A FULLY AUTOMATED STAGE FOR OPTICAL WAVEGUIDE MEASUREMENTS

E. THOMAS KENSKY

WRIGHT LABORATORY
MATERIALS DIRECTORATE
3005 P STREET, SUITE 6
WRIGHT-PATTERSON AFB OH 45433-7707

SEPTEMBER 93

FINAL REPORT FOR 09/09/92-09/09/93

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.
NOTICE

WHEN GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA ARE USED FOR ANY PURPOSE OTHER THAN IN CONNECTION WITH A DEFINITELY GOVERNMENT-RELATED PROCUREMENT, THE UNITED STATES GOVERNMENT INCURS NO RESPONSIBILITY OR ANY OBLIGATION WHATSOEVER. THE FACT THAT THE GOVERNMENT MAY HAVE FORMULATED OR IN ANY WAY SUPPLIED THE SAID DRAWINGS, SPECIFICATIONS, OR OTHER DATA, IS NOT TO BE REGARDED BY IMPLICATION, OR OTHERWISE IN ANY MANNER CONSTRUED, AS LICENSING THE HOLDER, OR ANY OTHER PERSON OR CORPORATION; OR AS CONVEYING ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY IN ANY WAY BE RELATED THERETO.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

E. THOMAS KENSKY
E. THOMAS KENSKY,
Electronic & Optical Mat'ls Branch
Electromagnetic Mat'ls & Surv. Div.

ROBERT L. DENISON, Chief
ROBERT L. DENISON, Chief
Electronic & Optical Mat'ls Branch
Electromagnetic Mat'ls & Surv. Div.

WILLIAM R. WOODY, Chief
WILLIAM R. WOODY, Chief
Electromagnetic Mat'ls & Surv. Div.
Materials Directorate

IF YOUR ADDRESS HAS CHANGED, IF YOU WISH TO BE REMOVED FROM OUR MAILING LIST, OR IF THE ADDRESSEE IS NO LONGER EMPLOYED BY YOUR ORGANIZATION PLEASE NOTIFY WL/MLPQ, BLDG 651, 3005 P ST STE 6, WPAFB OH 45433-7707, TO HELP MAINTAIN A CURRENT MAILING LIST.

COPIES OF THIS REPORT SHOULD NOT BE RETURNED UNLESS RETURN IS REQUIRED BY SECURITY CONSIDERATIONS, CONTRACTUAL OBLIGATIONS, OR NOTICE ON A SPECIFIC DOCUMENT.
**Title and Subtitle:** A Fully Automated Stage for Optical Waveguide Measurements

**Authors:** E. Thomas Kensky

**Performing Organization:**
- Materials Directorate
- Wright Laboratory
- Air Force Materiel Command
- Wright Patterson AFB OH 45433-7734

**Sponsoring/Monitoring Agency:**
- Materials Directorate
- Wright Laboratory
- Air Force Materiel Command
- Wright Patterson AFB OH 45433-7734

**Abstract (Maximum 200 words):**

(Blank)

**Subject Terms:**
- Waveguides, Nonlinear Optics, Optical Loss Measurements

**Security Classification:**
- Unclassified
- Unclassified
- Unclassified

**Distribution/Availability Statement:**
Approved for public release; distribution is unlimited.
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2.0 EXPERIMENTAL BACKGROUND</td>
<td>3</td>
</tr>
<tr>
<td>2.1 THE OUT-OF-PLANE TECHNIQUE</td>
<td>3</td>
</tr>
<tr>
<td>2.2 THE THREE-PRISM TECHNIQUE</td>
<td>5</td>
</tr>
<tr>
<td>3.0 SYSTEM REQUIREMENTS</td>
<td>8</td>
</tr>
<tr>
<td>3.1 SPECIFIC REQUIREMENTS</td>
<td>8</td>
</tr>
<tr>
<td>4.0 EXPERIMENTAL APPARATUS</td>
<td>11</td>
</tr>
<tr>
<td>5.0 SYSTEM DESIGN</td>
<td>14</td>
</tr>
<tr>
<td>5.1 MECHANICAL HARDWARE</td>
<td>14</td>
</tr>
<tr>
<td>5.1.1 THE RAIL TABLE</td>
<td>14</td>
</tr>
<tr>
<td>5.1.2 THE BASE PLATE</td>
<td>15</td>
</tr>
<tr>
<td>5.1.3 THE INPUT, OUTPUT, AND MEASUREMENT ARMS</td>
<td>17</td>
</tr>
<tr>
<td>5.2 ELECTRONICS AND OPTICS</td>
<td>19</td>
</tr>
<tr>
<td>5.2.1 PRISMS AND DETECTORS</td>
<td>20</td>
</tr>
<tr>
<td>5.2.2 THE LOAD CELLS</td>
<td>25</td>
</tr>
<tr>
<td>5.2.3 THE MEASUREMENT STEPPER MOTOR</td>
<td>27</td>
</tr>
<tr>
<td>6.0 EXPERIMENTAL RESULTS</td>
<td>30</td>
</tr>
<tr>
<td>7.0 REFERENCES</td>
<td>32</td>
</tr>
<tr>
<td>APPENDIX A - ELECTRONIC SCHEMATICS</td>
<td>33</td>
</tr>
<tr>
<td>APPENDIX B - MECHANICAL DRAWINGS</td>
<td>37</td>
</tr>
<tr>
<td>APPENDIX C - SOFTWARE LISTING</td>
<td>84</td>
</tr>
</tbody>
</table>
# LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Out-of-Plane Scattering Technique Schematic</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Three-Prism Measurement Technique Schematic</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Overall Waveguide Measurement System Schematic</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>Photograph of the Waveguide Stage</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>The Waveguide Stage Base Plate</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>The Input/Output Arm</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>Input/Output Arm Schematic</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>Prism Holder and Prism</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>Three-Prism Detector Holder Arrangement</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>Out-of-Plane Detector Holder Arrangement</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>Detector Electronics</td>
<td>24</td>
</tr>
<tr>
<td>12</td>
<td>Load Cell Electronics</td>
<td>26</td>
</tr>
<tr>
<td>13</td>
<td>Custom Interface Electronics</td>
<td>28</td>
</tr>
<tr>
<td>14</td>
<td>Three-Inch Sample Measurements</td>
<td>31</td>
</tr>
<tr>
<td>15</td>
<td>Two-Inch Sample Measurement</td>
<td>31</td>
</tr>
<tr>
<td>16</td>
<td>Load Cells and Detectors Schematic</td>
<td>33</td>
</tr>
<tr>
<td>17</td>
<td>Load Cells/Detectors Printed Circuit Board</td>
<td>34</td>
</tr>
<tr>
<td>18</td>
<td>Measurement Arm Stepper Motor Interface</td>
<td>35</td>
</tr>
<tr>
<td>19</td>
<td>Meas. Arm Stepper Motor Printed Circuit Board</td>
<td>36</td>
</tr>
<tr>
<td>20</td>
<td>Base Plate Top View</td>
<td>37</td>
</tr>
<tr>
<td>21</td>
<td>Base Plate Side View</td>
<td>38</td>
</tr>
<tr>
<td>22</td>
<td>Base Plate End View</td>
<td>39</td>
</tr>
<tr>
<td>23</td>
<td>Base Plate Isometric View</td>
<td>40</td>
</tr>
<tr>
<td>24</td>
<td>Slide Bracket B I Top View</td>
<td>41</td>
</tr>
<tr>
<td>Page</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Slide Bracket B I Side View</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Slide Bracket B I Front View</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Slide Bracket B I Isometric View</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Slide Bracket B II Top View</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Slide Bracket B II Front View</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Slide Bracket B II End View</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>Slide Bracket B II Isometric View</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>Slide Bracket A Top View</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>Slide Bracket A Side View</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>Slide Bracket A Front View</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>Slide Bracket A Isometric View</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>Upper Guide Bracket Top View</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>Upper Guide Bracket Front View</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>Upper Guide Bracket Isometric View</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>Lower Guide Bracket Top View</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>Lower Guide Bracket Front View</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>Lower Guide Bracket Isometric View</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>Prism Holder A Top View</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>Prism Holder A End View</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>Prism Holder A Front View</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Prism Holder A Isometric View</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>Prism Holder B Top View</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>Prism Holder B End View</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>Prism Holder B Front View</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>Prism Holder B Isometric View</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>Detector Holder A Top View</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>Detector Holder A Side View</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>62</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>65</td>
<td></td>
<td></td>
</tr>
<tr>
<td>66</td>
<td></td>
<td></td>
</tr>
<tr>
<td>67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>68</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

I would like to thank Angela McPherson for allowing the use of data taken for previous work for verification of the operation of the waveguide stage. I would also like to thank David Zelmon and Ken Hopkins for their many comments, suggestions and reviews pertaining to the initial drafts of this technical report.
Nonlinear optical (NLO) materials are becoming increasingly important to the Air Force, and their expanding use in future systems is inevitable. Applications ranging from communications and optical computing to lasers and laser hardening will all be affected by the development of these materials. Because of this need, basic research and development of these materials are extremely important. Within the Materials Directorate of Wright Laboratory, the Electronic and Optical Materials Branch (WL/MLPO) is performing basic R&D in various areas of nonlinear optical materials. Central to this research is the measurement of the optical properties of these materials.

In order to apply these materials to device and systems, it is important to understand how they behave under the influence of various forces and conditions. These include applied voltages and currents, incident electromagnetic radiation and temperature changes. How they behave under these various conditions can be either useful or detrimental depending on the application. In the case of optical transmission, changes in material properties due to applied voltage are useful for applications where modulation is important. On the other hand, changes in characteristics due to temperature fluctuations can be detrimental when operating in harsh environments. Various methods and techniques have been developed to characterize these materials and predict their behavior in a wide variety of applications.

One important parameter of NLO materials is the attenuation of light. High loss materials can be useful for filtering applications especially when the loss is wavelength dependent. Low loss materials are very useful for transparent window applications and waveguiding. In addition, knowing the loss of a material can help determine the nature of its structure.
The losses in NLO materials can occur from many sources. For guided light structures, these include scattering due to bulk and surface defects and absorption due to resonance, losses due to substrate coupling and losses from irregularities in the interface between the guiding layer and the substrate. There are various methods that can be used to measure these losses and two of the primary methods used in WL/MLPO involve measuring the propagation loss of planar waveguides fabricated from the material being tested. This report describes the design and implementation of a fully automated waveguide stage that allows the measurement of these propagation losses.
2.0 EXPERIMENTAL BACKGROUND

Two techniques used in WL/MLPO for measuring the propagation loss in planar waveguides are the out-of-plane scattering technique\(^1\) and the three-prism measurement technique\(^2\). These two techniques are similar in their measurement method and their geometrical requirements. Both employ channel or planar waveguides, and in both, the relative intensity of light is measured at points along the propagation path. In the out-of-plane technique, the relative intensity of the light scattered perpendicular to the plane of propagation is measured at many points along the guide, while in the three-prism method, guided light is coupled out from the waveguide at various points and its intensity is measured. In order to understand the design of the waveguide stage, it is useful to explore the experimental basis of these two techniques.

2.1 THE OUT-OF-PLANE TECHNIQUE

The out-of-plane scattering technique (graphically shown in Figure 1) is primarily used to measure waveguides with propagation losses higher than about 1 dB/cm. When using the visible wavelengths, these waveguides usually show a characteristic bright streak along the guide path. This is indicative of light being scattered from confined waveguide modes from scattering centers within the waveguide layer and at the interfaces. The intensity of the scattered light can be measured using a detector which is suspended above the streak and is moved parallel to the streak.
The scattering centers are assumed to be homogeneous and evenly distributed over the length of the waveguide. This is required if a reasonable comparison is to be made of the scattered light intensity at various points along the propagation path. Inhomogeneities in the distribution of scattering centers cause substantial nonuniformities in the scattered light intensity along the direction of propagation. The result is an unreliable figure for propagation loss for the waveguide.

The actual propagation loss for this technique is calculated using Beer's Law. This is given in equation 1. The loss value is obtained by taking the slope of the log of the intensities (this is $\alpha$ in equation 1) between the initial intensity reading and each subsequent reading of the scattered output intensities at points along the guide path.
\[ l(z) = l_0e^{-\alpha z} \]  

Propagation Loss in dB/cm = 10 log \( \frac{l(z)}{l_0} \)

\[
= 10 \log l_0e^{-\alpha z} = 10 \log(e^{-\alpha z}) = -10\alpha z \log(e) = -10\alpha z \frac{\ln(e)}{\ln(10)}
\]

\[ = -4.343\alpha z \]

Where

- \( l_0 \) = relative out-scattered intensity at the first measured point
- \( l(z) \) = relative out-scattered intensity at point \( z \) along the path
- \( \alpha \) is the attenuation factor
- \( z \) is the distance in cm (set to 1 cm)

\( \alpha \) is determined by a least-squares curve fit or by plotting the data (i.e. \( l(z) \)) on a log scale and taking the slope.

A distinct advantage of this technique is the ability to measure losses in "softer" waveguide materials such as polymers and organics. Since the detector never comes in contact with the waveguide the surface is never disturbed. This is not the case with the three-prism technique.

### 2.2 THE THREE-PRISM TECHNIQUE

The three-prism technique, as the name implies, employs three prisms to couple light into and out of a waveguide. This is shown in Figure 2. The first prism, called the input prism, couples light energy into the waveguide. The other two prisms, called the measurement and exit prisms, respectively, are used in conjunction with the appropriate detectors to measure the intensity of the out-coupled light along the propagation path of the waveguide. This technique works well when the waveguides are sufficiently long (1-2 inches) and it is uniquely suited for measurements where the losses for the materials are relatively low (<1 dB/cm).
The three-prism measurement is performed by using the input prism to couple light into the waveguide. The relative intensity of the light inside the guided path is then measured by using the measurement and exit prisms. First, a quiescent light intensity is measured using the exit prism with the measurement prism off of the waveguide. This value, $P_{eo}$, is used as the reference for all measurements taken along the points of the guide path. To make the loss measurements, the measurement prism is pressed onto the waveguide at equally spaced points along the guide path and intensity readings from both the measurement ($P_m$) and exit prisms ($P_e$) are taken. These values and $P_{eo}$ are used to calculate the relative decrease in intensity at each point along the waveguide. The loss in dB/cm given in equation 2, is then calculated by taking the slope of these calculated values when plotted on a log scale. The advantage of using the exit prism is that it eliminates the need to know the coupling coefficient of the measurement prism.
3.0 SYSTEM REQUIREMENTS

An automated system that can be used to make out-of-plane and three-prism loss measurements for optical waveguides must meet certain requirements. For example, there are certain geometric requirements that affect the accuracy of the measurements. One of these is the need for a known reference for distance measurements. In addition, accurate light intensity measurements have certain conditions that must be met. In the case of the out-of-plane measurement, only a narrow portion of the guiding streak can be measured at each point if an accurate loss measurement is to be made. These requirements necessarily affect the design of the measurement system, and they will be explored more fully in the following sections.

3.1 SPECIFIC REQUIREMENTS

There are many geometrical, electrical, and optical requirements for a successful waveguide stage system. First, a planar surface is required to support waveguide substrates. This is needed to prevent damage to the wafers when pressure is applied from the coupling prisms. Secondly, motion along the waveguide length must be strictly controlled. The travel path must be parallel to the direction of the propagation path. The positions along the path where measurements are to be taken must be accurately known, and they must be repeatable. This is necessary if multiple measurements of the same sample are to be taken to verify operation. In addition, smooth, equal pressure must be applied to the prisms when coupling to the waveguide. This is necessary to ensure proper coupling and to prevent damage to the wafer.
Propagation Loss in dB/cm

\[ = -4.343\alpha z \]  

Where

\[ I(z) = \frac{P_m P_{eo}}{P_e} = e^{-\alpha z} \]

\( \alpha \) is the attenuation factor
\( z \) is the distance in cm (set to 1 cm)
\( P_m \) is the measurement prism intensity
\( P_e \) is the exit prism intensity
\( P_{eo} \) is the quiescent exit prism intensity (measurement prism off the waveguide)

\( \alpha \) is determined by a least-squares curve fit of \( I(z) \) or by plotting the data (i.e. \( I(z) \)) on a log scale and taking the slope.

The three-prism method, as in the case of the out-of-plane method, also relies on a certain level of uniformity in the waveguide. Accurate loss measurements over a distance still require a uniform distribution of loss centers. However, nonuniformities in the waveguides such as bright spots are easily detectable with this technique, and the information obtained can be useful in determining the overall physical quality of the waveguide. The main disadvantage of this technique is that it requires "harder" waveguide materials. This is due to the fact that the prisms must be pressed onto the surface of the guide in order to obtain sufficient contact to allow coupling. Softer materials may be damaged by this technique, and any data taken on a damaged surface would be useless.

Both of these techniques employ planar waveguides and make intensity measurements at points along the guide path. These commonalities lend themselves to a single test setup that could easily be changed to accommodate either method.
Interchangeability is also an issue. The system must be able to be converted from the three-prism technique to the out-of-plane scattering technique with a minimum of effort. To meet this requirement, parts of the system must be designed to allow easy disassembly and reassembly without affecting the ultimate accuracy and repeatability of the system. In addition, all parts that are not removable must be designed to allow various attachments to be added with ease. This accommodates such things as various size prisms and detectors.

The electrical and optical requirements are also numerous. For example, coupling prisms must have as much available space as possible for launching light into and out of the waveguide. This is required in order to facilitate as many coupling angles as possible for a wide variety of guiding modes. Additionally, the detectors used for measuring the out-coupled or out-scattered light must be able to capture light at any of the out-coupling angles or at any position along the waveguide. Another requirement is the need to minimize backscattered light detection. The detectors must be designed to eliminate unwanted light from input coupling and from the surroundings. This light adversely affects the measurements.

Although not necessary for the three-prism technique, it is advantageous to know the force which is being applied to the coupling prisms. This allows repeatability in coupling pressure and can also be used to prevent excessive force from being applied to sensitive waveguides. This is best accomplished through the use of force detectors such as load cells.

A final requirement for a waveguide stage system is computer control for movement and data acquisition. In the modern laboratory setting, this is essential. These measurements are tedious and time consuming. Doing these types of measurements "by hand" also greatly increases the chance for errors. Computer control of all motion and data acquisition yields vastly improved accuracy and
repeatability while greatly reducing the chance for error. In addition, allowing a computer to take control frees the researcher for other more important tasks.

The waveguide stage measurement system has been designed to meet these requirements by combining commercially available equipment and custom designed parts.
4.0 EXPERIMENTAL APPARATUS

An overall schematic of the waveguide measurement system is shown in Figure 3. The heart of the system is the waveguide stage and its associated electronic and mechanical components. These are explained fully in Section 5. The remaining supporting equipment consists of the computer and electronic equipment necessary to take data from the waveguide stage.

The data acquisition portion of the waveguide measurement system is designed around a Hewlett Packard 9000 330 MMA computer using the HP BASIC 5.0 programming language. This allows easy programming for automated data acquisition. In addition, this computer is connected to a local area network which allows easy transfer of data to other systems more suitable for analysis and presentation.

The HP computer controls various external equipment necessary for supplying stimulus and acquiring response to and from the waveguide stage. In order to measure optical response from detectors on the waveguide stage as well as pressure readings from load cells, a Hewlett Packard HP3457A Digital Multimeter (DMM) is used. This DMM has an HP44492 10-channel Relay Input Card option that allows the DMM to be switched between 1-of-10 input channels. This allows a single DMM to be used to measure all signals from the waveguide stage.

In order to drive the rail table portion of the waveguide stage (rail table explained in Section 5), a Daedal MC 2000/MD 23 Linear Motion Controller/NEMA Motor Driver is employed. This combination allows accurate computer control and measurement of the out-coupling position of the measurement prism of the waveguide stage.
Also employed in the waveguide measurement system is an HP6942A Multiprogrammer. This device is a multipurpose card cage that allows various special purpose cards such as relay cards or D/A converter cards to be accessed by a computer. As used in this system, an HP69735 Pulse Train/Stepper Motor
card is used in the multiprogrammer to supply pulses to a Hurst stepper motor to control the measurement arm of the waveguide stage (see Section 5).

The rest of the system is comprised of the custom mechanics and electronics required to interface between the DMM, the Multiprogrammer and the waveguide stage. These custom components provide measurement arm drive and I/O signal condition necessary for data acquisition. Their purpose and function is explained fully in Section 5.
5.0 SYSTEM DESIGN

The system surrounding the waveguide stage (Figure 4) contains mechanical hardware, electronics and optics, and computer software. This is a mixture of commercial and custom components, and each of these will be discussed in the context of full system operation.

5.1 MECHANICAL HARDWARE

The mechanical hardware of the waveguide is comprised of a commercially available rail table to which custom-designed components have been added. The custom components are designed for specific application to three-prism or out-of-plane scattering loss measurements. The main components are the stage base plate, the input and output arms and the measurement arm. All parts are machined from aluminum, and they are black-anodized for use in laser environments.

5.1.1 THE RAIL TABLE

The rail table (Daedal Model 506041S-LH) has 4 inches of linear travel and is equipped with a five-pitch double nut ball screw mechanism to minimize backlash. The rail table is driven by a standard NEMA 23 frame stepper motor having 10,000 steps per revolution of rotational resolution. This allows the rail table to have ±0.0002-inch position repeatability and ±0.00025 in/in position accuracy. The NEMA 23 motor is driven by a Daedal single axis motion controller complete with IEEE-488 interface (Daedal Model MC 2000). This provides complete computer control of the motion of the rail table including the speed,
acceleration/deceleration rate and the step resolution between measurements positions.

5.1.2 THE BASE PLATE

The base plate for the waveguide stage is shown in Figure 5. The base plate serves as the mounting seat for the waveguide substrates and as a mount for the input and output arms. The base plate is mounted to the rail table using $\frac{1}{4}$ - 20 NC screws. The plate has been precision machined for flatness and it is aligned to the rail table using set pins and offset mounting holes for the $\frac{1}{4}$ - 20 screws. This ensures that the base plate stays aligned parallel to the rail table and that it can only be mounted with one unique orientation in the event disassembly is required.
In order to ensure maximum flatness, the rail table was machined "true" at the time of manufacture of the base plate.

![Figure 5 - The Waveguide Stage Base Plate](image)

FIGURE 5 - The Waveguide Stage Base Plate

The base plate has a set of 8-32 NC mounting holes that are equally spaced along two \( \frac{1}{4} \) inch troughs that run along each side of the base plate. These troughs hold the input and output arm mechanisms and allow the arms to be positioned and secured anywhere along the length of the base plate. This allows maximum flexibility while maintaining parallel positioning with respect to the sides of the base plate.

The base plate is raised up from the surface of the rail table \( \frac{3}{4} \) inch to allow the measurement arm to be mounted to the rail table. This in turn allows the measurement arms to move along the length of the base plate with the accuracy of the rail table.
5.1.3 THE INPUT, OUTPUT, AND MEASUREMENT ARMS

The input and output arms, shown in Figure 6, are identical in construction. Each arm is positioned over the surface of the waveguiding substrate. The input arm is placed at the front of the waveguide, and it is used to launch light into the guide. The output arm is placed at the end of the guide, and it is used to measure the exit intensity of the light exiting from the waveguide during three-prism measurements. It is not used for out-of-plane measurements and can be removed.

![The Input/Output Arm](image)

FIGURE 6 - The Input/Output Arm

The input/output arms are comprised of various moving parts that facilitate proper placement of the prisms onto the waveguide substrate. This is shown in Figure 7. Each arm is identical with the exception of a removable detector attachment for the output arm.

The operation of the input/output arm is straightforward. A set of steel rods is used to control the motion of the various moving parts of the arm. Referring to
Figure 7, the two outer steel rods are firmly press-fit into the slide bracket. This slide bracket slip-fits into the base plate and is screwed down into position over the waveguide substrate. The two inner steel rods are press-fit into the upper guide bracket and are allowed to slip freely through the slide bracket. The lower guide bracket has a prism attachment or and/or a detector attachment (depending on which measurement technique is being used), and it is also allowed to slip freely over the two inner steel rods. The whole mechanism is raised and lowered onto the waveguide substrate by a micrometer screw assembly and two springs located on the outer steel rods.

A unique feature of the arms is the use of compression load cells in the axis of applied force of the prisms. These load cells (Sensotec Model 13, AL322BN) are placed in all three arms of the waveguide stage, and they are located in the cup of the lower guide bracket. Due to the design of the arms, the load cell will only
measure the force being applied to the prisms. This is extremely useful when the stage is being used to measure polymer and other soft material waveguides. Excessive pressure can damage these types of guides making the measured data useless. In addition, these load cells allow the coupling pressure to be maintained and duplicated between measurements on the same sample and from sample to sample.

The measurement arm is identical to the input/output arms in construction and operation with two exceptions. First, the measurement arm does not have a slide bracket but rather has a two-piece bracket that attaches to the rail table instead of the base plate. This allows the measurement arm to be moved along the waveguide path to make measurements with the positioning accuracy of the rail table. Second, the micrometer screw assembly of the measurement arm is driven by a computer-controlled stepper motor. This allows complete control of the data acquisition by software programming.

Complete drawings for the input/output and measurement arms, as well as all other custom-designed mechanical parts, can be found in the Appendices.

5.2 ELECTRONICS AND OPTICS

The electronics and optics for the waveguide stage consist of the voltage references, amplifiers, prisms, etc. that allow the actual acquisition of the data pertaining to the waveguide under test. This can be classified into three groups: (1) the prisms and detectors and their associated electronics, (2) the load cells and their electronics, and (3) the measurement stepper motor. The first two are custom implementations, and the last is a commercially available unit.
5.2.1 PRISMS AND DETECTORS

The prisms are the only strictly optical component for the stage and their operational requirements are mechanical in nature, however, their discussion with the detectors is warranted. As mentioned previously, the prisms must be applied to the waveguide with a steady and even pressure. To accomplish this, a special prism holder was developed. A representative holder is shown in Figure 8. These holders come in various sizes to accommodate various size prisms.

The prism holders are interchangeable and attach to the lower guide brackets using 2-56 NC screws. This allows all holders to fit into the lower guide bracket making it very convenient to change from one type of prism to the other.

The holders were originally lined entirely on the inside with felt. This provided both a friction grip as well as protection from cracking for the prisms. However, the felt placed on the inside top produced excessive cushioning to the
prism when pressure was applied. This caused a classic exponential relaxation in the pressure applied to the prism as the felt continued to compress. This made accurate pressure measurements with the load cells very difficult. To remedy this problem, the felt on the inside top was replaced with thin teflon sheeting. This allowed protection of the prism top surface while greatly reducing the compression relaxation.

The detectors associated with the three-prism and the out-of-plane techniques have both mechanical and electronic fixturing. The physical setup for the detectors is different for each technique. For the out-of-plane technique there is only one detector and it must be suspended over the waveguide on the measurement arm. For the three-prism technique, two detectors are required, one for the measurement arm and one for the output arm. These detectors must be placed as close as possible to the prisms on each arm in order to capture the exiting light regardless of the out-coupling angle. In addition, the fixturing for mounting the detectors for both techniques must allow easy interchangeability.

The configurations for detector mounting for both techniques are shown in Figures 9 and 10. For the three-prism technique, the detectors are fixed to an angled mounting bracket using silicone adhesive. The bracket is then attached to the lower guide bracket using the same 2-56 NC screws that attach the prism holder. For the out-of-plane technique, a special detector mount was fabricated that attaches to the lower guide bracket in place of the prism holder on the measurement arm. This holder is a two-piece unit and the detector fits in between the pieces. A small slit in the bottom piece that is positioned over the detector is the input for the scattered light from the waveguide.
The detectors used for both techniques are Hamamatsu model S1227-66BQ silicon photodiodes. These detectors have a thin, rectangular shape that is ideally suited for this application. The overall size of these detectors is 10 mm x 9 mm x 2
mm with an active detector area of 33 mm². The S1227-66BQ has a response from 190 nm to 1000 nm with a peak at 720 nm and a typical radiant sensitivity of 0.35 A/W at the peak wavelength³.

The electronic circuitry for the detectors is shown in Figure 11. The detectors form part of a variable gain current-to-voltage converter. The gain for each amplifier is controlled by a series of resistors connected to a rotary switch. This allows gain adjustment for different light intensity levels. The current generated in the detector due to incident light is converted to a voltage at the output of the operational amplifier (op-amp) according to equation 3.

\[ V_o = -I_r R_f \text{ volts} \]  
where  
\( I_r = \) total photodiode current in amps  
\( R_f = \) feedback resistor in ohms

The photocurrent, \( I_r \), for the S1227-66BQ is given in equation 4⁴. The second and third terms of this equation can be ignored because the series resistance of the photodiode is small (\( R_s < 10 \Omega \)) and because the shunt resistance is large (\( R_{sh} > 10^7 \Omega \))⁵. This makes the current and, thus, the output voltage from the op-amp, very nearly directly proportional to the light intensity striking the detector.

\[ I_r = I_L - I_s \left( e^{\frac{qR_s}{kT}} - 1 \right) - \frac{I_s R_s}{R_{sh}} \]  
where  
\( I_r = \) total current pulled through the photodiode  
\( I_L = \) Current due to to the incident light on the detector  
\( I_s = \) photodiode reverse saturation current  
\( R_s = \) photodiode series resistance  
\( R_{sh} = \) photodiode shunt resistance  
\( q = \) electronic charge  
\( k = \) Boltzmann's constant  
\( T = \) Absolute temperature of the photodiode
FIGURE 11 - Detector Electronics
The 74121, the relays and the transistor in the circuit of Figure 11 have been added to prevent latching of the op-amps when the power is first applied to the stage electronics. It has been experimentally determined that the high input impedance of the LF351N, coupled with the low reverse bias current of the photodiodes and the stray capacitance of the cabling connecting the photodiodes to the op-amp, causes a latched and/or oscillating condition. The 74121 is a timed one-shot that shorts the inputs of the op-amps until the power supplies have settled. This allows the op-amps to have a stable reference when the inputs are released which eliminates the latching or oscillating.

5.2.2 THE LOAD CELLS

The load cell signal-conditioning electronics employed in the waveguide stage are shown in Figure 12. Each arm of the waveguide stage has a load cell that measures the pressure being applied to the prisms. These load cells operate in a standard compensated Wheatstone bridge configuration. The bridge is driven by a voltage reference, and the resistance in one of the arms changes as force is applied. This change causes a voltage differential to appear across the bridge that is related to the force that is being applied. This voltage is amplified using a differential amplifier, and the output is available as a measure of the force being applied to the prisms.
In the circuit of Figure 12, the reference voltage to the load cells is produced by a temperature-compensated zener diode (LM336Z-5.0) buffered by an
operational amplifier (LM351N). The zener voltage is adjusted using the potentiometer connected to pin 3. The differential voltage produced by the load cell is amplified by a Burr-Brown INA101CM Instrumentation Amplifier. The gain of this amplifier is set by resistors R1, R2 and R3. This allows precision adjustment of the full scale output of the load cell.

5.2.3 THE MEASUREMENT STEPPER MOTOR

The stepper motor assembly employed to drive the measurement arm is comprised of a commercially available stepper motor mounted on a custom-designed bracket. The motor is a Hurst SAS Series geared stepper motor with a rotation speed of 0.42 rpm providing 600:1 reduction with a typical torque of 200 oz-in. The motor is driven by a Hurst EPC-013 Stepper Motor Controller board. This board provides the drive current for the motor as well as the input/output to allow manual and digital control for the motor. The EPC-013 is interfaced to custom-designed electronics that allows switching between computer control and manual control. Ultimately, the computer control is accomplished through a Hewlett Packard HP6942A Multiprogrammer Interface Pulse Train Stepper Motor Controller card.

The EPC-013 provides inputs that allow control of various aspects of the stepper motor. This card controls the acceleration/deceleration of the motor, the rotation direction, the phase setting (3 or 4) and the run/hold status. In addition, this card has a free running pulse train to allow manual control of the motor and an input for external pulse trains to allow other circuitry to control the motor.

The custom electronics that serves as an interface between the Hewlett Packard multiprogrammer and the EPC-013 is shown in Figure 13. This circuit functions as a data router that allows the user to either control the motor manually
or by the computer. The circuit has two digital data selectors (74157) that allow either pulses from the computer or a pulse train from the EPC-013 to control the motion of the stepper motor. Manual/Computer control is determined by the setting of a DPDT switch connected to the select line of one of the 74157s. Another data selector chooses from either up or down pulses being supplied from the computer. The multiprogrammer pulse train card sends pulses out on one of two lines depending on which rotation direction is desired. These pulses are distinguished by which line is active. A 74123 One Shot is triggered by the pulses and acts as a timer for the select lines to the data selector. This ensures proper pulse routing and duration to the motor driver card.

FIGURE 13 - Custom Interface Electronics
This custom interface has two special features. First, manual control automatically overrides computer control. If anything goes wrong, the motor can be controlled or shutdown manually using one of two switches. In addition, a special comparator circuit provides auto-shutdown if the load cell pressure exceeds 40 lbs of force.
6.0 EXPERIMENTAL RESULTS

The operation of the waveguide stage has been verified by the three-prism, the two-prism, and the out-of-plane scattering techniques. Loss measurements using each of the techniques were taken on planar waveguides made from hexafluoro isopropylidene-polybenzoxazole (6F-PBO-1E) on oxidized Si wafers. The waveguides were fabricated by spin coating 1.0-μ-thick layers of 6F-PBO-1E on 1.4-μ-SiO$_2$ buffer layers on 2- and 3-inch silicon substrates. Loss measurements were taken on a 3-inch sample at a wavelength of $\lambda = 900$ nm using the out-of-plane and the three-prism and techniques for comparison. A 2-inch sample was measured using the two-prism method at a wavelength of $\lambda = 900$ nm to compare losses between samples. The resulting data for all samples were fitted to the Beer's law relationship given in equations 1 and 2. The plots of the data are shown in Figures 14 and 15. The loss for the 3-inch waveguide was found by averaging the loss for the three-prism and the out-of-plane techniques. The loss for the two-inch sample was taken from the data provided by the out-of-plane scattering technique. The average loss for the 3-inch sample was 3.93±0.565 dB/cm and the measured loss for the 2-inch sample was 3.523 dB/cm.
Comparison of Loss Measurements
6F-PBO-1E on 3" Silicon Wafer

\[ y = 0.92034 e^{-0.099678x} R = 0.99013 \]
\[ y = 1.7212 e^{-0.08125x} R = 0.99202 \]

Wavelength = 900 nm
Loss in dB/cm = 3.93±.565

FIGURE 14 - Three-Inch Sample Measurements

Two Prism Method Loss Measurement
6F-PBO-1E on 2" Silicon Wafer

\[ y = 1.0091 e^{-0.081172x} R = 0.99887 \]

Wavelength = 900 nm
dB/cm = 3.523

FIGURE 15 - Two-Inch Sample Measurement
REFERENCES


7. Angela L. McPherson, and Jeffery W. Baur, "Guided Wave Loss Measurements for End-capped 6F-Polybenzoxazole," unpublished work
APPENDIX A - ELECTRONIC SCHEMATICS

FIGURE 16 - Load Cells and Detectors Schematic

33
FIGURE 18 - Measurement Arm Stepper Motor Interface
FIGURE 19 - Meas. Arm Stepper Motor Printed Circuit Board
FIGURE 20 - Base Plate Top View
FIGURE 21 - Base Plate Side View
Some hidden lines and surfaces are not shown for clarity.
FIGURE 23 - Base Plate Isometric View
FIGURE 24 - Slide Bracket B I Top View

3/8" Diameter

Drill to Allow Passage of 1/4 - 20 Screw

Wave Guide Stage Assembly
Slide Bracket B Part I
Top View
Tom Kensky
WL/MLPO
Not to Scale
18 Jul 92
FIGURE 25 - Slide Bracket B I Side View
FIGURE 26 - Slide Bracket B I Front View
FIGURE 27 - Slide Bracket B I Isometric View
Drill to allow press fit of a 5/32" steel rod

Drill to allow slip fit of a 5/32" steel rod

Drill to allow press fit of a 3/8" dia. micrometer screw assembly

Drill to allow slip fit of a 5/32" steel rod

Drill to allow press fit of a 5/32" steel rod

FIGURE 28 - Slide Bracket B II Top View
Split pin to hold micrometer screw

0.125° D semi-circle
(for guide pins)

0.1° D

0.175°

0.01" E

0.01" O

0.01" U

0.01" LL

0.01" Cu

0.01" Cy

0.01" *6

0.01" Fb

0.01" E4

0.01" rce 1FotVe

0.01" -6
FIGURE 30 - Slide Bracket B II End View

1/4" 1/4"
3/16"
1/8"
3/16"

(Some hidden surfaces are not shown for clarity)
FIGURE 31 - Slide Bracket B II Isometric View
FIGURE 32 - Slide Bracket A Top View
FIGURE 33 - Slide Bracket A Side View

1/2"

1.318"

1/4"

3 1/2"

Wave Guide Stage Assembly
Slide Bracket A
Side View
Tom Kensky
WL/MLPO
Not to Scale
21 Jul 92
FIGURE 34 - Slide Bracket A Front View

Adjust for slip fit as per top view drawing

<table>
<thead>
<tr>
<th>Wave Guide Stage Assembly</th>
<th>3 of 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slide Bracket A</td>
<td></td>
</tr>
<tr>
<td>Front View</td>
<td></td>
</tr>
<tr>
<td>Tom Kensky</td>
<td></td>
</tr>
<tr>
<td>WL/MLPO</td>
<td></td>
</tr>
<tr>
<td>Not to Scale</td>
<td></td>
</tr>
<tr>
<td>21 Jul 92</td>
<td></td>
</tr>
</tbody>
</table>
FIGURE 35 - Slide Bracket A Isometric View
Adjust dimension to allow slip fit into the cup of the Lower Guide Bracket.

A. Drill to allow sliding passage (slip fit) of 5/32" diameter steel rod.

B. Drill to allow press fit of 5/32" diameter steel rod.
FIGURE 37 - Upper Bracket Front View
FIGURE 38 - Upper Guide Bracket Isometric View
A  Drill to allow slip fit of a 5/32 diameter steel rod

B  Drill and tap for 2-56 NC screw.

4 decimal place tolerances are for consistency in dimensioning only. Normal machining tolerances are allowable for actual fabrication.
4 decimal place tolerances are for consistency in dimensioning only. Normal machining tolerances are allowable for actual fabrication.
FIGURE 41 - Lower Guide Bracket Isometric View
Drill holes to allow passage of 2-56 screws.

* Adjust these dimensions for slip fit into Lower Guide Bracket
FIGURE 43 - Prism Holder A End View
FIGURE 45 - Prism Holder A Isometric View
Drill holes to allow passage of 2-56 screws.

*2.7 cm
(1.063")
+.000" 
-.003"

These lengths should be adjusted to allow slip fit into the lower guide bracket.
FIGURE 47 - Prism Holder B End View
FIGURE 48 - Prism Holder B Front View
FIGURE 49 - Prism Holder B Isometric View
FIGURE 50 - Detector Holder A Top View

Drill to allow passage of 2-56 NC screws

Aluminum thickness is 0.0365 inch or nearest
FIGURE 51 - Detector Holder A Side View
FIGURE 52 - Detector Holder A Isometric View
FIGURE 53 - Detector Holder B I Top View
FIGURE 54 - Detector Holder B I End View
FIGURE 55 - Detector Holder B I Front View
FIGURE 56 - Detector Holder B I Isometric View
FIGURE 57 - Detector Holder B II Top View
FIGURE 58 - Detector Holder B II End View
FIGURE 59 - Detector Holder B II Front View
FIGURE 60 - Detector Holder B II Isometric View
FIGURE 61 - Rods and Caps
FIGURE 62 - Upper Guide Bracket Assembly
Steel rod should be mounted firmly in Slide Bracket A. End of bar should be flush with top surface of Bracket.
FIGURE 64 - Slide Bracket B Assembly

Steel rods should be mounted firmly in Slide Bracket B. End of bar should be flush with top surface of Bracket.

ROD - B

1/2" 7/8"

ROD - B
FIGURE 65 - Stage Assembly Exploded View
FIGURE 66 - Bracket Assembly

- Micrometer Screw
- Steel Rods
- Slide Bracket
- Upper Guide Bracket
- Prism Holder
- Lower Guide Bracket
- Load Cell goes into this chamber
APPENDIX C - SOFTWARE LISTING

10 !*****************************************************************************
20 ** OPTICAL WAVEGUIDE LOSS MEASUREMENT PROGRAM **
30 ** TOM KENSKY **
40 ** MATERIALS DIRECTORATE WL/MLPO WPAFB, OHIO **
50 ** VERSION 6 2 SEP 93 **
60 !*****************************************************************************

70 !
80 OPTION BASE 0
90 GINIT
100 GCLEAR
110 !
120 !*****************************************************************************
130 ** DIM ARRAYS FOR USE BY SUBROUTINES **
140 !*****************************************************************************
150 !
160 DIM Value$(10)(80)
170 DIM Choices$(10)(80)
180 DIM Distance$(10)(80)
190 DIM Constraints$(10)(80)
200 DIM Stage_pos$(30)
210 DIM Filename$(80)
220 DIM Steps$(3)(10)
230 !
240 !*****************************************************************************
250 ** SET THE ADDRESS OF THE EQUIPMENT **
260 ** MOTION CONTROLLER ADDRESS IS 16 **
270 ** HP 3457A VOLTMETER ADDRESS IS 22 **
280 ** HP 6942A MULTIPROGRAMMER ADDRESS IS 12 **
290 !*****************************************************************************
300 !
310 ON ERROR GOTO 330
320 CSIZE 5
330 CLEAR SCREEN
340 MOVE 15,80
350 LABEL "********************************************************************************
360 LABEL ** SELECT THE BASE ADDRESS FOR **
370 LABEL ** THE EQUIPMENT (i.e. which **
380 LABEL ** optical table are you on?) **
390 LABEL "********************************************************************************
400 INPUT "ENTER THE BASE ADDRESS: (7 OR 8)*",Address$
410 IF VAL(Address$)=7 THEN
420 ASSIGN @Daedal TO 716
430 ASSIGN @Voltmeter TO 722
440 ASSIGN @Multi TO 712
450 GOTO 540
460 END IF
470 IF VAL(Address$)=8 THEN
480 ASSIGN @Daedal TO 816
490 ASSIGN @Voltmeter TO 822
500 ASSIGN @Multi TO 812
510 GOTO 540
520 END IF
530 GOTO 330
540 OFF ERROR
550 !
560 甲醛 ****************************
570 ** PROMPT USER FOR CHOICE OF **
580 ** 3-PRISM OR OUT-OF-PLANE **
590 ** TECHNIQUES **
600 ****************************
610  
620 ON ERROR GOTO 630
630 CLEAR SCREEN
640 MOVE 15,80
650 LABEL ****************************
660 LABEL ** ARE YOU DOING OUT OF PLANE **
670 LABEL ** MEASUREMENTS? **
680 LABEL ****************************
690 INPUT "ENTER 'Y' OR 'N',Out_of_plane$"
700 IF Out_of_plane$="Y" OR Out_of_plane$="y" THEN
710   Out_of_plane=1
720 ELSE
730   Out_of_plane=0
740 END IF
750 CLEAR SCREEN
760 OFF ERROR
770  
780 ****************************
790 ** CAN'T USE THE STEPPER MOTOR FOR **
800 ** THE MEASUREMENT ARM WHEN DOING **
810 ** OUT-OF-PLANE MEASUREMENTS **
820 ****************************
830  
840 IF Out_of_plane=1 THEN
850   Use_motor=0
860   GOTO 1090
870 END IF
880  
890 ****************************
900 ** PROMPT USER FOR CHOICE OF USING **
910 ** THE STEPPER MOTOR FOR DRIVING **
920 ** THE MEASUREMENT ARM **
930 ****************************
940  
950 ON ERROR GOTO 960
960 CLEAR SCREEN
970 MOVE 15,80
980 LABEL ****************************
990 LABEL ** ARE YOU USING THE STEPPER **
1000 LABEL ** MOTOR FOR THE MEASUREMENT **
1010 LABEL ** ARM? **
1020 LABEL ****************************
1030 INPUT "ENTER 'Y' OR 'N',Use_motor$
1040 IF Use_motor$="Y" OR Use_motor$="y" THEN
1050   Use_motor=1
1060 ELSE
1070   Use_motor=0
1080 END IF
1090 CLEAR SCREEN
1100 OFF ERROR
1110  
1120 ****************************
1130 ** INITIALIZE DAEDAL MOTION CONTROLLER:**

85
1140 ** ENABLE IEEE-488, NORMAL MODE, SET DISPLAY TO ZERO **
1150 ** ACCELERATION=5 REV/SEC/SEC, VELOCITY=1 REV/SEC **
1160 ** COUNTER ZERO RESET **
1170 Outputs @Daedal USING "$K$" E MN DA A5.0 V1 X0 *, END
1180 Outputs @Voltmeter; "$DCV AUTO,.01"
1190 Outputs @Voltmeter; "$TRIG AUTO"
1200 Outputs @Voltmeter; "$TERM REAR"
1210 Outputs @Voltmeter; "$FIXEDZ ON"
1220 ** INITIALIZE VALUES FOR SCROLL MENU **
1230 ** MANUAL ADJUSTMENT OF STAGE POSITION **
1240 Label ***********
1250 Choices$(0)="10,000 STEP MOVEMENT"
1260 Choices$(1)="1,000 STEP MOVEMENT"
1270 Choices$(2)="100 STEP MOVEMENT"
1280 Choices$(3)="10 STEP MOVEMENT"
1290 Choices$(4)="1 STEP MOVEMENT"
1300 Distance$(0)="10000"
1310 Distance$(1)="1000"
1320 Distance$(2)="100"
1330 Distance$(3)="10"
1340 Distance$(4)="1"
1350 Num_of_choices=5
1360 Stage_pos$="0"
1370 Label **
1380 ** ALLOW MONITORING OF THE LOAD CELLS **
1390 ** AND DETECTORS TO SETUP THE EXPERIMENT **
1400 ** ALLOW ADJUSTMENT OF THE STAGE POSITION **
1410 Label ***********
1420 ON KEY 7 Label "CELL 1 F1" GOTO 1780
1430 ON KEY 8 Label "CELL 2 F2" GOTO 1850
1440 ON KEY 9 Label "CELL 3 F3" GOTO 1920
1450 ON KEY 4 Label "EXIT P. F4" GOTO 1990
1460 ON KEY 5 Label "MEAS P. F5" GOTO 2060
1470 ON KEY 6 Label "STAGE F6" GOTO 2480
1480 ON KEY 7 Label "PROCEED F7" GOTO 2510
1490 CLEAR SCREEN
1500 MOVE 10,80
1510 Label **********
1520 Label *** PRE-EXPERIMENT SETUP SECTION ***
1530 Label ***
1540 Label *** ADJUST LOAD CELLS AND DETECTORS ***
1/20 LABEL **
1730 LABEL *** CHOOSE AN ITEM USING THE SOFTKEYS ***
1740 LABEL *** OR CONTINUE WITH THE EXPERIMENT ***
1750 LABEL *** BY PRESSING 'PROCEED' ***
1760 LABEL ***********************************************
1770 GOTO 1770
1/80 Message$="** ADJUST PRISM 1 ***
1790 CLEAR SCREEN
1800 CSIZE 5
1810 Multiplier=10
1820 Tail$=" lbs"
1830 OUTPUT @Voltmeter;"CHAN 0"
1840 GOTO 2120
1850 Message$="** ADJUST PRISM 2 ***
1860 CLEAR SCREEN
1870 CSIZE 5
1880 Multiplier=10
1890 Tail$=" lbs"
1900 OUTPUT @Voltmeter;"CHAN 1"
1910 GOTO 2120
1920 Message$="** ADJUST PRISM 3 ***
1930 CLEAR SCREEN
1940 CSIZE 5
1950 Multiplier=10
1960 Tail$=" lbs"
1970 OUTPUT @Voltmeter;"CHAN 2"
1980 GOTO 2120
1990 Message$="** EXIT PRISM INTENSITY ***
2000 CLEAR SCREEN
2010 CSIZE 5
2020 Multiplier=1
2030 Tail$=" V"
2040 OUTPUT @Voltmeter;"CHAN 3"
2050 GOTO 2120
2060 Message$="** MEAS PRISM INTENSITY ***
2070 CLEAR SCREEN
2080 CSIZE 5
2090 Multiplier=1
2100 Tail$=" V"
2110 OUTPUT @Voltmeter;"CHAN 4"
2120 MOVE 10,90
2130 LABEL ***********************************************
2140 LABEL Message$
2150 LABEL ***********************************************
2160 LABEL
2170 LABEL " PRESS ANY KEY TO CONTINUE"
2180 LABEL "AFTER ADJUSTMENT OR MONITORING"
2190 !
2200 ***********************************************
2210 ** ON SCREEN MONITOR OF MEASURED VALUE FROM VOLTMETER **
2220 ***********************************************
2230 !
2240 ON KBD GOTO 2380
2250 MOVE 5,50
2260 LABEL "YOUR CHOSEN ITEM HAS THIS CURRENT VALUE:"
2270 CSIZE 15
2280 Old_imp_value$=Imp_value$
2290 ENTER @Voltmeter;Imp_value
2300  imp_values=VALS(imp_value*multiplier)&tail$
2310  move 5,30
2320  pen -1
2330  label old_imp_values
2340  move 5,30
2350  pen 1
2360  label imp_values
2370  goto 2280
2380  off kbd
2390  clear screen
2400  csize 5
2410  goto 1660
2420  ****************************************************
2430  !** call subroutine that allows 'manual' positioning **
2440  ****************************************************
2450  call manual(num_of_choices,choices$(*)
2460  goto 1660
2470  off kbd
2480  off key
2490  clear screen
2500  csize 5
2510  ****************************************************
2520  !** call subroutine that allows 'manual' positioning **
2530  ****************************************************
2540  call manual(num_of_choices,choices$(*)
2550  goto 1660
2560  if use_motor=0 then goto 3160 ! skip when not using hurst
2570  ****************************************************
2580  print "num_of_choices=4"
2590  choices$(0)="1000 steps"
2600  choices$(1)="100 steps"
2610  choices$(2)="10 steps"
2620  choices$(3)="1 step"
2630  steps$(0)="1000"
2640  steps$(1)="100"
2650  steps$(2)="10"
2660  steps$(3)="1"
2670  total_steps=0
2680  call hurst(num_of_choices,choices$(*)
2690  !
2700  !
2710  !
2720  !
2730  !
2740  !
2750  !
2760  !
2770  !
2780  !
2790  !
2800  !
2810  !
2820  !
2830  !
*** SET THE TARGET PRESSURE FOR THE AUTOMATED MEASUREMENTS ***

Target_press=Pressure2

*** REDUCE INITIAL NUMBER OF STEPS SLIGHTLY ***

IF Initial_steps<50 THEN GOTO 3160
IF Initial_steps>5000 THEN
    Initial_steps=Initial_steps-100
ELSE
    Initial_steps=Initial_steps-50
END IF

*** END OF MANUAL SETUP ***

*** BEGIN EXPERIMENT ***

*** INITIALIZE VALUES FOR SCROLL MENU ***

Choices$(0)="MOVE TOWARD OR AWAY FROM MOTOR? (T/A)"
Choices$(1)="TRAVEL INCREMENT IN mm (.1 - 10)"
Choices$(2)="TOTAL MEAS. DIST. IN mm (1<D<100)"
Choices$(3)="ENTER THE FILENAME"
Choices$(4)="EXIT"
Num_of_choices=5

Constraints$(0)="T,0,0,T,A,!"
Constraints$(1)="N,10,.1,0,0,!
Constraints$(2)="N,100,1,0,0,!
Constraints$(3)="T,0,0,,!

*** CALL SUBROUTINE FOR INPUTING EXPERIMENT PARAMETERS ***

CALL Chooser(Num_of_choices,Choices$(0),Constraints$(0),Value$(0))
CLEAR SCREEN
Filename$=Filevalue$(3)

*** RESET POSITION COUNTER AND DISPLAY TO ZERO ***

OUTPUT @Daedal USING "+,K";*X0 DA ",END
3420 !** CONVERSION AND PARAMETER SETUP ROUTINES **
3430 !**********************************************************
3440 !
3450 !**********************************************************
3460 !** SET COMMAND FOR CONTROLLER FOR FORWARD OR **
3470 !** REVERSE TRAVEL DIRECTION **
3480 !**********************************************************
3490 !
3500 IF Value$(0)="A" THEN
3510   Direction$=""
3520 ELSE
3530   Direction$="--"
3540 END IF
3550 !
3560 !**********************************************************
3570 !** CONVERT MILLIMETERS TO INCHES **
3580 !** GET MOTOR STEPS PER INCREMENT **
3590 !**********************************************************
3600 !
3610 Increment=VAL(Value$(1))/25.4
3620 Total_dist=VAL(Value$(2))/25.4
3630 Inc_step=INT(Increment/.00002)
3640 Step_count=INT(Total_dist/Increment)
3650 !**********************************************************
3660 !**********************************************************
3670 !** INFORM USER OF THE NUMBER OF MEASUREMENT STEPS **
3680 !**********************************************************
3690 !
3700 PRINT "THE TOTAL NUMBER OF MEASUREMENT STEPS WILL BE ";Step_count
3710 WAIT 2
3720 CLEAR SCREEN
3730 !
3740 !**********************************************************
3750 !** DIMENSION AN ARRAY FOR StORING THE DATA **
3760 !**********************************************************
3770 !
3780 ALLOCATE Datums(Step_count-1,8)
3790 !
3800 !**********************************************************
3810 !** INFORM USER THAT MEASUREMENTS WILL BEGIN **
3820 !**********************************************************
3830 !
3840 MOVE 15,70
3850 LABEL "MEASUREMENT WILL NOW BEGIN"
3860 LABEL "MEASUREMENT WILL NOW BEGIN"
3870 LABEL "MEASUREMENT WILL NOW BEGIN"
3880 WAIT 2
3890 CLEAR SCREEN
3900 !
3910 !**********************************************************
3920 !** GENERATE COMMAND STRING FOR TRAVEL DIRECTION **
3930 !** AND INCREMENTAL DISTANCE STEP. THIS MOVE IS **
3940 !** EXECUTED IN THE UPCOMING FOR/NEXT LOOP **
3950 !**********************************************************
3960 !
3970 Command$="D"&Direction$&VAL$(Inc_step)&" "
3980 !
3990 !**********************************************************
4000 ** THIS SENDS THE COMMAND TO THE CONTROLLER **
4010 ** NO ACTUAL MOVE IS DONE YET THOUGH  **
4020 *******************************************************
4030
4040 OUTPUT @Daedal USING "#,K";CommandS,END
4050
4060 *******************************************************
4070 ** INITIALIZE STAGE POSITION VARIABLE TO ZERO **
4080 ******************************************************
4090
4100 Position=0
4110
4120 *******************************************************
4130 ** MOVE/MEASUREMENT FOR/NEXT LOOP **
4140 *******************************************************
4150
4160 FOR I=0 TO Step_count-1
4170 IF Out_of_plane=l THEN GOTO 4690 ! SKIP FOR OUT OF PLANE
4180 IF Use_motor=l THEN GOTO 4540 ! SKIP WHEN USING HURST MOTOR
4190
4200 MOVE 15,70
4210 LABEL ******************************
4220 LABEL ** TIGHTEN MEASUREMENT PRISM **
4230 LABEL *****************************************************
4240 LABEL
4250 LABEL "PRESS ANY KEY TO CONTINUE AFTER"
4260 LABEL "AFTER PRISM PRESSURE STABILIZES"
4270
4280 *******************************************************
4290 ** ON SCREEN MONITOR OF MEASUREMENT PRISM PRESSURE **
4300 *******************************************************
4310
4320 ON KBD GOTO 4470
4330 OUTPUT @Voltmeter;"CHAN 1"
4340 CSIZE 15
4350 Old_pressure2=Pressure2
4360 Old_pressure2$=VAL$(Old_pressure2)&" Ibs"
4370 ENTER @Voltmeter;Pressure2
4380 Pressure2=10*Pressure2
4390 Pressure2$=VAL$(Pressure2)&" ibs"
4400 PEN -1
4410 MOVE 5,30
4420 LABEL Old_pressure2$
4430 PEN 1
4440 MOVE 5,30
4450 LABEL Pressure2$
4460 GOTO 4350
4470 OFF KBD
4480 GOTO 4690
4490
4500 ********************************************************
4510 ** STEPPER MOTOR ROUTINE GOES HERE **
4520 ********************************************************
4530
4540 Total_steps=Initial_steps
4550 OUTPUT @Voltmeter;"CHAN 1"
4560 OUTPUT @Multi;"WF,8,1,0,8.2,5000T,OP,8,"&VAL$(Initial_steps)&"T"
4570 WAIT .005*Initial_steps ! WAIT UNTIL FINISHED
4580 ENTER @Voltmeter;Pressure2 !GET NEW PRESSURE
4590 !
4600 IF Pressure2>Target_press THEN
4610 GOTO 4690
4620 ELSE
4630 OUTPUT @Multi;"WF,8.1,0,8.2,5000T,OP,8,10T"
4640 WAIT .05 ! WAIT UNTIL FINISHED
4650 Total_steps=Total_steps+10
4660 GOTO 4580
4670 END IF
4680 !
4690 CSIZE 5
4700 CLEAR SCREEN
4710 !
4720 !***************************
4730 !** TAKE MEASUREMENTS **
4740 !***************************
4750 !
4760 OUTPUT @Voltmeter;"CHAN 3"
4770 PRINT "Reading Exit Prism Intensity..."
4780 ENTER @Voltmeter;Exit_prism
4790 DISP "EXIT PRISM READING WAS ";Exit_prism
4800 WAIT 1
4810 OUTPUT @Voltmeter;"CHAN 4"
4820 PRINT "Reading Measurement Prism Intensity..."
4830 ENTER @Voltmeter;Meas_prism
4840 DISP "MEASUREMENT PRISM READING WAS ";Meas_prism
4850 WAIT 1
4860 OUTPUT @Voltmeter;"CHAN 0"
4870 PRINT "Reading Load Cell 1 Pressure..."
4880 ENTER @Voltmeter;Pressure1
4890 DISP "LOAD CELL 1 READING WAS ";Pressure1
4900 WAIT 1
4910 !
4920 IF Out_of-plane=1 THEN GOTO 4990 ! SKIP FOR OUT OF PLANE
4930 !
4940 OUTPUT @Voltmeter;"CHAN 1"
4950 PRINT "Reading Load Cell 2 Pressure..."
4960 ENTER @Voltmeter;Pressure2
4970 DISP "LOAD CELL 2 READING WAS ";Pressure2
4980 WAIT 1
4990 OUTPUT @Voltmeter;"CHAN 2"
5000 PRINT "Reading Load Cell 3 Pressure..."
5010 ENTER @Voltmeter;Pressure3
5020 DISP "LOAD CELL 3 READING WAS ";Pressure3
5030 WAIT 1
5040 Datums(I,0)=ABS(Position*.00002*25.4)! STAGE POSITION
5050 Datums(I,1)=Exit_prism! EXIT PRISM
5060 Datums(I,2)=Meas_prism! MEAS PRISM
5070 Datums(I,3)=10*Pressure1! LDCELL1
5080 Datums(I,4)=10*Pressure2! LDCELL2
5090 Datums(I,5)=10*Pressure3! LDCELL3
5100 CLEAR SCREEN
5110 PRINT "THE LAST MEASUREMENT POSITION WAS ";Datums(I,0);" mm"
5120 !
5130 !***************************
5140 !** END OF TAKE MEASUREMENTS SECTION **
5150 !***************************
IF Out-of-plane=1 THEN GOTO 5720 ! SKIP FOR OUT OF PLANE

IF Use_motor=1 THEN GOTO 5690 ! SKIP WHEN USING HURST MOTOR

PEN 1

MOVE 15, 10

LABEL "***************

** LOOSEN MEASUREMENT PRISM **

***************

LABEL "PRESS ANY KEY TO CONTINUE AFTER"

LABEL "THE PRISM IS OFF THE SAMPLE"

! ***************

** ON SCREEN MONITOR OF MEASUREMENT PRISM PRESSURE **

! ***************

ON KBD GOTO 5530

OUTPUT @Voltmeter; "CHAN 1"

CSIZE 15

Old_no_press= No_press

Old_no_press$= VAL$(Old_no_press) &" lbs"

ENTER @Voltmeter; No_press

No_press= 10* No_press

No_press$= VAL$(No_press) &" lbs"

PEN -1

MOVE 5, 30

LABEL Old_no_press$

PEN 1

MOVE 5, 30

LABEL No_press$

GOTO 5350

PEN -1

MOVE 5, 30

LABEL Old_no_press$

GOTO 5350

! NO MOVEMENT ALLOWED UNTIL PRISM

IS OFF OF THE SAMPLE

! ***************

IF No_press>.08 THEN

PEN -1

MOVE 5, 30

LABEL Old_no_press$

GOTO 5350

END IF

OFF KBD

PEN -1

MOVE 5, 30

LABEL No_press$

GOTO 5720

PEN -1

! ***************

** RAISE HURST MOTOR ROUTINE HERE **

***************

OUTPUT @Multi; "WF,8.1,0,8.2,5000T,OP,8,-"& VAL$(Total_steps)&"T"

WAIT .005*Total_steps

PEN 1

CSIZE 5
5740 CLEAR SCREEN
5750 !
5760 !*************************************************************************************
5770 ** READ EXIT PRISM QUIESCENT INTENSITY **
5780 !*************************************************************************************
5790 !
5800 OUTPUT @Voltmeter:"CHAN 3"
5810 ENTER @Voltmeter:Exit_prism
5820 Datums(I,6)-Exit_prism! EXIT PRISM QUIESCENT
5830 !
5840 !*************************************************************************************
5850 ** CALCULATE (P3o*P2)/(P3o-P3) AND ITS LOG **
5860 !*************************************************************************************
5870 !
5880 Datums(I,7)=((Datums(I,6)*Datums(I,2))/(Datums(I,6)-Datums(I,1)))
5890 ON ERROR GOTO 5920 :CANT'T TAKE LOG OF NEG NUMBER
5900 Datums(I,8)=LGT(Datums(I,7))
5910 GOTO 5930
5920 Datums(I,8)=0
5930 OFF ERROR
5940 !
5950 !*************************************************************************************
5960 ** PREVENTS LAST MOVE COMMAND **
5970 !*************************************************************************************
5980 !
5990 IF I=Step_count-1 THEN GOTO 6210
6000 OUTPUT @Daedal USING ",","G ",END!** MOVE COMMAND **
6010 !
6020 !*************************************************************************************
6030 ** WAIT UNTIL MOVE IS COMPLETE **
6040 !*************************************************************************************
6050 !
6060 IF SPOLL(@Daedal)=1 THEN GOTO 6140
6070 GOTO 6060
6080 !
6090 !*************************************************************************************
6100 ** REQUEST AND RECEIVE POSITION INFO **
6110 ** FROM THE STAGE CONTROLLER **
6120 !*************************************************************************************
6130 :
6140 OUTPUT @Daedal USING ",",X1 ",END
6150 Test=SPOLL(@Daedal)
6160 IF Test>8 THEN GOTO 6180!GET FROM CONTROLLER?
6170 GOTO 6150
6180 WAIT .5
6190 ENTER @Daedal;Position$!RECEIVE POSITION
6200 Position=VAL(Position$)
6210 NEXT I
6220 !
6230 !*************************************************************************************
6240 ** END OF ACQUISITION FOR/NEXT LOOP **
6250 !*************************************************************************************
6260 !
6270 CLEAR SCREEN
6280 !
6290 !*************************************************************************************
6300 ** DISABLE IEEE-488 FOR CONTROLLER **
6310 !*************************************************************************************
CALL Data_converter(Datums(*),Filename$)

END

*****************************************************************
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
THE CURRENT POSITION OF THE STAGE IS TAKEN AS ZERO. THIS VARIABLE WILL GIVE THE USER THE RELATIVE POSITION SINCE BEGINNING THE PROGRAM.

*******************************************************************
THE NEXT 4 LINES CLEAR THE SCREEN, SET THE STARTING POINT FOR LABELS, SET THE PEN TO WRITE AND SET THE CHARACTER SIZE.

**************
CLEAR SCREEN
PEN 1
CSIZE 4,.5

**************
THIS LINE MAKES THE CHOOSING CURSOR A WHITE BLOCK.

**************
FOR I=0 TO Num of choices-1
MOVE 10,95-5*I
LABEL Choices$(I)
NEXT I

**************
THE NEXT 3 LINES PLACES THE CURSOR BESIDE THE 1st CHOICE.

**************
MOVE 5,95
I=0
LABEL Cursors

**************
THE NEXT TWO LINES PRINTS THE CURRENT POSITION OF THE STAGE.

**************
MOVE 10,20
LABEL "THE CURRENT POSITION IS ";Stage_pos$

**************
THE NEXT 4 LINES PRINTS AN OPERATING INSTRUCTION ON THE SCREEN.

**************
MOVE 10,30
LABEL "PRESS RETURN WHEN MANUAL ADJUSTMENT IS COMPLETE"
MOVE 10,35
LABEL "USE LEFT/RIGHT ARROWS TO MOVE MOTOR"
MOVE 10,40
LABEL "USE UP/DOWN ARROWS TO SCROLL MENU"
WHEN A KEY IS STRUCK GO TO THE MOVE CURSOR ROUTINE

** THIS ROUTINE ONLY TAKES ACTION WHEN THE UP, DOWN OR RETURN KEY IS PRESSED. GO TO Up ON UP KEY, GO TO Down ON DOWN KEY, GO TO Take_data ON RETURN

** MOVEMENT ROUTINE

** Up MOVEMENT ROUTINE **

** END OF UP MOVEMENT ROUTINE **

** END OF MOVE CURSOR ROUTINE **

** END OF MOVE CURSOR ROUTINE **

** END OF MOVE CURSOR ROUTINE **

** END OF MOVE CURSOR ROUTINE **

** END OF MOVE CURSOR ROUTINE **
8050 ** DOWN MOVEMENT ROUTINE **
8060 ********************************************
8070 !
8080 ** THIS SCROLLS THE CURSOR DOWN **
8090 ********************************************
8100 !
8110 !
8120 Down:
8130 PEN -1
8140 MOVE 5,95-5*I
8150 LABEL Cursors
8160 IF I=Num_of_choices-1 THEN
8170 I=0
8180 ELSE
8190 I=I+1
8200 END IF
8210 PEN 1
8220 MOVE 5,95-5*I
8230 LABEL Cursors
8240 GOTO Diddle_loop
8250 !
8260 ********************************************
8270 ** END OF DOWN MOVEMENT ROUTINE **
8280 ********************************************
8290 !
8300 ********************************************
8310 ** BEGIN STAGE MOVEMENT ROUTINES **
8320 ********************************************
8330 !
8340 ********************************************
8350 ** THIS SUBROUTINE MOVES THE RAIL TABLE STAGE TO **
8360 ** THE LEFT (TOWARD THE STEPPER MOTOR) THE NUMBER **
8370 ** OF STEPS INDICATED BY Distance$(I). THE **
8380 ** COMPUTER THEN POLLS THE MOTION CONTROLLER TO **
8390 ** DETERMINE WHEN THE MOVE HAS BEEN COMPLETED. **
8400 ** AFTER THE MOVE IS COMPLETE THE **
8410 ** COMPUTER REQUESTS THE CURRENT RELATIVE **
8420 ** POSITION OF THE STAGE AND PRINTS IT ON THE **
8430 ** SCREEN **
8440 ********************************************
8450 !
8460 Move_stage_lft:
8470 !
8480 ********************************************
8490 ** THE OUTPUT STATEMENT PERFORMS THE FOLLOWING: '##,K'IN USING **
8500 ** SUPPRESSES EOL AT THE END OF THE COMMAND AND STRIPS BLANKS **
8510 ** FROM ANY CHARACTERS, THE CHARACTER STRING SENT IS **
8520 ** 'D-XXXXXXX G' WHERE XXXXXXX IS A NUMBER. THE ',END' IS **
8530 ** USED TO SUPPRESS EOL WHEN THERE IS NO DATA SENT AND TO SEND **
8540 ** THE END-OR-IDENTIFY WHEN A COMMAND IS SENT. SEE PAGE 450 **
8550 ** OF THE BASIC 5.0/5.1 LANGUAGE REFERENCE VOL. 2:O-Z FOR AN **
8560 ** EQUALLY USELESS EXPLANATION. THIS IS ALL DUE **
8570 ** TO DAEDAL'S PISS-POOR IMPLEMENTATION OF THE GPIB COMMAND **
8580 ** SET **
8590 ********************************************
8600 !
8610 !
8620 !
8630 !
8640 !
8650 !
8660 !
8670 !
8680 !
8690 OUTPUT @Daedal USING "##,K","D-"&Distance$(I)&" G ",END
8700
8630 !***********************************************************
8640 ** SERIAL POLL THE MOTION CONTROLLER TO DETERMINE WHEN STAGE **
8650 ** MOVEMENT IS COMPLETE. CONTROLLER WILL RETURN A 1 **
8660 !***********************************************************
8670 !
8680 Move_done=SPOLL(@Daedal)
8690 IF Move_done<>1 THEN GOTO 8680
8700 !
8710 !***********************************************************
8720 ** THIS OUTPUT STATEMENT REQUESTS THE CURRENT RELATIVE **
8730 ** POSITION OF THE STAGE FROM THE MOTION CONTROLLER **
8740 !***********************************************************
8750 !
8760 !***********************************************************
8770 ** SERIAL POLL THE MOTION CONTROLLER TO DETERMINE WHEN THE **
8780 ** CURRENT POSITION INFORMATION IS AVAILABLE **
8790 !***********************************************************
8800 !
8810 Answer_back=SPOLLU(@Daedal)
8820 IF Answer_back<8 THEN GOTO 8830
8830 !
8840 !***********************************************************
8850 ** CLEAR THE OLD CURRENT POSITION FROM THE SCREEN THEN READ **
8860 ** THE NEW CURRENT POSITION INFORMATION AND PRINT IT ON THE **
8870 ** SCREEN **
8880 !***********************************************************
8890 !
8900 !
8910 !
8920 Move-stage-rgt:
8930 PEN -1
8940 LABEL "THE CURRENT POSITION IS ";Stage_pos$
8950 ENTER @Daedal USING "k";Stage_pos$
8960 MOVE 10,20
8970 PEN 1
8980 LABEL "THE CURRENT POSITION IS ";Stage_pos$
8990 GOTO Diddle_loop
9000 !
9010 !***********************************************************
9020 ** THIS SUBROUTINE MOVES THE RAIL TABLE STAGE TO **
9030 ** THE RIGHT (AWAY FROM THE STEPPER MOTOR) THE **
9040 ** NUMBER OF STEPS INDICATED BY DistanceS(I). THE **
9050 ** COMPUTER THEN POLLS THE MOTION CONTROLLER TO **
9060 ** DETERMINE WHEN THE MOVE HAS BEEN COMPLETED. **
9070 ** AFTER THE MOVE IS COMPLETE THE **
9080 ** COMPUTER REQUESTS THE CURRENT RELATIVE **
9090 ** POSITION OF THE STAGE AND PRINTS IT ON THE **
9100 ** SCREEN **
9110 !***********************************************************
9120 !
9130 Move-stage-rgt: !
9140 !
9150 !***********************************************************
9160 ** THE OUTPUT STATEMENT PERFORMS THE FOLLOWING: '"k'IN USING **
9170 ** SUPPRESSES EOL AT THE END OF THE COMMAND AND STRIPS BLANKS **
9180 ** FROM ANY CHARACTERS. THE CHARACTER STRING SENT IS **
9190 ** 'DXXXXXXX G ' WHERE XXXXXXX IS A NUMBER. THE ',END' IS **
9200 ** USED TO SUPPRESS EOL WHEN THERE IS NO DATA SENT AND TO SEND **
99
THE END-OF-IDENTIFY WHEN A COMMAND IS SENT. SEE PAGE 450

** OF THE BASIC 5.0/5.1 LANGUAGE REFERENCE VOL. 2:O-Z FOR AN

** EQUALLY USELESS EXPLANATION. THIS IS ALL DUE

** TO DAEDAL'S PISS-POOR IMPLEMENTATION OF THE GPIB COMMAND

** SET

******************************************************************************

OUTPUT #Daedal USING "#,K";"D"&Distance$1);" G ",END

******************************************************************************

** SERIAL POLL THE MOTION CONTROLLER TO DETERMINE WHEN STAGE

** MOVEMENT IS COMPLETE. CONTROLLER WILL RETURN A 1

******************************************************************************

Move_done=SPOLL(#Daedal)

IF Move_done<>1 THEN GOTO 9350

******************************************************************************

** THIS OUTPUT STATEMENT REQUESTS THE CURRENT RELATIVE

** POSITION OF THE STAGE FROM THE MOTION CONTROLLER

******************************************************************************

OUTPUT #Daedal USING "#,K";"X1 ",END

******************************************************************************

** SERIAL POLL THE MOTION CONTROLLER TO DETERMINE WHEN THE

** CURRENT POSITION INFORMATION IS AVAILABLE

******************************************************************************

Answer_back=SPOLL(#Daedal)

IF Answer_back<8 THEN GOTO 9500

******************************************************************************

** CLEAR THE OLD CURRENT POSITION FROM THE SCREEN THEN READ

** THE NEW CURRENT POSITION INFORMATION AND PRINT IT ON THE

** SCREEN

******************************************************************************

MOVE 10,20

PEN -1

LABEL "THE CURRENT POSITION IS ";Stage_pos$ 

ENTER #Daedal USING "K";Stage_pos$

MOVE 10,20

PEN 1

LABEL "THE CURRENT POSITION IS ";Stage_pos$

GOTO Diddle_loop

******************************************************************************

** END STAGE MOVEMENT ROUTINES **

******************************************************************************

** IF THE RETURN KEY IS PRESSED THE SUBPROGRAM TERMINATES **

******************************************************************************

Finished:

SUBEND
SUB Chooser(Num_of_choices,Choices$(*) ,Constraints$(*) ,Value$(*) )

** THIS SUBROUTINE ALLOWS MENU-LIKE INPUT OF INFORMATION **

** VARIABLE DESCRIPTIONS: **

** Num_of_choices - THIS IS THE TOTAL NUMBER OF ITEMS **

** Choices$(*) - THIS IS A SERIES OF STRINGS CONTAINING **

** THE TEXT ASKING FOR INPUT INFORMATION **

** OPTION BASE 0 MUST BE USED e.g. WHEN **

** Num_of_choices=3 then Choices$(*) MUST **

** BE NUMBERED Choices$(0), Choices$(1), AND **

** Choices$(2) **

** Constraints$(*) - THIS IS A SERIES OF STRINGS THAT **

** CONTAIN TESTING VALUES AND LIMITS **

** Validity of entered values. These **

** Strings are used to verify input **

** Values as number or text, within **

** High and low limits if a number **

** or a legal response if text **

** Value$(*) - THESE STRINGS CONTAIN THE ENTERED INFO **

** That will be returned to the main program **

** Erase$ IS USED WHEN CORRECTING A MISTAKE IN **

** An entered value **

DIM Erase$[30]

Erase$=""

** Temp$ IS A TEMPORARY STORAGE STRING THAT IS USED **

** When the user is inputting information. When **

** Temp$ IS COMPLETED IT IS TRANSFERRED TO Value$(1) **

DIM Temp$[30]

Temp$=""

** The next 4 lines clear the screen, set the starting **

** point for labels, set the pen to write and set the **

** character size **

CLEAR SCREEN

LORG 2

PEN 1

CSIZE 4,.5

** This line makes the choosing cursor a white block **

Cursor$=CHR$(127)

101
**THE FOR NEXT LOOP PUTS THE CHOICES ON THE SCREEN**

```
FOR I=0 TO Num_of_choices-1
    MOVE 10,95-5*I
    LABEL Choices$(I)
NEXT I
```

```
IF Value$(I)="" THEN GOTO 10520
```

```
FOR I=0 TO Num_of_choices-2
    MOVE 85,95-5*I
    LABEL Value$(I)
NEXT I
```

**THE NEXT 3 LINES PLACES THE CURSOR_BESIDE THE 1st CHOICE**

```
MOVE 5,95
I=0
LABEL Cursor$
```

**WHEN A KEY IS STRUCK GO TO THE MOVE CURSOR ROUTINE**

```
Diddle-loop: ON KBD GOTO Movement
GOTO 10590
```

**MOVE CURSOR ROUTINE**

```
Movement:!
```

**WHEN A KEY IS PRESSED, THIS ROUTINE DETERMINES**

```
Where$=KBD$
```

**IF THE KEYSTROKE WAS AN UP OR DOWN ARROW OR IF IT**

```
OFF KBD
```

**WAS A CHARACTER KEY. THE ARROWS MEAN MOVE THE**

```
IF Where$=CHR$(255)&"V" THEN Down
```

**CURSOR UP OR DOWN THROUGH THE CHOICES ON THE**

```
IF Where$=CHR$(255)&"^" THEN Up
```

**SCREEN. A CHARACTER KEY MEANS THAT INFO IS BEING**

```
IF LEN(Where$)=1 THEN DISP Where$
```

**ENTERED. THE "DISP Where$" PLACES THE FIRST CHAR.**

```
IF Where$<>"" THEN
```

**ON THE SCREEN BEFORE EXITING TO THE Take_data**

```
GOTO Take_data
```

**IN ORDER TO MAKE INPUT VIEWING MORE SMOOTH**

```
END IF
```

**END OF MOVE CURSOR ROUTINE**

```
GOTO Diddle_loop
```

**UP MOVEMENT ROUTINE**

```
```

**THIS SCROLLS THE CURSOR UP**

```
```
10950 Up:
10960 PEN -1
10970 MOVE 5,95-5*I
10980 LABEL Cursor$
10990 IF I=0 THEN
11000 I=Num_of_choices-1
11010 ELSE
11020 I=I-1
11030 END IF
11040 PEN 1
11050 MOVE 5,95-5*I
11060 LABEL Cursor$
11070 GOTO Diddle_loop
11080 !*****************************************************
11090 !** END OF UP MOVEMENT ROUTINE **
11100 !*****************************************************
11110!
11120 !*****************************************************
11130 !** DOWN MOVEMENT ROUTINE **
11140 !*****************************************************
11150!
11160 !*****************************************************
11170 !** THIS SCROLLS THE CURSOR DOWN **
11180 !*****************************************************
11190 Down:!
11200 PEN -1
11210 MOVE 5,95-5*I
11220 LABEL Cursor$
11230 IF I=Num_of_choices-1 THEN
11240 I=0
11250 ELSE
11260 I=I+1
11270 END IF
11280 PEN 1
11290 MOVE 5,95-5*I
11300 LABEL Cursor$
11310 GOTO Diddle_loop
11320 !*****************************************************
11330 !** END OF DOWN MOVEMENT ROUTINE **
11340 !*****************************************************
11350!
11360 !*****************************************************
11370 !** TAKE DATA ROUTINE **
11380 !*****************************************************
11390 !*****************************************************
11400 !** THIS LOOP TAKES YOUR ENTRY FOR A **
11410 !** PARTICULAR PARAMETER AND DETERMINES **
11420 !** IF IT IS VALID GIVEN THE CONSTRAINTS **
11430 !** SET FORTH IN THE CONSTRAINTS$ STRING **
11440 !*****************************************************
11450!
11460 !*****************************************************
11470 !** CHECKS TO SEE IF "EXIT" WAS CHOSEN **
11480 !*****************************************************
11490 Take_data: IF I=Num_of_choices-1 THEN
11500 GOTO Done
11510 END IF
11520 !*****************************************************
11530 !** THE FIRST CHARACTER OF THE TempS STRING **
11540 !** IS TAKEN FROM THE WhereS STRING SINCE **
11550 !** THIS WAS THE FIRST CHARACTER PRESSED **
11560 !** OTHER THAN AN UP OR DOWN ARROW KEY **
11570 !** Where$ IS THEN CLEARED **
11580 *********************************************
11590 Temp$=Where$
11600 Where$=""
11610 *********************************************
11620 !** SET PEN TO ERASE, ERASE CURSOR, SET PEN TO WRITE, **
11630 !** MOVE TO THE LOCATION WHERE THE VALID ENTRY WILL **
11640 !** BE DISPLAYED **
11650 *********************************************
11660 PEN -1
11670 LABEL CursorS
11680 PEN 1
11690 MOVE 85,95-5*I
11700 *********************************************
11710 !** GO TO THE SUBROUTINE THAT STRIPS THE CONSTRAINT **
11720 !** PARAMETERS FROM THE PROPER CONSTRAINTS STRING **
11730 !** FOR THE CHOSEN ENTRY ITEM **
11740 *********************************************
11750 GOSUB Extract:
11760 *********************************************
11770 !** UPDATE THE ERASE STRING, GENERATE AN ATTENTION TONE **
11780 !** AND REQUEST AN INPUT **
11790 *********************************************
11800 Erase$=Value$(I)
11810 ON KBD GOTO Build_string
11820 GOTO 11820
11830 Build_string:
11840 Char$=KBD$  
11850 IF Char$=CHR$(255)&"E" THEN GOTO 12050
11860 IF Char$=CHR$(255)&"B" THEN
11870 ON ERROR GOTO 11890
11880 TempS=TempS[1,LEN(TempS)-1]
11890 OFF ERROR
11900 GOTO 12030
11910 END IF
11920 IF Char$=CHR$(255)&"^" THEN
11930 Temp$=""
11940 DISP TempS
11950 GOTO Up
11960 END IF
11970 IF Char$=CHR$(255)&"V" THEN
11980 Temp$=""
11990 DISP TempS
12000 GOTO Down
12010 END IF
12020 TempS=TempS&Char$
12030 DISP TempS
12040 GOTO 11820
12050 OFF KBD
12060 Value$(I)=Temp$
12070 Temp$=""
12080 Where$=""
12090 *********************************************
12100 !** VALID ENTRY CHECK ROUTINE **
12110 !**********************************************************************
12120 !
12130 !**********************************************************************
12140 !** IF THE ENTRY IS SUPPOSED TO BE A NUMBER **
12150 !** CHECK TO SEE THAT IT IS AND STRIP OFF **
12160 !** TRAILING TEXT. THEN TEST TO SEE IF THE **
12170 !** NUMBER IS BETWEEN THE SPECIFIED HIGH **
12180 !** AND LOW VALUES. IF ANY OF THESE **
12190 !** CONDITIONS ARE NOT MET THEN REJECT **
12200 !**********************************************************************
12210 IF N_t$="N" THEN
12220    ON ERROR GOTO 12490
12230    Number=VAL(Value$(I))
12240    OFF ERROR
12250    Value$(I)=VALS(Number)
12260    IF Number>VAL(High$) THEN
12270       Value$(I)=Erase$
12280       GOTO 12490
12290    END IF
12300    IF Number<VAL(Low$) THEN
12310       Value$(I)=Erase$
12320       GOTO 12490
12330    END IF
12340 ELSE
12350 !**********************************************************************
12360 !** IF THE ENTRY IS SUPPOSED TO BE TEXT THEN CHECK **
12370 !** TO SEE THAT IT IS ONE OF THE VALID RESPONSES **
12380 !** IF NOT THEN REJECT **
12390 !**********************************************************************
12400    IF Text1$<>"" THEN GOTO 12430
12410    IF Text2$<>'" THEN GOTO 12430
12420    GOTO 12610
12430    IF Value$(I)=Text1$ THEN GOTO 12610
12440    IF Value$(I)=Text2$ THEN GOTO 12610
12450    Value$(I)=Erase$
12460    GOTO 12490
12470    END IF
12480    GOTO 12610
12490    BEEP 83.1,.2
12500    Temp$=""
12510    DISP Temp$
12520    GOTO 11810
12530 !**********************************************************************
12540 !** END ENTRY CHECK ROUTINE **
12550 !**********************************************************************
12560 !
12570 !**********************************************************************
12580 !** IF THE ENTRY WAS VALID THEN ERASE THE OLD VALUE **
12590 !** AND REPLACE IT WITH THE NEW VALUE **
12600 !**********************************************************************
12610 PEN -1
12620 LABEL Erase$
12630 PEN 1
12640 MOVE 85,95-5*I
12650 LABEL Value$(I)
12660 GOTO Diddle_loop
12670 !**********************************************************************
12680 !** END OF TAKE DATA ROUTINE **
12690 !*****************************************************************************
12700 !
12710 !*****************************************************************************
12720 !!! EXIT ROUTINE BEGINS HERE **
12730 !*****************************************************************************
12740 Done:!!
12750 !*****************************************************************************
12760 !!! ERASE THE CURSOR FROM ITS PRESENT LOCATION **
12770 !*****************************************************************************
12780 MOVE 5,95-5*I
12790 PEN -1
12800 LABEL Cursor$
12810 PEN 1
12820 !*****************************************************************************
12830 !!! THIS FOR NEXT LOOP TESTS FOR ANY MISSING **
12840 !!! ENTRIES. IF ANY ARE FOUND THEN THE EXIT **
12850 !!! IS IGNORED UNTIL ALL ASKED FOR VALUES **
12860 !!! ARE ENTERED **
12870 !*****************************************************************************
12880 OFF KBD
12890 FOR K=0 TO Num_of_choices-2
12900 IF Value$(K)="*" THEN
12910 MOVE 5,95-5*K
12920 LABEL Cursor$
12930 I=K
12940 Where$=""
12950 GOTO 11490
12960 END IF
12970 NEXT K
12980 CLEAR SCREEN
12990 !*****************************************************************************
13000 !!! END OF EXIT ROUTINE **
13010 !*****************************************************************************
13020 !*****************************************************************************
13030 !!! IF THIS POINT IS REACHED THEN THE SUBPROGRAM **
13040 !!! IS EXITED **
13050 !*****************************************************************************
13060 GOTO 13620
13070 !*****************************************************************************
13080 !!! STRING EXTRACTION SUB-Routine **
13090 !*****************************************************************************
13100 !!! THIS ROUTINE EXTRACTS THE INFORMATION **
13110 !!! FROM THE CONSTRAINTS$ STRING AND PLACES **
13120 !!! IT IN A SERIES OF STRINGS THAT CAN BE USED **
13130 !!! TO TEST THE ENTERED VALUES STRING TO SEE **
13140 !!! IF IT IS VALID **
13150 !!!*****************************************************************************
13160 Extract:!!
13170 N_LS=""
13180 HighS=""
13190 LowS=""
13200 Text1S=""
13210 Text2S=""
13220 TS=""
13230 P=1
13240 KS=""
13250 T$=Constraints$(I)[P,P]
13260 IF T$=""," THEN GOTO Next_string
13270 IF TS="!" THEN GOTO Finish
13280 K$=K$&TS
13290 P=P+1
13300 GOTO 13250
13310 Next_string:!
13320 IF N_t$=='" THEN
13330 N_t$=K$
13340 P=P+1
13350 GOTO 13240
13360 END IF
13370 IF High$="*" THEN
13380 High$=K$
13390 P=P+1
13400 GOTO 13240
13410 END IF
13420 IF Low$="-" THEN
13430 Low$=K$
13440 P=P+1
13450 GOTO 13240
13460 END IF
13470 IF Text1$="" THEN
13480 Text1$=K$
13490 P=P+1
13500 GOTO 13240
13510 END IF
13520 IF Text2$="" THEN
13530 Text2$=K$
13540 P=P+1
13550 GOTO 13240
13560 END IF
13570 Finish:!
13580 RETURN
13590 !*****************************************************************************
13600 ! END OF THE STRING EXTRACTION SUB-Routine **
13610 :*****************************************************************************
13620 SUBEND
13630 !
13640 !*****************************************************************************
13650 ! DATA ARRAY CONVERTER AND STORAGE SUBROUTINE **
13660 !*****************************************************************************
13670 !
13680 SUB Data_converter(Datums(*),Filename$)
13690 DIM Build_string$(1000)
13700 Build_string$="*
13710 Dimen=RANK(Datums)
13720 IF Dimen<2 THEN Finish
13730 Row_lower_bound=BASE(Datums,1)
13740 Row_upper_bound=BASE(Datums,1)+SIZE(Datums,1)-1
13750 Col_lower_bound=BASE(Datums,2)
13760 Col_upper_bound=BASE(Datums,2)+SIZE(Datums,2)-1
13770 CREATE Filename$,1
13780 ASSIGN @File_path TO Filename$;FORMAT ON
13790 !*****************************************************************************
13800 ! BUILD THE DATA COLUMN HEADINGS **
13810 !*****************************************************************************
13820 Build_string$="Stage Position"&CHR$(9)&"Exit Prism"&CHR$(9)&"Meas
Prism"&CHR$(9)&"Cell 1"&CHR$(9)&"Cell 2"&CHR$(9)&"Cell 3"&CHR$(9)
107
13830 Build_string$=Build_string$&"Exit
13840 FOR J=Row_lower_bound TO Row_upper_bound STEP 1
13850 FOR I=Col_lower_bound TO Col_upper_bound STEP 1
13860 Build_string$=Build_string$&VAL$(Datums(J,I))&CHR$(9)
13870 NEXT I
13880 Build_string$=Build_string$&VAL$(Datums(J,I))&CHR$(10)
13890 OUTPUT @File_path USING ";Build-string$";
13900 Build_string$=""
13910 NEXT J
13920 ASSIGN @File_path TO *
13930 GOTO 13970
13940 Finish:
13950 BEEP 1708.98,1
13960 DISP "YOUR DATA ARRAY IS NOT TWO DIMENSIONAL"
13970 SUBEND
13980 **************************************
14000 **************************************
14020 SUB Hurst(Num_of_choices, Choices$(*), Steps$(*), @Multi, @Voltmeter, Initial_steps, Pressure2)
14030 **************************************
14050 **************************************
14060 **************************************
14070 **************************************
14080 **************************************
14090 **************************************
14100 **** VARIABLE DESCRIPTIONS: 
14130 **** Num_of_choices - THIS IS THE TOTAL NUMBER OF ITEMS LISTED IN THE MENU ON THE SCREEN
14150 **** Choices$(*) - THIS IS A SERIES OF STRINGS CONTAINING THE TEXT ASKING FOR INPUT INFORMATION 
14180 **** e.g. "1000 STEPS" 
14190 **** OPTION BASE 0 MUST BE USED e.g. WHEN Num_of_choices=3 then Choices$(*) MUST BE NUMBERED Choices$(0),Choices$(1),AND Choices$(2)
14200 **** Steps$(*) - THIS SET OF STRINGS CONTAINS THE NUMBER OF STEPS THAT WILL BE MOVED WHEN THE LEFT OR RIGHT ARROW KEY IS PRESSED
14210 **** @MULTI - THIS IS THE HPIB ADDRESS OF THE MULTIPROGRAMMER. THIS IS ASSIGNED AT THE BEGINNING OF THE PROGRAM
14220 **** @VOLT - THIS IS THE HPIB ADDRESS OF THE HP3457A VOLTOMETER. THIS IS ASSIGNED AT THE BEGINNING OF THE PROGRAM
14230 **** Initial_steps - THIS IS THE TOTAL NUMBER OF STEPS
108
THAT THE MOTOR HAS ROTATED SINCE THE MANUAL POSITIONING OF THE PRISM.

AFTER THE FINAL PRESSURE IS DETERMINED BY THE USER THIS NUMBER IS USED AS A BASE REFERENCE FOR THE AUTOMATED MEASUREMENTS.
14960!
14970 !********************************************************************
14980 ** THE NEXT 4 LINES PRINTS AN OPERATING INSTRUCTION ON THE SCREEN **
14990 !********************************************************************
15000!
15010 MOVE 10,30
15020 LABEL "PRESS RETURN WHEN MANUAL ADJUSTMENT IS COMPLETE"
15030 MOVE 10,35
15040 LABEL "USE LEFT/RIGHT ARROWS TO MOVE MOTOR (Left=Up, Right=Down)"
15050 MOVE 10,40
15060 LABEL "USE UP/DOWN ARROWS TO SCROLL MENU"
15070!
15080 !********************************************************************
15090 ** WHEN A KEY IS STRUCK GO TO THE MOVE CURSOR ROUTINE **
15100 !********************************************************************
15110!
15120 Diddle_loop: ON KBD GOTO Movement
15130 PEN -1
15140 MOVE 10,20
15150 LABEL
15160 LABEL "";VAL$(Pressure2*10)&" Lbs"
15170 PEN 1
15180 ENTER @Voltmeter;Pressure2
15190 MOVE 10,20
15200 LABEL ""
15210 LABEL "";VAL$(Pressure2*10)&" Lbs"
15220 GOTO 15130
15230!
15240 !********************************************************************
15250 ** MOVE CURSOR ROUTINE **
15260 !********************************************************************
15270!
15280 !********************************************************************
15290 ** THIS ROUTINE ONLY TAKES ACTION WHEN THE UP,DOWN OR **
15300 ** RETURN KEY IS PRESSED. GO TO Up ON UP KEY, GO TO **
15310 ** Down ON DOWN KEY, GO TO Take_data ON RETURN **
15320 !********************************************************************
15330!
15340 Movement::!
15350 Where$=KBDS
15360 OFF KBDS
15370 IF Where$=CHR$(255)&"^" THEN Up
15380 IF Where$=CHR$(255)&"V" THEN Down
15390 IF Where$=CHR$(255)&"<" THEN Move_arm_up
15400 IF Where$=CHR$(255)&">" THEN Move_arm_down
15410 IF Where$=CHR$(255)&"E" THEN Finished
15420 GOTO Diddle_loop
15430!
15440 !********************************************************************
15450 ** END OF MOVE CURSOR ROUTINE **
15460 !********************************************************************
15470!
15480 !********************************************************************
15490 ** UP MOVEMENT ROUTINE **
15500 !********************************************************************
15510!
15520 !********************************************************************
15530 ** THIS SCROLLS THE CURSOR UP **
15540 !**********************************************************************
15550 :
15560 Up:!
15570 PEN -1
15580 MOVE 5,95-5*I
15590 LABEL Cursor$
15600 IF I=0 THEN
15610     I=Num_of_choices-1
15620 ELSE
15630     I=I-1
15640 END IF
15650 PEN 1
15660 MOVE 5,95-5*I
15670 LABEL Cursor$
15680 GOTO Diddle_loop
15690 :
15700 !**********************************************************************
15710 !** END OF UP MOVEMENT ROUTINE **
15720 !**********************************************************************
15730 :
15740 !**********************************************************************
15750 !** DOWN MOVEMENT ROUTINE **
15760 !**********************************************************************
15770 :
15780 !**********************************************************************
15790 !** THIS SCROLLS THE CURSOR DOWN **
15800 !**********************************************************************
15810 :
15820 Down:!
15830 PEN -1
15840 MOVE 5,95-5*I
15850 LABEL Cursor$
15860 IF I=Num_of_choices-1 THEN
15870     I=0
15880 ELSE
15890     I=I+1
15900 END IF
15910 PEN 1
15920 MOVE 5,95-5*I
15930 LABEL Cursor$
15940 GOTO Diddle_loop
15950 :
15960 !**********************************************************************
15970 !** END OF DOWN MOVEMENT ROUTINE **
15980 !**********************************************************************
15990 :
16000 !**********************************************************************
16010 !** BEGIN STAGE MOVEMENT ROUTINES **
16020 !**********************************************************************
16030 :
16040 !**********************************************************************
16050 !** THIS SUBROUTINE MOVES THE MEASUREMENT ARM UP **
16060 !** BY THE NUM. INDICATED BY THE CURRENT VALUE OF **
16070 !** Steps$(I). **
16080 !** THE COMPUTER THEN POLLS THE VOLTMETER TO **
16090 !** DETERMINE THE NEW PRESSURE ON THE PRISM AND **
16100 !** PRINTS IT ON THE SCREEN **
16110 !**********************************************************************
Move_arm_up:

IF VAL(Steps$(I)) > Initial_steps THEN
  OUTPUT @Multi; "WF, 8.1, 0, 8.2, 5000T, OP, 8, "& VAL$(Initial_steps) & "T"
  WAIT .005*Initia_steps! WAIT UNTIL FINISHED
ELSE
  OUTPUT @Multi; "WF, 8.1, 0, 8.2, 5000T, OP, 8, "& Steps$(I) & "T"
  WAIT .005*VAL(Steps$(I)) ! WAIT UNTIL FINISHED
END IF

****************************************************************
** CLEAR THE OLD NUMBER OF STEPS FROM THE SCREEN THEN PRINT **
** THE NEW NUMBER OF STEPS AND CURRENT PRESSURE ON THE SCREEN **
****************************************************************

MOVE 10, 20
PEN -1
LABEL "; Initial_steps
LABEL "; VAL$(Pressure2*10) & " Lbs"
ENTER @ Voltmeter; Pressure2 ! GET NEW PRESSURE

*******************************************************************************
** CALCULATE NEW TOTAL STEPS AFTER MOVE **
*******************************************************************************

IF VAL(Steps$(I)) > Initial_steps THEN
  Initial_steps = 0
ELSE
  Initial_steps = Initial_steps - VAL(Steps$(I))
END IF

*******************************************************************************
** LABEL NEW VALUES ON THE SCREEN **
*******************************************************************************

Move_arm_down:

OUTPUT @ Multi; "WF, 8.1, 0, 8.2, 5000T, OP, 8, "& Steps$(I) & "T"
WAIT .005*VAL(Steps$(I)) ! WAIT UNTIL FINISHED
!** CLEAR THE OLD NUMBER OF STEPS FROM THE SCREEN THEN PRINT **
!** THE NEW NUMBER OF STEPS AND CURRENT PRESSURE ON THE SCREEN **
!***********************************************************************
16690 MOV 10,20
16700 PEN -1
16710 LABEL " ;Initial_steps
16720 LABEL " ;VAL$(Pressure2*10)&" Lbs"
16730 ENTER @Voltmeter;Pressure2 !GET NEW PRESSURE
16740 Initial_steps=Initial_steps+VAL(Steps$(I))
16750 MOV 10,20
16760 PEN 1
16770 LABEL " ;Initial_steps
16780 LABEL " ;VAL$(Pressure2*10)&" Lbs"
16790 GOTO Diddle_loop
16800 !
16810 !***********************************************************************
16820 !** END ARM MOVEMENT ROUTINES **
16830 !***********************************************************************
16840 !
16850 !***********************************************************************
16860 !** IF THE RETURN KEY IS PRESSED THE SUBPROGRAM TERMINATES **
16870 !** AND RETURNS THE LAST PRESSURE READING TO THE MAIN PROGRAM **
16880 !***********************************************************************
16890 !
16900 !***********************************************************************
16910 !** READ PRISM PRESSURE FOR USE IN MAIN PROGRAM **
16920 !** MOVE PRISM BACK TO HOME POSITION **
16930 !***********************************************************************
16940 !
16950 !***********************************************************************
16960 !
16970 !***********************************************************************
16980 !
16990 Finished:!!
17000 ENTER @Voltmeter;Pressure2 !GET NEW PRESSURE
17010 OUTPUT @Multi;"WF,8.1,0,8.2,5000T,OP,8,-"&VAL$(Initial_steps)&"T"
17020 WAIT .005*Initial_steps
17030 CLEAR SCREEN
17040 SUREND