reported value (containing bias)

\hspace{1cm} $x =$ actual value imparted on the cell

Rearranging this equation using the same process shown previously the expression for the reported value as a function of the actual value is:

$$y = x + (-0.1436(x-3.0) + 0.1077)$$  \hspace{1cm} (3-8)$$

where: $y =$ reported value by cell
x = actual value being measured

-0.1436 = slope of eq. 3-7 minus 1.0

3.0 = x-axis offset

1.077 = y intercept of eq 3-7 at x-axis offset

Rearranging equation 3-8 to yield the desired actual load (x) as a function of the measured cell value containing bias (y) yields the following:

\[
x = \frac{(y - 0.5385)}{0.8564}
\]

(3-9)

where: x = actual value being measured

y = value reported by cell containing bias

Equation 3-9 has been substituted into the jr2.m code to account for the bias in the 3.0 to 3.7 lb range.

For the entire 0 to 20 lb range of measured values for the 6 sweeps of the axial force Fx, the average deviations ranged from 0.07 – 0.71 lb and the average of the average deviations was 0.29 lb. In the 4 to 20 lb range, the average percent error ranged from 0.14 – 3.96 %. Prior to accounting for the bias in the average values below 4 lb the average percent error ranged from 3.6 % at 3.5 lb actual value up to 36.6% error at 0.5 lb actual value. The modifications made to the jr2.m code reduce this error in the less than 5.0 lb range to less than 5% for the average sweep values.
Figure 3-7 shows the results of the complete 0 to 20 lb range for the 6 data sweeps. The 14 to 20 lb range would suggest the possibility of a bias however unlike the case of the less than 5 lb range, this hypothesis was not supported by the other tests that were conducted. Figure 3-8 displays the results of one such earlier test in which both the 15 and 20 lb measurements were within 0.25 lb of the actual value (1.67 & 1.25 % error respectively).

![Figure 3-7 Fx data sweep, 20 lb range](image-url)
This suggests the discrepancy could be the result of hysteresis (successive measurements are influenced by the previous ones) stemming from a lack of sufficient time between measurements, or the discrepancy could just be the result of the natural error within the cell. If the later of the previous conclusions is true, the worse single-measurement percent error occurs at the 16.5 lb load in which the cell reported only 15.7427 lb (4.59% error), and the average percent error in the 14 to 20 lb range was less than 3%. Figure 3-7 also shows a peak reading for the 16.5 lb measurement above the actual value and the average value of 16.38 lb is very close (<1% error) to the actual value of 16.5 lb.

Figure 3-8 Fx calibration, 5 lb decrements
These statistics support a slight increase in the delay time between measurements taken by the calibration code (jr1.m) for future testing rather than making additional modifications to the jr2.m code. The error found in the calibration results was always less than 5% and is considered acceptable for the purposes of experimenting with supercirculation test subjects. A one second pause and cell re-zero loop have been added to the calibration code jr1.m to eliminate any influence from hysteresis. This additional code increases the test time for each single point measurement by just over 1 second. The plotted results for the remaining data sweeps are found in Appendix G. The moment plot of the 6 data sweeps for Mx support the findings already discussed. The average values reported are nearly identical to the actual values applied to the cell. The worse case single measurement error was 8.125% resulting from 73.5 lb reported for 80 lb actual load, however the average percent error for the sweeps was less than 2%.

In summary, an observed bias was noticed in two of the six degrees of freedom measured by the cell. An investigation procedure was developed in order to either confirm or deny the presence of the bias using a calibration code (jr1.m). The code was developed in order to guide the user through an organized process of taking successive but independent measurements to more thoroughly investigate the response of the degrees of freedom (axial force and/or moment) in question. Once run, calibration code and test procedure generated data for the Fx and Fy axes in the less than five pound range. The results were statistically analyzed and plotted
confirming the presence of the bias, or consistent difference between the measured load and the load actually applied to the cell.

Since the difference was repeatable and therefore dependable, mathematical expressions were developed to interpolate the correct or actual applied load from the load indicated by the cell. In this case the relationship was linear, and negating the bias required determining its offsets and slopes (linear equations) for the three distinct regions and how they departed from the actual load \( (y = x) \) equation. However, the calibration code and test procedure are independent of the nature of the bias. If the bias had been non-linear in form, the same calibration code and process would have been used to collect the data. The difference would have come down to affecting the form of the equations added to the jr2.m code in order to negate the bias. In other words, the mathematical form of the bias (linear, exponential, hyperbolic, etc.) will dictate the form of the equations used to nullify its affect on the data collected for each load range in question. The step by step calibration procedure is documented in the ARSTS User’s Manual in Appendix C of this document on page 76.

3.4 Experimentation by the User

The basic functions supplied by the manufacturer in the MATJR3PCI module are listed in Appendix F. They are rudimentary in their operation allowing the user to give only one command at a time to the cell. For instance the user can
use the commands to sequentially sample the full scales, offsets, and readout of the cell and then change the full scales or offsets as desired. This is inadequate for the purposes of experimenting and collecting meaningful data in real time. The script in $jr2.m$ was developed around these basic functions to make the cell useable.

The written $jr2.m$ file permits the user to quickly collect a large amount of data in real time and interpret its meaning. The file can be used to display the data as it is collected, or store it in a Matlab `.mat-file` for analysis or manipulation at a later date. The $ViewData.m$ file displays data previously collected using the $jr2.m$ file in two separate figures for the axial forces and moments. Finally, the $UnusedCode.m$ file contains additional code that can be quickly added to $jr2.m$ file for alternative means of storing the data in a text file as well as performing other statistical calculations on the collected data. A manual has been written and included in Appendix C to instruct the user on executing these codes properly as well as using the stand for experimentation.

### 3.5 Vehicle Testing

Although a lack of time and continued development of the ¼-scale model prevented it from being tested at this time with the stand, a test of one of the model's engines was made to compare it with previously obtained test data. In a previous test run for the ¼-scale model, the engine thrust was measured using the same model T-20 spring scale from the calibration process. That test yielded a
static thrust of 13 lb (6.5 lb thrust per engine) using three-bladed propellers as the thrust model which the engines could turn at 21,000 rpm. Figure 3-9 shows one of the engines mounted to the intended thrust module composed of a 5 bladed ducted fan which runs at 28,000 rpm. The thrust was measured during the test run using the calibrated test stand and the output that was generated is shown in Figure 3-10. With this intended thrust module, the engine is claimed by the manufacturer to produce 11 lb of static thrust.

Figure 3-9 Aeromobile 1/4-scale engine test run
Figure 3-10 shows the program’s axial force output plot for one of the test runs. The test run gave thrust measurements for the ducted fan of 4 to 6.5 lb depending on the fuel air mixture with 6.5 being produced when the mixture was fully leaned out (high air content). In addition to the ducted fan, the engine is intended to run while fully enclosed in a streamlined duct within a small aircraft. The inability of the engine to meet the manufacturer’s claim of 11 lb static thrust is possibly due to the lack of this enclosure which was not available at the time of the test.
The sharp decline and rise in thrust at 62 seconds into the test displays the cell's measurement of an engine stall and restart. The negative Fx values at the very start of the test indicate the backpressure imparted on the fan using a motorized starter to start the engine. The ability of the system to capture these events demonstrates its ability to measure test subjects with dynamic behavior in a timely fashion. This proves that the stand is functioning according to the design requirements set forth at the beginning of this paper in section 1.3 and further asserts that it will provide reliable analysis for the user experimenting with supercirculation.
4. Conclusions and Recommendations

In this thesis, the need to create a test stand that can measure the performance of non-conventional lift generating vehicles and test subjects has been addressed and met. The three preceding chapters collectively demonstrate that the stand will enable the research required to progress the flying car project under development by Ohio University. In Chapter one, the specifics of this research and how it should be achieved were defined. The impact of that definition in generating the design specifications for the test stand was also discussed and itemized as presented in Appendix B. Chapter two illustrated the concept that would meet the design specifications and provide the researcher of supercirculation with the most cost effective test stand. This stand was then tested, calibrated and validated in Chapter three and the results presented for the future user and researcher. This chapter will now re-examine those results and include additional discussion about them.

The calibration procedure involved a wide range of tests in order to determine the stand was working accurately and reliably. The initial experimentation in section 3.2 demonstrated the proper execution of the Matlab script codes used to run the stand. It also proved functionality of the stand with respect to accurately timed data collection and storage, but clearly bias was present in the low range of x and y axial forces and additional calibration in the form of discrete data sweeps would be necessary. These data sweeps were conducted as
illustrated in section 3.3 and have generated useful and more reliable information which was then used to rewrite the computer codes where applicable. The data collected in the calibration was made more reliable and consistent with the use of hardware clamps to apply the loads imposed on the cell during calibration. This process minimized the calibration time and assured that the 32 and 40 independent moment and axial force measurements respectively were imparted on the cell in a consistent fashion.

The calibration further confirmed the cell bias in measuring x and y axial forces under 5 lb in both the positive and negative directions. This bias was found to contain three distinct linear regions for which mathematical relationships were developed and added to the code. Once these additions were made, the computer codes used to operate the stand nullified the bias and accurate operation in the less than 5 lb range was achieved. For the entire 0 to 20 lb range of measured values for the 6 sweeps of the axial force Fx, the average deviations ranged from 0.07 – 0.71 lb and the average of the average deviations was 0.29 lb. In the 4 to 20 lb range, the average percent error ranged from 0.14 – 3.96 %. Prior to accounting for the bias in the average values below 4 lb the average percent error ranged from 3.6 % at 3.5 lb actual value up to 36.6% error at 0.5 lb actual value. The modifications made to the jr2.m code reduced this error in the less than 5.0 lb range to less than 5% for the average sweep values.
The revolute joint assembly created for the upper portion of the stand appears to be functioning adequately for leveling the cell and test subjects. The base change to the welded wide-flange beam minimized deflection from applied loads at the cell location. The fine threaded hex bolts used throughout the stand assembly have maintained rigid connections throughout the duration of the tests and resisted the vibration imposed by the tested engine. With the exception of the welded base members, the bolted connections have also afforded room for modification to the height of the stand should research demonstrate it to be a desirable option. The coordinate system of the measured forces with respect to the cell appears to lie in the geometric center of the bolt surface with the exception of the moments about the x and y axis. Their zero moment axis extends through the geometric center of the cell and will need to be accounted for in interpreting the data collected during testing.

In conclusion, a working Aeromobile Regenerative Supercirculation Test Stand has been developed, constructed, calibrated and tested. The stand accurately measures and records the lift, drag, weight, and thrust forces generated by the desired range of test subjects and is operating successfully for the purpose of experimenting with supercirculation. The stand can also measure pitch, roll, and yaw capabilities for interpretation and analysis. Accessories enabling stand mobility within a laboratory setting and beyond to an outside location, as well as two simple load cell mounting pieces for adaptation of test subjects have been
created and tested. Since the test stand created within this thesis possesses these qualities, it will permit the development of supercirculation and other lift-generating processes of interest for the flying car and has therefore met the intended objectives.

The next step along this line of research can be taken in several directions. Although the single data sweeps performed on the remaining axis and moments did establish definite bias elsewhere in the range of measurements that will be performed by the cell, additional data sweeps could be made to confirm this conclusion or better define the cell’s performance if necessary for their individual testing application.

In addition to testing new types of lift generating devices, the researcher may also desire to fabricate a nacelle or housing for the wind tunnel facility of the senior lab to accommodate the ARSTS. This housing could be designed to immerse the top of the stand, the load cell, and a small test subject into the free stream flow created by the tunnel. This would permit further analysis of the test subjects once they have transitioned from VTOL operation to forward flight. It could also be used in parallel with the tunnel’s force-balance system for the purpose of comparing two independent sources of measurement of the test subjects.

Since the process of building and testing several or more models in two different methods as described above is time consuming and may become tedious, the researcher is reminded of the purpose of this stand. The stand is a tool for experimenting with general trends in performance and is accurate only to the extent
described in the previous paragraphs of this chapter and chapter three. Many types of models and processes may need to be tested in many ways to discover and qualify the driving forces of supercirculation. Once the trends have been discovered and the process understood, the development of supercirculation and therefore VTOL can be pursued in finer and finer levels of measurement and quantification. The researcher undertaking this task is encouraged in their pursuit of a more thorough understanding of complex air flows and lift generating devices.
Bibliography


Appendix A
Arizona State University Wind Tunnel Facility

Six-Component Force Balance:
A six component force balance, fabricated by Modern Machine and Tool, Newport News, has been operational since April 1999. This can be used for sensitive measurements in wind tunnels.
The six component values are:
Normal Force = 10 lb.
Axial Force = 2 lb.
Side Force = 10 lb.
Pitch Moment = 30 in-lb.
Roll Moment = 300 in-lb.
Yaw Moment = 90 in-lb.
Typical resolution is 0.1% full scale.

- X is horizontal and parallel to the freestream-flow direction
- Y is vertical
- Z is horizontal and normal to the freestream-flow direction

The tunnel is controlled using Lab View software running on a standard Intel P-III 800 MHz MS Windows based PC. Data acquisition is accomplished using two PCI DAQ boards. One of the boards has 16 channels (single-ended) and can sample at a maximum frequency of 333 KHz. It has two analog outputs which are used to control the butterfly valves in response to changes in the stagnation pressure reading. The second board provides four channels (differential) and can sample at a maximum frequency of 5 MHz.

The tunnel is a low-turbulence, closed-return facility with a test section in which oscillatory flows can be generated for the study of unsteady problems in low-speed aerodynamics. It can also be operated as a conventional low-turbulence wind tunnel with a steady speed range of 1 m/s to 36 m/s (Mach 0.1) that can be controlled with 0.1% accuracy.
ARSTS Design Specifications

- Measures aerodynamic performance
- Permits analysis both indoors and outside
- Tests ¼-scale Aeromobile model and smaller test subjects
- Includes semi-universal fixture to help minimize subject mounting time
- Tests a variety of different wing configurations
- Measures weight, lift and drag/thrust properties of a test subject (<= 5% error)
- Allows force data sampling and collection to be viewed in real time
- Stores this data in a retrievable archive
- Produces calibration data to verify accuracy and proper functioning
- Permits level testing on non-level surfaces of +/- 5 degree slope
- Possesses structural rigidity
- Enables portability without mechanical assistance within a reasonable time period
Users’ Manual for ARSTS Matlab Programs

Purpose:
Supplemental instruction for using the commented script codes in Appendix D. These codes can be used to operate the load cell and record data using the ARSTS
Jason J. Fink
jf352597@ohio.edu

Purpose for each program:

\textit{Jr1.m} – Calibrate the stand and investigate any suspected bias
\textit{Jr2.m} – Operate the cell and take data in real time which can be stored and retrieved at a later time.
\textit{ViewData.m} – Retrieve previously generated test data for plotting and/or manipulation
\textit{UnusedCode.m} – Additional lines of script which may be useful to the user.

Instructions for \textit{jrl.m}:

1) Locate the ARSTS PC & Stand in any vacant lab with relatively static ambient air and dynamic inactivity in order to minimize disturbances during the calibration. Be sure to position the stand near an adjacent rigid body of similar height to the cell as shown in the illustration below, or in a manner that permits imparting perpendicular or normal forces on the cell as needed to check the specific axis or moment of interest.

2) Create an anchor on the rigid body using C-clamps that permits repetitive application and removal of the T-20 Accu-weigh spring scale between the anchor and the aluminum bracket as shown below.

3) Check that the cell is level by placing a spherical air bubble on the cell. If the air bubble is centered inside the centering circle without overlap the calibration process will ensure reliability.

4) Mount the aluminum calibration bracket to the cell by tightening the four 6.0x1.0 mm hex bolts in a 1-2-3-4 star formation as illustrated until each reaches 40-65 in-lbs. Do not exceed 88 in-lb of torque! The total cell mounting process should cycle through the 1-2-3-4 star formation at least 3 times to be sure the cell is evenly loaded by the tightened bolts. This is accomplished by first tightening the bolts until they begin to become finger tight one at a time. Once all four bolts are to the point where they require a few ounce-inches of torque to further tighten, they can be tightened in star formation as mentioned.

\textbf{CAUTION:} the four 6.0x1.0mm bolts should not exceed 8 mm penetration into the cell. Measure the length of protrusion of the bolt shanks extending out from the mounting apparatus. Be sure to
account for a decrease in the mounting apparatus thickness resulting from the tightening of the bolts. In other words, the shank length protruding through a wood mounting will be longer after the bolts have compressed the wood.

5) On the ARSTS PC, logon to the user, “Super.”
   Note: any other user on the PC can run the program, but only
   “Super” will permit the data storage functions in the program to store
   the data since Matlab has been installed on the “C” partition of the
   hard drive for which “Super” is the user and is password protected.

4) Open the Matlab command window and check that the path of the working
directory is set to C:jr3pci\jr3pci_soft\jr3pci_app\Release

5) Before running the program in the command window, open the jrl.m file in
the Matlab editor and set the number of independent measurements to be
taken for the degree of freedom in question (32 for a moment calibration or
40 for an axial force) as shown in the code below. Once the number has
been set, save and close the m-file, and then open the Matlab command
window if it is not already open.

   for i=1:32;
   % the number in the command (for i=1:#;) above indicates the number of
   reading to be taken
   % # = 32 measurements for moment calibration
   % # = 40 measurements for axial force calibration

6) Type “jrl” and hit enter.
Note: This will execute the jrl.m script code located in Appendix D. The very first and very last commands called to the cell by the script are to “open” and “close” the cell respectively. If the program ever terminates prematurely due to a modification of the script, the last line will not be executed and the cell will not close. It must be closed manually by executing the last line of code. Otherwise it will cause problems in the mex file and “crash” the current matlab command window resulting in a loss of unsaved data. The two lines are repeated here for clarity.

```
a = matjr3pci('init_jr3', 5986, 4369, 1, 1, 1);  %This function
detects your operating system (OS) and calls the jr3pci driver.

matjr3pci('close_jr3')  %cell command to un-initialize cell/close the cell
```

7) When the prompt shown below appears, verify that nothing other than the aluminum bracket is contacting the cell and that the scale is NOT connected to the bracket and then follow the instructions of the prompt.

'Check that test subject is "Off" and there are no external loads, then hit enter:'
('activating cell, please wait... ')

8) When the prompt shown below appears, mount the spring scale between the rigid anchor and the aluminum bracket and tighten the lead screw on the anchor until the desired force is developed in the scale. For an axial force calibration, the tension should begin with \( \frac{1}{2} \) lb applied inline with the cell (center of aluminum bracket) as opposed to out on the end of the bracket. If a moment is to be calibrated, tighten the scale to \( \frac{1}{2} \) lb and connect it out on the end of the bracket at the 8” moment arm location as illustrated to develop the starting torque of 4 in-lb. When the scale is appropriately in place, hit enter to measure the force/torque.

'Begin operation of test subject and hit enter when you are ready to record data...'

9) The program will now measure the current calibration load and store the data in an array. It will then cycle back through the same process until the designated number of loops have been run as dictated by the user adjusted number code previously set to either 40 (axial force) or 32 (moment) for the
number of measurements to be taken. In other words, you will now repeat steps 7 & 8 as listed above until the full range of interest has been measured. 

NOTE: You may broaden or narrow the range of measurements taken by adjusting the number in the "for" loop as desired, and the specific values are determined by the starting load and increments, however increments greater than \( \frac{1}{2}\) lb may not achieve the resolution necessary to obtain a reliable calibration. The user should consider the intent of the test and the expected or possible bias when using any other increment.

10) When all the measurements for the test have been completed, the program will exit and a Matlab mat file containing an array of the recorded calibration data will be created. This mat file defaults to the generic name:

Raw_Data

Which contains the test run data in an array named:

raw_dat

The program saves this file in the working directory as previously set in step 4. Open this directory and rename the generically named mat file to new name not already in use that is more meaningful for the axis or moment that was just tested.

11) After renaming the mat file from the previous test run. Repeat steps 5-10 for the same axis or moment just tested until a sufficient number of arrays have been created to validate statistical significance. In section 3-3 of ARSTS thesis, 6 sweeps were performed to confirm the bias in the low range Fx and Fy axial force degrees of freedom.

12) Repeat steps 5-11 until each axis or moment of interest has been evaluated in the load range of interest. Once the data has been collected into the now renamed files containing the "raw_dat" arrays, apply basic statistical analysis to the data and plot the results using the script code in the ViewData.m file to confirm or deny the presence of a bias.

13) If a bias is detected. Apply the necessary mathematical relationships between the measured and actual load values to the jr2.m script code in order to nullify the bias. A quick re-sweep of the axis or moment in question will validate the change.

Instructions for jr2.m:

1) Move/locate the ARSTS PC & Stand to the desired test site and check that the cell is level by placing a spherical air bubble level on the cell. If the air bubble is not touching the outside of the circle the stand will provide accurate results.
CAUTION: be advised that testing small engines requires proper ventilation and protection for eyes and ears. Be courteous to those around you.

2) Mount the test subject to the cell by tightening the four 6.0x1.0 mm hex bolts in a 1-2-3-4 star formation as illustrated until each reaches 40-65 in-lbs. Do not exceed 88 in-lb of torque! The total cell mounting process should cycle through the 1-2-3-4 star formation at least 3 times to be sure the cell is evenly loaded by the tightened bolts. This is accomplished by first tightening the bolts until they begin to become finger tight one at a time. Once all four bolts are to the point where they require a few ounce-inches of torque to further tighten, they can be tightened in star formation as mentioned.

CAUTION: the four 6.0x1.0 mm bolts should not exceed 8 mm penetration into the cell. Measure the length of protrusion of the bolt shanks extending out from the mounting apparatus. Be sure to account for a decrease in the mounting apparatus thickness resulting from the tightening of the bolts. In other words, the shank length protruding from a wood mounting will be longer after the bolts have compressed the wood.

3) On the ARSTS PC, logon to the user, “Super.”
   Note: any other user on the PC can run the program, but only “Super” will permit the data storage functions in the program to store the data since Matlab has been installed on the “C” partition of the hard drive for which “Super” is the user and is password protected.

4) Open the Matlab command window and check that the path of the working directory is set to C:\jr3pci\jr3pci_soft\jr3pci_app\Release

5) Type “jr2” and hit enter.
Note: This will execute the jr2.m script code located in Appendix D. The very first and very last commands called to the cell by the script are to “open” and “close” the cell respectively. If the program ever terminates prematurely due to a modification of the script, the last line will not be executed and the cell will not close. It must be closed manually by executing the last line of code. Otherwise it will cause problems in the mex file and “crash” the current matlab command window resulting in a loss of unsaved data. The two lines are repeated here for clarity.

\[
a = \text{matjr3pci('init Jr3', 5986, 4369, 1, 1, 1);}
\]
%This function detects your operating system (OS) and calls the jr3pci driver.

\[
\text{matjr3pci('close Jr3')}
\]
%cell command to un-initialize cell/close the cell

6) At the first prompt, enter “1” and hit enter as directed to continue the program.
   Note: simply hitting enter will just exit the program.

7) The next prompt in the program directs the user to be sure that the test subject is “off.” In other words, if an engine is being tested for thrust or a supercirculation device for lift, it should have been already mounted and not currently in operation.
   Hit enter to continue the program.
   Note: the program will now display the default full-scale settings and that the cell is zero-ed. The UnusedCode.m file contains script to change the full-scale settings, but testing has shown it will not change the accuracy of the cell in taking measurements.

8) Be sure the readout given for the zero-ed force moment values are small; typical acceptable values from experiments have been in the -0.05:0.05 range. If values significantly different from these are displayed.
   Repetitively hit the enter key to cycle through the program and start over.

9) At the next prompt, enter the amount of time in seconds over which you would like to collect data. Be sure to pick a reasonable amount of time. A single measurement may only require 10-30 seconds. If testing a full range of engine rpm vs. thrust it may be more appropriate to chose 5-10 minutes, but the time must be entered in seconds.

10) The next prompt tells user to hit enter when ready. Turn on or begin the operation of the test subject and then hit enter. Be advised there is a four second delay after hitting the space bar before the cell begins to record data.
Note: Time can be tracked using [www.time.gov](http://www.time.gov) if connected to the internet, or the user may choose to remove the semi-colon in the jr2.m script code after the command:

\[ \text{tim(ii) = toc; \quad \%time in seconds from time tic was called (1/100 resolution)} \]

11) When the amount of time specified by the user has passed. The program will ask if the user would like to display the data for that test run. Type "1" and hit enter to graphically display the data on two separate figures. The first figure plots the axial forces and the second plots the moment forces, both are with respect to time.

12) The next prompt asks if the user would like to display basic statistical information. Be advised that the response characteristics of the cell will influence these values and close inspection of the two plotted figures is required in collaboration with the statistical readouts to assess their meaningfulness. For instance, a sudden load input to the cell will cause the cell to overshoot the measurement because of its inherent response characteristics. As long as the load is not removed for a few seconds, its true value can be interpreted otherwise the maximum value statistic will be misleading.

13) The next prompt asks if the user would like to store the data from the test run. Enter "1" and enter key to save the data.

14) The two last prompts ask the user if they would like to run another test and then reminds the user to re-name the .mat files for the stored data as they will be re-written in the next test run. Enter "1" to save data and then re-run the program, or enter to not save data and then terminate the program as instructed.

Caution: failure to rename the Raw_Data_Mat.mat files will result in the data from the previous test being lost.

**Instructions for ViewData.m:**

1) Open the folder C:\jr3pci\jr3pci_soft\jr3pci_app\Release
   Note: shortcut to this file is located on the desktop of user, “Super.”

2) Click on the Raw_Data_Mat.mat file containing the data of interest.
   Note: this file may have been renamed at an earlier time.

3) Click on the ViewData.m file to launch the .m-file window editor.

4) Highlight the entire code (about 20 lines) beginning with:
   \[ x = \text{size(Raw_data)}; \]
   Right click, and click “evaluate selection.” This will recreate the force/moment arrays for the 6 degrees of freedom and plot them on two separate figures just as would be done by the jr2.m program.

5) Repeat as desired for all previously collected sets of data.
Matlab Script Codes for ARSTS Matlab Programs

% 1804 Regenerative Supercirculation Test Stand Data Acquisition Program
% File name: jrl.m
% Purpose: Code to calibrate the Aeromobile Regenerative Supercirculation Test %
% Regenerative Supercirculation Thesis Project
% Author: Jason J. Fink
% Organization: Ohio University
% Date: 04/05/2004
% Additional Notes:
% BE SURE THE WORKING DIRECTORY IS:
C:\jr3pci\jr3pci_soft\jr3pci_app\Release

% note: the dll file: dll- Dynamic Link Library, must be in the same directory as the softcode (this m.file)
% initialize PCI Board using decimal address, obtained from hex decimal % address conversion, ie. 1762 hex dec = 5986 dec, 1111 hex dec = 4369 dec
% note: this conversion easily obtained using accessories/calculator

a = matjr3pci('init_jr3', 5986, 4369, 1, 1, 1); %This function detects your operating system (OS) and calls the jr3pci driver.
% If returns 0 - Your OS is Windows NT 4.0 / 2000 i.e. system operational
% If returns -95 Already in use i.e. close jr3 interface

raw_dat = [];
stat_dat = [];

disp('activating cell, please wait...');
pause(4);
%Cell needs to be addressed by program for internal electronics to begin proper operation
fprintf('
'); fprintf('
');
b = matjr3pci('read_offsets', 0); %Read offsets in use.
% Returns actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]

c = matjr3pci('read_ftdata', 6, 0); %Read offsets in use.
d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);
e=d+b; %offsets negate current readings to zero cell values
f = matjr3pci('set_offsets', 'e', 0); %cell now reset
g = matjr3pci('read_ftdata', 6, 0);
ci = matjr3pci('get_full_scales', 0);
%Reads actual/current full scales into [1x8] matrix

disp('Current full scale values are (lbs & in-lbs): ')
disp(' Fx  Fy  Fz  Mx  My  Mz'); format short
fprintf('%d', ci(1), ci(2), ci(3), ci(4), ci(5), ci(6)); fprintf('
');
disp('zero-ing cell...');
i=1;
for i=1:4; % Loop to ZERO the load cell
b = matjr3pci('read_offsets', 0); %Read offsets in use.
% Returns actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]

c = matjr3pci('read_ftdata', 6, 0); %Read offsets in use.
d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);
e=d+b;
f = matjr3pci('set_offsets', 'e', 0);
g = matjr3pci('read_ftdata', 6, 0);
i=i+1;
end

g = matjr3pci('read_ftdata', 6, 0);
Fx=(g(1)/16384)*ci(1); Fy=(g(2)/16384)*ci(2); Fz=(g(3)/16384)*ci(3);
Mx=(g(4)/16384)*ci(4); My=(g(5)/16384)*ci(5); Mz=(g(6)/16384)*ci(6);
disp('Zero-ed Force,Moment values in units of Lbs & in-Lbs are:
')
disp(' Fx  Fy  Fz  Mx  My  Mz')
fprintf('%4.2f', Fx, Fy, Fz, Mx, My, Mz); fprintf('
');

for i=1:32;
% the number in the command (for i=1:#;) above indicates the number of reading to be taken
    % # = 32 measurements for moment calibration
    % # = 40 measurements for axial force calibration
fprintf('W');
    format ('short'); fprintf('W'); fprintf('W');
check = input('Check that test subject is "Off" and there are no external loads,
then hit enter: '); disp('activating cell, please wait... ');}
pause(4);
% Cell needs to be addressed by program for internal electronics to begin proper operation
fprintf('
'); fprintf('
'); %Read offsets in use.
\texttt{b = matjr3pci('read_offsets', 0); \%Read offsets in use.}
\texttt{c = matjr3pci('read_ftdata', 6, 0); \%Read offsets in use.}
d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);
e=d+b; \texttt{\%offssets negate current readings to zero cell values}
f = matjr3pci('set_offsets', 'e', 0); \texttt{\%cell now reset}
g = matjr3pci('read_ftdata', 6, 0);

\texttt{b = matjr3pci('read_offsets', 0); \%Read offsets in use.}
\texttt{c = matjr3pci('read_ftdata', 6, 0); \%Read offsets in use.}
d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);
e=d+b; \texttt{\%offssets negate current readings to zero cell values}
f = matjr3pci('set_offsets', 'e', 0); \texttt{\%cell now reset}
g = matjr3pci('read_ftdata', 6, 0);

\texttt{ci = matjr3pci('get_full_scales', 0); \%Reads actual/current full_scales into [1x8] matrix}
disp('Current full scale values are (lbs & in-lbs): ')
disp(' Fx Fy Fz Mx My Mz'); format short
fprintf(' = %d',ci(1),ci(2),ci(3),ci(4),ci(5),ci(6)); fprintf('
'); fprintf('
');
time_to_read = 1; fprintf('
'); fprintf('
');
disp('There is 4 second delay before cell records data.');
num_to_read = time_to_read*8000; \texttt{\% 8khz capacity}
dat = zeros(num_to_read, 8);
\texttt{\%Pre-Initialize storage space for data arrays to allow accurate time sampling}
tim = zeros(num_to_read, 1);
\texttt{\%Initialize counter to record actual number of 6DOF arrays recorded}
disp('Begin operation of test subject and hit enter when you are ready to record data...')
reply = input('Measure F/T Data? Y/N [Y]: ','s');
if isempty(reply)
    reply = 'Y';
end
pause(4); \texttt{\%Power and initialize cell for operation}
disp('Clock starts now. Recording 6 DOF data....'); fprintf('
');
tic;  %starts timer in seconds (s)
while toc <= time_to_read
    dat(ii,:) = matr3pci('read_ftdata', 6, 0);
    %Read 6DOF F/T data into an array

    tim(ii) = toc;  %time in seconds from time tic was called (1/100 resolution)
    ii=ii+1; pause(1);
    % NOTE: Pause() can be adjusted to delay sampling
    % 0.01s is fastest if pause() is used
end
disp('Force balance data recorded. Processing...');  fprintf('
);

%NOTE: next character is "letter" l, and not "number" 1.
l=ii-1;  %counter started at time zero, but array begins with number 1
data = zeros(l, 8);  %NOTE: "letter" l, not #1) Pre-Initialize storage space
for data arrays actually recorded; get rid of zeros for plotting & data
manipulation
    time = zeros(l, 1);

% Following force/moment arrays are "letter l" deep by "number 1" long
Fx=zeros(l,1); Fy=zeros(l,1); Fz=zeros(l,1); Mx=zeros(l,1); My=zeros(l,1); Mz=zeros(l,1);

j=1; k=1;
while j < ii
    data(j,:) = dat(k,:);  %Create new data matrix to contain sampled data
    time(j) = tim(k);  %Create new time matrix to contain sampled time
    j=j+1; k=k+1;
end
m=1; n=l+1; o=(2*l)+1; p=(3*l)+1; q=(4*l)+1; r=(5*l)+1;

while m <= l
    %Bit to Force(lbs) & Bit to Moment(in-lbs) Conversions. 14 bit, 2^14 = 16384 byte res.
    Fx(m)=(data(m)/16384)*ci(1); Fy(m)=(data(n)/16384)*ci(2);
    Fz(m)=(data(o)/16384)*ci(3);
    Mx(m)=(data(p)/16384)*ci(4); My(m)=(data(q)/16384)*ci(5);
    Mz(m)=(data(r)/16384)*ci(6);
x = Fx(m); y = Fy(m);
m = m + 1; n = n + 1; o = o + 1; p = p + 1; q = q + 1; r = r + 1;
end

raw_dat = [raw_dat; [time, Fx, Fy, Fz, Mx, My, Mz]];

fprintf('
');
end

save Raw_Data raw_dat;
% Stores data in a MAT file and keeps all info and precision
% Can be loaded in Matlab for processing

matjr3pci('close_jr3')  % cell command to un-initialize cell
1804 Regenerative Supercirculation Test Stand Data Acquisition Program

File name: jr2.m

Purpose: Code to collect, analyze and manipulate force balance data as measured by the JR3 single channel, 6 DOF load cell

Regenerative Supercirculation Thesis Project

Author: Jason J. Fink

Organization: Ohio University

Date: 04/05/2004

Additional Notes:

BE SURE THE WORKING DIRECTORY IS:

C:\jr3pci\jr3pci_soft\jr3pci_app\Release

note: the dll file: dll- Dynamic Link Library, must be in the same directory as the soft code (this m.file)

initialize PCI Board using decimal address, obtained from hex decimal address conversion, ie. 1762 hex dec = 5986 dec, 1111 hex dec = 4369 dec

note: this conversion easily obtained using accessories/calculator

% This function detects your operating system (OS) and calls the jr3pci driver.
% If returns 0 - Your OS is Windows NT 4.0 / 2000 i.e. system operational
% If returns -95 Already in use i.e. close jr3 interface

fprintf(\n'); fprintf('n');
disp('Program to run Regenerative Supercirculation Test Stand')
run_program = input('Type "1" and hit enter to continue, or just enter to exit program: '); fprintf('n'); fprintf('n');
while run_program == 1
   clc; clear; %clear the cursor and all previously defined variables
   format ('short'); fprintf('n'); fprintf('n');
   check = input('Check that test subject is "Off" and there are no external loads, then hit enter: ');
   disp('Activating cell, please wait...'); pause(4); %Cell needs to be addressed by program for internal electronics to begin proper operation
   fprintf('n'); fprintf('n');
   b = matjr3pci('read_offsets', 0); %Read offsets in use.
   % Returns actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]
   c = matjr3pci('read_ftdata', 6, 0); %Read offsets in use.
   d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);
   e=d+b; %offsets negate current readings to zero cell values
   f = matjr3pci('set_offsets', 'e', 0); %cell now reset
   g = matjr3pci('read_ftdata', 6, 0);
ci = matjr3pci('get_full_scales', 0); %Reads actual/current full_scales into [1x8] matrix

disp('Current full scale values are (lbs & in-lbs): ')
disp(' Fx Fy Fz Mx My Mz'); format short
fprintf(' = %d',ci(1),ci(2),ci(3),ci(4),ci(5),ci(6)); fprintf('
'); fprintf('
');

% place code to reset full-scales here if
% desired
% end location of code to reset full-scales

% Portion of Program to record test data
disp('zero-ing cell...'); pause(1);
i=1;
for i=1:4; % Loop to ZERO the load cell
b = matjr3pci('read_offsets', 0); %Read offsets in use.
% Returns actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]
c = matjr3pci('read_ftdata', 6, 0); %Read offsets in use.
d(1)=c(1); d(2)=c(2); d(3)=c(3); d(4)=c(4); d(5)=c(5); d(6)=c(6);

e=d+b;
f = matjr3pci('set_offsets', 'e', 0);
g = matjr3pci('read_ftdata', 6, 0);
i=i+1;
end

g = matjr3pci('read_ftdata', 6, 0);
Fx=(g(1)/16384)*ci(1); Fy=(g(2)/16384)*ci(2); Fz=(g(3)/16384)*ci(3);
Mx=(g(4)/16384)*ci(4); My=(g(5)/16384)*ci(5); Mz=(g(6)/16384)*ci(6);
disp('Zero-ed Force,Moment values in units of Lbs & in-Lbs are: ')
disp(' Fx Fy Fz Mx My Mz')
fprintf(' = %.4f,Fx,Fy,Fz,Mx,My,Mz); fprintf('
'); fprintf('
');
disp('There is 4 second delay before cell records data.);
num_to_read = time_to_read*8000;  % 8kHz capacity
dat = zeros(num_to_read, 8);  % Pre-Initialize storage space for data arrays to allow accurate time sampling
tim = zeros(num_to_read, 1);
ii=1;  % Initialize counter to record actual number of 6DOF arrays recorded
disp('Begin operation of test subject and hit enter when you are ready to record data...')
reply = input('Measure F/T Data? Y/N [Y]: ','s');
if isempty(reply)
   reply = 'Y';
end
pause(4);  % Power and initialize cell for operation
disp('Clock starts now. Recording 6 DOF data....');  fprintf('
');
tic;  % Starts timer in seconds (s)
while toc <= time_to_read
   dat(ii,:) = matjr3pci('readAftdata', 6, 0);  % Read 6DOF F/T data into an array
tim(ii) = toc;  % Time in seconds from time tic was called (1/100 resolution)
ii=ii+1;  pause(1);  % NOTE: Pause() can be adjusted to delay sampling
   %% 0.01s is fastest if pause() is used
end
disp('Force balance data recorded. Processing....');  fprintf('
');

% NOTE: next character is "letter" l, and not "number" 1.
l=ii-1;  % Counter started at time zero, but array begins with number 1
data = zeros(l, 8);  % (NOTE: "letter" l, not #1) Pre-Initialize storage space for data arrays actually recorded; get rid of zeros for plotting & data manipulation
time = zeros(l, 1);
% Following force/moment arrays are "letter" l deep by "number" 1 long
Fx=zeros(l,1);  Fy=zeros(l,1);  Fz=zeros(l,1);  Mx=zeros(l,1);  My=zeros(l,1);  Mz=zeros(l,1);

% ii=1;
j=1;  k=1;
while j < ii
   data(j,:) = dat(k,:);  % Create new data matrix to contain sampled data only
time(j) = tim(k);  % Create new time matrix to contain sampled time only
   j=j+1;  k=k+1;
end
m=1;  n=l+1;  o=(2*l)+1;  p=(3*l)+1;  q=(4*l)+1;  r=(5*l)+1;
while m <= l

% Bit to Force(lbs) & Bit to Moment(in-lbs) Conversions. 14 bit, $2^{14} = 16384$ byte res.
Fx(m) = (data(m)/16384)*ci(1); Fy(m) = (data(n)/16384)*ci(2);
Fz(m) = (data(o)/16384)*ci(3);
Mx(m) = (data(p)/16384)*ci(4); My(m) = (data(q)/16384)*ci(5);
Mz(m) = (data(r)/16384)*ci(6);
x = Fx(m); y = Fy(m);
if abs(y) < 2.25  % Calibration of Fy axial force for loads under 2.25 lbs.
  if y > 0
    y = Fy(m) - 0.15;
  end
  if y < 0
    y = Fy(m) + 0.15;
  end
  Fy(m) = y;
end
if abs(x) < 2.25  % Calibration of Fx axial force for loads under 2.25 lbs.
  if x > 0
    x = Fx(m) - 0.15;
  end
  if x < 0
    x = Fx(m) + 0.15;
  end
  Fx(m) = x;
end
m = m + 1; n = n + 1; o = o + 1; p = p + 1; q = q + 1; r = r + 1;
end

disp('Plot data? If not just hit enter. ')
plot_opt = input('To display plot, type "1" and hit enter: '); fprintf('
');
fprintf('
');
if plot_opt == 1  % plotColor = {'b','g','r','c','m','y','k'};
  figure; plot(time, Fx, 'b', time, Fy, 'g', time, Fz, 'r'); grid minor; % Plot Axial Forces with time
  legend('Fx', 'Fy', 'Fz', -1)  %(-1) locates legend outside plot so all data is unobstructed
  axis('square'); set(gca, 'FontSize', 18); % axis([0 max(time) min() max()])
  xlabel('it time (s)'); ylabel('it Axial Forces (Lbs)');
figure; plot(time,Mx,'c',time,My,'m',time,Mz,'y'); grid minor; %Plot

Moments with time
legend('Mx','My','Mz','-1') %(-1) locates legend outside plot so all data is
unobstructed
axis('square'); set(gca,'FontSize',18); % axis([0 max(time) min() max()])

xlabel('it time (s)'); ylabel('it Moments (in-Lbs'));
end

%Basic Stats info created/stored here:
maxFx=max(Fx); minFx=min(Fx); maxFy=max(Fy); minFy=min(Fy);
maxFz=max(Fz); minFz=min(Fz);
maxMx=max(Mx); minMx=min(Mx); maxMy=max(My);
minMy=min(My); maxMz=max(Mz); minMz=min(Mz);

peakFx=maxFx; peakFy=maxFy; peakFz=maxFz; peakMx=maxMx;
peakMy=peakMx; peakMz=maxMz;
if abs(minFx)>maxFx %%%%\%\%\%\% IF statements to determine highest
nominal value
peakFx=minFx;end %%%%\%\%\%\% for each axis' Force/Moments
if abs(minFy)>maxFy
peakFy=minFy;end
if abs(minFz)>maxFz
peakFz=minFz;end
if abs(minMx)>maxMx
peakMx=minMx;end
if abs(minMy)>maxMy
peakMy=minMy;end
if abs(minMz)>maxMz
peakMz=minMz;end

av_Fx = mean(Fx); av_Fy = mean(Fy); av_Fz = mean(Fz);
av_Mx = mean(Mx); av_My = mean(My); av_Mz = mean(Mz);
disp('View max, min, avg statistics? If not just hit enter."
stats_opt = input('To view stats, type "1" and hit enter: '); fprintf('n');

if stats_opt == 1
    disp('AVERAGE Force, Moment values in units of Lbs & in-Lbs are:')
    disp('Fx_avg Fy_avg Fz_avg Mx_avg My_avg Mz_avg')
    fprintf(‘ = %.4f,av_Fx,av_Fy,av_Fz,av_Mx,av_My,av_Mz); fprintf(‘n’);
    fprintf(‘n’);
disp('PEAK Force, Moment values in units of Lbs & in-Lbs are:')
disp('  Fx_peak  FY_peak  Fz_peak  Mx_peak  My_peak  Mz_peak')
fprintf('% .4f,peakFx,peakFy,peakFz,peakMx,peakMy,peakMz);
end
% .4f above limits output to 4 dec places

disp('Would you like to store data from this test? ');
store_t = input('If not, hit enter. Otherwise, type "1" and hit enter: ');
if store_t == 1
  Raw_data = [time, Fx, Fy, Fz, Mx, My, Mz];
  Stat_data = [av_Fx, av_Fy, av_Fz, av_Mx, av_My, av_Mz,
               peakFx,peakFy,peakFz,peakMx,peakMy,peakMz];
  %Stores data in a MAT file and keeps all info and precision
  save Raw_Data_Mat Raw_data;  save Stat_Data_Mat Stat_data; %Can be
  % loaded in Matlab for processing
end

%%% place code to create text data files here
%%% end location of code
end
fprintf('%n');
run_program = 2;
run_program = input('Type "1" and hit enter to RE-RUN program, or just "enter"
to quit: ');  fprintf('%n');  fprintf('%n');
if run_program ==1
  disp('You will need to rename data files in:
C:\jr3pci\jr3pci_soft\jr3pci_app\Release. ');
  disp('This is to retain previous test data. ');
  check = input('Re-name Text & Mat files, "Raw-data" & "Stat_data" and hit enter... ')
end

datajr3pci('close Jr3')  %cell command to un-initialize cell
% 1804 Regenerative Supercirculation Test Stand Data Acquisition Program
% File name: ViewData.m
% Purpose: % Code to re-plot old test data
% Regenerative Supercirculation Thesis Project
% Author: Jason J. Fink
% Organization: Ohio University
% Date: 04/05/2004

%%%%%% INSTRUCTIONS %%%%%%
% First load the mat file of interest in the matlab command window
% Now, execute the script below to plot 6DOF Data from any previous test
% from which data was created using either the jrl or jr8k script codes
% NOTE this is easily done by highlighting all below, right click,
% and click to "evaluate selection"

x = size(Raw_data);
x1=x(1); % x1 is number of data points taken; number of seconds of test time for jrl.m program
x2=x(2);
total=x1*x2;
n0=x1+1; o=(2*x1)+1; p=(3*x1)+1; q=(4*x1)+1; r=(5*x1)+1; s=(6*x1)+1;
for i=1:x1
    time(i)=Raw_data(i);
    Fx(i)=Raw_data(n); Fy(i)=Raw_data(o); Fz(i)=Raw_data(p);
    Mx(i)=Raw_data(q); My(i)=Raw_data(r); Mz(i)=Raw_data(s);
    n=n+1; o=o+1; p=p+1; q=q+1; r=r+1; s=s+1;
end

figure; plot(time,Mx,'b',time,My,'g',time,Mz,'r'); grid minor; %Plot Moments with time
legend('Mx','My','Mz','-1') %(-1) locates legend outside plot so all data is unobstructed
axis('square'); set(gca,'FontSize',18); % axis([0 max(time) min() max()])
xlabel('it time (s)'); ylabel('it Moments (in-Lbs)');

figure; plot(time,Fx,'b',time,Fy,'g',time,Fz,'r'); grid minor; %Plot Axial Forces with time
legend('Fx','Fy','Fz','-1') %(-1) locates legend outside plot so all data is unobstructed
axis('square'); set(gca,'FontSize',18); % axis([0 max(time) min(time)])
xlabel('time (s)'); ylabel('Axial Forces (Lbs)');

% 1804 Regenerative Supercirculation Test Stand Data Acquisition Program
% File name: UnusedCode.m
% Purpose: Unused code from working jrl, jr8k m-files
% Regenerative Supercirculation Thesis Project
% Author: Jason J. Fink
% Organization: Ohio University
% Date: 04/05/2004

%%%%
Code to reset full scales-original testing showed it did not affect
% accuracy of test results or have any change on results
However,
should the user find a need to reset full scales, just past the
following commented code into the specified location in the jr
program scripts

%%%% Code to reset full-scales
%%%%
disp('Would you like to change the full scale values?'); fprintf('
');
reply_1 = input('If not, hit enter. Otherwise, type "1" and hit enter: ');
if reply_1 == 1
decision = 1;
while decision == 1
Fx=input('Enter new full scale Fx between 1 and 50 lbs and hit enter: ');
while Fx<1 || 50<Fx
Fx=input('Enter new full scale Fx BETWEEN 1 and 50 lbs!! and hit enter: ');
end
Fy=input('Enter new full scale Fy between 1 and 50 lbs and hit enter: ');
while Fy<1 || 50<Fy
Fy=input('Enter new full scale Fy BETWEEN 1 and 50 lbs!! lbs and hit enter: ');
end
Fz=input('Enter new full scale Fz between 1 and 100 lbs and hit enter: ');
while Fz<1 || 100<Fz
Fz=input('Enter new full scale Fz BETWEEN 1 and 100 lbs!! and hit enter: ');
}
end
Mx=input('Enter new full scale Mx between 1 and 130 in-lbs and hit enter: ');
while Mx<1 || 130<Mx
    Mx=input('Enter new full scale Mx BETWEEN 1 and 130 in-lbs!!! and hit enter: ');
end
My=input('Enter new full scale My between 1 and 130 in-lbs and hit enter: ');
while My<1 || 130<My
    My=input('Enter new full scale My BETWEEN 1 and 130 in-lbs!!! and hit enter: ');
end
Mz=input('Enter new full scale Mz between 1 and 130 in-lbs and hit enter: ');
while Mz<1 || 130<Mz
    Mz=input('Enter new full scale Mz BETWEEN 1 and 130 in-lbs!!! and hit enter: ');
end
b=[Fx, Fy, Fz, Mx, My, Mz]; %array containing new user-defined full scales

v=matjr3pci('set_full_scales', 'b', 0); % user-defined full scales set
ci = matjr3pci('get_full_scales', 0); %call & define ci: full scales from cell (to be certain they have been set)
    fprintf('
'); fprintf('
');
    disp('You have chosen the following full-scale values: ');
    disp(' Fx Fy Fz Mx My Mz');
    fprintf(' = %d',ci(1),ci(2),ci(3),ci(4),ci(5),ci(6));
    fprintf('
');
    decision=input('Enter 1 to change these values again, otherwise just enter: ');
end

%%% The following can also be added to the jr scripts if the user desires
%%% to be able to create test data in a text file for later analysis

%%% code to create text files  %%%%
    Raw_Data_Text = fopen('Raw_data.txt','w'); %Stores data in a text file only to 4 places, can be opened in excel
    fprintf(Raw_Data_Text,'%.4f %.4f %.4f %.4f %.4f %.4f
', Raw_data);
fclose(Raw_Data_Text);

Stat_Data_Text = fopen('Stat_data.txt','w'); %Stores data in a text file only to 4 places, can be opened in excel
fprintf(Stat_Data_Text,'%.4f %.4f %.4f %.4f %.4f %.4f %.4f %.4f %.4f %.4f 
Sta

close(Stat_Data_Text);
% Additional Unused code from original file
% code contains basic statistical analysis commands
% disp('Would you like to re-zero the stand?')
fprintf('
_reply = input('If not, hit enter. Otherwise, type "1" and hit enter:');
if reply_1 == 1
    i=1; %Loop to ZERO the load cell
    b = matjr3pci('read_offsets', 0); %Read offsets in use.
    % Returns actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]
    c = matjr3pci('read_ftdata', 6, 0); %Read offsets in use.
    d1 = c(1); d2 = c(2); d3 = c(3); d4 = c(4); d5 = c(5); d6 = c(6);
    e = d + b;
    f = matjr3pci('set_offsets', 'e', 0);
    g = matjr3pci('read_ftdata', 6, 0);
    i = i + 1;
end
end
plot('time(s)',time, 'force(lb)',Fx); plot(time, Fy); plot(time, Fz);
plot(time, Mx); plot(time, My); plot(time, Mz);

reply = input('Hit enter for statistical analysis: ', 's');
if isempty(reply)
    reply = 'Y';
end
% mean and standard deviation calcs using matlab functions
fx_mean = mean(fx); fy_mean = mean(fy); fz_mean = mean(fz); mx_mean = mean(mx);
my_mean = mean(my); mz_mean = mean(mz);
\[ f_{x} \text{stddev}=\text{std}(f_{x}); \quad f_{y} \text{stddev}=\text{std}(f_{y}); \quad f_{z} \text{stddev}=\text{std}(f_{z}); \quad m_{x} \text{stddev}=\text{std}(m_{x}); \quad m_{y} \text{stddev}=\text{std}(m_{y}); \quad m_{z} \text{stddev}=\text{std}(m_{z}); \]

\% Fx Confidence Interval calculations for recorded values
\[
\begin{align*}
\text{rtn} & = n^{0.5}; \\
\text{fxstdd}_{\text{error}} & = \text{fxstddev}/\text{rtn} \\
\text{fxeta} & = 0.0045/\text{fxstdd}_{\text{error}} \\
\text{fxerror}_{\text{margin}} & = 4.587*\text{fxstdd}_{\text{error}}; \\
\text{fxcon}_{\text{int1}} & = \text{fx}_{\text{mean}}+\text{fxerror}_{\text{margin}}; \\
\text{fxcon}_{\text{int2}} & = \text{fx}_{\text{mean}}-\text{fxerror}_{\text{margin}}; \\
\text{fxcon}_{\text{int}} & = [\text{fxcon}_{\text{int1}}, \text{fxcon}_{\text{int2}}]
\end{align*}
\]

\% % Fy Confidence Interval calculations for recorded values
\[
\begin{align*}
\text{rtn} & = n^{0.5}; \\
\text{fystdd}_{\text{error}} & = \text{fystddev}/\text{rtn} \\
\text{fyeta} & = 0.0045/\text{fystdd}_{\text{error}} \\
\text{fyerror}_{\text{margin}} & = 4.587*\text{fystdd}_{\text{error}}; \\
\text{fycon}_{\text{int1}} & = \text{fy}_{\text{mean}}+\text{fyerror}_{\text{margin}}; \\
\text{fycon}_{\text{int2}} & = \text{fy}_{\text{mean}}-\text{fyerror}_{\text{margin}}; \\
\text{fycon}_{\text{int}} & = [\text{fycon}_{\text{int1}}, \text{fycon}_{\text{int2}}]
\end{align*}
\]

\% % Fz Confidence Interval calculations for recorded values
\[
\begin{align*}
\text{rtn} & = n^{0.5}; \\
\text{fzstdd}_{\text{error}} & = \text{fzstddev}/\text{rtn} \\
\text{fzeta} & = 0.0045/\text{fzstdd}_{\text{error}} \\
\text{fzerror}_{\text{margin}} & = 4.587*\text{fzstdd}_{\text{error}}; \\
\text{fzcon}_{\text{int1}} & = \text{fz}_{\text{mean}}+\text{fzerror}_{\text{margin}}; \\
\text{fzcon}_{\text{int2}} & = \text{fz}_{\text{mean}}-\text{fzerror}_{\text{margin}}; \\
\text{fzcon}_{\text{int}} & = [\text{fzcon}_{\text{int1}}, \text{fzcon}_{\text{int2}}]
\end{align*}
\]

\% % Mx Confidence Interval calculations for recorded values
\[
\begin{align*}
\text{rtn} & = n^{0.5}; \\
\text{moxstdd}_{\text{error}} & = \text{moxstddev}/\text{rtn} \\
\text{mxeta} & = 0.0045/\text{moxstdd}_{\text{error}} \\
\text{mxerror}_{\text{margin}} & = 4.587*\text{moxstdd}_{\text{error}}; \\
\text{mxcon}_{\text{int1}} & = \text{mx}_{\text{mean}}+\text{mxerror}_{\text{margin}}; \\
\text{mxcon}_{\text{int2}} & = \text{mx}_{\text{mean}}-\text{mxerror}_{\text{margin}}; \\
\text{mxcon}_{\text{int}} & = [\text{mxcon}_{\text{int1}}, \text{mxcon}_{\text{int2}}]
\end{align*}
\]

\% % My Confidence Interval calculations for recorded values
\[
\begin{align*}
\text{rtn} & = n^{0.5}; \\
\text{mystdd}_{\text{error}} & = \text{mystddev}/\text{rtn}
\end{align*}
\]
myeta=0.0045/mystdd_error
myerror_margin=4.587*mystdd_error;
mycon_int1=my_mean+myerror_margin;
mycon_int2=my_mean-myerror_margin;
mycon_int=[mycon_int1,mycon_int2]

Yo

%% Mz Confidence Interval calculations for recorded values
rtn=n^0.5;
mzstdd_error=mzstdddev/rtm
mzeta=0.0045/mzstdd_error
mzerror_margin=4.587*mzstdd_error;
mzcon_int1=mz_mean+mzerror_margin;
mzcon_int2=mz_mean-mzerror_margin;
mzcon_int=[mzcon_int1,mzcon_int2]

matjr3pci('close_jr3')  %This function closes the handle to the JR3 PCI board driver.
Appendix E
2/9/2004

JR3 PCI Software – how to install
Win98

If you have previous versions installed.

1. Uninstall all the JR3PCI boards you have installed. Go to the Window control Panel, System, Hardware, Device Manager. Delete all the JR3PCI boards. Find them under JNP.

2. Go to the Windows INF directory and remove any INF files related with JR3PCI boards.

3. Go to Windows\system32\drivers and look for “windrvr.sys file and Wdpnp.sys. DELETE both files.


5. Next, you remove the following registry keys using “regedit”. Extreme caution must be taken any time you work with the registry!

6) HKEY_LOCAL_MACHINE\SYSTEM\CURRENTCONTROLSET\SERVICES\Windriver
7) HKEY_LOCAL_MACHINE\SYSTEM\CURRENTCONTROLSET\SERVICES\Wdpnp

6. Power off and restart your system.

First time installation

6.1 Create a Folder under the name jr3pci on the C:\ drive.

7. Download the software from our site into the jr3pci folder you created, (go to jr3.com, and then click on software, J. Norberto of the University of Coimbra.).

On J. Norberto’s site click on software, then click on JR3PCI. (This is a zip file, you will need a winzip type program.)

7.1 Download the .dll file for the jr3pci into the jr3pci folder you created in 6.1 and be sure to extract the .dll file from the folder you downloaded and place it directly in the jr3pci folder you created in 6.1.
8. Unzip the jr3pci_soft.zip file within the jr3pci folder you created in instruction 6.1.

9. Use the utility wdreg to install windriver’s kernel on to the target computer. The command should be:
   C:\jr3pci\jr3pci_soft\driver\wdreg -inf
c:\jr3pci\jr3pci_soft\driver\pci\windrvr6.inf install
   The full path must be used for files, wdreg and windrvr.inf. ‘I.E.
c:\jr3pci\jr3pci_soft\driver’, is the full path for wdreg.

10. Go to Start, run and type in “command” to go to a dos screen. Type in the above command line, making any changes necessary. This will take a few seconds to run and should give you an “install completed successfully” message.

11. This will install windrvr6 driver...

   **Now proceed to install your board(s)**

12. Now power down and install your board(s). If board(s) are already installed go to step 14.

13. Power up system, windows should find new hardware. Skip to step 15.

14. In the Device Manager right click on “Scan for New Hardware”. Windows will find your JR3PCI boards.

15. When prompted, select “search for a suitable driver”, select “specify a location”, and give the location of the INF file corresponding to your board. You can find the under the drivers folder of the JR3PCI Folder. Click on the board style you have.

16. Windows will install your drivers.

17. If you have multiple boards, windows will again find new hardware.

18. Repeat step 15 till all boards are installed.

19. Your installation should be complete.

20. There are additional installation hints on Norberto’s web site.
Appendix F
MATJR3PCI

for MATLAB 5

Version 4.01
J. Norberto Pires
Mechanical Engineering Department
Robotics and Control Laboratory
3030 Coimbra Portugal
norbento@robotics.dem.uc.pt
http://robotics.dem.uc.pt/norbento/

MATJR3PCI is a module of a general Toolbox for Robotics & Automation applications. It enables real access to JR3 Force/Torque sensors from within MATLAB 5. The complete toolbox includes support for:
- ABB S4 robots and attached sensors;
- Siemens VS710 CCD Cameras;
- Z-World BL1100 controllers;
- JR3 Force/Torque


Introduction
This document briefly introduces module MATJR3PCI of the MATROBCOM toolbox.

It is mainly a user guide written for new users.

MATROBCOM is a collection of MEX files that enable access to several Robotic and Automation equipment's, including robots, sensors, actuators, CCD cameras, etc., from within MATLAB 5 (the popular math software package from MathWorks Inc).

Each individual equipment has its own module. This document describes module MATJR3, designed to operate with JR3 force/torque sensors (using PCI DSP receivers).

The module MATJR3PCI can be used with MATLAB 5 under Win32 operating systems (Windows). Other options are available with the same basic functionality, including ActiveX controls, DDE server and a C++ library. In fact, the current MEX functions were built using the C++ library.

Authorization
You are authorized to use this module on whatever applications you desire, and you may distribute it freely, taking in consideration that the module is supplied "AS IS". You are not authorized to remove the module copyright notices and/or sell the module to others.
Any reference to this work should be done as:

**Author:** J. Norberto Pires  
**Title:** MATJR3PCI for MATLAB 5  
**Publication:** Internal Report N° GCG.RCL.001.02  
**Institution:** Mechanical Engineering Department, Robotics and Control Laboratory, University of Coimbra, Portugal.  
**Date:** 12 of December 2001.

**Functions included in the MATJR3PCI module**

Current version (2.01) includes the following functions:
- 'init_jr3', 'close_jr3', 'read', 'write', 'system_warnings', 'system_errors',  
- 'command', 'get_threshold_status', 'reset_threshold', 'read_ftdata',  
- 'set_transforms', 'use_transforms', 'read_offsets', 'set_offsets',  
- 'change_offset_num', 'reset_offsets', 'use_offset', 'peak_data',  
- 'peak_data_reset',  
- 'read_peaks', 'bit_set', 'set_full_scales', 'get_full_scales',  
- 'get_recommended_full_scales', 'sensor_info'.

With these functions users can explore completely the sensor and receiver board functionality. Other functions can be built using the presented basic functions. In the next sections each function will be explained, considering that you are using the MATLAB 5 shell. Please refer also to the JR3 Users Manual for details and to the 'jr3pci_ft.hl' file that you can download from http://www.dem.uc.pt/norberto/jr3pci/.

**INIT_JR3**

This function detects your operating system (OS) and calls the jr3pci driver.

**Usage:**

```matlab
a = matjr3pci('init_jr3', vendor_ID, device_ID, n_board, n_processors, downl);
```

**Input Value:**

- `vendor_ID` – PCI vendor ID.  
- `device_ID` – PCI device ID.  
- `n_board` – board number (1 for single board system).  
- `n_processors` – number of processors in the board  
- `downl` – Download DSP Code (1) or not (0).

**Note:** Values should be entered in decimal format. If user sets ZERO on any parameter, then default values will be used:

- `Vendor_ID` = 0x1762  
- `Device_ID` = 0x111  
- `Number_board` = 1

**Return Value:**

- 0 - Your OS is Windows NT 4.0 / 2000  
- 1 - Your OS is Windows 95/98  
- 2 - Your OS is Windows 3.X  
- 3 - Your OS is unknown - SERIOUS ERROR  
- -90 – IDM DSP File was NOT FOUND (it should be in the same directory of the running program)  
- -91 – Handle to Windrv failed ... run wdreg.  
- -92 – Windrv version is incorrect.  
- -93 – PCI card not found.
-94 – Card out of range.
-95 – Error locking device. Already in use.
-96 – Download error.

CLOSE_JR3
This function closes the handle to the JR3 PCI board driver.

Usage:
matjr3pci('close_jr3');

Return Value:
0 – If successful

READ
Reads from a receiver board memory address.

Usage:
a = matjr3pci('read', addr, pnum);

Input Value:
addr - address to read.
pnum – processor number

Returned Value:
Data from addressed considered as 'short'.

WRITE
Writes to a receiver board memory address.

Usage:
matjr3pci('write', addr, data, pnum);

Input Value:
addr - address to write.
data - value to write ('short')
pnum – processor number

Returned Value:
None

SYSTEM_WARNINGS
Reads system saturation warnings (board memory address WARNINGS).

Usage:
a = matjr3pci('system_warnings', pnum);

Input Value:
pnum – processor number

Returned Value:
A 1x6 matrix with the values of the warnings:
[warning.fx_sat, warning.fy_sat, warning.fz_sat, warning.mx_sat, warning.my_sat warning.mz_sat]

SYSTEM_ERRORS
Reads system errors (board memory address ERRORS).

Usage:
a = matjr3pci('system_errors', pnum);

Input Value:
pnum – processor number

Returned Value:
A 1x12 matrix with the values of the errors:
[error.fx_sat, error.fy_sat, error.fz_sat, error.mx_sat, error.my_sat error.mz_sat,
error.memory_error, error.sensor_change, error.system_busy,
error.cal_crc_bad,
error.watch_dog2, error.watch_dog

COMMAND
Commands JR3 receiver board.
Usage:
a=matjr3pci('command', addr, data, pnum);
Input Value:
addr - address to command.
data - value to command ('short')
pnum – processor number
Returned Value:
0- Command was successful.
1- Command Failed.

GET_THRESHOLD
Gets the value of the threshold bits (board address THRESHOLD).
Usage:
a=matjr3pci('get_threshold', pnum);
Input Value
pnum – processor number
Return Value:
Value of the threshold bits.

RESET_THRESHOLD
Reset the threshold bits.
Usage:
matjr3pci('reset_threshold', pnum);
Input Value
pnum – processor number
Returned Value:
None

READ_FTDATA
Reads force/torque data from receiver board.
Usage:
a=matjr3pci('read_ftdata', filter, pnum);
Input Value:
filter - filter number (0...6).
pnum - processor number
Returned Value:
A 1x8 matrix with the force/torque data: [fx, fy, fz, mx, my, mz, v1, v2]

SET_TRANSFORMS
Sets a new transformation definition.
Usage:
a=matjr3pci('set_transforms', index, name, pnum);
Input Value:
index - Index of transformation (see JR3 Users Manual)
name - name of matrix existing in the MATLAB workspace that contains the
transformation structure. Use a row matrix with the same structure as defined in
the JR3 Users Manual, i.e., [link_type, link_amount, ..., 0]
pnum – processor number

Returned Value:
0 - If successful.

USE_TRANSFORMS
Selects the transformation to use.
Usage:
a = matjr3pci('use_transforms', index, pnum);
Input Value:
index - index of transform to use.
pnum – processor number
Returned Value:
0 - if successful.

READ_OFFSETS
Read offsets in use.
Usage:
a = matjr3pci('read_offsets', pnum);
Input Value:
pnum – processor number
Returned Value:
Actual offsets using a six_axis_array matrix: [fx, fy, fz, mx, my, mz]

SET_OFFSETS
Set actual offsets, using the current offset index.
Usage:
a = matjr3pci('set_offsets', name, pnum);
Input Value:
name - name of matrix existing in the MATLAB workspace that contains the offsets. Use a row matrix with the same structure as defined in the JR3 Users Manual, i.e., [fx, fy, fz, mx, my, mz]
pnum – processor number
Returned Value:
0 - If successful.

CHANGE_OFFSET_NUM
Changes actual offset index (num).
Usage:
a = matjr3pci('change_offset_num', index, pnum);
Input Value:
index - Offset_num to use.
pnum – processor number
Returned Value:
0 - If successful.

RESET_OFFSETS
Set actual offsets to the current values read from FILTER_2.
Usage:
a = matjr3pci('reset_offsets', pnum);
Input Value:
pnum – processor number
Returned Value:
0 - If successful.

USE_OFFSET
Changes actual offsets to the ones defined by index.
Usage:
a=matjr3pci('use_offset', index, pnum);
Input Value:
index - Offset_num to use.
pnum - processor number
Returned Value:
0 - If successful.

PEAK_DATA
Set address to watch for peaks.
Usage:
a=matjr3pci('peak_data', filter, pnum);
Input Value:
filter - filter to use (0...6).
pnum - processor number
Return Value:
0 - Successful.

PEAK_DATA_RESET
Set address to watch for peaks and resets internal values to current data.
Usage:
a=matjr3pci('peak_data_reset', filter, pnum);
Input Value:
filter - filter to use (0...6).
pnum - processor number
Return Value:
0 - Successful.

READ_PEAKS
Reads current peak values.
Usage:
a=matjr3pci('read_peak', type, pnum);
Input Value:
type - Minimum (0) or Maximum values (1).
pnum - processor number
Return Value:
Peak data using a force_array matrix: [fx, fy, fz, mx, my, mz, v1, v2 ].

BIT_SET
Set bits on defined bit-map
Usage:
a=matjr3pci('bit_set', bit_map, addr, pnum);
Input Value:
bit_map - value to set on bit map.
addr - address of bit map on memory board.
pnum - processor number
Output Value:
0 - If successful.

SET_FULL_SCALES
Set JR3 Full_Scales.
Usage:
a=matjr3pci('set_offsets', name, pnum);
Input Value:
name - name of matrix existing in the MATLAB workspace that contains the full_scales. Use a row matrix with the same structure as defined in the JR3 Users Manual, i.e., [fx, fy, fz, mx, my, mz].
pnum – processor number

Returned Value:
0 - If successful.

GET_FULL_SCALES
Reads actual full_scales.

Usage:
a=matjr3pci('get_full_scales', pnum);

Input Value
pnum – processor number

Returned Value:
Actual full scales using a force_array matrix: [fx, fy, fz, mx, my, mz, v1, v2].

GET_RECOMMENDED_FULL_SCALES
Reads recommended full_scales.

Usage:
a=matjr3pci('get_full_scales', type, pnum);

Input Value
type - Minimum Full Scales (0) or Maximum Full Scales (1).
pnum – processor number

Returned Value:
Recommended full scales using a force_array matrix: [fx, fy, fz, mx, my, mz, v1, v2].

SENSOR_INFO
Reads information from the sensor and from the receiver board. Use this function to test your setup.

Usage:
matjr3pci('sensor_info', pnum);

Input Value
pnum – processor number

Returned Value:
Sensor Information (model, serial number, copyright, etc.).

© J. Norberto Pires, Robotics and Control Laboratory, Mechanical Engineering Department, University of Coimbra, 2001. MATJR3PCI for MATLAB 5 is protected by copyright international laws. All rights reserved. MATLAB is a registered trademark of MathWorks Inc (USA), http://www.mathworks.com. Windows is a registered trademark of Microsoft Corporation (USA), http://www.microsoft.com.
Appendix G
Additional Stand Calibration Data and Figures

Additional Data Sweep and Calibration Test Results:

![Graph showing measured moment vs. actual moment](image-url)
Additional Calibration Data Plots
Test run to verify multiple loadings with time
### Test 4: Factory Shipped Full Scale test of Fx,Fz,Mx,Mz

**Multiple Loadings**

<table>
<thead>
<tr>
<th>time (s)</th>
<th>Fx (lb)</th>
<th>Fz (lb)</th>
<th>Mx (in-lb)</th>
<th>Mz (in-lb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10-20</td>
<td>10</td>
<td></td>
<td>-11.02</td>
<td></td>
</tr>
<tr>
<td>20-30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30-40</td>
<td>-10</td>
<td></td>
<td>-11.02</td>
<td></td>
</tr>
<tr>
<td>40-50</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50sec</td>
<td></td>
<td>10.00</td>
<td>88.18</td>
<td></td>
</tr>
<tr>
<td>60-70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1min-10</td>
<td></td>
<td>10.00</td>
<td>-88.18</td>
<td></td>
</tr>
<tr>
<td>80-90</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90-100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1min-40</td>
<td></td>
<td></td>
<td>88.18</td>
<td>80</td>
</tr>
<tr>
<td>110-120</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>120-130</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2min-10</td>
<td></td>
<td></td>
<td>-88.18</td>
<td>80</td>
</tr>
<tr>
<td>140-150</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>150-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2min-40</td>
<td></td>
<td></td>
<td>88.18</td>
<td>-80</td>
</tr>
<tr>
<td>170-180</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>180-190</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3min-20</td>
<td></td>
<td></td>
<td>-88.18</td>
<td>-80</td>
</tr>
<tr>
<td>200-210</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Test run 4, multiple combined loadings
Test run 4, multiple combined loadings
Test run 13, check low scale Fy
Additional testing of low scale Fx
Additional testing of low scale Fx
Additional testing of low scale Fy