Report No. MM-64-9
(Dept. 870)

DOCUMENTATION
OF
DIGITAL COMPUTER PROGRAMS
FOR
LUNAR LANDING
DYNAMICS SYSTEM INVESTIGATION

BELLCOMM SUBCONTRACT NO. 10002 WITH
BENDIX PRODUCTS AEROSPACE DIVISION

November 1964

Analytical Mechanics Department
Bendix Products Aerospace Division
The Bendix Corporation
South Bend, Indiana 46620
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>REFERENCES</td>
<td>vii</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>ix</td>
</tr>
<tr>
<td>I LANDING DYNAMICS COMPUTER PROGRAM</td>
<td>1-1</td>
</tr>
<tr>
<td>II DIGITAL COMPUTER PROGRAM FOR THE APPROXIMATE DETERMINATION OF NOZZLE CLEARANCE</td>
<td>2-1</td>
</tr>
<tr>
<td>A Digital Computer Program for the Approximate Determination of Nozzle Clearance</td>
<td>2-1</td>
</tr>
<tr>
<td>III WEIGHT ANALYSIS DIGITAL COMPUTER PROGRAM</td>
<td>3-1</td>
</tr>
<tr>
<td>IV DIGITAL COMPUTER PROGRAMS FOR THE CONDENSED ANALYSIS OF REAR LEG TOUCHDOWN AND FREE FLIGHT</td>
<td>4-1</td>
</tr>
<tr>
<td>Description</td>
<td>4-1</td>
</tr>
<tr>
<td>Computer Input Data</td>
<td>4-1</td>
</tr>
<tr>
<td>Computer Output Data</td>
<td>4-2</td>
</tr>
<tr>
<td>V PROGRAM FOR COMPUTER SOLUTION OF EQUATIONS FROM CONDENSED ANALYSIS OF FRONT LEG IMPACT</td>
<td>5-1</td>
</tr>
<tr>
<td>VI LANDING DYNAMICS COMPUTER PROGRAM FOR NON-PLANAR LANDINGS WITH INFINITE GROUND COEFFICIENT OF FRICTION AND ZERO FOOTPAD MASS</td>
<td>6-1</td>
</tr>
<tr>
<td>Theoretical Analysis of Non-Planar Motion</td>
<td>6-1</td>
</tr>
<tr>
<td>Discussion of Mathematical Model</td>
<td>6-1</td>
</tr>
<tr>
<td>Coordinate Transformations</td>
<td>6-7</td>
</tr>
<tr>
<td>Initial Orientation of the Vehicle</td>
<td>6-8</td>
</tr>
<tr>
<td>Description of Subroutine INTEQM</td>
<td>6-9</td>
</tr>
<tr>
<td>Description of Subroutine FORCE</td>
<td>6-13</td>
</tr>
<tr>
<td>Stability Determination</td>
<td>6-14</td>
</tr>
<tr>
<td>VII GENERAL NON-PLANAR LUNAR LANDING COMPUTER PROGRAM</td>
<td>7-1</td>
</tr>
<tr>
<td>Introduction</td>
<td>7-1</td>
</tr>
<tr>
<td>Method</td>
<td>7-1</td>
</tr>
<tr>
<td>Nozzle Clearance</td>
<td>7-4</td>
</tr>
<tr>
<td>Stability Check</td>
<td>7-4</td>
</tr>
<tr>
<td>Conclusions</td>
<td>7-6</td>
</tr>
</tbody>
</table>
LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-1</td>
<td>Landing Dynamics Computer Program Flow Diagram</td>
<td>1-4</td>
</tr>
<tr>
<td>1-2</td>
<td>Strut Force Vs. Strut Length</td>
<td>1-5</td>
</tr>
<tr>
<td>1-3</td>
<td>Input Data Format</td>
<td>1-8</td>
</tr>
<tr>
<td>1-4</td>
<td>Main Lunar Landing Dynamics Program</td>
<td>1-10</td>
</tr>
<tr>
<td>1-5</td>
<td>Subroutine INPUT1</td>
<td>1-17</td>
</tr>
<tr>
<td>1-5A</td>
<td>Subroutine INCON</td>
<td>1-18</td>
</tr>
<tr>
<td>1-6</td>
<td>Subroutine INITSE</td>
<td>1-19</td>
</tr>
<tr>
<td>1-7</td>
<td>Subroutine INITRU</td>
<td>1-20</td>
</tr>
<tr>
<td>1-8</td>
<td>Subroutine STRANG</td>
<td>1-22</td>
</tr>
<tr>
<td>1-9</td>
<td>Subroutine STGEOM</td>
<td>1-23</td>
</tr>
<tr>
<td>1-10</td>
<td>Subroutine FOTPAD</td>
<td>1-24</td>
</tr>
<tr>
<td>1-11</td>
<td>Subroutine VEHFOR</td>
<td>1-26</td>
</tr>
<tr>
<td>1-12</td>
<td>Subroutine VEHMOT</td>
<td>1-27</td>
</tr>
<tr>
<td>1-13</td>
<td>Subroutine STABAN</td>
<td>1-28</td>
</tr>
<tr>
<td>1-14</td>
<td>Subroutine STOREP</td>
<td>1-29</td>
</tr>
<tr>
<td>1-15</td>
<td>Subroutine PRINT1</td>
<td>1-30</td>
</tr>
<tr>
<td>1-16</td>
<td>Subroutine ENERGY</td>
<td>1-33</td>
</tr>
<tr>
<td>1-17</td>
<td>Subroutine PROFIL</td>
<td>1-34</td>
</tr>
<tr>
<td>1-18</td>
<td>Main Program</td>
<td>1-36</td>
</tr>
<tr>
<td>1-19</td>
<td>Subroutine INPUT1</td>
<td>1-45</td>
</tr>
<tr>
<td>1-20</td>
<td>Subroutine INCON</td>
<td>1-48</td>
</tr>
<tr>
<td>1-21</td>
<td>Subroutine INITSE</td>
<td>1-50</td>
</tr>
<tr>
<td>1-22</td>
<td>Subroutine INITRU</td>
<td>1-52</td>
</tr>
<tr>
<td>1-23</td>
<td>Subroutine STRANG</td>
<td>1-59</td>
</tr>
<tr>
<td>1-24</td>
<td>Subroutine STGEOM</td>
<td>1-62</td>
</tr>
<tr>
<td>1-25</td>
<td>Subroutine FOTPAD</td>
<td>1-65</td>
</tr>
<tr>
<td>1-26</td>
<td>Subroutine VEHFOR</td>
<td>1-72</td>
</tr>
<tr>
<td>1-27</td>
<td>Subroutine VEHMOT</td>
<td>1-74</td>
</tr>
<tr>
<td>1-28</td>
<td>Subroutine STABAN</td>
<td>1-77</td>
</tr>
<tr>
<td>1-29</td>
<td>Subroutine STOREP</td>
<td>1-79</td>
</tr>
<tr>
<td>1-30</td>
<td>Subroutine PRINT1</td>
<td>1-82</td>
</tr>
<tr>
<td>1-31</td>
<td>Subroutine ENERGY</td>
<td>1-91</td>
</tr>
<tr>
<td>1-31A</td>
<td>Subroutine PROFIL</td>
<td>1-93</td>
</tr>
<tr>
<td>1-32</td>
<td>Data Printout and &quot;On-Line&quot; Printout</td>
<td>1-95</td>
</tr>
<tr>
<td>1-33</td>
<td>Summary Data Printout</td>
<td>1-98</td>
</tr>
<tr>
<td>1-34</td>
<td>Final Data Printout (Part 1)</td>
<td>1-99</td>
</tr>
<tr>
<td>1-34</td>
<td>Final Data Printout (Part 2)</td>
<td>1-101</td>
</tr>
<tr>
<td>1-34</td>
<td>Final Data Printout (Part 3)</td>
<td>1-103</td>
</tr>
</tbody>
</table>

2-1    | Geometric Computer Program                                                  | 2-2  |
2-2    | Geometry Program                                                           | 2-5  |
# LIST OF ILLUSTRATIONS (CONT.)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-3</td>
<td>Geometry Program</td>
<td>2-6</td>
</tr>
<tr>
<td>2-4</td>
<td>Subroutine to Geometry Program</td>
<td>2-8</td>
</tr>
<tr>
<td>2-5</td>
<td>Subroutine to Geometry Program</td>
<td>2-9</td>
</tr>
<tr>
<td>2-6</td>
<td>Subroutine to Geometry Program</td>
<td>2-10</td>
</tr>
<tr>
<td>2-7</td>
<td>Nozzle Clearance Program</td>
<td>2-11</td>
</tr>
<tr>
<td>2-8</td>
<td>Subroutine INPUT</td>
<td>2-15</td>
</tr>
<tr>
<td>2-9</td>
<td>Subroutine GEOM</td>
<td>2-17</td>
</tr>
<tr>
<td>2-10</td>
<td>Subroutine INTER</td>
<td>2-20</td>
</tr>
<tr>
<td>2-11</td>
<td>Output Data for Mode IFLAG</td>
<td>2-22</td>
</tr>
<tr>
<td>2-12</td>
<td>Output Data for Mode IFLAG +1</td>
<td>2-23</td>
</tr>
<tr>
<td>3-1</td>
<td>Input Data Format</td>
<td>3-3</td>
</tr>
<tr>
<td>3-2</td>
<td>Computer Program: Design of Legs to Minimize Weight</td>
<td>3-4</td>
</tr>
<tr>
<td>3-3</td>
<td>Subroutine INPUT</td>
<td>3-6</td>
</tr>
<tr>
<td>3-4</td>
<td>Subroutine GEDM</td>
<td>3-7</td>
</tr>
<tr>
<td>3-5</td>
<td>Subroutine CURVE</td>
<td>3-8</td>
</tr>
<tr>
<td>3-6</td>
<td>Subroutine VECTOR</td>
<td>3-11</td>
</tr>
<tr>
<td>3-7</td>
<td>Subroutine EQBCD</td>
<td>3-12</td>
</tr>
<tr>
<td>3-8</td>
<td>Subroutine EQE</td>
<td>3-13</td>
</tr>
<tr>
<td>3-9</td>
<td>Main Program</td>
<td>3-14</td>
</tr>
<tr>
<td>3-10</td>
<td>Subroutine INPUT</td>
<td>3-24</td>
</tr>
<tr>
<td>3-11</td>
<td>Subroutine GEOM</td>
<td>3-26</td>
</tr>
<tr>
<td>3-12</td>
<td>Subroutine CURVE</td>
<td>3-29</td>
</tr>
<tr>
<td>3-13</td>
<td>Subroutine VECTOR</td>
<td>3-33</td>
</tr>
<tr>
<td>3-14</td>
<td>Subroutine EQBCD</td>
<td>3-35</td>
</tr>
<tr>
<td>3-15</td>
<td>Subroutine EQE</td>
<td>3-36</td>
</tr>
<tr>
<td>3-16</td>
<td>Sample Output - Case A</td>
<td>3-38</td>
</tr>
<tr>
<td>3-17</td>
<td>Sample Output - Case B</td>
<td>3-39</td>
</tr>
<tr>
<td>3-18</td>
<td>Sample Output - Case C</td>
<td>3-41</td>
</tr>
<tr>
<td>3-19</td>
<td>Sample Output - Case D</td>
<td>3-43</td>
</tr>
<tr>
<td>3-20</td>
<td>Sample Output - Case E</td>
<td>3-45</td>
</tr>
<tr>
<td>4-1</td>
<td>Input Data Format</td>
<td>4-4</td>
</tr>
<tr>
<td>4-2</td>
<td>Computer Program - Condensed Analysis for Downhill Landing Dynamics</td>
<td>4-5</td>
</tr>
<tr>
<td>4-3</td>
<td>Subroutine INCONT</td>
<td>4-7</td>
</tr>
<tr>
<td>4-4</td>
<td>Subroutine FFIT</td>
<td>4-8</td>
</tr>
<tr>
<td>4-5</td>
<td>Subroutine FTHREE</td>
<td>4-11</td>
</tr>
<tr>
<td>4-6</td>
<td>Subroutine N30N44</td>
<td>4-12</td>
</tr>
<tr>
<td>4-7</td>
<td>Main Program</td>
<td>4-13</td>
</tr>
</tbody>
</table>

---

iii
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-8</td>
<td>Subroutine INCONT</td>
<td>4-18</td>
</tr>
<tr>
<td>4-9</td>
<td>Subroutine FFIT</td>
<td>4-20</td>
</tr>
<tr>
<td>4-10</td>
<td>Subroutine FTHREE</td>
<td>4-23</td>
</tr>
<tr>
<td>4-11</td>
<td>Subroutine N30N44</td>
<td>4-25</td>
</tr>
<tr>
<td>4-12</td>
<td>Sample Output</td>
<td>4-28</td>
</tr>
<tr>
<td>5-1</td>
<td>Sample Input Data</td>
<td>5-6</td>
</tr>
<tr>
<td>5-2</td>
<td>Flow Diagram of Main Program</td>
<td>5-7</td>
</tr>
<tr>
<td>5-3</td>
<td>Flow Diagram of Subroutine INCOND</td>
<td>5-10</td>
</tr>
<tr>
<td>5-4</td>
<td>Flow Diagram of Subroutine GUESS</td>
<td>5-10</td>
</tr>
<tr>
<td>5-5</td>
<td>Flow Diagram of Subroutine FEVAL</td>
<td>5-11</td>
</tr>
<tr>
<td>5-6</td>
<td>Flow Diagram of Subroutine SOLN</td>
<td>5-12</td>
</tr>
<tr>
<td>5-7</td>
<td>Flow Diagram of Subroutine DEVAL</td>
<td>5-13</td>
</tr>
<tr>
<td>5-8</td>
<td>Main Program Listing</td>
<td>5-14</td>
</tr>
<tr>
<td>5-9</td>
<td>Subroutine INCOND</td>
<td>5-19</td>
</tr>
<tr>
<td>5-10</td>
<td>Subroutine GUESS</td>
<td>5-21</td>
</tr>
<tr>
<td>5-11</td>
<td>Subroutine FEVAL</td>
<td>5-23</td>
</tr>
<tr>
<td>5-12</td>
<td>Subroutine SOLN</td>
<td>5-25</td>
</tr>
<tr>
<td>5-13</td>
<td>Subroutine DEVAL</td>
<td>5-27</td>
</tr>
<tr>
<td>5-14</td>
<td>Output Data</td>
<td>5-30</td>
</tr>
<tr>
<td>6-1</td>
<td>Moving (Vehicle) Coordinate System and Leg Numbering System</td>
<td>6-2</td>
</tr>
<tr>
<td>6-2</td>
<td>Fixed (Ground) Coordinate System and Output Variables</td>
<td>6-3</td>
</tr>
<tr>
<td>6-3</td>
<td>Nomenclature for Stability Determination</td>
<td>6-15</td>
</tr>
<tr>
<td>6-4</td>
<td>Input Data Format</td>
<td>6-18</td>
</tr>
<tr>
<td>6-5</td>
<td>Three Dimensional Landing Dynamics</td>
<td>6-19</td>
</tr>
<tr>
<td>6-6</td>
<td>Subroutine INPUT</td>
<td>6-23</td>
</tr>
<tr>
<td>6-7</td>
<td>Subroutine INIT</td>
<td>6-24</td>
</tr>
<tr>
<td>6-8</td>
<td>Subroutine LEGTYP</td>
<td>6-25</td>
</tr>
<tr>
<td>6-9</td>
<td>Subroutine STAB</td>
<td>6-26</td>
</tr>
<tr>
<td>6-10</td>
<td>Subroutine DATA</td>
<td>6-27</td>
</tr>
<tr>
<td>6-11</td>
<td>Subroutine CONFIG</td>
<td>6-28</td>
</tr>
<tr>
<td>6-12</td>
<td>Subroutine FORCE</td>
<td>6-30</td>
</tr>
<tr>
<td>6-13</td>
<td>Subroutine INTEQM</td>
<td>6-32</td>
</tr>
<tr>
<td>6-14</td>
<td>Main Program</td>
<td>6-34</td>
</tr>
<tr>
<td>6-15</td>
<td>Subroutine INPUT</td>
<td>6-41</td>
</tr>
<tr>
<td>6-16</td>
<td>Subroutine INIT</td>
<td>6-44</td>
</tr>
<tr>
<td>6-17</td>
<td>Subroutine STAB</td>
<td>6-49</td>
</tr>
<tr>
<td>6-18</td>
<td>Subroutine DATA</td>
<td>6-51</td>
</tr>
<tr>
<td>6-19</td>
<td>Subroutine LEGTYP</td>
<td>6-55</td>
</tr>
<tr>
<td>6-20</td>
<td>Subroutine CONFIG</td>
<td>6-57</td>
</tr>
<tr>
<td>6-21</td>
<td>Subroutine FORCE</td>
<td>6-62</td>
</tr>
</tbody>
</table>
LIST OF ILLUSTRATIONS (CONT.)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-22</td>
<td>Subroutine INTEQM</td>
<td>6-65</td>
</tr>
<tr>
<td>6-23</td>
<td>Input Data Printout</td>
<td>6-69</td>
</tr>
<tr>
<td>6-24</td>
<td>&quot;On Line&quot; Data Printout</td>
<td>6-70</td>
</tr>
<tr>
<td>6-25</td>
<td>Final Printout of Stored Data</td>
<td>6-71</td>
</tr>
<tr>
<td>7-1</td>
<td>Initial Vehicle Orientation</td>
<td>7-2</td>
</tr>
<tr>
<td>7-2</td>
<td>Input Data Form</td>
<td>7-8</td>
</tr>
<tr>
<td>7-3</td>
<td>Main Computer Program</td>
<td>7-10</td>
</tr>
<tr>
<td>7-4</td>
<td>Subroutine INCON</td>
<td>7-21</td>
</tr>
<tr>
<td>7-5</td>
<td>Subroutine STABAN</td>
<td>7-22</td>
</tr>
<tr>
<td>7-6</td>
<td>Subroutine INTEQM</td>
<td>7-23</td>
</tr>
<tr>
<td>7-7</td>
<td>Main Program</td>
<td>7-24</td>
</tr>
<tr>
<td>7-8</td>
<td>Subroutine INCON</td>
<td>7-48</td>
</tr>
<tr>
<td>7-9</td>
<td>Subroutine STABAN</td>
<td>7-50</td>
</tr>
<tr>
<td>7-10</td>
<td>Subroutine INTEQM</td>
<td>7-52</td>
</tr>
<tr>
<td>7-11</td>
<td>Input Data Printout</td>
<td>7-54</td>
</tr>
<tr>
<td>7-12</td>
<td>Calculated Input Values</td>
<td>7-55</td>
</tr>
<tr>
<td>7-13</td>
<td>Output Data (Part 1)</td>
<td>7-56</td>
</tr>
<tr>
<td>7-14</td>
<td>Summary Output Data</td>
<td>7-58</td>
</tr>
<tr>
<td>7-15</td>
<td>Output Data (Part 2)</td>
<td>7-59</td>
</tr>
<tr>
<td>7-16</td>
<td>Output Data (Part 3)</td>
<td>7-61</td>
</tr>
<tr>
<td>7-17</td>
<td>Output Data (Part 4)</td>
<td>7-64</td>
</tr>
<tr>
<td>7-18</td>
<td>Output Data (Part 5)</td>
<td>7-65</td>
</tr>
</tbody>
</table>

LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-1</td>
<td>Summary of Important Parameters Used In Condensed Analysis of Front Leg Impact</td>
<td>5-2</td>
</tr>
</tbody>
</table>
REFERENCES

INTRODUCTION

This report supplements the Final Report on Lunar Landing Dynamics Systems Investigation (Reference 1) and provides detailed information on the Digital Computer Programs involved in that study.

The information supplied herein should be sufficient to permit the use of any of the computer programs. All the programs were written in Fortran IV and will execute on a Univac 1107 Digital Computer.

This report includes, in addition to program flow diagrams and listings, a sample run for checkout purposes. This sample run may be used as an aid to the checkout of the programs on any other digital computer. The sample runs were performed using single precision on the Univac 1107 computer. Minor differences in computer results may be experienced when using these programs on other computers because of the differences in precision which may exist.

These programs have been checked out to the extent discussed in the particular section pertaining to that program and in Reference 1.
SECTION I

LANDING DYNAMICS COMPUTER PROGRAM

This program was written to study the detailed vehicle motions from the instant of touch-down until the vehicle comes to rest. The program will handle only planar motion although footpad motions and positions are determined in three dimensions. The general program flow is illustrated in Figure 1-1.

The basic mathematical approach used in this program is as follows:

1. Using the time history of the forces and torques acting on the vehicle center of gravity at the start of an interval $\Delta T$, determine the c.g. motions during and at the end of the time interval.

2. Using the time history of the forces acting on the footpad masses at the start of the time interval, determine individual footpad motions during and at the end of the time interval.

3. Knowing the positions of the vehicle c.g. and all footpads at the end of a time interval $\Delta T$, determine the lengths of all the landing gear struts and, knowing the strut lengths, determine the stroking forces developed by each individual strut.

4. Having determined the strut forces existing at the end of the time interval, bring the time history of the forces up to date and repeat the procedure for the next time interval.

A parabolic type of integration was used for the determination of both basic vehicle motion and footpad motion.

In general form, the force during the time interval $t$ to $t + \Delta t$ can be expressed in the form

$$F(t) = f + a t + b t^2$$

let

$$f = \text{general force at time } t$$

$$f_{-1} = \text{general force at time } t - \Delta t$$
\( f_{-2} \) = general force at time \( t - 2\Delta t \)

then from (1)

\[
F(0) = f
\]

\[
F(-\Delta t) = f_{-1} = f + a(-\Delta t) + b(-\Delta t)^2
\]

\( = f - a\Delta t + b\Delta t^2 \) \( \quad (2) \)

\[
F(-2\Delta t) = f_{-2} = f + a(-2\Delta t) + b(-2\Delta t)^2
\]

\( = f - 2a\Delta t + 4b\Delta t^2 \) \( \quad (3) \)

Solving equations (2) and (3) simultaneously for \( b \) gives:

\[
b = \frac{f - 2f_{-1} + f_{-2}}{2\Delta t^2} \quad (4)
\]

and solving (2) and (3) for \( a \) gives:

\[
a = \frac{3f - 4f_{-1} + f_{-2}}{2\Delta t} \quad (5)
\]

The general form for the equation of motion is:

\[
M\ddot{q} = F(t)
\]

or by substitution of equation (1)

\[
\dddot{q} = \frac{1}{M} \left\{ F(t) \right\} = \frac{1}{M} \left\{ f + a t + b t^2 \right\} \quad (6)
\]

Integrating to obtain velocity

\[
\dot{q} = \frac{1}{M} \left\{ f \Delta t + \frac{a \Delta t^2}{2} + \frac{b \Delta t^3}{3} \right\} + \dot{q}_0 \quad (7)
\]

which, upon substitution for \( a \) and \( b \) from equations (4) and (5) gives

\[
\dot{q} = \frac{\Delta t}{12M} \left\{ 23f - 16f_{-1} + 5f_{-2} \right\} + \dot{q}_0 \quad (8)
\]

Integrating equation (7) to obtain displacement

\[
q = \frac{1}{M} \left\{ \frac{f \Delta t^2}{2} + \frac{a \Delta t^3}{6} + \frac{b \Delta t^4}{12} \right\} + \dot{q}_0 \Delta t + q_0 \quad (9)
\]

1-2
and substituting for \( a \) and \( b \) from equations (4) and (5) gives

\[
q(t) = \frac{\Delta t^2}{24M} \left\{ 19f - 10f_{-1} + 3f_{-2} \right\} + \dot{q}_0 \Delta t + q_0
\]

(10)

A comparison of this force representation with simple rectangular integration shows the similarity

\[
\dot{q} = \dot{q}_0 + \frac{\Delta t}{M} f \quad \text{rectangular}
\]

\[
\ddot{q} = \ddot{q}_0 + \frac{\Delta t}{12M} \left\{ 23f - 16f_{-1} + 5f_{-2} \right\} \quad \text{parabolic}
\]

and

\[
q(t) = q_0 + \dot{q}_0 \Delta t + \frac{\Delta t^2}{2M} f \quad \text{rectangular}
\]

\[
q(t) = q_0 + \dot{q}_0 \Delta t + \frac{\Delta t^2}{24M} \left\{ 19f - 10f_{-1} + 3f_{-2} \right\} \quad \text{parabolic}
\]

Using this representation of the forces existent during the time interval \( t \) to \( t + \Delta t \) gives a much better approximation to the closed form solution than could be obtained using rectangular integration.

These equations are used for determining the motions of both the main vehicle mass and the footpad masses.

Referring to Figure 1-1, the general flow through the computer program is as follows:

1. Input data is read in. The initial detailed vehicle geometry is determined and the input data printed for future reference.

2. The initial strut lengths of all the struts is determined from the input data.

3. The vehicle orientation relative to the ground surface is determined.

4. A test is performed to determine which footpads are in contact with the ground surface. If any footpads are not in contact with the ground, a further test is performed to determine if the footpad is moving relative to the vehicle. The strut force vs. strut length is illustrated in Figure 1-2. As is shown, the strut may develop an elastic force prior to actual plastic crushing of the honeycomb. Thus, the footpads may be off the ground but still possess both kinetic and potential energy relative to the vehicle itself. If this total energy is less than ten percent of the maximum possible potential energy due to the elasticity of the strut, the footpad is assumed to be fixed to the vehicle and its motions described by rigid body equations.
BENDIX COMPUTER PROGRAM FOR LUNAR LANDING STUDY

Read Input Data
Initialize Program
Print Program Checks

Determine Strut Lengths
Find Vehicle Orientation

---

YES

Is Footpad (i) On The Ground?

NO

Determine Strut Forces
Determine Footpad Motion And Position After Time \( \Delta T \)

NO

Is Footpad (i) Moving With Vehicle?

YES

Iterate N Times

Determine Forces And Torques Acting On Vehicle
Determine Vehicle Motion And Position After Time \( \Delta T \)

Set Parameters For Next Calculation

---

YES

Test If Vehicle Is Stopped

NO

Print Output Data
Prepare For Next Landing

Continued Program For Next \( \Delta T \)

Figure 1-1. Landing Dynamics Computer Program Flow Diagram
If any footpads are on the ground or moving relative to the vehicle, the forces developed in the landing gear struts are determined using the force profile illustrated in Figure 1-2. The computer program is sufficiently general so that all parameters of stroke force (i.e. friction, spring rates, crush stroke force, etc.) may be varied from one strut to another as required.

The strut forces acting on each footpad and gravity, ground reaction forces and ground friction plus the time history of these forces are considered in determining the individual footpad motions. Since the footpad masses are usually small and the forces large, it is necessary to use a smaller time increment in integrating footpad motions. For this reason, an iteration loop is used at this step in the program so that time intervals of $\Delta T/N$ may be used. In practice, it has been determined that three iterations of footpad motion ($N = 3$) are sufficient to produce accurate results.

Using the strut forces determined in step 5, it is possible to define all forces acting on the vehicle c.g. Since geometry is known, torques can also be determined. Again, using these forces plus their time history, the c.g. motions can be determined during and at the end of the time interval $\Delta T$. Parabolic integration is used for vehicle motion.

Next, all parameters are set for the start of the next time interval and tests are performed to determine if the vehicle is stopped. Here, all velocities $\dot{x}$, $\dot{y}$ and rotational velocity $\psi$ must be within epsilon's of zero. In addition,
all footpads must be either in contact with the ground or within a distance epsilon of the ground.

9. If all these tests cannot be met simultaneously, the vehicle is still moving and the program is repeated for another iteration.

It has been determined that a time increment $\Delta T = 0.002$ seconds is sufficiently small to adequately describe the vehicle motions for reasonable values of all the parameters. If very small footpad masses (less than 0.5% of total vehicle mass) are used, a smaller time interval would be required. Since, depending on the accuracy of the particular computer used, smaller time increments may result in excessive computer round-off error; this is not recommended for the program described here.

Refer to Figure 7-1 for system nomenclature used in this program.

Figure 1-3 illustrates the input data format required for this program. The input variables are defined under "Input Definitions" in Figure 1-18.

When this program is used to define a complete stability profile as discussed in Reference 1, the following programming procedure should be followed.

In order to run a complete stability profile, an array of X and Y velocities is used. Both XVEL$\phi$ and YVEL$\phi$ are doubly subscripted variables where the subscripts NQ and NS represent the column and row of the velocity array. This velocity array must be rectangular (the same number of values in each row and the same number in each column). The starting point in the array is defined by setting the input parameters NS$\phi$ and NQ$\phi$. This permits starting anywhere in the velocity array and not just at (1, 1).

The computer program is designed to develop a stability profile in as few runs as possible. The following table illustrates a typical sequence of runs which the program will follow automatically in defining the stability profile. As is indicated, the choice of the velocities for succeeding runs is dependent upon the stability (or instability) of the preceding runs.

<table>
<thead>
<tr>
<th>NS</th>
<th>NQ</th>
<th>Stability</th>
<th>Program Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>Stable</td>
<td>Program sets NQ = 8</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>Unstable</td>
<td>Program sets NS = 2</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>Stable</td>
<td>Program sets NQ = 9</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>Unstable</td>
<td>Program sets NS = 3</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>Unstable</td>
<td>Program sets NQ = 8</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>Stable</td>
<td>Program sets NS = 4</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>Stable</td>
<td>Program sets NQ = 9</td>
</tr>
</tbody>
</table>

Figures 1-4 through 1-17 are flow diagrams for the Landing Dynamics Computer Program and its subroutines.
Figures 1-18 through 1-31 are complete program listings of the program and its sub-routines.

Figures 1-32 illustrates the "on-line" printout of a typical program run. The initial printout is a summary of the input data for identification purposes. The following pages contain printout of pertinent information during the run. Following completion of the run, two additional outputs are printed. These include Figure 1-33 which is a summary of pertinent information concerning vehicle stability. This includes:

Line 1 - Identification and summary of input conditions
Line 2 - Conversion of X and Y velocities to vertical and horizontal velocities
Line 3 - Problem running time
Lines 4 - 6 Maximum stroke of all struts.

Note - When symmetry is used, results for struts 2 and 4 are identical to those for struts 1 and 3 respectively but the printout indicates zeros.

Lines 7 - 9 Self explanatory

Lines 10 - 12 Energy balance. If little or no sliding occurs, the "energy based on vehicle velocities and C.G. drop" should be approximately equal to the "energy dissipated" (lines 11 and 12).

Line 13 - Angle between vehicle centerline and a normal to the ground surface
Line 14 - Final stability angle and its rate of change
Line 15 - Percentage of vehicle energy absorbed by each legset.

Figure 1-34 illustrates the printout of information stored during the program run and printed upon completion of the run.
<table>
<thead>
<tr>
<th>Statement Number</th>
<th>FORTRAN Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A(1)</td>
</tr>
<tr>
<td>2</td>
<td>ALPHA(1)</td>
</tr>
<tr>
<td>3</td>
<td>D(1)</td>
</tr>
<tr>
<td>4</td>
<td>D11(1)</td>
</tr>
<tr>
<td>5</td>
<td>F11(1)</td>
</tr>
<tr>
<td>6</td>
<td>F22(1)</td>
</tr>
<tr>
<td>7</td>
<td>F33(1)</td>
</tr>
<tr>
<td>8</td>
<td>GRDMU(1)</td>
</tr>
<tr>
<td>9</td>
<td>P1(1)</td>
</tr>
<tr>
<td>10</td>
<td>P2(1)</td>
</tr>
<tr>
<td>11</td>
<td>P3(1)</td>
</tr>
<tr>
<td>12</td>
<td>R1(1)</td>
</tr>
<tr>
<td>13</td>
<td>R2(1)</td>
</tr>
<tr>
<td>14</td>
<td>RP(1)</td>
</tr>
<tr>
<td>15</td>
<td>SKS(1)</td>
</tr>
<tr>
<td>16</td>
<td>THETA(1)</td>
</tr>
<tr>
<td>17</td>
<td>DELTAP</td>
</tr>
<tr>
<td>18</td>
<td>EPS3</td>
</tr>
<tr>
<td>19</td>
<td>FINT</td>
</tr>
<tr>
<td>20</td>
<td>PMASS</td>
</tr>
</tbody>
</table>

Figure 1-3. Input Data Format (Sheet 1 of 2)
<table>
<thead>
<tr>
<th>STATEMENT NUMBER</th>
<th>FORTRAN STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identification</td>
<td>(Cont.)</td>
</tr>
<tr>
<td>21</td>
<td>RMFI</td>
</tr>
<tr>
<td></td>
<td>RUNN</td>
</tr>
<tr>
<td></td>
<td>RN</td>
</tr>
<tr>
<td>22</td>
<td>SK1</td>
</tr>
<tr>
<td></td>
<td>SK2</td>
</tr>
<tr>
<td></td>
<td>SK3</td>
</tr>
<tr>
<td>23</td>
<td>SKE1</td>
</tr>
<tr>
<td></td>
<td>SKE2</td>
</tr>
<tr>
<td></td>
<td>SK3</td>
</tr>
<tr>
<td>24</td>
<td>SERN</td>
</tr>
<tr>
<td></td>
<td>VMASS</td>
</tr>
<tr>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>25</td>
<td>ZETA</td>
</tr>
<tr>
<td></td>
<td>H</td>
</tr>
<tr>
<td>26</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>JJ</td>
</tr>
<tr>
<td></td>
<td>J</td>
</tr>
<tr>
<td></td>
<td>K</td>
</tr>
<tr>
<td></td>
<td>KPRINT</td>
</tr>
<tr>
<td></td>
<td>LAND</td>
</tr>
<tr>
<td>27</td>
<td>N</td>
</tr>
<tr>
<td></td>
<td>NS</td>
</tr>
<tr>
<td></td>
<td>NQ</td>
</tr>
<tr>
<td></td>
<td>NSMAX</td>
</tr>
<tr>
<td></td>
<td>NQMAX</td>
</tr>
<tr>
<td></td>
<td>KCONMX</td>
</tr>
<tr>
<td>28</td>
<td>INPLD</td>
</tr>
<tr>
<td></td>
<td>LTPRT</td>
</tr>
<tr>
<td>29</td>
<td>VVEL(1, 1)</td>
</tr>
<tr>
<td></td>
<td>HVEL(1, 1)</td>
</tr>
<tr>
<td></td>
<td>VVEL(2, 1)</td>
</tr>
<tr>
<td></td>
<td>HVEL(2, 1)</td>
</tr>
<tr>
<td></td>
<td>VVEL(3, 1)</td>
</tr>
<tr>
<td></td>
<td>HVEL(3, 1)</td>
</tr>
<tr>
<td>30</td>
<td>VVEL(NQMAX, 1)</td>
</tr>
<tr>
<td></td>
<td>HVEL(NQMAX, 1)</td>
</tr>
<tr>
<td></td>
<td>VVEL(1, 2)</td>
</tr>
<tr>
<td></td>
<td>HVEL(1, 2)</td>
</tr>
<tr>
<td></td>
<td>VVEL(2, 2)</td>
</tr>
<tr>
<td></td>
<td>HVEL(2, 2)</td>
</tr>
<tr>
<td>31</td>
<td>ETC</td>
</tr>
<tr>
<td></td>
<td>ETC</td>
</tr>
<tr>
<td></td>
<td>VVEL(NQMAX, 1)</td>
</tr>
<tr>
<td></td>
<td>HVEL(NQMAX, 1)</td>
</tr>
<tr>
<td></td>
<td>NSMAX</td>
</tr>
<tr>
<td></td>
<td>NSMAX</td>
</tr>
</tbody>
</table>

Figure 1-3. Input Data Format (Sheet 2 of 2)
This program determines detailed planar motions from touchdown to rest or instability.

Figure 1-4. Main Lunar Landing Dynamics Program
Figure 1-4. Main Lunar Landing Dynamics Program (Continued)
Figure 1-4. Main Lunar Landing Dynamics Program (Continued)
Figure 1-4. Main Lunar Landing Dynamics Program (Continued)
CALL SUBROUTINE ENERGY
Compute vehicle energy dissipated

DO 775 I = 1, 2
CALL SUBROUTINE PRINT1(4)
Print summary (Unstable)

CALL SUBROUTINE PRINT1(7)
Print general summary

CALL SUBROUTINE PRINT1(8)
Print stored output

IPRO = 1

CALL SUBROUTINE PROFIL(IPRO)
Choose new velocities from array for profile

GO TO (1, 20, 40), IPRO
Figure 1-4. Main Lunar Landing Dynamics Program (Continued)
Figure 1-4. Main Lunar Landing Dynamics Program (Concluded)
This subroutine reads input data from cards.

Figure 1-5. Subroutine-INPUT1
This subroutine converts the input data to a form that is used in the program.

Figure 1-5A. Subroutine-INCON
This subroutine initializes the program for a series of runs used in determining a stability profile.

Figure 1-6. Subroutine-INITSE
This subroutine initializes program for each run.

Figure 1-7. Subroutine-INITRU
INITRU-INITIALIZE FOR EACH RUN

1. Initialize stability angle functions
2. Initialize program for strut length calculations
3. Determine trigonometric relationships for strut length calculations
4. Determine initial strut lengths
5. Compute stability angles at touchdown

RETURN

Figure 1-7. Subroutine-INITRU(Concluded)
This subroutine determines new strut angles if footpad is on the ground.

**Figure 1-8. Subroutine-STRANG**

```
ENTRY

DO 240 I = 1, N, K
Is this footpad on ground?

YES - Determine new strut angles for leg

NO - 240 Continue

RETURN
```
This subroutine determines new strut geometry if footpad is on ground.

Figure 1-9. Subroutine-STGEOM
This subroutine determines strut forces and forces acting on footpads. It also obtains footpad position and motion using parabolic integration.

Figure 1-10. Subroutine-FOTPAD
Footpad is moving independently of the vehicle - calculate strut forces and footpad energy

Calculate components of strut force

Determine forces acting on footpad and footpad motions

Index previous forces for next integration of footpad motion

RETURN
This subroutine determines the forces at c.g. of vehicle to be used in determination of vehicle motion.

Figure 1-11. Subroutine-VEHFOR
This subroutine determines vehicle c.g. motion using parabolic integration.

Figure 1-12. Subroutine-VEHMOT
This subroutine computes the stability angle and orients it in proper quadrant.

**Figure 1-13. Subroutine-STABAN**
This subroutine stores variables for print to be made at the end of the run.

**Figure 1-14. Subroutine-STOREP**
This subroutine prints all output. Point of entry depends on value of IPR in call statement - CALL PRINT1 (IPR)

 ENTRY

 Go to (10, 20, 30, 40, 50, 60, 70, 80), IPR

 10  Print input data  RETURN

 20  Print on line data with headings  RETURN

Figure 1-15. Subroutine-PRINT1
Figure 1-15. Subroutine-PRINT1 (Continued)
Figure 1-15. Subroutine-PRINT1 (Concluded)
This subroutine computes energy dissipated based on vehicle velocities and c.g. drop, based on plastic stroke and based on plastic and full elastic stroke.

Figure 1-16. Subroutine-ENERGY
This subroutine determines new index values to choose new velocities from inputed array for the next run.

Figure 1-17. Subroutine-PROFIL
PROFIL—DETERMINE INDEX VALUES FOR VELOCITIES

Figure 1-17. Subroutine—PROFIL (Concluded)
MAIN PROGRAM LUNAR LANDING DYNAMICS COMPUTER PROGRAM

R. BLACK, J. CADORET, J. GIBSON THE BENDIX CORPORATION

I0-25-64

THIS PROGRAM COMPUTES THE DETAILED VEHICLE MOTIONS
FOR A PLANAR LANDING

THIS PROGRAM WAS WRITTEN IN FORTRAN IV

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

INPUT THROUGH LABLED COMMON

OUTPUT THROUGH LABLED COMMON

NOTE AN INPUT PARAMETER FOLLOWED BY A O MEANS THE INITIAL VALUE
AS B10(I). THE INSTANTANEOUS VALUE OF THE PARAMETER
IS DEFINED WITHOUT THE O AS B1(I).

SYMBOL DEFINITION

A SYMMETRY FACTOR -- IF TWO VEHICLE LEGS ARE SYMMETRIC,
SET A=2.0 FOR ONE OF THE SYMMETRIC PAIRS AND A=0.0 FOR
THE OTHER TO SAVE ON COMPUTER TIME. IE FOR 22 LANDING
A=2.0 FOR A 121 LANDING A= 1.21.0

ALPHA ANGLE IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE
SUBTENDED BY THE LOWER HARDPOINTS AND VEHICLE C.G.

BEPR VALUE OF BETA AT THE END OF THE PREVIOUS TIME INCREMENT.
USED TO ASSIGN BETA TO THE PROPER QUADRANT

BETA VEHICLE STABILITY ANGLE

BETADT RATE OF CHANGE OF BETA WITH TIME

BETAPR SIMILAR TO BEPR. USED IN THE DETERMINATION OF VEHICLE
STABILITY

B1MIN MINIMUM STABILITY ANGLE FOR LEG SET 11

B10 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE

B20 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 2 AND VEHICLE CENTERLINE

B2PREV VALUE OF B2 AT THE END OF THE PREVIOUS TIME INCREMENT

B30 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 3 AND VEHICLE CENTERLINE

B3PREV VALUE OF B3 AT THE END OF THE PREVIOUS TIME INCREMENT

CONS COUNTER FOR DETERMINING ON LINE PRINT FREQUENCY

COSA COS(Psi)

COSB COS(B1-PSI)

COSC COS(THETA)

COSD COS(B1)

COSE COS(THETA-ALPHA/2.0)

COSG COS(THETA+ALPHA/2.0)

C10*C20*C30 ANGLE , IN PLANE FORMED BY STRUT AND A NORMAL TO
THE DIRECTION OF MOTION, BETWEEN STRUT AND A PLANE NORMAL
TO THE VEHICLE CENTERLINE - FOR STRUTS 1, 2, 3 RESPECTIVELY

C1PREF VALUE OF C1 AT THE END OF THE PREVIOUS TIME INCREMENT

C2PREF VALUE OF C2 AT THE END OF THE PREVIOUS TIME INCREMENT

C3PREF VALUE OF C3 AT THE END OF THE PREVIOUS TIME INCREMENT

D VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS

DEL PROGRAM CONSTANT EQUAL TO ZERO

Figure 1-18. Main Program

1-36

BENDIX PRODUCTS AEROSPACE DIVISION
DISTANCE FROM BOTTOM OF FOOTPAD TO INTERSECTION OF THE LEG STRUTS.

TIME INTERVAL BETWEEN PROGRAM CALCULATIONS.

TIME INCREMENT USED IN THE INTEGRATION OF FOOTPAD MOTION.

INITIAL LENGTH (PROJECTED IN PLANE NORMAL TO DIRECTION OF VEHICLE MOTION) OF THE COMPONENT (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE) OF STRUT NO. 1 LENGTH.

VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT.

POTENTIAL ENERGY STORED IN STRUT NO. 1 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG.

POTENTIAL ENERGY STORED IN STRUT NO. 2 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG.

POTENTIAL ENERGY STORED IN STRUT NO. 3 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG.

ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND C.G.

ENERGY DISSIPATED BASED ON PLASTIC STROKE OF ALL STRUTS.

ENERGY DISSIPATED BASED ON PLASTIC AND FULL ELASTIC STROKE OF ALL STRUTS.

PERCENT OF TOTAL ENERGY ABSORBED BY STROKING OF THE STRUTS OF LEG SET I.

PROGRAM CONSTANT EQUAL TO 10 PERCENT OF THE POSSIBLE POTENTIAL ENERGY WHICH COULD BE STORED IN A FOOTPAD AS THE RESULT OF ELASTIC STROKING OF THE UPPER STRUT.

MINIMUM ALLOWABLE FOOTPAD SLIDING VELOCITY.

LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN X DIRECTION.

LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN Y DIRECTION.

LIMITING MINIMUM ANGULAR VELOCITY OF VEHICLE C.G.

IF XVEL, YVEL AND PSIVEL ARE SIMULTANEOUSLY LESS THAN EPS3, EPS4 AND EPS5 RESPECTIVELY AND THE FOOTPADS ARE ALL LESS THAN 1 FT. FROM THE GROUND, THE PROGRAM TERMINATES.


FORCE, PARALLEL TO THE VEHICLE CENTERLINE ACTING ON THE VEHICLE C.G. AS THE RESULTANT OF THE STRUT FORCES IN THE THREE STRUTS OF THE I TH LEG SET.


MAXIMUM ALLOWABLE TIME FOR COMPUTER RUN.

TOTAL FORCE ACTING ON THE VEHICLE IN THE LATERAL DIRECTION (NORMAL TO THE VEHICLE CENTERLINE).

TOTAL FORCE ACTING ON THE VEHICLE IN THE VERTICAL DIRECTION (PARALLEL TO THE VEHICLE CENTERLINE).

FORCE ON THE FOOTPAD I IN THE X DIRECTION IN THE FIXED COORDINATE SYSTEM.

SAME AS FXP.

AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

FX VALUE FOR LEG II STORED FOR FINAL PRINT.

FX VALUE FOR LEG JJ STORED FOR FINAL PRINT.

FXPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT.

FXPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT.

Figure 1-18. Main Program (Continued)
FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G., DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE.

FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G., DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE.

FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G., DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO.1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO.2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO.3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

PLASTIC STROKE FORCE IN STRUT NO.1 OF LEG I

PLASTIC STROKE FORCE IN STRUT NO.2 OF LEG I

PLASTIC STROKE FORCE IN STRUT NO.3 OF LEG I

PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 11)

PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2

PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO.3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.

LOCAL GRAVITY
G3FR2  F2 VALUE FOR LEG JJ STORED FOR FINAL PRINT
G3FR3  F3 VALUE FOR LEG JJ STORED FOR FINAL PRINT
H     DISTANCE FROM THE BOTTOM OF THE FOOTPAD TO THE VEHICLE CENTER OF GRAVITY
HN    VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND THE LOWEST POINT ON THE NOZZLE CONE
HVELO  HORIZONTAL VELOCITY PERPENDICULAR TO THE GRAVITAL FIELD.
IFLAG  FLAG FOR INITIALIZING THE PROGRAM
II    PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE PRINTED AS OUTPUT
IP    PRINT INDICATOR
IQ    PRINT INDICATOR
IR    PRINT INDICATOR
J     INDEX ASSOCIATED WITH FIRST LEG TO STRIKE GROUND DURING INITIAL IMPACT
JJ    SIMILAR TO II
K     FLAG FOR LANDING MODE  IF 1-2-1 LANDING , K=1  IF 2-2 LANDING , K=2
KCONMX NUMBER OF ITERATIONS OF FOOTPAD CALCULATIONS PER (DELTAT) TIME INTERVAL
KM    COUNTER FOR DETERMINING THE FREQUENCY OF THE STORED PRINTING
KPRINT COMPUTATION INCREMENTS BETWEEN PRINTOUT INTERVALS
LAND SIGNIFIES LANDING MODE - SET FOR 22 OR 121 LANDING
LINE PRINTOUT LINE COUNTER
LOTPTT FLAG TO DETERMINE IF COMPLETE SUMMARY PRINTING IS TO BE DESIRED. COMPLETE OUTPUT PRINTING IS NOT NORMALLY USED THEREFORE SET LOTPTT=0. IF SET LOTPTT=1 , THE COMPLETE HISTORY OF STRUT STROKES, AND STRUT FORCES WILL BE PRINTED.
M INDEX USED TO STORE VARIABLES FOR PRINT AT END OF RUN
MULT INDICATOR FLAG USED TO PRINT STORED OUTPUT DATA AT LESS FREQUENT INTERVALS IF FINT IS SUCH THAT THE DIMENSIONED STORAGE CAPACITY FOR THE STORED VARIABLES WILL BE EXCEEDED.
N NUMBER OF LEGS ON THE VEHICLE
NOGR(I) INDICATES IF FOOTPAD (I) IS MOVING WITH THE VEHICLE. IF NOGR(I)=-1, FOOTPAD IS MOVING WITH THE VEHICLE. IF NOGR(I) =+1, FOOTPAD IS MOVING INDEPENDENTLY
NGO STARTING COLUMN IN VELOCITY INPUT ARRAY
NOMAX ENDING COLUMN IN VELOCITY ARRAY
NSO STARTING ROW IN VELOCITY INPUT ARRAY
NSMAX ENDING ROW IN VELOCITY ARRAY
NOTE SEE WRITEUP FOR DISCUSSION OF VELOCITY ARRAY
NST FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE
NUN FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE
PHI ANGLE BETWEEN VEHICLE CENTERLINE AND GRAVITY VECTOR
PMASS MASS OF EACH FOOTPAD
PRBE SIMILAR TO BEPR. USED IN THE DETERMINATION OF BETADT IN MULTIPLE RUNS, THIS IS THE INITIAL X VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.
PRXVEL IN MULTIPLE RUNS, THIS IS THE INITIAL Y VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.
PSI INSTANTANEOUS VALUE OF PSIO
PS10 INITIAL PITCH ANGLE
PSIVEL INSTANTANEOUS PITCH VELOCITY OF THE VEHICLE C.G.
PSVELO INITIAL VEHICLE PITCH RATE
P1  FRICTION FORCE IN STRUT NO. 1
P2  FRICTION FORCE IN STRUT NO. 2

Figure 1-18. Main Program (Continued)
Friction force in strut no. 3

Moment of inertia of the individual vehicle footpads. This term includes only those footpads which are off the ground at the instant under investigation.

Vehicle moment of inertia.

Total moment of inertia of the vehicle mass and those footpads which are off the ground.

Torque acting on the vehicle C.G. in the X-Y plane during the current time interval. This torque is used in the parabolic integration procedure.

Torque acting on the vehicle C.G. in the X-Y plane during the N-1 time interval. This torque is used in the parabolic integration procedure.

Torque acting on the vehicle C.G. in the X-Y plane during the N-2 time interval. This torque is used in the parabolic integration procedure.

Exhaust nozzle radius.

Radius of footpad (i).

Run number (for identification only).

Radius of upper hardpoint mounting circle.

Radius of lower hardpoint mounting circle.

Series number (for identification only).

Sine of psi.

Sine of (b1 - psi).

Sine of theta.

Sine of (b1).

Sine of (c1).

Sine of (c3).

Sine of (theta - alpha/2.0).

Sine of (theta + alpha/2.0).

Sine of (c2).

Tensile elastic spring rate of strut no. 1 (upper).

Tensile elastic spring rate of strut no. 2 (lower).

Tensile elastic spring rate of strut no. 3 (lower).

Spring rate under vehicle footpads.

Compressive elastic spring rate of strut no. 1 (upper).

Compressive elastic spring rate of strut no. 2 (lower).

Compressive elastic spring rate of strut no. 3 (lower).

Projected length of strut no. 1 in plane parallel to direction of motion.

Minimum length to which strut no. 1 has been compressed during this run.

Initial value of sl1.

Length of strut no. 1 at the end of the previous time increment.

True instantaneous length of strut no. 1.

True initial length of strut no. 1.

Instantaneous length of strut no. 2, projected in a plane parallel to the direction of motion.

Minimum length to which strut no. 2 has been compressed during this run.

Length of strut no. 2 at the end of the previous time increment.

True instantaneous length of strut no. 2.

True initial length of strut no. 2.

Instantaneous length of strut no. 3, projected in a plane parallel to the direction of motion.

Minimum length to which strut no. 3 has been compressed during this run.

Length of strut no. 3 at the end of the previous time increment.
Figure 1-18. Main Program (Continued)
1 ENPRO(5)
COMMON/IBLOCK/BETA(5), BPR(5), BETAPR(5), LAND
COMMON/JBLOCK/BETADT, FLL, FVV, LINE, FIN, RUNNO, RUNNOO, SERNO, XN,
1 XNMIN, TMINXN, BJJMIN, TMINBI, HN, RN, KPRINT, NSO, NQO,
1 NS, NQ, NSMAX, NQMAX, LOTPRT, NST, NUN, JJ, JJ, CONS, IFLAG, PRBE, KM,
1 MULT, PRXVEL, PRYVEL
COMMON/KBLOCK/M, TIME, TME(402), XP1(402), YP1(402), XP3(402),
1 YP3(402), ZP3(402), FX1(402), FX3(402), FXPL31(402), FYPL31(402),
1 FZPL31(402), FXPL33(402), FYPL33(402), FZPL33(402), G1STR1(402),
1 G1STR2(402), G1STR3(402), G3STR1(402), G3STR2(402), G3STR3(402),
1 G1FRC1(402), G1FRC2(402), G1FRC3(402), G3FRC1(402), G3FRC2(402),
1 G3FRC3(402)

1 CALL INPUT1
C CONVERT INPUT DATA TO PROPER FORM TO BE USED BY THE PROGRAM
C
C CALL INCON
C CALL INITSE
C
20 NST=0
NUN=0
40 RUNNO=RUNNO+1.0
C CALL INITRU
C CALL PRINT1(1)
C
140 CALL STRANG
C IF(IFLAG)250,250,280
C
250 CALL STOREP
C CALL PRINT1(2)
C
280 CALL FOTPAD
C CALL VEHFOR
C CALL VEHMOT
C INCREMENT TIME FOR NEXT CALCULATION
C
640 TIME=TIME+DELTAT
CONS=CONS+1.0
PRBE=BETA(II)
C
TEST NOZZLE = GROUND CLEARANCE AND STORE MINIMUM VALUE
C
XN=X-HN*COSA-RN*ABS (SINA)
IF (XN-XNMIN) 650,651,651
650 XNMIN=XN
TMINXN=TIME-DELTAT
IF (TIME-2.0*DELTAT) 653,653,651
C
TEST STABILITY ANGLE AND STORE MINIMUM VALUE
C
651 IF (ABS (BETA(II))-ABS (BIJMIN)) 652,1653,1653

Figure 1-18. Main Program (Continued)
652 BIIMIN=BETA(II)
TMINBI=TIME-DELTAT
1653 IF(ABS (BETA(JJ))-ABS (BIJMIN)) 1654, 653, 653
1654 BIJMIN=BETA(JJ)
TMINBJ=TIME-DELTAT

C TEST IF VEHICLE IS STOPPED
C
653 IF(ABS (XVEL)-EPS3) 656, 656, 668
656 IF(ABS (YVEL)-EPS4) 660, 660, 668
660 IF(ABS (PSIVEL)-EPS5) 663, 663, 668
663 DO 665 I=1,N,K
665 CONTINUE
GO TO 810
668 CALL STABAN

C TEST IF VEHICLE IS UNSTABLE
C
DO 674 I=1,N,K
PROD=BETA(I)*BETAPR(I)
BETAPR(I)=BETA(I)
IF(PROD) 6732, 674, 674

C TEST WHETHER LANDING IS 1=2=1 OR 2=2 . IF IT IS 1=2=1 *DISREGARD
STABILITY ANGLE DEFINED FOR MIDDLE LEGS
C
6732 IF(LAND-100)1770,770,6734
6734 IF(I=1)770,770,6736
6736 IF(I=3)674,770,674
674 CONTINUE
IF(TIME-FINT)675,BOO,BOO

C SET LINE COUNT AND STORAGE FOR PLOT ROUTINE
C
675 KM=KM+1
IF(CONS-KPRINT)710,680,680
680 CONS=O.O
BETADT=(BETA(II)-PRBE)/DELTAT
687 IF(KM=MULT*KPRINT)700,688,688

C 688 CALL STOREP
C
KM=0
C 700 CALL PRINTI(3)
C 710 CALL STGEOM
C
IFLAG=I
GO TO 140
C
C VEHICLE IS UNSTABLE PRINT OUTPUT DATA
C
770 PHI=ZETA+PSI
C
CALL ENERGY
C
DO 775 I=1,2
C
CALL PRINTI(4)

Figure 1-18. Main Program (Continued)
775 CALL PRINT1(7)
C
CALL PRINT1(8)
C
IPRO=1
GO TO 830
800 PHI=ZETA+PSI
C
CALL ENERGY
C
DO 805 I=1,2
C
CALL PRINT1(5)
C
805 CALL PRINT1(7)
C
GO TO 825
810 PHI=ZETA+PSI
C
CALL ENERGY
C
DO 815 I=1,2
C
CALL PRINT1(6)
C
815 CALL PRINT1(7)
C
825 CALL PRINT1(8)
C
IPRO=2
C
830 CALL PROFIL(IPRO)
C
GO TO (1,20,40), IPRO
END

Figure 1-18. Main Program (Concluded)
THE BENDIX CORPORATION

READ INITIAL DATA

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

INPUT I NPUT1

NOTE AN INPUT PARAMETER FOLLOWED BY A 0 IMPUTES THE INITIAL VALUE
AS B1O(I). THE INSTANTANEOUS VALUE OF THE PARAMETER
IS DEFINED WITHOUT THE 0 AS B1(I).

SYMBOL DEFINITION

A SYMMETRY FACTOR -- IF TWO VEHICLE LEGS ARE SYMMETRIC,
SET A=2.0 FOR ONE OF THE SYMMETRIC PAIRS AND A=0.0 FOR
THE OTHER TO SAVE ON COMPUTER TIME IE FOR 22 LANDING
A=2.0,2.0,0.0,0.0 FOR A 121 LANDING A= 1.2,1.0

ALPHA ANGLE(IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE)
SUBTENDED BY THE LOWER HARDPOINTS AND VEHICLE C.G.

D VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS

DELTAP DISTANCE FROM BOTTOM OF FOOTPAD TO INTERSECTION OF THE
LEG STRUTS

DELTAT TIME INTERVAL BETWEEN PROGRAM CALCULATIONS

D11 VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT

EPS2 MINIMUM ALLOWABLE FOOTPAD SLIDING VELOCITY

EPS3 LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN X DIRECTION

EPS4 LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN Y DIRECTION

EPS5 LIMITING MINIMUM ANGULAR VELOCITY OF VEHICLE C.G.

NOTE IF XVEL, YVEL AND PSIVEL ARE SIMULTANEOUSLY LESS THAN
EPS3, EPS4 AND EPS5 RESPECTIVELY AND THE FOOTPADS ARE ALL
LESS THAN 1 FT. FROM THE GROUND, THE PROGRAM TERMINATES

FINT MAXIMUM ALLOWABLE TIME FOR COMPUTER RUN

F11 PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)

F22 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2

F33 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3

GRAV LOCAL GRAVITY

GRDMU COEFFICIENT OF FRICTION BETWEEN VEHICLE FOOTPADS AND GROUND

H DISTANCE FROM THE BOTTOM OF THE FOOTPAD TO THE VEHICLE
CENTER OF GRAVITY

HN VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND THE LOWEST
POINT ON THE NOZZLE CONE

HVELO HORIZONTAL VELOCITY PERPENDICULAR TO THE GRAVITATIONAL
FIELD

I I PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE
PRINTED AS OUTPUT

J INDEX ASSOCIATED WITH FIRST LEG TO STRIKE GROUND DURING
INITIAL IMPACT

JJ SIMILAR TO II

Figure 1-19. Subroutine INPUT1
Figure 1-19. Subroutine INPUT1 (Continued)
Figure 1-19. Subroutine INPUT1 (Concluded)
SUBROUTINE INCON

Figure 1-20. Subroutine INCON
COMMON/ABLOCK/N,K,NOGR(5)
COMMON/BBLOCK/H,DELTA,P,D1(5),R1(5),R2(5),RP(5),THETA(5),ALPHA(5),
1 D(5),SL10(5),D10(5),B10(5),B20(5),B30(5),C10(5),C20(5),C30(5)
DIMENSION CIANG(5),C2ANG(5),C3ANG(5)

DETERMINE VEHICLE GEOMETRIC RELATIONSHIPS
1 DO 100 I=1,N,K
  SL10(I)=SQRT((H-DELTA,P-D1(I))*(H-DELTA,P-D1(I))+(RP(I)-R1(I))*
1*COS(THETA(I)))+((RP(I)-R1(I))*COS(THETA(I))))
  D10(I)=(RP(I)-R1(I))*SIN(THETA(I))
  B10(I)=ATAN((RP(I)-R1(I))*COS(THETA(I))/(H-DELTA,P-D1(I)))
  B20(I)=ATAN((RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)-ALPHA(I)/2.0)
1)/(H-DELTA,P-D1(I)-D(I)))
  B30(I)=ATAN((RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)+ALPHA(I)/2.0)
1)/(H-DELTA,P-D1(I)-D(I)))
  CIANG(I)=ATAN(ABS(SL10(I)/D10(I)))
  S2X=RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)-ALPHA(I)/2.0)
  SL200=SQRT(S2X*S2X+(H-DELTA,P-D1(I)-D(I))*(H-DELTA,P-D1(I)-D(I)))
  C2ANG(I)=ATAN(ABS(SL200/(RP(I)*SIN(THETA(I)))-R2(I)*SIN(THETA(I)-
1*ALPHA(I)/2.0))))
  S3X=RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)+ALPHA(I)/2.0)
  SL300=SQRT(S3X*S3X+(H-DELTA,P-D1(I)-D(I))*(H-DELTA,P-D1(I)-D(I)))
  C3ANG(I)=ATAN(ABS(SL300/(RP(I)*SIN(THETA(I)))-R2(I)*SIN(THETA(I)+
1*ALPHA(I)/2.0))))

C TEST WHICH QUADRANT CONTAINS THETA TO DETERMINE PROPER QUADRANT
C
64 IF (THETA(I)} 68, 70, 72
68  C10(I)=3.14159265359-C1ANG(I)
   GO TO 76
70  C10(I)=1.5707963
   GO TO 76
72  C10(I)=C1ANG(I)
76  IF (RP(I)*SIN(THETA(I))-R2(I)*SIN(THETA(I)-ALPHA(I)/2.0)) 80, 82, 84
80  C20(I)=3.14159265359-C2ANG(I)
   GO TO 88
82  C20(I)=1.5707963
   GO TO 88
84  C20(I)=C2ANG(I)
88  IF (RP(I)*SIN(THETA(I))-R2(I)*SIN(THETA(I)+ALPHA(I)/2.0)) 92, 94, 96
92  C30(I)=3.14159265359-C3ANG(I)
   GO TO 100
94  C30(I)=1.5707963
   GO TO 100
96  C30(I)=C3ANG(I)
100 CONTINUE
RETURN
END

Figure 1-20. Subroutine INCON (Concluded)
INITIALIZE PROGRAM CONSTANTS AND VARIABLES FOR SERIES OF RUNS

THIS PROGRAM WAS WRITTEN IN FORTRAN IV

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS B10(I), THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS B1(I).

TIME INTERVAL BETWEEN PROGRAM CALCULATIONS
MAXIMUM ALLOWABLE TIME FOR COMPUTER RUN
PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)
FLAG FOR LANDING MODE IF 1-2-1 LANDING, K=1 IF 2-2 LANDING, K=2
COMPUTATION INCREMENTS BETWEEN PRINTOUT INTERVALS
NUMBER OF LEGS ON THE VEHICLE
RUN NUMBER (FOR IDENTIFICATION ONLY)
COMPRESSIONELASTIC SPRING RATE OF STRUT NO. 1(UPPER)

PROGRAM CONSTANT EQUAL TO 10 PERCENT OF THE POSSIBLE POTENTIAL ENERGY WHICH COULD BE STORED IN A FOOTPAD AS THE RESULT OF ELASTIC STROKING OF THE UPPER STRUT
INDICATOR FLAG USED TO PRINT STORED OUTPUT DATA AT LESS FREQUENT INTERVALS IF FINT IS SUCH THAT THE DIMENSIONED STORAGE CAPACITY FOR THE STORED VARIABLES WILL BE EXCEEDED.
STARTING COLUMN IN VELOCITY INPUT ARRAY
STARTING ROW IN VELOCITY INPUT ARRAY
SEE WRITEUP FOR DISCUSSION OF VELOCITY ARRAY
IN MULTIPLE RUNS, THIS IS THE INITIAL X VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.
IN MULTIPLE RUNS, THIS IS THE INITIAL Y VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.

1-50

Figure 1-21. Subroutine INITSE
COMMON/HBLOCK/VVELO(14,8),HVELO(14,8),XVELO(14,8),YVELO(14,8),
1 XVELO0,YVELO0,PSI,PSI0,PSVELO,XO,YO,XVEL,YVEL,PSIVEL,GRAV,
1 ZETA,KCONMX,DELTAT,DELTAT,GRDMU(5),A(5),FXX(3),FYY(3),RM00(3),
1 VMAS5,PMASS,RMOMIT,RMOMGR,RMOMI,FA(5),FB(5),FC(5),IP,IP1,IR,
1 EGBAL1,EGBAL2,EGBAL3,P1(5),P2(5),P3(5),DEL(5),EPS F(5),PHI,
1 ENPRO(5)

COMMON/JBLOCK/BETADT,FLL,FVV,LINE,FINT,RUNNO, RUNNO0, SERNO,XN,
1 XNMIN,TMINXN, BI MIN,BJMIN,TMINBI,TMINBJ,HN,RN, KPRINT,NSO,NQO,
1 NS,NQ,NSMAX,NQMAX, LOPRT-NST, NUM,IIJJ,CONS,IFLAG,PRBE,KM,
1 MULT,PRXVEL,PRYVEL

INITIALIZE PROGRAM CONSTANTS

DO 5 I=1,N,K
5 EPSEN(I)=F11(I)*F11(I)/(SK1*20.0)

DETERMINE PRINTOUT FREQUENCY OF STORED OUTPUT

FACTOR=FINT/402.0
IF(FACTOR-(DELTAT*KPRINT))7,8,8
7 MULT=1
GO TO 15
8 IF(FACTOR-2.0*(DELTAT*KPRINT))9,12,12
9 MULT=2
GO TO 15
12 MULT=5

SET STARTING POINT IN VELOCITY ARRAY

15 NS=NSO
NO=NQO

INITIALIZE PROGRAM VARIABLES

RUNNO=RUNNO0-1.0
PRXVEL=100.0
PRYVEL=100.0
RETURN
END
Initialize program constants and variables for each run

This program was written in FORTRAN IV

This program executes on the UNIVAC 1107 computer

Input through labeled common

An input parameter followed by a 0 implies the initial value as B10(I). The instantaneous value of the parameter is defined without the 0 as B1(I).

Symbol definition

**ANGLE (in plane perpendicular to the vehicle centerline)**
- **B10** Angle (projected in plane parallel to vehicle motion) between strut no. 1 and vehicle centerline
- **B20** Angle (projected in plane parallel to vehicle motion) between strut no. 2 and vehicle centerline
- **B30** Angle (projected in plane parallel to vehicle motion) between strut no. 3 and vehicle centerline

**COS**
- **C10,C20,C30** Angle in plane formed by strut and a normal to the direction of motion, between strut and a plane normal to the vehicle centerline - for struts 1,2,3 respectively

**DISTANCE FROM BOTTOM OF FOOTPAD TO INTERSECTION OF THE LEG STRUTS**

**TIME INTERVAL BETWEEN PROGRAM CALCULATIONS**

**INITIAL LENGTH (in plane normal to direction of vehicle motion) of the component (in plane perpendicular to the vehicle centerline) of strut no. 1 length**

**VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT**

**FORCE IN THE X DIRECTION ON THE FOOTPAD (I)**

**PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)**

**PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2**

**PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3**

**HORIZONTAL VELOCITY PERPENDICULAR TO THE GRAVITATIONAL FIELD**

**NUMBER OF ITERATIONS OF FOOTPAD CALCULATIONS PER (DELTAT)**

**TIME INTERVAL**

**IN MULTIPLE RUNS, THIS IS THE INITIAL X VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.**

**IN MULTIPLE RUNS, THIS IS THE INITIAL Y VELOCITY OF THE PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.**

**INITIAL PITCH ANGLE**

**INITIAL VEHICLE PITCH RATE**

**RADIUS OF UPPER HARDPOINT MOUNTING CIRCLE**

**RADIUS OF LOWER HARDPOINT MOUNTING CIRCLE**

**SIN(PSI)**

**COMPRESSIVE ELASTIC SPRING RATE OF STRUT NO. 1(UPPER)**

Figure 1-22. Subroutine INITRU
Figure 1-22. Subroutine INITRU (Continued)

C SK2 COMPRESSIVE ELASTIC SPRINGRATE OF STRUT NO. 2 (LOWER)
C SK3 COMPRESSIVE ELASTIC SPRINGRATE OF STRUT NO. 3 (LOWER)
C SL10 INITIAL VALUE OF SL1
C THEETA ANGLE BETWEEN PLANE PARALLEL TO VEHICLE CENTERLINE IN DIRECTION OF VEHICLE MOTION AND PLANE THROUGH VEHICLE CENTERLINE AND UPPER HARDPOINT
C TIME TIME AFTER TOUCHDOWN
C VVELO INITIAL VEHICLE VELOCITY PARALLEL TO THE GRAVITATIONAL FIELD, POSITIVE DOWNWARD
C YO INITIAL HORIZONTAL POSITION OF VEHICLE C.G.
C YVEL INSTANTANEOUS Y VELOCITY OF THE VEHICLE C.G.
C ZETA GROUND SLOPE
C OUTPUT THROUGH LABELED COMMON
C SYMBOL DEFINITION
C BEPR VALUE OF BETA AT THE END OF THE PREVIOUS TIME INCREMENT.
C BETAPR USED TO ASSIGN BETA TO THE PROPER QUADRANT
C BETADT VALUE OF BETA AT THE END OF THE PREVIOUS TIME INCREMENT.
C BETADT USED IN THE DETERMINATION OF VEHICLE STABILITY
C BIIMIN MINIMUM STABILITY ANGLE FOR LEG SET II
C BJMIN MINIMUM STABILITY ANGLE FOR LEG SET JJ
C B2PREV VALUE OF B2 AT THE END OF THE PREVIOUS TIME INCREMENT
C B3PREV VALUE OF B3 AT THE END OF THE PREVIOUS TIME INCREMENT
C CONS COUNTER FOR DETERMINING ON LINE PRINT FREQUENCY
C COSB COS(B1-PSI)
C COSC COS(THETA)
C COSD COS(B1)
C COSG COS(THETA+ALPHA/2.0)
C C1PREV VALUE OF C1 AT THE END OF THE PREVIOUS TIME INCREMENT
C C2PREV VALUE OF C2 AT THE END OF THE PREVIOUS TIME INCREMENT
C C3PREV VALUE OF C3 AT THE END OF THE PREVIOUS TIME INCREMENT
C DELI4 TIME INCREMENT USED IN THE INTEGRATION OF FOOTPAD MOTION.
C ELI4 DELTAT/CONMX
C E1(I) POTENTIAL ENERGY STORED IN STRUT NO. 1 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG
C E2(I) POTENTIAL ENERGY STORED IN STRUT NO. 2 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG
C E3(I) POTENTIAL ENERGY STORED IN STRUT NO. 3 OF LEG I DUE TO COMPRESSION OR EXTENSION OF THE LEG
C FLL TOTAL FORCE ACTING ON THE VEHICLE IN THE LATERAL DIRECTION (NORMAL TO THE VEHICLE CENTERLINE)
C FVV TOTAL FORCE ACTING ON THE VEHICLE IN THE VERTICAL DIRECTION (PARALLEL TO THE VEHICLE CENTERLINE)
C FX SAME AS FXP
C FXP(I) FORCE ON THE FOOTPAD I IN THE X DIRECTION IN THE FIXED COORDINATE SYSTEM
C FXPLG1 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE
C FXPLG2 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE
C FYPLG1 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE
C FYPLG2 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 2
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FYPLG3
AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO,3
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FZPLG1
AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.1
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FZPLG2
AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.2
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FXPLG3
AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO.3
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FZPLG3
AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO.3
DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
THE PARABOLIC INTEGRATION PROCEDURE

FZPL31
FZPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT

FXX(3)
FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(2)
FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(1)
FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(3)
FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(2)
FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(1)
FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

F1(1)
PLASTIC STROKE FORCE IN STRUT NO.1 OF LEG I

F2(1)
PLASTIC STROKE FORCE IN STRUT NO.2 OF LEG I

F3(1)
PLASTIC STROKE FORCE IN STRUT NO.3 OF LEG I

IFLAG
FLAG FOR INITIALIZING THE PROGRAM

KM
COUNTER FOR DETERMINING THE FREQUENCY OF THE STORED
PRINTING

LINE
PRINTOUT LINE COUNTER

M
INDEX USED TO STORE VARIABLES FOR PRINT AT END OF RUN

MULT
INDICATOR FLAG USED TO PRINT STORED OUTPUT DATA AT LESS
FREQUENT INTERVALS IF FINT IS SUCH THAT THE DIMENSIONED
STORAGE CAPACITY FOR THE STORED VARIABLES WILL BE EXCEEDED.

N
NUMBER OF LEGS ON THE VEHICLE

NOGR(1)
INDICATES IF FOOTPAD (1) IS MOVING WITH THE VEHICLE. IF
NOGR(1)= 1, FOOTPAD IS MOVING WITH THE VEHICLE. IF
NOGR(1) =+1, FOOTPAD IS MOVING INDEPENDENTLY

PSI
INSTANTANEOUS VALUE OF PS10

PSIVEL
INSTANTANEOUS PITCH VELOCITY OF THE VEHICLE C.G.

RMOMGR
MOMENT OF INERTIA OF THE INDIVIDUAL VEHICLE FOOTPADS.
 THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE
GROUND AT THE INSTANT UNDER INVESTIGATION

RM00(3)
TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE CURRENT TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

RM00(2)
TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-1 TIME INTERVAL. THIS TORQUE IS USED

Figure 1-22. Subroutine INITRU (Continued)
IN THE PARABOLIC INTEGRATION PROCEDURE
TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-2 TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

SINB  SIN(B1-PSI)
SINC  SIN(THETA)
SIND  SIN(B1)
SINJ  SIN(THETA+ALPHA/2.0)
SINK  SIN(THETA-ALPHA/2.0)
SL1   PROJECTED LENGTH OF STRUT NO.1 IN PLANE PARALLEL TO
       DIRECTION OF MOTION
SL1M  MINIMUM LENGTH TO WHICH STRUT NO.1 HAS BEEN COMPRESSED
       DURING THIS RUN
SL1PRE LENGTH OF STRUT NO.1 AT THE END OF THE PREVIOUS TIME
       INCREMENT
SL1T  TRUE INSTANTANEOUS LENGTH OF STRUT NO.1
SL1I0 TRUE INITIAL LENGTH OF STRUT NO.1
SL2   INSTANTANEOUS LENGTH OF STRUT NO.2 PROJECTED IN A PLANE
       PARALLEL TO THE DIRECTION OF MOTION
SL2M  MINIMUM LENGTH TO WHICH STRUT NO.2 HAS BEEN COMPRESSED
       DURING THIS RUN
SL2PRE LENGTH OF STRUT NO.2 AT THE END OF THE PREVIOUS TIME
       INCREMENT
SL2T  TRUE INSTANTANEOUS LENGTH OF STRUT NO.2
SL2I0 TRUE INITIAL LENGTH OF STRUT NO.2
SL3   INSTANTANEOUS LENGTH OF STRUT NO.3 PROJECTED IN A PLANE
       PARALLEL TO THE DIRECTION OF MOTION
SL3M  MINIMUM LENGTH TO WHICH STRUT NO.3 HAS BEEN COMPRESSED
       DURING THIS RUN
SL3PRE LENGTH OF STRUT NO.3 AT THE END OF THE PREVIOUS TIME
       INCREMENT
SL3T  TRUE INSTANTANEOUS LENGTH OF STRUT NO.3
SL3I0 TRUE INITIAL LENGTH OF STRUT NO.3
X    INSTANTANEOUS X POSITION OF THE VEHICLE C.G.
XMIN MINIMUM VALUE OF NOZZLE CLEARANCE
XO   INITIAL VERTICAL POSITION OF VEHICLE CENTER OF GRAVITY
XP(1) X POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
XPVEL(1) VELOCITY OF THE I TH FOOTPAD IN THE X DIRECTION IN THE
         FIXED COORDINATE SYSTEM
XVELO INITIAL VERTICAL VELOCITY OF VEHICLE C.G.
XVELO0 SAME AS XVELO
YP(1) Y POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
YPVEL(1) VELOCITY OF THE I TH FOOTPAD IN THE Y DIRECTION IN THE
         FIXED COORDINATE SYSTEM
YVELO INITIAL HORIZONTAL VELOCITY OF VEHICLE C.G.
ZP(1) Z POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
ZPVEL(1) VELOCITY OF THE I TH FOOTPAD IN THE Z DIRECTION IN THE
         FIXED COORDINATE SYSTEM

SUBROUTINE INITRU

COMMON/ABLOCK/N,K,NOGR(5)
COMMON/BBLOCK/H,DELTAP,D11(5),R1(5),R2(5),R(P5),THETA(5),ALPHA(5),
1 COSG(5),SINK(5),COS(5),SIN(5),SINL(5),SINI(5),COSA,SINA
1 SL2(5),SL3(5),SL1T(5),SL2T(5),SL3T(5),SL1TO(5),SL2TO(5),SL3TO(5),

Figure 1-22. Subroutine INITRU (Continued)
CONVERT INPUT VELOCITIES FROM GRAVITY COORD. SYSTEM TO SURFACE COORD. SYSTEM

CHOOSE INITIAL TOUCHDOWN CONDITIONS

XVELO(NO,NS)= HVELO(NO,NS)*SIN(ZETA)-VVELO(NO,NS)*COS(ZETA)
YVELO(NO,NS)=HVELO(NO,NS)*COS(ZETA)+VVELO(NO,NS)*SIN(ZETA)
XVELO0=XVELO(NO,NS)
YVELO0=YVELO(NO,NS)
XVEL=XVELO(NO,NS)
YVEL=YVELO(NO,NS)
Y=YO
X=SL10(J)*COS(B10(J)-PS10)+R1(J)*COS(THETA(J))*SIN(PS10)+D11(J)*
1 COS(PS10)+DELTAP
XO=X
PSI+=PS10
PSIVEL=PSVELO

TEST IF TOUCHDOWN VELOCITIES DUPLICATE THOSE OF PREVIOUS RUN. IF SO, STOP. OTHERWISE CONTINUE

IF(PRXXVEL-XVELO0)44,43,44
43 IF(PRYYVEL-YVELO0)44,845,44
44 PRXXVEL=XVELO0
PRYYVEL=YVELO0

INITIALIZE PROGRAM VARIABLES

TCONMX=KCONMX
DELTAT/DELTAT/TCONMX
SINA=SIN(PS10)
COSA=COS(PS10)
DO 45 I=1,2
FXX(I)=0,0
FY(I)=0,0
45 RM00(I)=0,0

Figure 1-22. Subroutine INITRU (Continued)
ZERO OUT LINE COUNT
LINE=0
KM=0

ZERO OUT INITIAL FORCES FOR PARABOLIC INTEGRATION OF FOOTPAD
FORCES

DO 70 I=1,N,K
E1(I)=0.0
E2(I)=0.0
E3(I)=0.0
F1(I)=0.0
F2(I)=0.0
F3(I)=0.0
FX(I)=0.0
FL1=0.0
FV1=0.0

INITIALIZE FORCES FOR PARABOLIC INTEGRATION PROCEDURE

FXPLG3(I)=0.0
FYPLG3(I)=0.0
FZPLG3(I)=0.0
FXPLG1(I)=0.0
FXPLG2(I)=0.0
FYPLG1(I)=0.0
FYPLG2(I)=0.0
FZPLG1(I)=0.0
FZPLG2(I)=0.0

DETERMINE INITIAL STRUT ANGLES AND TRIGONOMETRIC RELATIONSHIPS

B1(I)=B10(I)
COSB(I)=COS(B1(I)-PSI)
COSC(I)=COS(THETA(I))
SINB(I)=SIN(B1(I)-PSI)
SINC(I)=SIN(THETA(I))
SL1(I)=SL10(I)
B2PREV(I)=B20(I)
B3PREV(I)=B30(I)
C1PREV(I)=C10(I)
C2PREV(I)=C20(I)
C3PREV(I)=C30(I)

DETERMINE FOOTPAD POSITIONS AND VELOCITIES

DO 100 I=1,N,K
XP(I)=X-D11(I)*COSA-RI(I)_COSB(I)
YP(I)=Y-D11(I)*SINA+RI(I)_COSB(I)
ZP(I)=0.0
XPVEL(I)=XVEL+D11(I)*SINA_PSIVEL-RI(I)_COSB(I)*PSIVEL
YPVEL(I)=YVEL-D11(I)*COSA_PSIVEL+RI(I)_SINA_PSIVEL-COSB(I)*PSIVEL
ZPVEL(I)=0.0

DETERMINE STABILITY ANGLE FUNCTIONS

BETA(I)=0.0
BEPR(I)=ATAN((YP(I)-Y)/(X-XP(I)))-ZETA

Figure 1-22. Subroutine INITRU (Continued)
BETAPR(I)=0.0
NOGR(I)=1

c
INITIALIZE PROGRAM FOR STRUT LENGTH CALCULATIONS

c
SL1PRE(I)=0.0
SL2PRE(I)=0.0
SL3PRE(I)=0.0
SL1M(I)=F11(I)/SK1
SL2M(I)=F22(I)/SK2
100 SL3M(I)=F33(I)/SK3
TIME=0.0
CONS=0.0
XNMIN=100.0
BIJM=10.0
BJJMIN=-10.0
IFLAG=-1
BETADT=PSVELO
M=0
RMOMGR=0.0

c
DETERMINE TRIGONOMETRIC RELATIONSHIPS FOR STRUT LENGTH
CALCULATIONS.

c
110 DO 130 I=1,N,K
SIND(I)=SIN (B1(I))
COSD(I)=COS (B1(I))
COS(E(I)=COS (THETA(I)-ALPHA(I)/2.0)
COSG(I)=COS (THETA(I)+ALPHA(I)/2.0)
120 SINJ(I)=SIN (THETA(I)-ALPHA(I)/2.0)
SINK(I)=SIN (THETA(I)+ALPHA(I)/2.0)

c
DETERMINE INITIAL STRUT LENGTHS.

SL1TO(I)=SQRT (SL1(I)*SL1(I)+D1(I)*D1(I))
HALF3=(SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))
HALF4=(SL1(I)*COSD(I)-D(I))*(SL1(I)*COSD(I)-D(I))
SL2(I)=SQRT (HALF3+HALF4)
SL2TO(I)=SQRT (SL2(I)*SL2(I)+(D1(I)+R1(I)*SINC(I)-R2(I)*SINJ(I))*
(D1(I)+R1(I)*SINC(I)-R2(I)*SINJ(I))
HALF1=(SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))
HALF2=(SL1(I)*COSD(I)-D(I))*(SL1(I)*COSD(I)-D(I))
SL3(I)=SQRT (HALF1+HALF2)
SL3TO(I)=SQRT (SL3(I)*SL3(I)+(D1(I)-R2(I)*SINK(I)+R1(I)*SINC(I))*
(D1(I)-R2(I)*SINK(I)+R1(I)*SINC(I))
SL1T(I)=SL1TO(I)
SL2T(I)=SL2TO(I)
130 SL3T(I)=SL3TO(I)

c
DETERMINE INITIAL STRUT LENGTHS.

CMPLE STABILITY ANGLES AT TOUCHDOWN

c
270 BETA(I)=ATAN ((YP(I)-Y)/(X-XP(I)))-ZETA
BETA(J)=ATAN ((YP(J)-Y)/(X-XP(J)))-ZETA
RETURN

845 PRINT 950
PRINT 935
STOP
935 FORMAT(98H X AND Y VELOCITIES ARE IDENTICAL TO THOSE OF THE PREVIOUS RUN. CHECK THE INPUT DATA FOR ERRORS )
950 FORMAT(1H1)
END

Figure 1-22. Subroutine INITRU (Concluded)
TITLE STRANG

AUTHORS R. BLACK, J. CADORET, J. GIBSON THE BENDIX CORPORATION

DATE 10-25-64

PURPOSE DETERMINE STRUT ANGLES AND ASSIGN TO PROPER QUADRANTS

CALL STRANG

NOTE THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

INPUT THROUGH Labeled COMMON

NOTE AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS B1O(I). THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS B1(I).

SYMBOL DEFINITION

B2PREV VALUE OF B2 AT THE END OF THE PREVIOUS TIME INCREMENT
B3PREV VALUE OF B3 AT THE END OF THE PREVIOUS TIME INCREMENT
COSC COS(THETA)
COSD COS(B1)
COSE COS(THETA-ALPHA/2.0)
COSG COS(THETA+ALPHA/2.0)
C1PREV VALUE OF C1 AT THE END OF THE PREVIOUS TIME INCREMENT
C2PREV VALUE OF C2 AT THE END OF THE PREVIOUS TIME INCREMENT
C3PREV VALUE OF C3 AT THE END OF THE PREVIOUS TIME INCREMENT
D VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS
D10 INITIAL LENGTH (PROJECTED IN PLANE NORMAL TO DIRECTION OF VEHICLE MOTION) OF THE COMPONENT (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE) OF STRUT NO 1 LENGTH
K FLAG FOR LANDING MODE IF 1-2-1 LANDING, K=1 IF 2-2 LANDING, K=2
N NUMBER OF LEGS ON THE VEHICLE
NOGR(I) INDICATES IF FOOTPAD (I) IS MOVING WITH THE VEHICLE. IF NOGR(I)=-1, FOOTPAD IS MOVING WITH THE VEHICLE. IF NOGR(I)=+1, FOOTPAD IS MOVING INDEPENDENTLY
R1 RADIUS OF UPPER HARDPOINT MOUNTING CIRCLE
R2 RADIUS OF LOWER HARDPOINT MOUNTING CIRCLE
SINC SIN(THETA)
SIND SIN(B1)
SINJ SIN(THETA-ALPHA/2.0)
SINK SIN(THETA+ALPHA/2.0)
SL1 PROJECTED LENGTH OF STRUT NO.1 IN PLANE PARALLEL TO DIRECTION OF MOTION
SL2 INSTANTANEOUS LENGTH OF STRUT NO.2, PROJECTED IN A PLANE PARALLEL TO THE DIRECTION OF MOTION
SL3 INSTANTANEOUS LENGTH OF STRUT NO.3, PROJECTED IN A PLANE PARALLEL TO THE DIRECTION OF MOTION

OUTPUT THROUGH Labeled COMMON

SYMBOL DEFINITION

B2O ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 2 AND VEHICLE CENTERLINE

Figure 1-23. Subroutine STRANG
B30 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 3 AND VEHICLE CENTERLINE

C10, C20, C30 ANGLE, IN PLANE FORMED BY STRUT AND A NORMAL TO THE DIRECTION OF MOTION, BETWEEN STRUT AND A PLANE NORMAL TO THE VEHICLE CENTERLINE — FOR STRUTS 1, 2, 3 RESPECTIVELY

SUBROUTINE STRANG

COMMON/ABLOCK/N,K,NOGR(5)
COMMON/BBLOCK/H,DELTAP,D11(5),R1(5),R2(5),RP(5),THETA(5),ALPHA(5),
1 D(5),SL10(5),D10(5),B10(5),B20(5),B30(5),C10(5),C20(5),C30(5)
COMMON/CBLOCK/SIND(5),COSC(5),COSE(5),COSD(5),SINC(5),SINK(5),
1 COSG(5),SINB(5),COSB(5),SING(5),SINL(5),SINJ(5),COSA,SINA
1 SL2(5),SL3(5),SL1T(5),SL2T(5),SL3T(5),SL1T0(5),SL2T0(5),SL3T0(5),
1 B3PREV(5),C1PREV(5),C2PREV(5),C3PREV(5),EPS2, EPS3, EPS4, EPS5
DIMENSION B2MI(5),B2PL(5),B3MI(5),B3PL(5),C1MI(5),C1PL(5),
1 C2MI(5),C2PL(5),C3MI(5),C3PL(5)

DETERMINE STRUT ANGLES AND ASSIGN TO PROPER QUADRANTS

TEST IF FOOTPAD (I) IS MOVING WITH THE VEHICLE

140 DO 240 I=1,N,K
141 IF(NOGR(I))240,240,145
146 DETERMINE B2 ANGLE

145 B2(I)=ATAN(((SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSE(I))/(SL1(I)*
1 COSD(I)-D(I))))
D2MI(I)=B2(I)-3.14159265359
D2PL(I)=B2(I)+3.14159265359
DIFF1=ABS (B2MI(I)-B2PREV(I))
DIFF2=ABS (B2(I)-B2PREV(I))
DIFF3=ABS (B2PL(I)-B2PREV(I))
148 IF (DIFF1-DIFF2)150,152,152
150 IF (DIFF1-DIFF3)154,158,158
152 IF (DIFF2-DIFF3)160,158,158
154 B2(I)=B2MI(I)
GO TO 160
158 B2(I)=B2PL(I)

DETERMINE B3 ANGLE

165 B3(I)=ATAN(((SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))/(SL1(I)*
1 COSD(I)-D(I))))
D3MI(I)=B3(I)-3.14159265359
D3PL(I)=B3(I)+3.14159265359
DIFF1=ABS (B3MI(I)-B3PREV(I))
DIFF2=ABS (B3(I)-B3PREV(I))
DIFF3=ABS (B3PL(I)-B3PREV(I))
168 IF (DIFF1-DIFF2)170,172,172
170 IF (DIFF1-DIFF3)174,178,178
172 IF (DIFF2-DIFF3)180,178,178
174 B3(I)=B3MI(I)
GO TO 180
178 B3(I)=B3PL(I)

Figure 1-23. Subroutine STRANG (Continued)
Figure 1-23. Subroutine STRANG (Concluded)
STGEOM

THIS PROGRAM WAS WRITTEN IN FORTRAN IV

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

THROUGH LABELED COMMON

AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS BIO(I). THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS BI(I).

COSA = COS(PSI)
COSC = COS(THETA)
COSE = COS(THETA-ALPHA/2.0)
COSG = COS(THETA+ALPHA/2.0)
D = VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS
D10 = INITIAL LENGTH (PROJECTED IN PLANE normal TO DIRECTION OF VEHICLE MOTION) OF THE COMPONENT (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE) OF STRUT NO 1 LENGTH
D11 = VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT
K = FLAG FOR LANDING MODE IF 1-2-1 LANDING , K=1 IF 2-2 LANDING, K=2
N = NUMBER OF LEGS ON THE VEHICLE
NOGR(I) = INDICATES IF FOOTPAD (I) IS MOVING WITH THE VEHICLE. IF NOGR(I)=-1, FOOTPAD IS MOVING WITH THE VEHICLE. IF NOGR(I) =+1, FOOTPAD IS MOVING INDEPENDENTLY
PSI = INSTANTANEOUS VALUE OF PSIO
R1 = RADIUS OF UPPER HARDPOINT MOUNTING CIRCLE
R2 = RADIUS OF LOWER HARDPOINT MOUNTING CIRCLE
SINA = SIN(PSI)
SINC = SIN(THETA)
SINJ = SIN(THETA-ALPHA/2.0)
SINK = SIN(THETA+ALPHA/2.0)
X = INSTANTANEOUS X POSITION OF THE VEHICLE C.G.
XP(I) = X POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
Y = INSTANTANEOUS Y POSITION OF THE VEHICLE C.G.
YP(I) = Y POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
ZP(I) = Z POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM

ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE
B10 = ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE
COSB = COS(B1-PSI)
COSD = COS(B1)
SINB = SIN(B1-PSI)
SIND = SIN(B1)

Figure 1-24. Subroutine STGEOM

1-62

BENDIX PRODUCTS AEROSPACE DIVISION
SUBROUTINE STGEOM

COMMON/ABLOCK/N,K,NOGR(5)
  D(5),SL10(5),DI0(5),B10(5),B20(5),B30(5),C10(5),C20(5),C30(5)
COMMON/CBLOCK/SIND(5),COSC(5),COSD(5),COSD(5),SINC(5),SINJ(5),
  1 COSG(5),SINK(5),COSB(5),SINB(5),SING(5),SINL(5),SINI(5),COSA,SINA
  1 SL2(5),SL3(5),SL1T5,SL2T(5),SL3T(5),SL1TQ(5),SL2TO(5),SL3TO(5),
COMMON/EBLOCK/XP(5),YP(5),ZP(5),XPVEL(5),YPVEL(5),ZPVEL(5)
COMMON/HBLOCK/VVELO(14,8),HVELO(14,8),XVELO(14,8),YVELO(14,8),
  1 XVELOO,YVELO0,PSI,PSIO,PSVEL0,X,YO,XVEL,YVEL,PSI2,VEL,GRAV,
  1 ZETA,KCONMX,DELTAT,GRDMU(5),A(5),FXX(3),FYY(3),RMOO(3),
  1 VMASS,PMASS,FMOMIT,FMOMGR,FMOMI,FA(5),FB(5),FC(5),IP,IO1R,
  1 EGBAL1,EGBAL2,EGBAL3,P1(5),P2(5),P3(5),DEL(5),EPSEN(5),PHI,
  1 ENPRE(5)
COMMON/KBLOCK/M,TIME,TME(402),XPI(402),YPI(402),ZPI(402),XP3(402),
  1 YP3(402),ZP3(402),FXI(402),FX3(402),FXPL3I(402),FYPL3I(402),
  1 FZPL3I(402),FZPL3I(402),FYPL3I(402),FZPL3I(402),G1STRI(402),
  1 G2STRI(402),G3STRI(402),G3STR1(402),G3STR2(402),G3STR3(402),
  1 G1FCR1(402),G1FCR2(402),G1FCR3(402),G2FCR1(402),G2FCR2(402),
  1 G3FCR3(402),G3FCR3(402)

710 DO 730 I=IoN,K
    IF(NOGR(I))730,730,715
    IF FOOTPAD IS ON THE GROUND, DETERMINE STRUT GEOMETRY

715  AA=YP(RI(I))COSC(I)*COSA-D11(I)*SINA
     BB=X-R1(I)*COSC(I)*SINA-D11(I)*COSA
     SL1(I)=SRT (((YP(I)-AA)*(YP(I)-AA)+(BB-XP(I))((BB-XP(I))))
     B1(I)=ATAN (((YP(I)-AA)/(BB-XP(I)))+PSI

    RESET PROGRAM CONSTANTS

     COSB(I)=COS (B1(I)-PSI)
     COSD(I)=COS (B1(I))
     SINB(I)=SIN (B1(I)-PSI)
     SIND(I)=SIN (B1(I))

    CALCULATE NEW STRUT LENGTHS

    720  D1(I)=ZP(I)-R1(I)*SINC(I)
     SL1(T(I))=SRT (SL1(I))**2+D11(I)**2)
     HALF3=(SL1(I)**2+SIND(I)**2+C1(I)**2+COSC(I)**2-R2(I)**2)*COSE(I)
     SL1M(I)=HALF3+HALF3
     SL1V(I)=SRT ((SL1(T(I)**2)+(R1(I)**2-2**(I))**2)*SINC(I)**2)

    Figure 1-24. Subroutine STGEOM (Continued)
1D1(I) = R1(I) * SINC(I) - R2(I) * SINJ(I))
HALF1 = (SL1(I) * SIND(I) + R1(I) * COSC(I) - R2(I) * COSG(I)) * 
1(SL1(I) * SIND(I) + R1(I) * COSC(I) - R2(I) * COSG(I))
HALF2 = (SL1(I) * COSD(I) - D(I)) * (SL1(I) * COSD(I) - D(I))
SL3(I) = SQRT(HALF1 + HALF2)
SL3T(I) = SQRT ((SL3(I) * SL3(I)) + (D1(I) - R2(I) * SINK(I) + R1(I) * 
SINC(I)) * (D1(I) - R2(I) * SINK(I) + R1(I) * SINC(I)))
730 CONTINUE
RETURN
END
TITLE   FOTPAD
AUTHORS R.BLACK, J.CADORET, J.GIBSON  THE BENDIX CORPORATION
DATE   10-25-64
PURPOSE Compute strut forces and forces acting on footpads.
          Determine footpad position and motion.
METHOD Parabolic integration used in determining footpad motion.
CALL FOTPAD
NOTE  This program was written in Fortran IV.
NOTE  This program executes on the Univac 1107 computer.
INPUT Through labeled common.
NOTE An input parameter followed by a 0 implies the initial value
       as B10(I). The instantaneous value of the parameter
       is defined without the 0 as B1(I).

SYMBOL DEFINITION
B20 Angle (projected in plane parallel to vehicle motion)
       between strut No. 2 and vehicle centerline.
B30 Angle (projected in plane parallel to vehicle motion)
       between strut No. 3 and vehicle centerline.
COSA  Cos(PSI).
COSB  Cos(B1-PSI).
COSC  Cos(THETA).
COSD  Cos(B1).
C10+C20+C30 Angle, in plane formed by strut and a normal to
       the direction of motion, between strut and a plane normal
       to the vehicle centerline - for struts 1, 2, 3 respectively.
DELTAP Distance from bottom of footpad to intersection of the
       leg struts.
DELTAT Time increment used in the integration of footpad
       motion. DELTAT = DELTAT/KCONMX.
D10 Initial length (projected in plane normal to direction
       of vehicle motion) of the component (in plane perpendicular
       to the vehicle centerline) of strut No. 1 length.
D11 Vertical distance from vehicle C.G. to upper hardpoint.
EPS2 Program constant equal to 10 percent of the possible
       potential energy which could be stored in a footpad
       as the result of elastic stroking of the upper strut.
EPSN Minimum allowable footpad sliding velocity.
F11 Plastic stroke force for upper strut (No. 1).
F22 Plastic stroke force for lower strut No. 2.
F33 Plastic stroke force for lower strut No. 3.
GRAV Local gravity.
GRDMU Coefficient of friction between vehicle footpads and ground.
K Flag for landing mode. If 1-2-1 landing, K=1. If
       2-2 landing, K=2.
KCONMX Number of iterations of footpad calculations per (DELTAT)
       time interval.
N Number of legs on the vehicle.
PMAS Mass of each footpad.
PSIVEL Instantaneous pitch velocity of the vehicle C.G.

Figure 1-25. Subroutine FOTPAD 1-65
P1   friction force in strut no. 1
P2   friction force in strut no. 2
P3   friction force in strut no. 3
R1   radius of upper hardpoint mounting circle
SINA sin(PSI)
SINB sin(B1-PSI)
SIND sin(B1)
SKE1 tensile elastic springrate of strut no. 1 (upper)
SKE2 tensile elastic springrate of strut no. 2 (lower)
SKE3 tensile elastic springrate of strut no. 3 (lower)
SK5 springrate under vehicle footpads
SK1 compressive elastic springrate of strut no. 1 (upper)
SK2 compressive elastic springrate of strut no. 2 (lower)
SK3 compressive elastic springrate of strut no. 3 (lower)
SL1 projected length of strut no. 1 in plane parallel to
direction of motion
SL1M minimum length to which strut no. 1 has been compressed
during this run
SL1PRE length of strut no. 1 at the end of the previous time
increment
SL1T true instantaneous length of strut no. 1
SL1TO true initial length of strut no. 1
SL2M minimum length to which strut no. 2 has been compressed
during this run
SL2PRE length of strut no. 2 at the end of the previous time
increment
SL2T true instantaneous length of strut no. 2
SL2TO true initial length of strut no. 2
SL3M minimum length to which strut no. 3 has been compressed
during this run
SL3PRE length of strut no. 3 at the end of the previous time
increment
SL3T true instantaneous length of strut no. 3
SL3TO true initial length of strut no. 3
X instantaneous x position of the vehicle C.G.
XVEL instantaneous x velocity of the vehicle C.G.
Y instantaneous y position of the vehicle C.G.
YVEL instantaneous y velocity of the vehicle C.G.
ZETA ground slope

OUTPUT through labeled common

SYMBOL DEFINITION
E3(I) potential energy stored in strut no. 3 of leg i due to
compression or extension of the leg
E2(I) potential energy stored in strut no. 2 of leg i due to
compression or extension of the leg
E3(I) potential energy stored in strut no. 3 of leg i due to
compression or extension of the leg
FA(I) force in the z direction acting on the vehicle C.G. as
the resultant of the strut forces in the three struts
of the i th leg set
FB(I) force, parallel to the vehicle centerline acting on the
vehicle C.G. as the resultant of the strut forces in
the three struts of the i th leg set
FC(I) force, normal to vehicle centerline in the x-y plane
acting on the vehicle C.G. as the resultant of the strut
forces in the three struts of the i th leg set
FX same as FXP
FXP(I) force in the x direction on the footpad (i)

Figure 1-25. Subroutine FOTPAD (Continued)

1-66
FXPLG1 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FXPLG2 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FXPLG3 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FYPLG1 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FYPLG2 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FYPLG3 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FZPLG1 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 1 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FZPLG2 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 2 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
FZPLG3 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 3 DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY THE PARABOLIC INTEGRATION PROCEDURE.
NOGR(I) INDICATES IF FOOTPAD (I) IS MOVING WITH THE VEHICLE. IF NOGR(I) =-1, FOOTPAD IS MOVING WITH THE VEHICLE. IF NOGR(I) =+1, FOOTPAD IS MOVING INDEPENDENTLY MOMENT OF INERTIA OF THE INDIVIDUAL VEHICLE FOOTPADS. THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE GROUND AT THE INSTANT UNDER INVESTIGATION.
XP(I) X POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
YP(I) Y POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
ZP(I) Z POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
XPVEL(I) VELOCITY OF THE I TH FOOTPAD IN THE X DIRECTION IN THE FIXED COORDINATE SYSTEM
YPVEL(I) VELOCITY OF THE I TH FOOTPAD IN THE Y DIRECTION IN THE FIXED COORDINATE SYSTEM
ZPVEL(I) VELOCITY OF THE I TH FOOTPAD IN THE Z DIRECTION IN THE FIXED COORDINATE SYSTEM.

SUBROUTINE FOTPAD

COMMON/ABLOCK/N,K, NOGR(5)
COMMON/EBLOCK/XP(5), YP(5), ZP(5), XVEL(5), YVEL(5), ZVEL(5)

Figure 1-25. Subroutine FOTPAD (Continued)
1 F1(5), F2(5), F3(5), E1(5), E2(5), E3(5)
COMMON/GBLOCK/F11(5), F22(5), F33(5), SKS(5), SK1, SK2, SK3, SKE1, SKE2, 
1 SKE3
COMMON/HBLCK/BVELO(14,8), HVELO(14,8), XVELO(14,8), YVELO(14,8), 
1 XVLOO= YVLOO= PSI*PSIO*PSVELO*X+XO*Y+YO*YVLO+PSIVEL*GRAV. 
1 ZETA= KCONMX+DELT1+DELT1+GROMU(5)+A(5)+FXX(3)+FYX(3)+RMOD(3)
1 VMSS*RMSS*RMOMR, RMOMG, RMOMF, FB(5), FC(5), IP, IQ, IR,
1 EGBAL1, EGBAL2, EGBAL3, P1(5), P2(5), P3(5), DEL(5), EPSN(5), PHI
1 ENPRO(5)
DIMENSION PADENG(5), FXP(5)
280 DO 600 I=1, N, K
    TEST IF FOOTPAD IS ON THE GROUND
    IF(XP(I) - DELTAP)330, 330, 290
    COMPUTE FOOTPAD VELOCITY RELATIVE TO C.G. AND DETERMINE RELATIVE
    ENERGY OF FOOTPAD
    290 XPVELR = XVEL + D11(I)*SINA*PSIVEL - R1(I)*COSA*PSIVEL - SL1(I)*
        1 SINB(I)*PSIVEL
        YPVELR = YVEL + D11(I)*COSA*PSIVEL - R1(I)*SINA*PSIVEL - SL1(I)*
        1 COSB(I)*PSIVEL
        PADENG(I) = PMASS/2.0*(XPVELR - XPVEL(I))*(XPVELR - XPVEL(I)) + 
        (YPVELR - YPVEL(I))*(YPVELR - YPVEL(I)) + (ZPVEL(I))*(ZPVEL(I)) + 
        E1(I) + E2(I) + 
        E3(I)
    TEST IF FOOTPAD IS MOVING WITH THE VEHICLE
    IF(PADENG(I) - EPSN(I))300, 300, 330
    FOOTPAD IS MOVING WITH THE VEHICLE. SET FORCES EQUAL TO ZERO
    300 FA(I) = 0.0
        FB(I) = 0.0
        FC(I) = 0.0
        FXPLG1(I) = 0.0
        FXPLG2(I) = 0.0
        FXPLG3(I) = 0.0
        FYPLG1(I) = 0.0
        FYPLG2(I) = 0.0
        FYPLG3(I) = 0.0
        FYPLG3(I) = 0.0
        RMOGR = RMOGR + 2.0*PMASS*(X-XP(I))*(X-XP(I)) + 
        (Y-YP(I))*(Y-YP(I))
        NOGR(I) = -1
    FIND FOOTPAD POSITIONS AND VELOCITY BY RIGID BODY MOTION WITH
    THE VEHICLE
    310 XP(I) = X - D11(I)*COSA - R1(I)*COSB(I)
        YP(I) = Y - D11(I)*SINA + R1(I)*SINB(I)
        ZP(I) = R1(I)*SINB(I) + D11(I)
        XPVEL(I) = XVEL + D11(I)*SINA*PSIVEL - R1(I)*COSA*PSIVEL - 
        SL1(I)*SINB(I)*PSIVEL
        XPVEL(I) = XVEL + D11(I)*COSA*PSIVEL - R1(I)*SINA*PSIVEL - 
        SL1(I)*COSB(I)*PSIVEL
        XPVEL(I) = 0.0
        FX(I) = 0.0
        GO TO 600

Figure 1-25. Subroutine FOTPAD (Continued)
FOOTPAD IS MOVING INDEPENDENTLY OF THE VEHICLE • CALCULATE STRUT FORCES AND FOOTPAD ENERGY

330 SL=SL1T(1)-SL1T(1)
  NOGR(1)=I
  IF(SL-SL1M(1)+F11(I)/SK1)<334.334.375
  334 IF(SL)<340.367.367
  340 IF(SL-SL1PRE(I))<343.360.360
  343 F1(I)=SKE1*SL-P1(I)
  E1(I)=SKE1*SL/2.0
  GO TO 393
  350 F1(I)=-P1(I)
  E1(I)=0.0
  GO TO 393
  360 F1(I)=SKE1*SL+P1(I)
  E1(I)=SKE1*SL/2.0
  GO TO 393
  367 IF(SL-SL1PRE(I))<350.370.370
  370 F1(I)=P1(I)
  E1(I)=0.0
  GO TO 393
  375 IF(SL-SL1M(1))<378.390.390
  378 IF(SL-SL1PRE(I))<385.380.380
  380 F1(I)=SK1*SL-SL1M(1)+F11(I)/SK1-P1(I)
  E1(I)=(SK1/2.0)*(SL-SL1M(1)+F11(I)/SK1)*F11(I)/SK1
  GO TO 393
  385 F1(I)=SK1*SL-SL1M(1)+F11(I)/SK1-P1(I)
  E1(I)=(SK1/2.0)*(SL-SL1M(1)+F11(I)/SK1)*F11(I)/SK1
  GO TO 393
  390 F1(I)=F11(I)+P1(I)
  E1(I)=SKE1*SL/2.0
  GO TO 393
  393 IF(SL-SL1M(1))<400,400.396
  396 SL1M(I)=SL
  400 SL1PRE(I)=SL
  SL=SL2T(1)-SL2T(1)
  IF(SL-SL2M(I)+F22(I)/SK2)<403.403.433
  403 IF(SL)<406.425.425
  406 IF(SL-SL2PRE(I))<410.420.420
  410 F2(I)=SKE2*SL-P2(I)
  E2(I)=SKE2*SL/2.0
  GO TO 450
  412 F2(I)=-P2(I)
  E2(I)=0.0
  GO TO 450
  420 F2(I)=SKE2*SL+P2(I)
  E2(I)=SKE2*SL/2.0
  GO TO 450
  425 IF(SL-SL2PRE(I))<412.430.430
  430 F2(I)=P2(I)
  E2(I)=0.0
  GO TO 450
  433 IF(SL-SL2M(I))<436.446.446
  436 IF(SL-SL2PRE(I))<443.440.440
  440 F2(I)=SK2*SL-SL2M(I)+F22(I)/SK2-P2(I)
  E2(I)=(SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)*F22(I)/SK2
  GO TO 450
  443 F2(I)=SK2*SL-SL2M(I)+F22(I)/SK2-P2(I)
  E2(I)=(SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)*F22(I)/SK2
  GO TO 450
  446 F2(I)=F22(I)+P2(I)

Figure 1-25. Subroutine FOTPAD (Continued)
E2(I) = (SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)*(SL-SL2M(I)+F22(I)/SK2)
450 IF(SL-SL2M(I))454,454,452
452 SL2M(I) = SL
454 SL2PRE(I) = SL
   SL=SL3TO(I)-SL3T(I)
456 IF(SL-SL3M(I)+F33(I)/SK3)458,458,486
458 IF(SL)460,480,480
460 IF(SL-SL3PRE(I))465,475,475
465 F3(I)=SK3*SL-P3(I)
   E3(I)=SK3*SL*SL/2.0
   GO TO 505
470 F3(I)=-P3(I)
   E3(I)=0.0
   GO TO 505
475 F3(I)=SK3*SL+P3(I)
   E3(I)=SK3*SL*SL/2.0
   GO TO 505
480 IF(SL-SL3PRE(I))470,483,483
483 F3(I)=P3(I)
   E3(I)=0.0
   GO TO 505
486 IF(SL-SL3M(I))490,500,500
490 IF(SL-SL3PRE(I))496,493,493
493 F3(I)=SK3*(SL-SL3M(I)+F33(I)/SK3)+P3(I)
   E3(I)=SK3/2.0*(SL-SL3M(I)+F33(I)/SK3)*(SL-SL3M(I)+F33(I)/SK3)
   GO TO 505
496 F3(I)=SK3*(SL-SL3M(I)+F33(I)/SK3)-P3(I)
   E3(I)=SK3/2.0*(SL-SL3M(I)+F33(I)/SK3)*(SL-SL3M(I)+F33(I)/SK3)
   GO TO 505
500 F3(I)=F33(I)+P3(I)
   E3(I)=SK3/2.0*(SL-SL3M(I)+F33(I)/SK3)*(SL-SL3M(I)+F33(I)/SK3)
505 IF(SL-SL3M(I))510,510,507
507 SL3M(I) = SL
510 SL3PRE(I) = SL

CALCULATE COMPONENTS OF STRUT FORCE

SING(I)=SIN (C1(I))
SINL(I)=SIN (C2(I))
SINI(I)=SIN (C3(I))
FA(I)=F1(I)*COS (C1(I)) F2(I)*COS (C2(I)) F3(I)*COS (C3(I))
FB(I)=F1(I)*SING(I)*COSD(I)+F2(I)*SINL(I)*COS (B2(I)) F3(I)*
   SINI(I)*COS (B3(I))
FC(I)=F1(I)*SING(I)*SIND(I)+F2(I)*SINL(I)*SIN (B2(I)) F3(I)*
   SINI(I)*SIN (B3(I))

DETERMINE FORCES ACTING ON FOOTPAD AND FOOTPAD MOTIONS

514 KCON=0
515 KCON=KCON+1
520 IF(SLVEL-EP521530)530,535
530 SLVEL=EPS2
535 FXP(I)=SKS(I)*(DELTAP-XP(I))
540 FX(I)=0.0
545 TEMP2=GRDMU(1)*FX(I)/(PMASS*SLVEL)
550 FYPLG3(I)=(FC(I)*COSA-FB(I)*SINA)/PMASS+GRAV*SIN (ZETA(I))*PMASS
   YP(I)=YP(I)+YPVEL(I)*DELTTT+(19.0*FYPLG3(I)-10.0*FYPLG2(I)+3.0*

Figure 1-25. Subroutine FOTPAD (Continued)
IFYPLGI(1))*DELTIT*DELTIT/(24.0*PMASS)-TEMP2*YPVEL(1)*DELTIT* 
2DELTIT/2.0 
YPVEL(I)=YPVEL(I)+(23.0*FYPLG3(I)-16.0*FYPLG2(I)+5.0*FYPLG1(I))* 
1DELTIT/(12.0*PMASS)-TEMP2*YPVEL(I)*DELTIT* 
FZPLG3(I)=FA(I) 
580 ZP(I)=ZP(I)+ZPVEL(I)*DELTIT+(19.0*FZPLG3(I)-10.0*FZPLG2(I)+3.0* 
1FZPLG1(I))*DELTIT*DELTIT/(24.0*PMASS)-TEMP2*ZPVEL(I)*DELTIT* 
2DELTIT/2.0 
ZPVEL(I)=ZPVEL(I)+(23.0*FZPLG3(I)-16.0*FZPLG2(I)+5.0*FZPLG1(I))* 
1DELTIT/(12.0*PMASS)-TEMP2*ZPVEL(I)*DELTIT* 
FXPLG3(I)=-FB(I)*COSA-FC(I)*SINA-GRAV*COS (ZETA)*PMASS* 
XP(I)=XP(I)+XPVEL(I)*DELTAT+(19.0*FXPLG3(I)-10.0*FXPLG2(I)+3.0* 
1FXPLG1(I))*DELTIT*DELTIT/(24.0*PMASS)+FX(I)*DELTIT*DELTIT/ 
2(2.0*PMASS) 
XPVEL(I)=XPVEL(I)+(23.0*FXPLG3(I)-16.0*FXPLG2(I)+5.0*FXPLG1(I))* 
1DELTIT/(12.0*PMASS)+FX(I)*DELTIT/PMASS 
590 IF(KCON-KCONMX) 515,60C*600 
600 CONTINUE 
C 
DO 639 I=1,N,K 
IF(NEG(I))639,639,638 
638 FXPLG1(I)=FXPLG2(I) 
FXPLG2(I)=FXPLG3(I) 
FYPLG1(I)=FYPLG2(I) 
FYPLG2(I)=FYPLG3(I) 
FZPLG1(I)=FZPLG2(I) 
FZPLG2(I)=FZPLG3(I) 
639 CONTINUE 
RETURN 
END 

Figure 1-25. Subroutine FOTPAD (Concluded)
DETERMINE FORCES ACTING ON VEHICLE

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

THROUGH LABELED COMMON

SYMMETRY FACTOR -- IF TWO VEHICLE LEGS ARE SYMMETRIC, SET A=2.0 FOR ONE OF THE SYMMETRIC PAIRS AND A=0.0 FOR THE OTHER TO SAVE ON COMPUTER TIME IE FOR 22 LANDING A=2.0,2.0 FOR A 121 LANDING A= 1,2,1,0

COS(PSI) COS(THETA) COS(B1)

VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT FORCE, PARALLEL TO THE VEHICLE CENTERLINE ACTING ON THE VEHICLE C.G. AS THE RESULTANT OF THE STRUT FORCES IN THE THREE STRUTS OF THE I TH LEG SET


LOCAL GRAVITY

VEHICLE MOMENT OF INERTIA

MOMENT OF INERTIA OF THE INDIVIDUAL VEHICLE FOOTPADS. THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE GROUND AT THE INSTANT UNDER INVESTIGATION

RADIUS OF UPPER HARDPOINT MOUNTING CIRCLE

PROJECTED LENGTH OF STRUT NO.1 IN PLANE PARALLEL TO DIRECTION OF MOTION

VEHICLE MASS

GROUND SLOPE

FORCES ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FORCES ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FORCES ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FORCES ACTING IN THE Y DIRECTION ON THE VEHICLE C.G. DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED

VEHFOR

NOTE THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

VEHFOR

INPUT THROUGH LABELED COMMON

SYMBOL DEFINITION

FXX(3) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(2) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(1) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G. DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(3) FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G. DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED

VEHFOR

1-72
IN THE PARABOLIC INTEGRATION PROCEDURE
FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE CURRENT TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-1 TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-2 TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

TOTAL MOMENT OF INERTIA OF THE VEHICLE MASS AND THOSE
FOOTPADS WHICH ARE OFF THE GROUND

SUBROUTINE VEHFOR

COMMON/ABLOCK/N,K,NOGR(5)
1 D(5),SL10(5),D10(5),B10(5),B20(5),B30(5),C10(5),C20(5),C30(5)
COMMON/CBLOCK/SIND(5),COSC(5),COSE(5),COSD(5),SINC(5),SIND(5)
1 COSG(5),SINK(5),COSB(5),SINB(5),SING(5),SINK(5),COSA,SINA
COMMON/HBLOCK/VVELO(14,8),XVELO(14,8),YVELO(14,8),ZVELO(14,8),
1 XVELOO,YVELOO,PSII,PSIVO,PSVT,PSVTO,PSVT,PSVTO,PSVT,PSVTO,PSVT
1 ZETA,KCONM,DELTTT,DELTAT,GRDMU(5),A(5),FXX(3),FYY(3),RMOO(3),
1 VMAS,PMAS,RMOMIT,RMOMGR,RMOMI,FA(5),FB(5),FC(5),IP,IR,
1 EGBAL1,EGBAL2,EGBAL3,P1(5),P2(5),P3(5),DEL(5),EPSEN(5),PHI,
1 ENPRO(5)

DETERMINE FORCES ACTING ON THE VEHICLE

FXX(3) = 0.0
FYY(3) = 0.0
RMOO(3) = 0.0
RMOMIT = RMOM1 + RMOMGR

610 RMOMGR = 0.0
DO 620 I = 1,N*K
FXX(3) = FXX(3) + A(1)*FB(1)*COSA + A(1)*FC(1)*SINA
FYY(3) = FYY(3) + A(1)*FB(1)*SINA - A(1)*FC(1)*COSA

620 RMOO(3) = RMOO(3) + A(1)*FC(1)*D11(I) + SL1(I)*COSD(1) - A(1)*FB(1)*
1 (R1(I)*COSC(I) + SL1(I)*SIND(I))
FXX(3) = FXX(3) - VMAS*GRAV*COS(ZETA)
FYY(3) = FYY(3) - VMAS*GRAV*SIN(ZETA)
RETURN
END

Figure 1-26. Subroutine VEHFOR (Concluded)
VEHMOT

TITL

AUTHORS R. BLACK, J. CADORET, J. GIBSON

DATE 10-25-64

PURPOSE DETERMINE VEHICLE C.G. MOTION

METHOD PARABOLIC INTEGRATION USED IN DETERMINING VEHICLE MOTION

NOTE THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

INPUT THROUGH LABELED COMMON

SYMBOL DEFINITION

B1 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE

DELTAT TIME INTERVAL BETWEEN PROGRAM CALCULATIONS

FXX(3) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(2) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FXX(1) FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(3) FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE CURRENT TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(2) FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-1 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

FYY(1) FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
DURING THE N-2 TIME INTERVAL. THIS FORCE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

PSI INSTANTANEOUS VALUE OF PSI0

PSIVEL INSTANTANEOUS PITCH VELOCITY OF THE VEHICLE C.G.

RM00(3) TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE CURRENT TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

RM00(2) TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-1 TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

RM00(1) TORQUE ACTING ON THE VEHICLE C.G. IN THE X-Y PLANE
DURING THE N-2 TIME INTERVAL. THIS TORQUE IS USED
IN THE PARABOLIC INTEGRATION PROCEDURE

RM0MIT TOTAL MOMENT OF INERTIA OF THE VEHICLE MASS AND THOSE
FOOTPADS WHICH ARE OFF THE GROUND

X INSTANTANEOUS X POSITION OF THE VEHICLE C.G.

XVEL INSTANTANEOUS X VELOCITY OF THE VEHICLE C.G.

Y INSTANTANEOUS Y POSITION OF THE VEHICLE C.G.

YVEL INSTANTANEOUS Y VELOCITY OF THE VEHICLE C.G.

VMASS VEHICLE MASS

Figure 1-27. Subroutine VEHMOT
Determine vehicle C.G. motions

\[ x = x + x_{VEL} \Delta t + (19.0 F_{XX}(3) - 10.0 F_{XX}(2) + 3.0 F_{XX}(1)) \Delta t / (24.0 V_{MASS}) \]
\[ y = y + y_{VEL} \Delta t + (19.0 F_{YY}(3) - 10.0 F_{YY}(2) + 3.0 F_{YY}(1)) \Delta t / (12.0 V_{MASS}) \]

Reset program constants

\[ \cos(a) = \cos(PSI) \]
\[ \sin(a) = \sin(PSI) \]
\[ \cos(b) = \cos(B1-PSI) \]
\[ \sin(b) = \sin(B1-PSI) \]

Determine horizontal and vertical forces at the vehicle C.G.

\[ F_{VV} = F_{XX}(3) \cos(a) + F_{YY}(3) \sin(a) \]
\[ F_{LL} = F_{YY}(3) \cos(a) - F_{XX}(3) \sin(a) \]

Index previous forces for the next integration of vehicle motion

Figure 1-27. Subroutine VEHMOT (Continued)
C
DO 637 I=1,2
   FXX(I)=FXX(I+1)
   FYY(I)=FYY(I+1)
   RM00(I)=RM00(I+1)
RETURN
END
SUBROUTINE STABAN

COMMON/ABLOCK/N,K,NOGR(5)
COMMON/EBLOCK/XP(5),YP(5),XPVEL(5),YPVEL(5),ZPVEL(5)
COMMON/HBLOCK/VVELO(14,8),HVELO(14,8),XVELO(14,8),YVELO(14,8),
  1 XVELOC,YVELOC,PXI,PXO,XPVEL0,X,Y,XPVEL,YVEL,PSIVEL,GRAV,
  1 ZETA,KCONMX,DELTTT,DELTAT,GRDMU(5),A(5),FXX(3),FYY(3),RMOO(3),
  1 VMASS,PMASS,ROMIT,ROMGRI,ROMGI,FA(5),FB(5),FC(5),IP,IO,IR,
  1 EGBAL1,EGBAL2,EGBAL3,P1(5),P2(5),P3(5),DEL(5),EPSEN(5),PHI,
  1 ENPRO(5)
COMMON/IBLOCK/BETA(5),BEPR(5),BETAPR(5),LAND
DIMENSION BEMI(5),BEPL(5)

ORIENT STABILITY ANGLE IN PROPER QUADRANT

DO 673 I=1,N,K
BETA(I)=ATAN ((YP(I)-Y)/(X-XP(I)))-ZETA
BEMI(I)=BETA(I)-3.14159265359
BEPL(I)=BETA(I)+3.14159265359
DIFF1=ABS (BEMI(I)-BEPR(I))
DIFF2=ABS (BETAPR(I)-BEMI(I))
DIFF3=ABS (BEPL(I)-BEPR(I))
IF (DIFF1>DFF1) GO TO 672
IF (DIFF2>DFF2) GO TO 671
IF (DIFF3>DFF3) GO TO 670
670 BETA(I)=BEPL(I)
GO TO 673
671 BETA(I)=BEMI(I)
672 BETA(I)=BEPR(I)
673 CONTINUE

Figure 1-28. Subroutine STABAN
RESET BEPR FOR NEXT ITERATION STEP

BEPR(I)=BETA(I)
RETURN
END

Figure 1-28. Subroutine STABAN (Concluded)
Figure 1-29. Subroutine STOREP
SUBROUTINE STOREP

E1PREV(5),C2PREV(5),C3PREV(5),EPS2,EPS3,EPS4,EPS5

COMMON/EBLOCK/XP(5),YP(5),ZP(5),XPVEL(5),YPVEL(5),ZPVEL(5),

COMMON/FBLOCK/FX(5),FYPLG3(5),FZPLG3(5),FXPLG2(5),FYPLG2(5),
FZPLG2(5),FXPLG1(5),FYPLG1(5),FZPLG1(5),

COMMON/JBLOCK/BETADT,FLL,FVV,LINE,FINT,RUNNO,RUNNOO,SERNO,XN,

COMMON/KBLOCK/MPTIME,TME(k02),XPI(402),YPI(402),ZPI(402),XP3(402),
YP3(402),ZP3(402),FX1(402),FX3(402),FXPL31(402),FYPL31(402),
FZPL31(402),FXPL33(402),FYPL33(402),FZPL33(402),GISTR1(402),
GISTR2(402),GISTR3(402),G3STR1(402),G3STR2(402),G3STR3(402),
GIFRC1(402),GIFRC2(402),GIFRC3(402),G3FRC1(402),G3FRC2(402),
G3FRC3(402)

STORE VARIABLES FOR THE PRINT TO BE MADE AT THE END OF THE RUN

M=M+1
TME(M)=TIME
XP1(M)=XP(I1)
YP1(M)=YP(I1)
ZP1(M)=ZP(I1)
XP3(M)=XP(JJ)
YP3(M)=YP(JJ)
ZP3(M)=ZP(JJ)
FX1(M)=FX(I1)
FX3(M)=FX(JJ)
FXPL31(M)=FXPLG3(I1)
FYPL31(M)=FYPLG3(I1)
FZPL31(M)=FZPLG3(I1)
FXPL33(M)=FXPLG3(JJ)
FYPL33(M)=FYPLG3(JJ)
FZPL33(M)=FZPLG3(JJ)
GISTR1(M)=SL1T(I1)
GISTR2(M)=SL2T(I1)
GISTR3(M)=SL3T(I1)
G3STR1(M)=SL1T(JJ)
G3STR2(M)=SL2T(JJ)
G3STR3(M)=SL3T(JJ)
GIFRC1(M)=F1(I1)
GIFRC2(M)=F2(I1)
GIFRC3(M)=F3(I1)

Figure 1-29. Subroutine STOREP (Continued)
G1FRC2(M) = F2(IJ)
G1FRC3(M) = F3(IJ)
G3FRC1(M) = F1(JJ)
G3FRC2(M) = F2(JJ)
G3FRC3(M) = F3(JJ)
RETURN
END

Figure 1-29. Subroutine STOREP (Concluded)
TO PRINT THE OUTPUT DATA

PRINT I (IPR)

IPR=1 --- PRINT INITIAL DATA
IPR=2 --- PRINT ON LINE OUTPUT WITH HEADINGS
IPR=3 --- PRINT ON LINE OUTPUT
IPR=4 --- PRINT SUMMARY (UNSTABLE RUN)
IPR=5 --- PRINT SUMMARY (TIME EXCEEDED)
IPR=6 --- PRINT SUMMARY (STABLE RUN)
IPR=7 --- PRINT GENERAL SUMMARY
IPR=8 --- PRINT STORED OUTPUT

NOTE

THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

SYMBOL DEFINITION

NOTE

AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE
AS BI0(I). THE INSTANTANEOUS VALUE OF THE PARAMETER
IS DEFINED WITHOUT THE 0 AS BI(I).

A SYMMETRY FACTOR -- IF TWO VEHICLE LEGS ARE SYMMETRIC,
SET A=2.0 FOR ONE OF THE SYMMETRIC PAIRS AND A=0.0 FOR
THE OTHER TO SAVE ON COMPUTER TIME IE FOR 22 LANDING
A=2.0 2.0
FOR A 121 LANDING A=1.2 1.0

ALPHA
ANGLE (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE)
SUBTENDED BY THE LOWER HARDPOINTS AND VEHICLE C.G.

BETA
VEHICLE STABILITY ANGLE

BETADT
RATE OF CHANGE OF BETA WITH TIME

BIIMIN
MINIMUM STABILITY ANGLE FOR LEG SET II

BJMIN
MINIMUM STABILITY ANGLE FOR LEG SET JJ

B10
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE

B20
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 2 AND VEHICLE CENTERLINE

B30
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 3 AND VEHICLE CENTERLINE

C10, C20, C30
ANGLE IN PLANE FORMED BY STRUT AND A NORMAL TO
THE DIRECTION OF MOTION, BETWEEN STRUT AND A PLANE NORMAL
TO THE VEHICLE CENTERLINE -- FOR STRUTS 1, 2, 3 RESPECTIVELY

D
VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS

DELTAP
DISTANCE FROM BOTTOM OF FOOTPAD TO INTERSECTION OF THE
LEG STRUTS

DELTAT
TIME INTERVAL BETWEEN PROGRAM Calculations

D10
INITIAL LENGTH (PROJECTED IN PLANE NORMAL TO DIRECTION
OF VEHICLE MOTION) OF THE COMPONENT (IN PLANE PERPENDICULAR
TO THE VEHICLE CENTERLINE) OF STRUT NO 1 LENGTH

D11
VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT

EGBAL1
ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND C.G.
DROP

EGBAL2
ENERGY DISSIPATED BASED ON PLASTIC STROKE OF ALL STRUTS

EGBAL3
ENERGY DISSIPATED BASED ON PLASTIC AND FULL ELASTIC

Figure 1-30. Subroutine PRINT1
STROKE OF ALL STRUTS

ENPRO(I) PERCENT OF TOTAL ENERGY ABSORBED BY STROKING OF THE

STRUTS OF LEG SET I

EPS2

EPS3 LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN X DIRECTION

EPS4 LIMITING MINIMUM VELOCITY OF VEHICLE C.G. IN Y DIRECTION

EPS5 LIMITING MINIMUM ANGULAR VELOCITY OF VEHICLE C.G.

NOTE

FINT MAXIMUM ALLOWABLE TIME FOR COMPUTER RUN

FLL TOTAL FORCE ACTING ON THE VEHICLE IN THE LATERAL DIRECTION

(FOR NORMAL TO THE VEHICLE CENTERLINE)

FVV TOTAL FORCE ACTING ON THE VEHICLE IN THE VERTICAL

DIRECTION (PARALLEL TO THE VEHICLE CENTERLINE)

FX1 FX VALUE FOR LEG II STORED FOR FINAL PRINT

FX3 FX VALUE FOR LEG JJ STORED FOR FINAL PRINT

FXPL31 FXPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT

FXPL33 FXPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT

FYPL31 FYPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT

FYPL33 FYPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT

F11 PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)

F22 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2

F33 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3

FZPL31 FZPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT

FZPL33 FZPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT

GRAV LOCAL GRAVITY

GRDMU COEFFICIENT OF FRICTION BETWEEN VEHICLE FOOTPADS AND GROUND

G1FRC1 F1 VALUE FOR LEG II STORED FOR FINAL PRINT

G1FRC2 F2 VALUE FOR LEG II STORED FOR FINAL PRINT

G1FRC3 F3 VALUE FOR LEG II STORED FOR FINAL PRINT

G3FRC1 F1 VALUE FOR LEG JJ STORED FOR FINAL PRINT

G3FRC2 F2 VALUE FOR LEG JJ STORED FOR FINAL PRINT

G3FRC3 F3 VALUE FOR LEG JJ STORED FOR FINAL PRINT

G1STR1 SL1T VALUE FOR LEG II STORED FOR FINAL PRINT

G1STR2 SL2T VALUE FOR LEG II STORED FOR FINAL PRINT

G1STR3 SL3T VALUE FOR LEG II STORED FOR FINAL PRINT

G3STR1 SL1T VALUE FOR LEG JJ STORED FOR FINAL PRINT

G3STR2 SL2T VALUE FOR LEG JJ STORED FOR FINAL PRINT

G3STR3 SL3T VALUE FOR LEG JJ STORED FOR FINAL PRINT

H DISTANCE FROM THE BOTTOM OF THE FOOTPAD TO THE VEHICLE

CENTER OF GRAVITY

HN VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND THE LOWEST

POINT ON THE NOZZLE CONE

HVELO HORIZONTAL VELOCITY PERPENDICULAR TO THE GRAVITATIONAL

FIELD

II PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE

PRINTED AS OUTPUT

IP PRINT INDICATOR

IQ PRINT INDICATOR

IR PRINT INDICATOR

J INDEX ASSOCIATED WITH FIRST LEG TO STRIKE GROUND DURING

INITIAL IMPACT

JJ SIMILAR TO II

K FLAG FOR LANDING MODE IF 1-2-1 LANDING , K=1 IF

2-2 LANDING, K=2

KCONMX NUMBER OF ITERATIONS OF FOOTPAD CALCULATIONS PER (DELTAT)

TIME INTERVAL

KPRINT COMPUTATION INCREMENTS BETWEEN PRINTOUT INTERVALS

Figure 1-30. Subroutine PRINT1 (Continued)
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAND</td>
<td>Signifies landing mode - set for 22 or 121 landing</td>
</tr>
<tr>
<td>LINE</td>
<td>PRINTOUT LINE COUNTER</td>
</tr>
<tr>
<td>LOTPRT</td>
<td>Flag to determine if complete summary printing is to be desired. Complete output printing is not normally used therefore set LOTPRT=0. If set LOTPRT=1, the complete history of strut strokes, and strut forces will be printed.</td>
</tr>
<tr>
<td>M</td>
<td>Index used to store variables for print at end of run</td>
</tr>
<tr>
<td>N</td>
<td>Number of legs on the vehicle</td>
</tr>
<tr>
<td>NGO</td>
<td>Starting column in velocity input array</td>
</tr>
<tr>
<td>NGOX</td>
<td>Ending column in velocity array</td>
</tr>
<tr>
<td>NOTE</td>
<td>See writeup for discussion of velocity array</td>
</tr>
<tr>
<td>NSO</td>
<td>Starting row in velocity input array</td>
</tr>
<tr>
<td>NSMAX</td>
<td>Ending row in velocity array</td>
</tr>
<tr>
<td>PHI</td>
<td>Angle between vehicle centerline and gravity vector</td>
</tr>
<tr>
<td>PMASS</td>
<td>Mass of each footpad</td>
</tr>
<tr>
<td>PSI</td>
<td>Initial pitch angle</td>
</tr>
<tr>
<td>PSVELO</td>
<td>Initial vehicle pitch rate</td>
</tr>
<tr>
<td>P1</td>
<td>Friction force in strut No. 1</td>
</tr>
<tr>
<td>P2</td>
<td>Friction force in strut No. 2</td>
</tr>
<tr>
<td>P3</td>
<td>Friction force in strut No. 3</td>
</tr>
<tr>
<td>RMOM1</td>
<td>Vehicle moment of inertia</td>
</tr>
<tr>
<td>RM00(3)</td>
<td>Torque acting on the vehicle C.G. in the X-Y plane during the current time interval. This torque is used in the parabolic integration procedure.</td>
</tr>
<tr>
<td>RM00(2)</td>
<td>Torque acting on the vehicle C.G. in the X-Y plane during the N-1 time interval. This torque is used in the parabolic integration procedure.</td>
</tr>
<tr>
<td>RM00(1)</td>
<td>Torque acting on the vehicle C.G. in the X-Y plane during the N-2 time interval. This torque is used in the parabolic integration procedure.</td>
</tr>
<tr>
<td>RN</td>
<td>Exhaust nozzle radius</td>
</tr>
<tr>
<td>RP(1)</td>
<td>Radius of footpad (1)</td>
</tr>
<tr>
<td>PSI</td>
<td>Instantaneous value of PSIO</td>
</tr>
<tr>
<td>PSIVEL</td>
<td>Instantaneous pitch velocity of the vehicle C.G.</td>
</tr>
<tr>
<td>RUNNOO</td>
<td>Run number (for identification only)</td>
</tr>
<tr>
<td>R1</td>
<td>Radius of upper hardpoint mounting circle</td>
</tr>
<tr>
<td>R2</td>
<td>Radius of lower hardpoint mounting circle</td>
</tr>
<tr>
<td>SERNO</td>
<td>Series number (for identification only)</td>
</tr>
<tr>
<td>SKE1</td>
<td>Tensil elastic springrate of strut No. 1 (upper)</td>
</tr>
<tr>
<td>SKE2</td>
<td>Tensil elastic springrate of strut No. 2 (lower)</td>
</tr>
<tr>
<td>SKE3</td>
<td>Tensil elastic springrate of strut No. 3 (lower)</td>
</tr>
<tr>
<td>SKS</td>
<td>Springrate under vehicle footpads</td>
</tr>
<tr>
<td>SK1</td>
<td>Compressive elastic springrate of strut No. 1 (upper)</td>
</tr>
<tr>
<td>SK2</td>
<td>Compressive elastic springrate of strut No. 2 (lower)</td>
</tr>
<tr>
<td>SK3</td>
<td>Compressive elastic springrate of strut No. 3 (lower)</td>
</tr>
<tr>
<td>SL1M</td>
<td>Minimum length to which strut No. 1 has been compressed during this run</td>
</tr>
<tr>
<td>SL10</td>
<td>Initial value of SL1</td>
</tr>
<tr>
<td>SL2M</td>
<td>Minimum length to which strut No. 2 has been compressed during this run</td>
</tr>
<tr>
<td>SL3M</td>
<td>Minimum length to which strut No. 3 has been compressed during this run</td>
</tr>
<tr>
<td>THETA</td>
<td>Angle between plane parallel to vehicle centerline in direction of vehicle motion and plane through vehicle centerline and upper hardpoint</td>
</tr>
<tr>
<td>TIME</td>
<td>Time after touchdown</td>
</tr>
<tr>
<td>TME</td>
<td>Time stored for final print</td>
</tr>
<tr>
<td>TMINBI</td>
<td>Time when minimum stability for leg set II occurs</td>
</tr>
<tr>
<td>TMINBJ</td>
<td>Time when minimum stability for leg set JJ occurs</td>
</tr>
<tr>
<td>TMINXN</td>
<td>Time when minimum nozzle clearance occurs</td>
</tr>
</tbody>
</table>

Figure 1-30. Subroutine PRINT1 (Continued)
VMASS  VEHICLE MASS
VVELO  INITIAL VEHICLE VELOCITY PARALLEL TO THE GRAVITATIONAL
FIELD. POSITIVE DOWNWARD
X   INSTANTANEOUS X POSITION OF THE VEHICLE C.G.
XP1   XP VALUE FOR LEG II STORED FOR FINAL PRINT
XNMIN  MINIMUM VALUE OF NOZZLE CLEARANCE
XP3   XP VALUE FOR LEG JJ STORED FOR FINAL PRINT
XVEL  INSTANTANEOUS X VELOCITY OF THE VEHICLE C.G.
XVELO  INITIAL VERTICAL VELOCITY OF VEHICLE C.G.
XVELO0  SAME AS XVELO
Y   INSTANTANEOUS Y POSITION OF THE VEHICLE C.G.
YO   INITIAL HORIZONTAL POSITION OF VEHICLE C.G.
YP1   YP VALUE FOR LEG II STORED FOR FINAL PRINT
YP3   YP VALUE FOR LEG JJ STORED FOR FINAL PRINT
YVEL  INSTANTANEOUS Y VELOCITY OF THE VEHICLE C.G.
YVELO  INITIAL HORIZONTAL VELOCITY OF VEHICLE C.G.
YVELOO  SAME AS YVELO
ZETA  GROUND SLOPE
ZP1   ZP VALUE FOR LEG II STORED FOR FINAL PRINT
ZP3   ZP VALUE FOR LEG JJ STORED FOR FINAL PRINT
OUTPUT  PRINTED OUTPUT

SUBROUTINE PRINT1(IPR)
COMMON/ABLOCK/NK,NOGR(5)
COMMON/BBLOCK/H,DELTAP,D11(5),R1(5),R2(5),RP(5),THETA(5),ALPHA(5),
SL2(5),SL3(5),SL1T(5),SL2T(5),SL3T(5),SL1T0(5),SL2T0(5),SL3T0(5),
B3PREV(5),C1PREV(5),C2PREV(5),C3PREV(5),EPS2,EPS3,EPS4,EPS5
COMMON/GBLOCK/F11(5),F22(5),F33(5),SKS(5),SK1,SK2,SK3,SKE1,SKE2,
SKE3
COMMON/HBLOCK/VVELO(14,B),HVELO(14,B),XVELO(14,B),YVELO(14,B),
XVELO0,YVELO0,PS1,PS1O,PSVELO,X,XO,Y,YO,XVEL,YVEL,PSIVEL,GRav,
ZETA,KCONMX,DELTAT,DELTAT,GR0NU(5),AL1(5),FX1(3),FY1(3),RMOO(3),
VMASS,PMASS,RMOM1,CMOM1,FA(5),FB(5),FC(5),IP,IP'',IR,
EBAL1,EBAL2,EBAL3,P1(5),P2(5),P3(5),DEL1(5),EPS1(5),PHI,
ENPRO(5)
COMMON/IBLOCK/BETA(5),BEPR(5),BETA(5),LAND
COMMON/KBLOCK/BETADT,FLL,FVV,LINE,FINT,RUNNO,REKN,SRNO,XN,
XMIN,TMINXN,BIMIN,BJMIN,TMINB,TMINJ,HN,RN,KPRINT,NSO,NQO,
NS,NN,NSMAX,NMAX,LOTPRT,NST,NU0,II,JJ,J,CONS,IFLAG,PRBE,KM,
MULT,PRXVEL,PRVEL
COMMON/KBLOCK+TIME,TIME,TME(402),XP1(402),YP1(402),ZP1(402),XP3(402),
YP3(402),ZP3(402),FX1(402),Fx1(402),FXPL31(402),FXPL31(402),
FZPL31(402),FZPL31(402),FZPL31(402),G1STR1(402),G1STR1(402),
G1STR2(402),G1STR2(402),G3STR1(402),G3STR1(402),G3STR2(402),
G3STR2(402),G1FRC1(402),G1FRC2(402),G1FRC3(402),G3FRC1(402),G3FRC2(402),
G3FRC3(402)
GO TO (10,20,30,40,50,60,70,80),IPR

PRINT INPUT DATA

10  DO 3042 KKK=1,2
PRINT 950
PRINT 1050,(A(I),I=1,5)
PRINT 1061,(ALPHA(I),I=1,5)
PRINT 1062,(D(I),I=1,5)
PRINT 1063,(D(I),I=1,5)

Figure 1-30. Subroutine PRINT1 (Continued)
PRINT 1064, (F11(I), I=1,5)
PRINT 1065, (F22(I), I=1,5)
PRINT 1066, (F33(I), I=1,5)
PRINT 1058, (GRDMU(I), I=1,5)
PRINT 1067, (P1(I), I=1,5)
PRINT 1068, (P2(I), I=1,5)
PRINT 1069, (P3(I), I=1,5)
PRINT 1070, (R1(I), I=1,5)
PRINT 1071, (R2(I), I=1,5)
PRINT 1090, (RP(I), I=1,5)
PRINT 1091, (SKS(I), I=1,5)
PRINT 1060, (THETA(I), I=1,5)
PRINT 1072, (DELTAP, DELTAT, EPS2)
PRINT 1073, (EPS3, EPS4, EPS5)
PRINT 1074, (FIN, GRAY, HN)
PRINT 1075, (PMASS, PSIO, PSVELO)
PRINT 1076, (RMOMI, RUNNO, RN)
PRINT 1077, (SK1, SK2)
PRINT 1078, (SK3, SKE1, SKE2)
PRINT 1079, (SK3, SERO, VMASS)
PRINT 1080, (Y0, ZETA, H)
PRINT 1081, (I, JJ, J, K, KPRINT, LAND)
PRINT 1082, (NSO, N00, NSMX, NQMX, KCUNMX)
PRINT 1089, (LOT, RPRT)
PRINT 1083, (VVELO(ML, NL), HVELO(ML, NL), ML=1, NQMAX), NL=1, NSMAX)
PRINT 1059, (SL10(I), I=1,5)
PRINT 1057, (D10(I), I=1,5)
PRINT 1051, (B10(I), I=1,5)
PRINT 1052, (B20(I), I=1,5)
PRINT 1053, (B30(I), I=1,5)
PRINT 1054, (C10(I), I=1,5)
PRINT 1055, (C20(I), I=1,5)
PRINT 1056, (C30(I), I=1,5)
PRINT 1092, (XVELO0, YVELO0)
3042 CONTINUE
RETURN

PRINT ON LINE DATA

20 PRINT 950
PRINT 850, (SERNO, RUNNO, XVELO0, YVELO0, ZETA, PSIO, PSVELO
PRINT 851, (VVELO(NQ, NS), HVELO(NQ, NS)
PRINT 902, (II, JJ)
PRINT 903, (TIME, X, Y, PSI, XVEL, YVEL, PSIVEL, BETADT, BETA(II),
1 BETA(JJ), FLL, FVV, RM00(3)
RETURN

RECORD LINE COUNT FOR PRINT HEADINGS

30 LINE=LINE+1
IF(LINE=49) 705, 705, 707
705 PRINT 903, (TIME, X, Y, PSI, XVEL, YVEL, PSIVEL, BETADT, BETA(II),
1 BETA(JJ), FLL, FVV, RM00(3)
RETURN
707 PRINT 950
PRINT 902, (II, JJ)
LINE=0
PRINT 903, (TIME, X, Y, PSI, XVEL, YVEL, PSIVEL, BETADT, BETA(II),
1 BETA(JJ), FLL, FVV, RM00(3)
RETURN
40 PRINT 950

Figure 1-30. Subroutine PRINT1 (Continued)
C PRINT PAGE TITLES
C PRINT 850,SERNO,RUNNO,XVELOO,YVELOO,ZETA,PSIO,PSVELO
PRINT 851,VVELOO(NQ,NS),HVELOO(NQ,NS)
PRINT 905,TIME
RETURN
C PRINT 950
PRINT 850,SERNO,RUNNO,XVELOO,YVELOO,ZETA,PSIO,PSVELO
PRINT 851,VVELOO(NQ,NS),HVELOO(NQ,NS)
PRINT 906,TIME
RETURN
C VEHICLE IS STABLE AND HAS STOPPED. PRINT OUTPUT DATA
C PRINT 950
PRINT 850,SERNO,RUNNO,XVELOO,YVELOO,ZETA,PSIO,PSVELO
PRINT 904,TIME
RETURN
C PRINT SUMMARY OUTPUT
C PRINT 907,(SL1M(I),I=1,N)
PRINT 908,(SL2M(I),I=1,N)
PRINT 909,(SL3M(I),I=1,N)
PRINT 910,XNMIN,TMINXN
PRINT 936,I,BJMIN,TMINBI
PRINT 937,JJ,BJMIN,TMINBJ
PRINT 917,EGBAL1
PRINT 918,EGBAL2
PRINT 919,EGBAL3
PRINT 913,PHI
PRINT 938,II,BETA(II),II,BETADT
PRINT 939,(ENPRO(I),I=1,N)
RETURN
C PRINT STORED OUTPUT INFORMATION
C IF(M-IQ)1823,1824,1825
1822 IF(M-IQ)1823,1823,1824
1823 IQ=M
IR=1
C PRINT STORED OUTPUT INFORMATION
C PRINT 950
PRINT 931,(TME(I),XP1(I),YP1(I),ZP1(I),XP2(I),YP2(I),ZP2(I),
1FX1(I),FX3(I),I=IP,IQ)
IF(IR)1825,1825,1826
1825 IP=IP+50
IQ=IQ+50
GO TO 1822
C TEST FLAG LOTPRT TO SEE IF COMPLETE PRINTOUT IS DESIRED
C IF (LOTPRT) 1836,1836,2826
2826 IP=1
IQ=50
IR=-1
1827 IF(M-IQ)1828,1828,1829

Figure 1-30. Subroutine PRINT1 (Continued)

BENDIX PRODUCTS AEROSPACE DIVISION
Subroutine PRINT1 (Continued)

1828  IQ=M
1829  IR=1
1830  IF (IR) 1830,1830,1831
1831  IPI=IP+50
1832  GO TO 1827
1833  GO TO 1832
1834  PRINT 950
1835  IF (M-IQ) 1833,1833,1834
1836  IF (IR) 1835,1835,1836
1837  RETURN

FORMAT STATEMENTS FOR OUTPUT INFORMATION

850 FORMAT(11H SERIES NO., FT.2,9H RUN NO.,F6,2,14H XVELO
1=F6*2.11H YVELO=F6.2+10H ZETA=F5+3.10H PS10=F6.41
12H PSVELO=F6.4/)
851 FORMAT(47H)
902 FORMAT(11H TIME X Y PSI XVEL YVEL P
1SIVEL BETAVEL BETA*I1,6H BETA*I1,30H FL XVEL)
904 FORMAT(11H STABLE*F7.3*5HSECS///)
905 FORMAT(13H UNSTABLE*F7.3*5HSECS///)
906 FORMAT(19H TIME*F7.3*5HSECS///)
907 FORMAT(32H MAXIMUM STROKE NO. 1 STRUT*F12.3///)
908 FORMAT(32H NO. 2 STRUT*F12.3///)
909 FORMAT(32H NO. 3 STRUT*F12.3///)
910 FORMAT(11H MINIMUM CLEARANCE OF NOZZLE=F7.3,55H
1TIME WHEN THE MINIMUM CLEARANCE OCCURS =F7.3///)
911 FORMAT(54H MINIMUM CLEARANCE BETWEEN SHOCK STRUT AND FRAME =)
912 FORMAT(32H 5F12.3///)
913 FORMAT(43H FINAL ANGULAR ORIENTATION OF VEHICLE =F7.4///)
914 FORMAT(32H MAXIMUM STROKE NO. 1 STRUT*F12.3///)
915 FORMAT(32H NO. 2 STRUT*F12.3///)
916 FORMAT(32H NO. 3 STRUT*F12.3///)
917 FORMAT(16H ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND
1C.G. DROP=F11.3///)
918 FORMAT(48H ENERGY DISSIPATED BASED ON PLASTIC STROKE =F11.3///)
919 FORMAT(21H TIME XP(*I1,11H) YP(*I1,11H) ZP
1(*I1,12H) XP(*I1,11H) YP(*I1,11H) ZP(*I1,13H)
1 FX(*I1,14H) FX(*I1,14H) /

Figure 1-30. Subroutine PRINT1 (Continued)
Figure 1-30. Subroutine PRINT1 (Continued)
Figure 1-30. Subroutine PRINT1 (Concluded)
ENERGY

THIS PROGRAM WAS WRITTEN IN FORTRAN IV

THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

NOTE AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS BI(1). THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS B1(I).

SYMBOL

A  SYMMETRY FACTOR -- IF TWO VEHICLE LEGS ARE SYMMETRIC, SET A=2.0 FOR ONE OF THE SYMMETRIC PAIRS AND A=0.0 FOR THE OTHER TO SAVE ON COMPUTER TIME IE FOR 22 LANDING A=2.0+2.0 FOR A 121 LANDING A=1.2+1.0

F11  PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)
F22  PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2
F33  PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3

GRAV  LOCAL GRAVITY

PSIVEL  INSTANTANEOUS PITCH VELOCITY OF THE VEHICLE C.G.
PVELO  INITIAL VEHICLE PITCH RATE

P1  FRICTION FORCE IN STRUT NO. 1
P2  FRICTION FORCE IN STRUT NO. 2
P3  FRICTION FORCE IN STRUT NO. 3

RMOMIT  TOTAL MOMENT OF INERTIA OF THE VEHICLE MASS AND THOSE FOOTPADS WHICH ARE OFF THE GROUND

SK1  COMPRRESSIVE ELASTIC SPRING RATE OF STRUT NO. 1(UPPER)
SK2  COMPRRESSIVE ELASTIC SPRING RATE OF STRUT NO. 2(LOWER)
SK3  COMPRRESSIVE ELASTIC SPRING RATE OF STRUT NO.3 (LOWER)

SL1M  MINIMUM LENGTH TO WHICH STRUT NO.1 HAS BEEN COMPRESSED DURING THIS RUN
SL2M  MINIMUM LENGTH TO WHICH STRUT NO.2 HAS BEEN COMPRESSED DURING THIS RUN
SL3M  MINIMUM LENGTH TO WHICH STRUT NO.3 HAS BEEN COMPRESSED DURING THIS RUN

VMASS  VEHICLE MASS
X  INSTANTANEOUS X POSITION OF THE VEHICLE C.G.
XVEL  INSTANTANEOUS X VELOCITY OF THE VEHICLE C.G.
XVELO  INITIAL VERTICAL VELOCITY OF VEHICLE C.G.
XVELOO  SAME AS XVELO

Y  INSTANTANEOUS Y POSITION OF THE VEHICLE C.G.
YYEL  INSTANTANEOUS Y VELOCITY OF THE VEHICLE C.G.
YVELO  INITIAL HORIZONTAL VELOCITY OF VEHICLE C.G.
YVELOO  SAME AS YVELO

ZETA  GROUND SLOPE

OUTPUT  THROUGH LABELED COMMON

Figure 1-31. Subroutine ENERGY
SYMBOL DEFINITION

EGBAL1 ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND C.G. DROP
EGBAL2 ENERGY DISSIPATED BASED ON PLASTIC STROKE OF ALL STRUTS
EGBAL3 ENERGY DISSIPATED BASED ON PLASTIC AND FULL ELASTIC STROKE OF ALL STRUTS
ENPRO(I) PERCENT OF TOTAL ENERGY ABSORBED BY STROKEING OF THE STRUTS OF LEG SET I

SUBROUTINE ENERGY

COMMON/ABLACK/NK,NOGR(5)
B3PREV(5),C1PREV(5),C2PREV(5),C3PREV(5),EPS2, EPS3, EPS4, EPS5
COMMON/GBLOCK/F11(5),F22(5),F33(5),SKS(5),SK1,SK2,SK3,SKE1,SKE2,
SKE3
COMMON/HBLOCK/VVELO(14,8),VVELO(14,8),XVELO(14,8),YVELO(14,8),
XVELO0,YVELO0,PS1,PS10,PSVELO,X0,Y0,XVEL,YVEL,PSIVEL,PSIVEL,PSIVEL,
ZETA,KCONMX,DELTTT,DELTA,T,GROMU(5),A(5),FXX(3),FY(3),RMOC(3),
VMASS,PMASS,ROMMIT,ROMMGR,ROMMI,FA(5),FB(5),FC(5),IP, IQ, IR,
ENGBAL1,ENGBAL2,ENGBAL3,P1(5),P2(5),P3(5),DEL(5),EPSE1(5),PHI,
ENPRO(5)
DIMENSION ENGY(5)

COMPUTE ENERGY TERMS FOR PRINTOUT PURPOSES ONLY

EGBAL1=VMASS*XVELO0*XVELO0/2.0+VMASS*YVELO0*YVELO0/2.0+RMOMIT*
1PSVELO0+PSVELO/2.0+((X0-X)*COS(ZETA)+Y*SIN(ZETA))*VMASS*GRAV-
1(VMASS*XVEL*XVEL/2.0+VMASS*YVEL*YVEL/2.0+RMOMIT*PSIVEL*PSIVEL/12.0)
EGBAL2=0.0
EGBAL3=0.0
DO 821 I=1,N
821 ENGY(I)=0.0
DO 822 I=1,N
ENGY(I)=A(I)*(F11(I)+P1(I))*(SL1M(I)-F11(I)/SK1)+(F22(I)+P2(I))*
1 SL2M(I)-F22(I)/SK2)+(F33(I)+P3(I))*(SL3M(I)-F33(I)/SK3))
ENGBAL2=ENGBAL2+ENGY(I)
822 ENGBAL3=ENGBAL3+A(I)*(F11(I)+P1(I))*(SL1M(I)-F11(I)/2.0*SK1)+(1F22(I)+P2(I))*(SL2M(I)-F22(I)/(2.0*SK2)+(F33(I)+P3(I))*(SL3M(I)-
2F33(I)/(2.0*SK3))
DO 2822 I=1,N

DETERMINE ENERGY ABSORPTION DISTRIBUTION BETWEEN LEG SETS

2822 ENPRO(I)=(ENGY(I)/ENGBAL2)*100.0
RETURN
END

Figure 1-31. Subroutine ENERGY (Concluded)
TITLE
PROFIL

AUTHORS
R. BLACK, J. CADORET, J. GIBSON
THE BENDIX CORPORATION

DATE
10-25-64

PURPOSE
DETERMINE NEW INDEX VALUES TO CHOOSE NEW VELOCITY CONDITIONS FROM INPUTED ARRAY FOR NEXT RUN

CALL
PROFIL (IPRO)

NOTE
THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE
THIS PROGRAM EXECUTES ON THE UNIVAC 1107 COMPUTER

INPUT
THROUGH LABELED COMMON

NOTE
AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS BI(1). THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS BI(1).

SYMBOL
DEFINITION

IPRO
FLAG USED TO INDICATE ENTRANCE POINT

NQO
STARTING COLUMN IN VELOCITY INPUT ARRAY

NQMAX
ENDING COLUMN IN VELOCITY ARRAY

NSO
STARTING ROW IN VELOCITY INPUT ARRAY

NSMAX
ENDING ROW IN VELOCITY ARRAY

NOTE
SEE WRITEUP FOR DISCUSSION OF VELOCITY ARRAY

NST
FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE

NUN
FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE

OUTPUT
THROUGH LABELED COMMON

SUBROUTINE PROFIL(IPRO)

COMMON/JBLOCK/BETADT, FLL, FVV, LINE, FINT, RUI, RUNNO, RUNNOO, SERNO, XN,
XMIN, XMINX, XMINB, XMINI, XMINJ, XMINK, RN, KPRINT, NSO, NQO,
NS, NQ, NSMAX, NQMAX, LOPRT, NST, NUN, I, JJ, CONS, IFLAG, PRBE, KM,
MULT, PRXVEL, PRYVEL

DETERMINE NEW INDEX VALUES TO CHOOSE NEW VELOCITY CONDITIONS FROM INPUTED ARRAY FOR NEXT RUN

GO TO (785, 824) IPRO

785 IF(NST) 790, 790, 796

790 NUN = 1

NQ = NQ - 1

794 IF(NQ) 794, 794, 40

796 NS = NS + 1

IF(NS-NSMAX)20,20,840

824 IF(NUN) 825, 825, 796

825 NST = 1

NQ = NQ + 1

IF(NQ-NQMAX) 140, 40, 830

Figure 1-31A. Subroutine PROFIL
830 NQ=NQ-1
   NS=NS+1
   IF(NS-NSMAX)833,833,840
833 IF(NQ-NQMAX )20,20,836
836 NQ=NQ-1
   GO TO 833
840 PRINT 950
    IPRO=1
    RETURN
20 IPRO=2
    RETURN
40 IPRO=3
    RETURN
950 FORMAT(1H1)
   END

Figure 1-31A. Subroutine PROFIL (Concluded)
<table>
<thead>
<tr>
<th>SERIES ID</th>
<th>VELU</th>
<th>VEL0</th>
<th>ZLTA</th>
<th>PS10</th>
<th>PSVELO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-98</td>
<td>-5.74</td>
<td>2.21</td>
<td>+2.7</td>
<td>+2.68</td>
<td>+0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TIME</th>
<th>0.000</th>
</tr>
</thead>
</table>

| MAXIMUM STROKE | 2.5 | 1.00 | 0.00 |
| MINIMUM CLEARANCE OF STROKE | 2.5 |
| MINIMUM STABILITY ANGLE a1 | 2.5 |
| MINIMUM STABILITY ANGLE a2 | 2.5 |

| TIME WHEN THE MINIMUM CLEARANCE OCCURS | = 2.562 |
| TIME WHEN THE MINIMUM CLEARANCE OCCURS | = 2.500 |
| TIME WHEN THE MINIMUM CLEARANCE OCCURS | = 2.500 |

ENERGY DISSIPATED DUE TO VEHICLE VELOCITIES AND COG DROP = 2444.271

ENERGY DISSIPATED DUE TO PLASTIC STROKES = 23536.849

ENERGY DISSIPATED DUE TO PLASTIC AND FULL ELASTIC STROKES = 2444.271

FINAL ANGULAR ORIENTATION OF VEHICLE = 90°

FINAL STABILITY ANGLE a1 = 2.5 |

FINAL BETA RATE OF CHANGE = +0.06

ENERGY DISTRIBUTION BETWEEN LEGS - PERCENT = 33.553 66.647 1000

Figure 1-33. Summary Data Printout
<table>
<thead>
<tr>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>210.6</td>
<td>940.8</td>
<td>1403.4</td>
<td>2887.7</td>
<td>-3691.1</td>
<td>-4068.0</td>
<td>-4049.2</td>
<td>-4069.5</td>
</tr>
<tr>
<td>-1777.6</td>
<td>-3377.7</td>
<td>-4004.7</td>
<td>-3177.7</td>
<td>-4215.2</td>
<td>-4221.7</td>
<td>-3832.1</td>
<td>-3906.5</td>
</tr>
<tr>
<td>-7747.7</td>
<td>-8067.6</td>
<td>-1195.6</td>
<td>-3145.6</td>
<td>-6682.7</td>
<td>-6779.4</td>
<td>-1774.4</td>
<td>-1228.9</td>
</tr>
<tr>
<td>-1456.4</td>
<td>-2427.7</td>
<td>-1062.8</td>
<td>-2558.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Figure 1-34. Final Data Printout (Part 2)
<table>
<thead>
<tr>
<th>EPL3(1)</th>
<th>EPL4(1)</th>
<th>EPL5(1)</th>
<th>EPL6(1)</th>
<th>EPL7(1)</th>
<th>EPL8(1)</th>
<th>EPL9(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6A</td>
<td>1.6A</td>
<td>1.6A</td>
<td>1.6A</td>
<td>1.6A</td>
<td>1.6A</td>
<td>1.6A</td>
</tr>
<tr>
<td>1.6B</td>
<td>1.6B</td>
<td>1.6B</td>
<td>1.6B</td>
<td>1.6B</td>
<td>1.6B</td>
<td>1.6B</td>
</tr>
<tr>
<td>1.6C</td>
<td>1.6C</td>
<td>1.6C</td>
<td>1.6C</td>
<td>1.6C</td>
<td>1.6C</td>
<td>1.6C</td>
</tr>
<tr>
<td>1.6D</td>
<td>1.6D</td>
<td>1.6D</td>
<td>1.6D</td>
<td>1.6D</td>
<td>1.6D</td>
<td>1.6D</td>
</tr>
<tr>
<td>1.6E</td>
<td>1.6E</td>
<td>1.6E</td>
<td>1.6E</td>
<td>1.6E</td>
<td>1.6E</td>
<td>1.6E</td>
</tr>
<tr>
<td>1.6F</td>
<td>1.6F</td>
<td>1.6F</td>
<td>1.6F</td>
<td>1.6F</td>
<td>1.6F</td>
<td>1.6F</td>
</tr>
<tr>
<td>1.6G</td>
<td>1.6G</td>
<td>1.6G</td>
<td>1.6G</td>
<td>1.6G</td>
<td>1.6G</td>
<td>1.6G</td>
</tr>
<tr>
<td>1.6H</td>
<td>1.6H</td>
<td>1.6H</td>
<td>1.6H</td>
<td>1.6H</td>
<td>1.6H</td>
<td>1.6H</td>
</tr>
<tr>
<td>1.6I</td>
<td>1.6I</td>
<td>1.6I</td>
<td>1.6I</td>
<td>1.6I</td>
<td>1.6I</td>
<td>1.6I</td>
</tr>
<tr>
<td>1.6J</td>
<td>1.6J</td>
<td>1.6J</td>
<td>1.6J</td>
<td>1.6J</td>
<td>1.6J</td>
<td>1.6J</td>
</tr>
</tbody>
</table>

Figure 1-34. Final Data Printout (Part 2 - Continued)
SECTION II

DIGITAL COMPUTER PROGRAM FOR THE APPROXIMATE DETERMINATION OF NOZZLE CLEARANCE

This digital computer program was written to provide a simple means for determining final nozzle clearance as a function of initial vehicle C.G. height and upper strut stroke load.

In addition to this information, the program also defines additional vehicle geometrical dimensions required for the landing dynamics computer program (Section I) and the stowability index (STOW).

A DIGITAL COMPUTER PROGRAM FOR THE APPROXIMATE DETERMINATION OF NOZZLE CLEARANCE

The mathematical model used in this program is illustrated in Figure 2-1. In determining the approximate relationship between vehicle geometry, upper strut stroke force and nozzle clearance, it is assumed that total vehicle energy, both kinetic and potential, is absorbed by stroking of the upper struts only. Further, it is assumed that all upper struts, both on the front and rear legs, stroke equal amounts. This assumption approximates the real vehicle by one in which the lower struts are rigid or non-strokable, and the coefficient of friction is zero.

The kinetic energy of the vehicle is given by the equation

\[ KE = \frac{1}{2} M V^2 \]

where \( M \) is the total vehicle mass and \( V \) may be assumed to be the vehicle vertical velocity. (This assumes the vehicle comes to rest.)

The potential energy of the vehicle may be broken into two parts:

1. The potential energy loss resulting from the vehicle tipping from its initial contact configuration through an angle until all feet contact the ground. The potential energy loss for this phase of landing is approximated by

\[ P E_1 = M G H_1 \]

Where \( M \) is the mass of the vehicle c.g. only, and \( H_1 \) is the vertical distance the c.g. drops.

2. The potential energy loss resulting from stroking of the upper struts. This energy term is a function of the final allowable nozzle clearance (FNC) and is given by the equation.

\[ P E_2 = M G H_2 \]
UPPER STRUT STROKE FORCE DETERMINATION

Assume: Vehicle Energy = \( E = KE + PE1 + PE2 \)

Where:
\[ KE = \frac{1}{2} MTV^2 \]

\[ PE1 = MCGGH_1 \]

\[ PE2 = MCGGH_2 \text{ (FNC)} \]

Assume: Energy Absorbed by Upper Struts Only

\[ E = 4 \times F_{11} \times (LO-LF) \]

Therefore:

\[ F_{11} = \frac{E}{4 \times (LO-LF)} \]

or

\[ H = f(F_{11}, \text{FNC}, \text{Etc.)} \]

Figure 2-1. Geometric Computer Program
where $M$ is again the mass of the vehicle c.g. only, and $H_2$ is the vertical distance the c.g. drops.

Here, the distance $H_2$ is a function of FNC.

This total vehicle energy (which must be absorbed to bring the vehicle to rest) $E = K_E + P_E_1 + P_E_2$ is equated to the energy absorbed by stroking of the upper struts. The initial strut length is given by $L_0$ and its final length by $L_f$ as indicated in the lower right of Figure 2-1.

Equating the energy terms by the equation

$$E = 4 \times F_{11} \times (L_0 - L_f)$$

permits a solution for the required upper strut stroke load $F_{11}$ as a function of initial c.g. Height and final required nozzle clearance or, as was used for the parametric study, the initial c.g. height, $H_0$, required to result in the proper nozzle clearance for a given $F_{11}$. Thus:

$$H_0 = \text{function of } F_{11}, \text{ FNC and overall vehicle geometry.}$$

This computer program was also used to compute various geometric quantities for the vehicle which are required inputs to the landing dynamics and weight analysis programs. In addition to these parameters, the stowability for the vehicle is also determined.

It has been determined that a 2-2 landing configuration having infinite friction on the rear legs and zero friction on the front legs with $V_Y$ and $V_H$ the maximum values defined by the landing specification set produces the most critical nozzle clearance conditions. Repeated comparisons between the results of the geometric program and complete dynamic landing studies have shown good agreement.

Although not rigorous, this geometric computer program provided a consistent basis for the determination of $R$ and $H$ for any given ratio of $R/H$ during the parameter variation studies. This approximate relationship between $R$ and $H$ for a given $R/H$, final nozzle clearance and $F_{11}$ provided a convenient means of systematizing the choice of $R$ and $H$ during the parametric variation study.

Figure 2-2 illustrates the input data format required for use of the program. The input quantities are defined in Figure 2-7 under "input definitions."

The program has two modes of operation defined by the input parameter "IFLAG." If this flag is set equal to -1, the program will compute the upper strut stroke force $F_{11}$ necessary to produce a given final nozzle clearance for a given initial vehicle c.g. height.

If the flag is set equal to +1, the program will compute the vehicle c.g. height necessary to produce the required final nozzle clearance for a given upper strut stroke load.

Figures 2-3 through 2-6 are a flow diagram of the program and Figures 2-7 through 2-10 are complete listings of the program and its subroutines.
Figures 2-11 and 2-12 are summaries of the output data for a typical run. As is illustrated, the input data is printed as part of the output record. The first sample run has $\text{IFLAG} = -1$ as shown on card No. 8 of the input data. The output data is immediately after the input data.

The sample data of Figure 2-12 is a run for control mode $\text{IFLAG} = +1$. Again, the input data is printed out first. Since, in this mode, the vehicle c.g. height is interpolated to converge on the desired value of $F_{11}$, namely $F_{\phi \phi}$, the output data is printed during the convergence process for each new $H(I)$ chosen by the computer. The final output is the desired information and includes computation of the stowability index (STOW).

This program was written in Fortran IV and is currently being used on a Univac 1107 digital computer.
### Figure 2-2. Geometry Program

<table>
<thead>
<tr>
<th>Statement Number</th>
<th>FORTRAN Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>THETA</td>
</tr>
<tr>
<td>2</td>
<td>RH</td>
</tr>
<tr>
<td>3</td>
<td>AN</td>
</tr>
<tr>
<td>4</td>
<td>G</td>
</tr>
<tr>
<td>5</td>
<td>DELTAP</td>
</tr>
<tr>
<td>6</td>
<td>Z</td>
</tr>
<tr>
<td>7</td>
<td>FOO</td>
</tr>
<tr>
<td>8</td>
<td>NOH</td>
</tr>
<tr>
<td>9</td>
<td>H(1)</td>
</tr>
</tbody>
</table>
This program computes stroke force $F_{11}$ vs vehicle C.G. height $H(I)$ and final nozzle clearance $FNC$.

Figure 2-3. Geometry Program
Figure 2-3. Geometry Program (Continued)
This subroutine reads the input data cards and prints data as part of the output record.

Figure 2-4. Subroutine To Geometry Program
This subroutine computes the stroke force $F_{11}$ as a function of vehicle geometry.

Figure 2-5. Subroutine To Geometry Program
This subroutine determines by interpolation the $H(I)$ required to produce a given $F11$ and $FNC$.

**Figure 2-6. Subroutine To Geometry Program**

2-10
**Title:** Force - Strut Force Program

**Author:** Bendix Products Aerospace Division - South Bend, Indiana

**J.C. Gibson**

**Date:** August 3, 1964

**Purpose:** Lunar landing dynamics system investigation - This program relates vehicle C.G., height and strut stroke force to final nozzle clearance, computes the stowability index and basic geometric inputs for the landing dynamics and weight programs. Ref. Bendix Report No MM-64-9 Nov., 1964

**Method:** Program relates landing vehicle incoming kinetic and potential energy to energy absorbed in stroking main shock strut. This energy balance allows determination of final C.G., height as function of geometry and stroke force. Also computes supplemental geometric data by trigonometric relationships.

**Note:**

- If (IFLAG) = +, program computes C.G., height for a given nozzle clearance and stroke force.
- If (IFLAG) = -, program computes nozzle clearance for a given C.G., height and stroke force.

**Input:** By punch card

**Input Definitions:**

- AN = Number of legs on vehicle (usually 4)
- D = Vertical distance between upper and lower hard-points
- D11 = Vertical distance between vehicle C.G. and upper hardpoint
- DELTA = Angle projected in plane parallel to direction of motion for 1-2-1 landing, between lower struts and vehicle centerline
- DELTAP = Vertical thickness of footpad (from bottom surface to strut attach point)
- FNC = Final desired nozzle clearance
- FOO = Required upper strut stroke force
- G = Acceleration of local gravity at landing site
- GAMMA = True angle between upper (main) strut and vehicle centerline
- HN = Vertical distance between vehicle C.G. and lowest point on nozzle cone
- PMASS = Mass of one vehicle footpad
- PSI = Initial vehicle pitch angle, angle between

Figure 2-7. Nozzle Clearance Program
LOCAL VERTICAL NAD VEHICLE CENTERLINE

RH = R/H RATIO WHERE R = RADIUS OF VEHICLE FOOTPADS
      H = DEFINED BELOW

CRNO = IDENT. NO. COMPUTER RUN

THETA = ANGLE BETWEEN VERTICAL PLANE IN DIRECTION OF
      MOTION AND VERTICAL PLANE THROUGH VEHICLE C.G.
      AND UPPER HARDPOINT

VMASS = PRIMARY VEHICLE MASS CONCENTRATED AT VEHICLE
      C.G.

XVEL = VELOCITY NORMAL TO GROUND SURFACE. POSITIVE
      AWAY FROM SURFACE

YVEL = VELOCITY PARALLEL TO GROUND SURFACE

Z = WIDTH OF BASIC ATTACH POINTS ON VEHICLE FOR
   STRUTS ON NARROW TRACK VEHICLE OR TRUSSWORK ON
   WIDE TRACK VEHICLE. THIS IS A PROGRAM CONSTANT
   WITH VALUE = 5.0

ZETA = GROUND SLOPE ANGLE

IFLAG = FLAG TO DETERMINE PROGRAM MODE. IF +, CALCULATE
        C.G. HEIGHT FOR GIVEN STRUT FORCE AND FNC. IF-,
        CALCULATE STRUT FORCE FOR GIVEN C.G. HEIGHT AND
        FNC.

NO = NUMBER OF HI11 VALUES IN INPUT LIST. PERMITS
     MULTIPLE RUNS UNDER IFLAG = - CONTROL

HI11 = INITIAL VEHICLE C.G. HEIGHT. INPUT AS LIST FOR
      MULTIPLE RUNS UNDER IFLAG = - CONTROL

OUTPUT PRINTED OUTPUT

OUTPUT BY EQUIVALENCE TO COMMON STORAGE

OUTPUT DEFINITIONS

RP = FOOTPAD RADIUS

F11 = UPPER STRUT STROKE FORCE

ALPHA = ANGLE BETWEEN LOWER HARDPOINTS

SL0 = INITIAL UPPER STRUT LENGTH

E = TOTAL VEHICLE ENERGY

SL = FINAL UPPER STRUT LENGTH

EKE = VEHICLE INCOMING KINETIC ENERGY

PE1 = POTENTIAL ENERGY RESULTING FROM C.G DROP FROM
      INITIAL ORIENTATION TO ALL FOOTPADS ON SURFACE

PE2 = POTENTIAL ENERGY RESULTING FROM C.G DROP
      RESULTING FROM STROKING OF UPPER STRUT

R1 = UPPER HARDPOINT RADIUS

R2 = LOWER HARDPOINT RADIUS

STOW = STOWABILITY INDEX

COMMON AN,ALPHA,D11,D1TAP,DELTA,DELTA1,DIFF,E,EKE,F11,FNC,F00,
      G,GAMMA,BIPREL(120),HI11,IFLAG,IJK,JFLAG,LL,NO,PE1,PE2,PMASS,PS1,
      2PI,P2,R1,R2,RP,SL,SL0,ST=1,THETA,VMASS,XVEL,YVEL,Z,ZETA

1 CALL INPUT

   IJK=1
   I=1

   TEST PROGRAM MODF

Figure 2-7. Nozzle Clearance Program (Continued)
IF (IFLAG) 20,20,15

COMPUTE H(I) MINIMUM BASED ON VEHICLE GEOMETRY

15 H(I)=1.0*(FNC+MN+DELTAP)
   HPREV=FNC+MN+DELTAP
   IFLAG=!1

20 CALL GEOM

GO TO (20*1*100)*11

TEST PROGRAM MODE

100 IF (IFLAG) 110,110,120

PRINT OUTPUT DATA

110 PRINT 610,H(I),RP,F11,ALPHA
   PRINT 611,SLO,E,SL,KE
   PRINT 612,PE1,PE2,R1,R2
   PRINT 618,STOW
   PRINT 530

IF STROKE FORCE IS NEGATIVE, INCREASE H(I)

IF (F11) 200,200,112

112 IF (IFLAG) 115,1,1

CHOOSE NEW I INDEX AND RETURN TO 20 IF A MULTIPLE LIST OF H(I) IS PRESENT

115 I=I+1
   IF (I=NOH) 20,20,1

TEST DIFFERENCE OF F11 FROM FOO UNDER IFLAG = - MODE

120 DIFF=(F11-FOO)

IF DIFFERENCE IS LESS THAN 50 LBS, STOP AND PRINT OUTPUT DATA

IF (ABS(DIFF)-50.0) 110,110,125

125 PRINT 610,H(I),RP,F11,ALPHA
   PRINT 611,SLO,E,SL,KE
   PRINT 612,PE1,PE2,R1,R2
   PRINT 617

130 CALL INTER

GO TO 2C

200 PRINT 616,F11

H(I)=H(I)+0.5

GO TO 20

530 FORMAT(1H1)

610 FORMAT(18H) H =,F15.5,13H RP =,F15.5,12H
   1F11 =,F15.5,12H ALPHA =,F15.5)
   611 FORMAT(18H) SLO =,F15.5,13H E =,F15.5,12H
   1SL =,F15.5,12H KE =,F15.5)
   612 FORMAT(18H) PE1 =,F15.5,13H PE2 =,F15.5,12H
   1R1 =,F15.5,12H R2 =,F15.5)

Figure 2-7. Nozzle Clearance Program (Continued)
616 FORMAT(61H THE FORCE F11 IS NEGATIVE REPEAT USING A LARGER H)
     1 F15.5/F15.5/)
617 FORMAT(1HC)
618 FORMAT(18H STOW =F15.5)
END

Figure 2-7. Nozzle Clearance Program (Concluded)

BENDIX PRODUCTS AEROSPACE DIVISION
INPUT DATA AS PART OF THE OUTPUT RECORD
THIS SUBROUTINE READS THE INPUT CARDS AND PRINTS THE

INPUT DEFINITIONS

AN  = NUMBER OF LEGS ON VEHICLE (USUALLY 4)
D   = VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARD-POINTS
D11  = VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND
       UPPER HARDPOINT
DELTA = ANGLE PROJECTED IN PLANE PARALLEL TO DIRECTION
       OF MOTION FOR 1-2-1 LANDING BETWEEN LOWER
       STRUTS AND VEHICLE CENTERLINE
DELTAP = VERTICAL THICKNESS OF FOOTPAD (FROM BOTTOM
         SURFACE TO STRUT ATTACH POINT)
FNC  = FINAL DESIRED NOZZLE CLEARANCE
FDC  = REQUIRED UPPER STRUT STROKE FORCE
G   = ACCELERATION OF LOCAL GRAVITY AT LANDING SITE
GAMMA = TRUE ANGLE BETWEEN UPPER (MAIN) STRUT AND
        VEHICLE CENTERLINE
HN   = VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND
       LOWEST POINT ON NOZZLE CONE
PMass = MASS OF ONE VEHICLE FOOTPAD
PSI   = INITIAL VEHICLE PITCH ANGLE, ANGLE BETWEEN
       LOCAL VERTICAL AND VEHICLE CENTERLINE
RH   = R/H RATIO WHERE R = RADIUS OF VEHICLE FOOTPADS
       H = DEFINED BELOW
SERNO = IDENT. NO. COMPUTER RUN
THETA = ANGLE BETWEEN VERTICAL PLANE IN DIRECTION
        OF MOTION AND VERTICAL PLANE THROUGH VEHICLE C.G.
        AND UPPER HARDPOINT
VMass = PRIMARY VEHICLE MASS CONCENTRATED AT VEHICLE
        C.G.
VMass = PRIMARY VEHICLE MASS CONCENTRATED AT VEHICLE
        C.G.
XVEL = VELOCITY NORMAL TO GROUND SURFACE, POSITIVE
       AWAY FROM SURFACE
YVEL = VELOCITY PARALLEL TO GROUND SURFACE
Z    = WIDTH OF BASIC ATTACH POINTS ON VEHICLE FOR
       STRUTS ON NARROW TRACK VEHICLE OR TRUSSWORK ON
       WIDETRACK VEHICLE. THIS IS A PROGRAM CONSTANT
       WITH VALUE = 5.0
ZETA = GROUND SLOPE ANGLE
IFLAG = FLAG TO DETERMINE PROGRAM MODE. IF +, CALCULATE
        C.G. HEIGHT FOR GIVEN STRUT FORCE AND FNC. IF-,
        CALCULATE STRUT FORCE FOR GIVEN C.G. HEIGHT AND
        FNC.

Figure 2-8. Subroutine INPUT
**Figure 2-8. Subroutine INPUT (Concluded)**

```plaintext
SUBROUTINE INPUT
COMMON AN, ALPHA, D, D11, DELTAP, DIFF, E, EK, F11, FNC, FOO,
    I1, IGA, MI, PRE, M20, HN, IFLAG, JFLAG, L, NOH, PE1, PE2, PMASS, PSI,
    2R1, 2R2, RH, RP, SLU, STU, THETA, VMASS, XVEL, YVEL, ZETA
READ 500, THETA, PSI, ZETA, RH, VMASS, PMASS, AN, XVEL, YVEL, G, HN, FNC,
    DELTAP, D11, D2, GAMMA, DELTA, FOO, SERNO
READ 510, NOH, IFLAG
READ 520, (H(K), K=1, NOH)
PRINT 613, SERNO
10 PRINT 600, THETA, PSI, ZETA, RH
PRINT 601, VMASS, PMASS, AN, XVEL
PRINT 602, YVEL, G, HN, FNC
PRINT 603, DELTAP, D11, D2
PRINT 604, GAMMA, DELTA, FOO
PRINT 614, NOH, IFLAG
RETURN
500 FORMAT(10X, 3F15.5)
510 FORMAT(10X, 3I10)
520 FORMAT(10X, 6F10.5)
613 FORMAT(21H)
600 FORMAT(18H, SERIES NO =, F10.3//) THETA =, F15.5, 13H PSI =, F15.5, 12H
   1ZETA =, F15.5, 12H R/H =, F15.5
601 FORMAT(18H VMASS =, F15.5, 13H PMASS =, F15.5, 12H
   1AN =, F15.5, 12H XVEL =, F15.5)
602 FORMAT(18H YVEL =, F15.5, 13H G =, F15.5, 12H
   1HN =, F15.5, 12H FNC =, F15.5)
603 FORMAT(18H DELTAP =, F15.5, 13H D11 =, F15.5, 12H
   1D =, F15.5, 12H Z =, F15.5)
604 FORMAT(18H GAMMA =, F15.5, 13H DELTA =, F15.5, 12H
   1FOO =, F15.5)
614 FORMAT(18H NHO =, I15, 13H IFLAG =, I15///)
END
```

BENDIX PRODUCTS AEROSPACE DIVISION
BENDIX PRODUCTS AEROSPACE DIVISION

SUBROUTINE GEOM - GEOMETRY

CALL GEOM

INPUT

BY EQUIVALENCE TO COMMON STORAGE

INPUT DEFINITIONS

AN = NUMBER OF LEGS ON VEHICLE (USUALLY 4)
D = VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARD-POINTS
D11 = VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND UPPER HARDPOINT
DELTA = ANGLE PROJECTED IN PLANE PARALLEL TO DIRECTION OF MOTION FOR 1-2-1 LANDING BETWEEN LOWER STRUTS AND VEHICLE CENTERLINE
DELTAP = VERTICAL THICKNESS OF FOOTPAD (FROM BOTTOM SURFACE TO STRUT ATTACH POINT)
FNC = FINAL DESIRED NOZZLE CLEARANCE
FOG = REQUIRED UPPER STRUT STROKE FORCE
G = ACCELERATION OF LOCAL GRAVITY AT LANDING SITE
GAMMA = TRUE ANGLE BETWEEN UPPER (MAIN) STRUT AND VEHICLE CENTERLINE
HN = VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND LOWEST POINT ON NOZZLE CONE
PMASS = MASS OF ONE VEHICLE FOOTPAD
PSI = INITIAL VEHICLE PITCH ANGLE, ANGLE BETWEEN LOCAL VERTICAL AND VEHICLE CENTERLINE
RH = R/H RATIO WHERE R = RADIUS OF VEHICLE FOOTPADS H = DEFINED BELOW
SERNO = IDENT. NO. COMPUTER RUN
THETA = ANGLE BETWEEN VERTICAL PLANE IN DIRECTION OF MOTION AND VERTICAL PLANE THROUGH VEHICLE C.G. AND UPPER HARDPOINT
VMASS = PRIMARY VEHICLE MASS CONCENTRATED AT VEHICLE C.G.
XVEL = VELOCITY NORMAL TO GROUND SURFACE, POSITIVE AWAY FROM SURFACE
YVEL = VELOCITY PARALLEL TO GROUND SURFACE
Z = WIDTH OF BASIC ATTACH POINTS ON VEHICLE FOR STRUTS ON NARROW TRACK VEHICLE OR TRUSSWORK ON WIDETRACK VEHICLE. THIS IS A PROGRAM CONSTANT WITH VALUE = 5.0
ZETA = GROUND SLOPE ANGLE
IFLAG = FLAG TO DETERMINE PROGRAM MODE. IF +, CALCULATE C.G.HEIGHT FOR GIVEN STRUT FORCE AND FNC. IF-, CALCULATE STRUT FORCE FOR GIVEN C.G.HEIGHT AND FNC.

Figure 2-9. Subroutine GEOM
ACH = NUMBER OF H(I) VALUES IN INPUT LIST. PERMITS MULTIPLE RUNS UNDER IFLAG = - CONTROL
H(I) = INITIAL VEHICLE G.S. HEIGHT. INPUT AS LIST FOR MULTIPLE RUNS UNDER IFLAG = - CONTROL

OUTPUT MY EQUIVALENCES TO COMMON STORAGE

OUTPUT DEFINITIONS

PP = FOOTPAD RADIUS
PL = UPPER STRUT STROKE FORCE
ALPHA = ANGLE BETWEEN LOWER HARDFLOORS
CLC = INITIAL UPPER STRUT LENGTH
E = TOTAL VEHICLE ENERGY
SL = FINAL UPPER STRUT LENGTH
SKL = VEHICLE INCOMING KINETIC ENERGY
PE1 = POTENTIAL ENERGY RESULTING FROM C G 406P FROM INITIAL ORIENTATION TO ALL FOOTPAD ON SURFACE
PE2 = POTENTIAL ENERGY RESULTING FROM C G 406P RESULTING FROM STROKING OF UPPER STRUT
R1 = UPPER HARDFLOOR RADIUS
R2 = LOWER HARDFLOOR RADIUS
STOW = STOWABILITY INDEX

SUBROUTINE GEOM

COMMON AN, ALPHA, E, PHI, H(I), DELTAP, DIFF, EKE, H1, F1, F2, FAC, FOC, I1, GAMMA, IF1, H2(I), H1, IFLAG, IJK, IL(K), NUM, PE1, PE2, MASS, PSI, R1, R2, RH, RP, L, FLU, STOW, THETA, VRASS, XVEL, YVEL, Z, ZETA

DETERMINE VEHICLE GEOMETRY

1  PP=RH*H(I)
   A=PP*COS(THETA)
   VV=XVEL*COS(ZETA)
   VH=YVEL*SIN(ZETA)

DETERMINE KINETIC ENERGY

30 EKE=0.5*(VRASS*AN*PMASS)*(VH-VV)*(VH-VV)

DETERMINE PE1

   P1=H(I)*COS(PSI+ZETA)
   C1=A*SIN(PHI+ZETA)
   P2=H(I)*COS(ZETA)
   C2=A*SIN(ZETA)

40 PE1 = VRASS*G*(R1-C1-2+C2) * D1S = (H(I)-HN-FAC)

DETERMINE PE2

   PL2 = VRASS*G*COS(ZETA)*D1S
   R4 = (H(I)-DELTAP-O11-D)*TAN(DELTA)
   R3 = RP-R4

DETERMINE ANGLE BETWEEN LOWER HARDFLOORS

50 ALPHA=2.0*TAN(Z/R3)

Figure 2-9. Subroutine GEOM (Continued)
IF (ALPHA - 1.5706) < 57.5, 57.5
53  ALPHA = 1.5706
54
DETERMINE LOWER HANDPOINT RADIUS
55  R2 = 3/R3/COS(ALPHA/2.01
56  U0 = SQRT((R4*R4 + (H(I) - DELTAP-D11 - D)*(H(I) - DELTAP-D11 - D))
57  R5 = (H(I) - DELTAP-D11)*TAN(GAMMA)
58
DETERMINE RADIUS OF UPPER HANDPOINT
59  R1 = R.P - R5
60
COMPUTE STOAxABILITY INDEX
61  SOW = SQRT(RP*RP + (H(I) - DELTAP-D11)*(H(I) - DELTAP-D11))
62
FIND INITIAL LENGTH OF UPPER STRUT
63  SLO = (H(I) - DELTAP-D11)/COS(GAMMA)
64
FIND TOTAL VEHICLE ENERGY
65  E = SQRT(UO*UO - (H(I) - D11-D-DHS)*(H(I) - D11-D-DHS))
66  SL = SQRT((E*R3-R1)*(E+R3-R1)*(H(I) - D11-DHS)*(H(I) - D11-DHS))
67  DS = SLO - SL
68  IF (DS-0.005) BE, 85, 90
69  PRINT 615, DS
70  H(I) = H(I) + 0.1
71  IJK = IJK + 1
72  IF (IJK-20) 100, 100, 95
73
FIND UPPER STRUT STROKE LOAD
74  F11 = (EKE+PE1+PE2)/(AN*DS)
75  LL = 3
76  RETURN
77  LL = 2
78  RETURN
79  LL = 1
80  RETURN
81  615 FORMAT(70H THE DIFFERENCE SLO-SL IS TOO SMALL, REPEAT USING H=H + 0)
82  1.1  SLO-SL =, F15.5//)
83
END

Figure 2-9. Subroutine GEOM (Concluded)
**Figure 2-10. Subroutine INTER**

BENDIX PRODUCTS AEROSPACE DIVISION
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>THF30</td>
<td>1</td>
<td>0.7854</td>
<td></td>
</tr>
<tr>
<td>RH30</td>
<td>2</td>
<td>1.20</td>
<td></td>
</tr>
<tr>
<td>AN30</td>
<td>3</td>
<td>4.0</td>
<td></td>
</tr>
<tr>
<td>G30</td>
<td>4</td>
<td>5.32</td>
<td></td>
</tr>
<tr>
<td>DFL30</td>
<td>5</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>Z30</td>
<td>6</td>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>FOO30</td>
<td>7</td>
<td>7000.0</td>
<td>2126.0</td>
</tr>
<tr>
<td>NOH30</td>
<td>8</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>H30</td>
<td>9</td>
<td>14.10</td>
<td></td>
</tr>
<tr>
<td>THE30</td>
<td>1</td>
<td>0.7854</td>
<td></td>
</tr>
<tr>
<td>RH30</td>
<td>2</td>
<td>1.20</td>
<td></td>
</tr>
<tr>
<td>AN30</td>
<td>3</td>
<td>4.0</td>
<td></td>
</tr>
<tr>
<td>G30</td>
<td>4</td>
<td>5.32</td>
<td></td>
</tr>
<tr>
<td>DFL30</td>
<td>5</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>Z30</td>
<td>6</td>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>FOO30</td>
<td>7</td>
<td>7000.0</td>
<td>2126.0</td>
</tr>
<tr>
<td>NOH30</td>
<td>8</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>H30</td>
<td>9</td>
<td>14.10</td>
<td></td>
</tr>
</tbody>
</table>

FIN

Figure 2-10. Subroutine INTER (Concluded)
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta E$</td>
<td>$-4.35740$</td>
</tr>
<tr>
<td>$\Delta F$</td>
<td>$4.00000$</td>
</tr>
<tr>
<td>$\Delta G$</td>
<td>$5.31000$</td>
</tr>
<tr>
<td>$\Delta H$</td>
<td>$10.51000$</td>
</tr>
<tr>
<td>$\Delta I$</td>
<td>$5.00000$</td>
</tr>
</tbody>
</table>

Figure 2-11. Output Data for Mode IFLAG = -1
SECTION III

WEIGHT ANALYSIS DIGITAL COMPUTER PROGRAM

This computer program determines a gear system weight as a function of five landing gear parameters $R$, $H$, $\alpha$, $\beta$, and stroke loads $F_{11}$ in member 1-4 and $F_{22}$ in members 1-2 and 1-3 which are input data to the program.

The program will handle five different structural configurations referred to as Cases A, B, C, D, and E. See pages 3-1 through 3-5 in Section III of Reference 1.

NOTE

All references to page numbers in this write-up refer to pages in Section III of Reference 1.

The computer program first optimizes the design of the stroking members 1-2 and 1-4 (see pages 3-5 through 3-21), and second it optimizes the design of the truss members (see pages 3-22 through 3-28).

Please note that Case A has no truss structure. Case E has a modified truss which differs from Cases B, C, and D, and will require separate equilibrium equations.

The computer input data card consists of one set of the five basic parameters, a Flag B to designate the particular Case in question, and a serial number.

<table>
<thead>
<tr>
<th>INPUT DATA</th>
<th>COLUMNS IN CARD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Nos.</td>
<td>1 through 10</td>
</tr>
<tr>
<td>R</td>
<td>11 through 19</td>
</tr>
<tr>
<td>H</td>
<td>20 through 28</td>
</tr>
<tr>
<td>ALPHA</td>
<td>29 through 37</td>
</tr>
<tr>
<td>BETA</td>
<td>38 through 46</td>
</tr>
<tr>
<td>$F(1) = F_{22}$</td>
<td>47 through 55</td>
</tr>
<tr>
<td>$F(2) = F_{11}$</td>
<td>56 through 64</td>
</tr>
<tr>
<td>FLAGB</td>
<td>65 through 72</td>
</tr>
</tbody>
</table>

Note 1. Serial No. may be numeric and (or) alphabetic. Alphabetic is used for the five test sets of parameters only.
Note 2. Flag B = + 2.0 for Case A  
Flag B = + 1.0 for Cases B, C, D  
Flag B = - 1.0 for Case E

Note 3. If material other than aluminum is studied, then the five cards FTU(1), FTU(2), FTU(3), E, and DENS in the subroutine INPUT must be changed (see page 3-28 of Reference 1).

Note 4. See computer program listing (Figure 3-9) for identification of symbols.

COMPUTER OUTPUT DATA

1. The input data is printed first and then the density of the material.

2. A table of leg radius, wall thickness, and the product of the two is now printed as the program searches for the minimum point on the curve (see Figure 3-16) for member 1-2.

3. Then the weight, length, radius, wall thickness is printed for member 1-2.

4. Steps (2) and (3) are now printed for member 1-4.

5. For Case A only, a total leg weight, final total weight, weight of honeycomb, and weight of one footpad is now printed and program returns to read next set of data.

6. For the other four cases, the program now designs the truss members and prints the weight, force, length, radius, and wall thickness of each truss member for the particular case in question.

7. Then (5) is repeated for Case B, C, D, or E.

Note 1. The computer sample solution that follows gives the solution to a set of parameters for each of the five cases. The serial number is replaced by the words Test A, Test B, etc.
This program minimizes the leg weight based upon a given leg configuration and also calculates total leg weight including honeycomb and footpads. (See Reference 1.)

Figure 3-2. Computer Program: Design of Legs to Minimize Weight
Figure 3-2. Computer Program: Design of Legs to Minimize Weight (Concluded)
This subroutine reads and prints one data card and sets material constants

```
INPUT

101
Read and print one data card

Set material constants FTU(1) = FTU(3), FTU(2), E, Density → these 5 cards are now set for aluminum

Note: These five cards must be changed for different materials such as beryllium, etc.

Return
```

Figure 3-3. Subroutine Input
This subroutine initializes geometric constants, calculates lengths of members 1-2 and 1-4 and the weight of one footpad.

**Figure 3-4. Subroutine GEDM**
This subroutine finds the optimum leg radius and wall thickness to minimize leg weight.

Figure 3-5. Subroutine CURVE
Figure 3-5. Subroutine CURVE (Continued)
Figure 3-5. Subroutine CURVE (Concluded)
This subroutine finds the cosine of the angle between any two leg members. Before entering this subroutine N must be set between 1 and 33 so as to pick up the proper set of 5 nos. in the data table that determines the 2 leg member in question.

\[
\begin{align*}
I &= \Pi(N, 1) \\
J &= \Pi(N, 2) \\
K &= \Pi(N, 3) \\
L &= \Pi(N, 4) \\
M &= \Pi(N, 5)
\end{align*}
\]

Calculate cosine of the angle between the 2 leg members - COSANG (M)

Figure 3-6. Subroutine VECTOR
This subroutine solves a set of simultaneous equilibrium force equations for cases B or C or D. They are solved for forces in members 4-2, 4-8, 2-8, 2-6, 2-5. Sum locations are zeroed before 1st entry.

Figure 3-7. Subroutine EQBCD
This subroutine solves a set of simultaneous equilibrium force equations for Case E only. They are solved for forces in members 4-2, 4-8, 2-8, 2-7, 3-5, 2-5, 8-5, 8-10, 8-14, 5-13, and 5-10. The sum locations are zeroed before the 1st entry.

Figure 3-8. Subroutine EQE
TITLE OF MAIN PROGRAM DESIGN OF LEGS TO MINIMIZE WEIGHT

AUTHOR H. O. CARK

DATE 05-1-64

CALL LEGXT

PURPOSE TO MINIMIZE LEG WEIGHT BASED UPON A GIVEN INPUT
GEOMETRIC CONFIGURATION, AND THEN TO CALCULATE TOTAL LEG
WEIGHT INCLUDING WEIGHT OF HONEYCOMBS WITHIN EACH LEG
AND THE WEIGHT OF ALL FOUR FOOTPADS.

SEE BENDIX REPORT WP-8-64
SEE FIGURES 3-1, 3-2, 3-9, 3-11, 3-13, 3-14, 3-16

THIS PROGRAM WILL HANDLE FIVE DIFFERENT LEG TRUSS
DESIGNS AS FOLLOWS.

<table>
<thead>
<tr>
<th>CASE</th>
<th>SERIAL NO. RANGE</th>
<th>SEE DRAWING NO</th>
<th>FLAG B</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>22,000 TO 22,999</td>
<td>A</td>
<td>+2.0</td>
</tr>
<tr>
<td>B</td>
<td>23,000 TO 23,999</td>
<td>B</td>
<td>+1.0</td>
</tr>
<tr>
<td>C</td>
<td>24,000 TO 24,999</td>
<td>C</td>
<td>+1.0</td>
</tr>
<tr>
<td>D</td>
<td>25,000 TO 25,999</td>
<td>D</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>26,000 TO 26,999</td>
<td>E</td>
<td>-1.0</td>
</tr>
</tbody>
</table>

INPUT DATA CARD (ONE CARD ONLY)

SYMBOL COLUMNS USED

| SERIAL NO. | 1 THRU 10 |
| R           | 11 THRU 19 |
| H           | 20 THRU 28 |
| ALPHA       | 29 THRU 37 |
| BETA        | 38 THRU 46 |
| F(1) = F22  | 47 THRU 55 |
| F(2) = F11  | 56 THRU 64 |
| FLAGR       | 65 THRU 72 |

LIST OF SYMBOLS FOLLOWS.

X(N),Y(N),Z(N) COORDINATE LOCATIONS OF LEG JOINTS FOR N=1 THRU 14

SERIAL(1) INPUT CARD SERIAL NO
SERIAL(2) INPUT CARD SERIAL NO
R INPUT GEOMETRIC DATA - SEE DRAWINGS SAME
H SAME (IN DEGREES)
ALPHA SAME (IN DEGREES)
BETA

Figure 3-9. Main Program
F22
F11
FLAG R

FURTHER SYMBOLS

ALPRD
BEPRD
PHI
RONE
RTWO
GAMMA
XL(1)
XL(2)
PBAR
A1
A2
A4
RBAR
XY(1)
XY(2)
TMIN
UM1(1)
UM1(2)
UM2(1)
UM2(2)
XK1 = 0.65
XK2 = 2.0
VFP
FTU(1)
FTU(2)
FTU(3)
DENS
F
PRFV
TPREV
XL(I)
F(I)
XMTOT
TRMAX
COSANG(M)
R2X,R2Y,R2Z
R4X,R4Y
F42,F48,F28,F26,F25

DEFINITIONS

ALPHA IN RADIANS
BETA IN RADIANS
TRUSS ANGLE - SEE DRAWING FOR CASE NO.
GEOMETRIC DIMENSION - SEE DRAWING
SAME
LENGTH OF MEMBER 1-2
LENGTH OF MEMBER 1-4
VERTICAL REACTION ON THE FOOTPAD
DECELERATION FACTOR
SAME
SAME
RADIUS OF FOOTPAD IN INCHES
BEAM MASS
SAME
MINIMUM WALL THICKNESS OF LEGS
ALPRD (SEE ABOVE)
BEPRD (SEE ABOVE)
ALPRD (SEE ABOVE)
PI/2 RADIANS
SAME FOR MEMBER 1-2
SAME FOR MEMBER 1-4
WEIGHT OF ONE FOOTPAD
ULTIMATE TENSILE STRENGTH IN MEMBER 1-2
SAME IN MEMBER 1-4
SAME IN MEMBER 1-2
DENSITY OF MATERIAL
MODULUS OF ELASTICITY
FINAL RADIUS FOR MEMBERS 1-2 AND 1-4
FINAL WALL THICKNESS FOR SAME MEMBERS
LENGTH OF MEMBERS 1-2 AND 1-4 FOR I = 1,2
STROKE LOAD IN MEMBERS 1-2 AND 1-4
TOTAL STRUT MOMENT
MAXIMUM VALUE OF THE QUOTIENT OF WALL THICKNESS TO RADIUS
COSINE OF THE ANGLE BETWEEN TWO VECTORS IN SPACE (FOR THE TRUSS LOADINGS)
FORCE IN MEMBERS 4-2,4-8,2-8,2-6,2-5 FOR CASES B,C,D

Figure 3-9. Main Program (Continued)
THE FOLLOWING DATA STATEMENT CONTAINS ALL THE JOINT ENDPOINT NOS. REQUIRED TO CALCULATE THE COSINE OF THE ANGLE BETWEEN ALL THE NECESSARY MEMBERS FOR ALL FIVE CONFIGURATIONS.

A SET OF FIVE NOS. IS REQUIRED FOR EACH CALCULATION IN THE SUBROUTINE VECTOR. THE FIRST SET (2,4,4,8,1) FOR EXAMPLE MEANS --- THE ANGLE BETWEEN MEMBERS 2-4 AND 4-8 AND STORE IT AS COSANG(1)

CALL SUBROUTINE (INPUT) WHICH READS ONE DATA CARD AND THEN PRINTS THIS DATA.
THE NEXT SUBROUTINE (GEOM) SETS UP INITIAL VALUES, 
CALCULATES GEOMETRIC CONSTANTS FOR LEG CONFIGURATION, 
CASE A OR B OR C OR D OR E, CALCULATES LENGTH OF MEMBER 
1-2 AND 1-4 AND CALCULATES WEIGHT OF ONE FOOTPAD

102 CALL GEOM
110 DO 255 I=1,2,1
PRINT 902
T=0,028
TRMAX=FT(I)/(15*F)
X=1=X-(1)/(12*0*12*0*XL(I))*2/16*0*(A1+A2)*COS(U1(I))*XK2
IF (I-2) 113,112,112
X=TOT=X(I)
GO TO 114
112 X=TOT=X(I)
GO TO 114
113 UI=ATAN((5.0/RTWO*COS(PHI)))
SINLAM=(H-7.575)/XL(I)
COSLAM=5.0/((1)*SIN/UI))
XY2=0.20/3.1416*PBAR*RBAR*COSLAM*(TAN(GAMMA)+1.0/TAN(GAMMA))*XK2
XY2=0.60/3.1416*PBAR*RBAR*COSLAM*XK2
ETA=ATAN((XY2*XY2)/XY2)
XMTOT=(XM1+XM2)/SIN(FTA)
GO TO 114
114 CALL CURVE
GO TO (100,250),JJ

NOW CALCULATE WEIGHT OF LEG (I=2 FOR I=1 AND 1-4 FOR 
I=2), WEIGHT OF HONEYCOMB AND PRINT WEIGHT OF LEG, 
LENGTH, RADIUS = RIPREV, WALL THICKNESS = TPREV

250 WT(I)=(DENS)*2.0*RIPREV*TPREV*12.0*XL(I)*3.1416
WHC(I)=1.0*3.1416*RIPREV/RIPREV/1728.0*0.45*12.0*XL(I)/0.75/127.3
1*IF(I)/(1.0*3.1416*RIPREV/RIPREV+300.0)
PRINT 930,2(L)
255 PRINT 904,WT(I),XL(I),RIPREV,TPREV,P

NOW IF THIS IS A CASE A RUN THEN PRINT TOTAL LEG 
WEIGHT AND THEN GO TO 100 TO READ NEXT DATA CARD 
CASE A HAS NO TRUSS STRUCTURE
DUE TO SYMMETRY THERE ARE 8 MEMBERS LIKE 1-2 AND 4 
MEMBERS LIKE 1-4 INCLUDING ALL FOUR LEGS
FOR CASES B,C,D,E, GO ON TO PART TWO TO CALCULATE 
THE TRUSS LOADS AND LEG SIZES

260 IF (FLAGB-2.) 300,265,265
265 WTTOT=8.0*WT(1)+4.0*WT(2) 
PRINT 907,WTTOT
FINWT=WTTOT+4.0*WFP+4.0*WHC(2)+8.0*WHC(1)
PRINT 908,FINTOT+WFP*WHC(1)+WHC(2)
PRINT 1500
907 FORMAT(24HO TOTAL LEG WEIGHT =,F10,5/)

Figure 3-9. Main Program (Continued)
C:

NAME: FMT

105 FORMAT(1H1)
GO TO 100

SETUP POINTS 27,28,29,30 TO GIVE UNIT VECTORS

200 X(27)=0.0
Y(27)=0.0
Z(27)=0.0
X(28)=1.0
Y(28)=0.0
Z(28)=0.0
X(29)=0.0
Y(29)=1.0
Z(29)=0.0
X(30)=0.0
Y(30)=0.0
Z(30)=1.0

CALCULATE LEG LENGTHS

FOR CASES E OR C OR D

MEMBER SYMBOL
4-2 = DL(1)
4-6 = DL(2)
2-8 = DL(3)
2-6 = DL(4)
2-5 = DL(5)

205 DL(1) = SQRT((X(4)-X(2))**2+(Y(4)-Y(2))**2+(Z(4)-Z(2))**2)
DL(2) = SQRT((X(4)-X(8))**2+(Y(4)-Y(8))**2+(Z(4)-Z(8))**2)
DL(3) = SQRT((X(2)-X(8))**2+(Y(2)-Y(8))**2+(Z(2)-Z(8))**2)
DL(4) = SQRT((X(2)-X(6))**2+(Y(2)-Y(6))**2+(Z(2)-Z(6))**2)
DL(5) = SQRT((X(2)-X(5))**2+(Y(2)-Y(5))**2+(Z(2)-Z(5))**2)

CALCULATE THE COSINE OF THE ANGLE BETWEEN 9 PAIRS OF VECTORS IN SPACE (LEG MEMBERS)

310 CALL VECTOR
N=2
CALL VECTOR
N=3
CALL VECTOR
N=4
CALL VECTOR
N=5
CALL VECTOR
N=6
CALL VECTOR
N=7
CALL VECTOR
N=8
CALL VECTOR
N=9
CALL VECTOR

SET 20 LOCATIONS FOR SUM OF FORCES SF(N) EQUAL TO ZERO
FOR CASES E,C,D THE SUM SF(N) FOR N=1 THRU 5 EQUALS FORCES F42,
F48, F28, F26, F25 FOR LEG MEMBERS 4-2, 4-8, 2-8, 2-6, AND 2-5
FOR CASE E THE SUM SF(N) FOR N=1 THRU 11 EQUALS FORCES

Figure 3-9. Main Program (Continued)
IF (FLAGB) 500, 500, 325

NOW CALCULATE THE X, Y, Z COMPONENTS OF REACTIONS R2X, R2Y, R2Z, R4X, AND R4Y FOR JOINTS TWO AND FOUR FOR LOADINGS (1) UPPER LEG STROKE, (2) UPPER LEG BEAM ACTION, (3) LOWER LEG STROKES, (4) LOWER LEG BEAM ACTION, AND (5) ECCENTRIC IMPACT AND THEN ENTER SUBROUTINE EQBCD TO CALCULATE SUM OF FORCES.

325  R2X=0.0
     R2Y=0.0
     R2Z=0.0
     R4X=F(2)*COS(BETRD)
     R4Y=F(2)*SIN(BETRD)
     CALL EQBCD

320  R2X=0.0
     R2Y=0.0
     R2Z=0.0
     R4X=-XM(2)*XL(2)*COS(BETRD)*SIN(BETRD)*XK2/3.0*(A1/2.0*C+A4)
     R4Y=-XM(2)*XL(2)*(COS(BETRD)**2)*XK2/9.0e_(A1/2.0*C+A4)
     CALL EQBCD

335  R2X=0.0
     R2Y=0.0
     R2Z=0.0
     R4X=-F(1)*COS(ALPRD)*COS(GAMMA)
     R4Y=F(1)*COS(ALPRD)*SIN(ALPRD)
     R4Z=F(1)*COS(ALPRD)*SIN(GAMMA)
     CALL EQBCD

340  R2X=0.0
     R2Y=0.0
     R2Z=0.0
     R4X=-XM(1)*XL(1)*COS(ALPRD)*SIN(ALPRD)*XK2/3.0*(A1/2.0*C+A2)
     R4Y=-XM(1)*XL(1)*(COS(ALPRD)**2)*XK2/9.0e_(A1/2.0*C+A2)
     CALL EQBCD

345  R4X=0.0
     R4Y=0.0
     R4Z=0.0
     R2X=PBAR*RBR/((2.0*12.0*XL(1))*(COS(ALPRD)*SIN(ALPRD))*(TAN(GAMMA)+11.0/TAN(GAMMA)))*XK2)
     R2Y=PBAR*RBR/((2.0*12.0*XL(1))*(COS(ALPRD)**2)*(TAN(GAMMA)+11.0/TAN(GAMMA)))*XK2)
     CALL EQBCD

THIS SUM OF FORCES FOR EACH MEMBER 4-2, 4-6, 2-8, 2-6, AND 2-5 IS NOW USED TO DESIGN EACH MEMBER BY EQUATING ULTIMATE STRESS TO LOCAL BUCKLING ALLOWABLE STRESS TO FIND WALL THICKNESS T AND RADIUS RR.

Figure 3-9. Main Program (Continued)
C0 DO 365 I=1,5,1
C0 RTMIN=1.5*SF(I)/12.0*3.1416*FTU(31)
C0 TMN=SQR((1.5*SF(I))/13.0*3.1416*E11)
C0 IF (TMN-0.75) 360,365,365
C0 TMN=.025
C0 RR=(1.5*SF(I))*(12.0*DL(I)**2)/(E*TMN*13.1416)**3/(1.0/3.0)
C0 IF (RR*TMN-RTMIN) 375,380,380
C0 TPRIME=RTMIN/RR
C0 TMN=TPRIME
C0
C0 NOW CALCULATE AND PRINT WEIGHT OF MEMBER, FORCE IN MEMBER,
C0 LENGTH, RADIUS, AND WALL THICKNESS FOR THE FIVE TIMES THRU
C0 THE DO LOOP.
C0 CALCULATE AND PRINT TOTAL LEG WEIGHT (DUE TO SYMMETRY
C0 MEMBER 2-5 = 3-7, 2-6 = 3-6, AND 2-8 = 3-8) OF ALL FOUR LEGS.
C0 THEN PRINT FINAL WEIGHT INCLUDING TOTAL LEG WEIGHT, TOTAL
C0 HONEYCOMB WEIGHT, AND FOOTPAD WEIGHT
C0
C0 380 WTT(I)=DENS*2.0*3.1416*RR*TMIN*12.0*DL(I)
C0 PRINT 930,L3:1)
C0 385 PRINT 906,WTT(I),SF(I),DL(I),RR,TMIN
C0 390 WTTOT=4.0*(2.0*X(T1)+WTT(I)+WTT(3)+WTT(4)+WTT(5)+WTT(2)+WTT(2))
C0 391 PRINT 907,WTTOT
C0 FINWT=WTTOT+4.0*WFP+WHC(2)+2.0*WHC(1)
C0 PRINT 908,FINT,WFP,WHC(1),WHC(2)
C0 PRINT 1500
C0 400 GO TO 100

C0 CALCULATE MEMBER LENGTHS FOR CASE E MEMBERS ONLY
C0 MEMBER SYMBOL
C0 4-2 XLEN(1)
C0 4-8 XLEN(2)
C0 2-8 XLEN(3)
C0 2-7 XLEN(4)
C0 2-5 XLEN(5)
C0 8-5 XLEN(6)
C0 8-10 XLEN(7)
C0 8-14 XLEN(8)
C0 5-13 XLEN(9)
C0 5-10 XLEN(10)
C0 5-9 XLEN(11)
C0
C0 500 XLEN(1)=SQRT((X(4)-X(2))**2+(Y(4)-Y(2))**2+(Z(4)-Z(2))**2)
C0 XLEN(2)=SQRT((X(4)-X(8))**2+(Y(4)-Y(8))**2+(Z(4)-Z(8))**2)
C0 XLEN(3)=SQRT((X(2)-X(8))**2+(Y(2)-Y(8))**2+(Z(2)-Z(8))**2)
C0 XLEN(4)=SQRT((X(2)-X(7))**2+(Y(2)-Y(7))**2+(Z(2)-Z(7))**2)
C0 XLEN(5)=SQRT((X(2)-X(5))**2+(Y(2)-Y(5))**2+(Z(2)-Z(5))**2)
C0 XLEN(6)=SQRT((X(8)-X(5))**2+(Y(8)-Y(5))**2+(Z(8)-Z(5))**2)
C0 XLEN(7)=SQRT((X(5)-X(9))**2+(Y(5)-Y(9))**2+(Z(5)-Z(9))**2)
C0 XLEN(8)=SQRT((X(8)-X(10))**2+(Y(8)-Y(10))**2+(Z(8)-Z(10))**2)
C0 XLEN(9)=SQRT((X(5)-X(10))**2+(Y(5)-Y(10))**2+(Z(5)-Z(10))**2)
C0 XLEN(10)=SQRT((X(5)-X(10))**2+(Y(5)-Y(10))**2+(Z(5)-Z(10))**2)
C0
C0 NOW CALCULATE THE COSINE OF THE ANGLE BETWEEN 11 PAIRS OF
C0 VECTORS (LEG MEMBERS) IN SPACE.
C0 505 N=10

Figure 3-9. Main Program (Continued)
CALL VECTOR
N=11
CALL VECTOR
N=12
CALL VECTOR
N=13
CALL VECTOR
N=14
CALL VECTOR
N=15
CALL VECTOR
N=16
CALL VECTOR
N=17
CALL VECTOR
N=18
CALL VECTOR
N=19
CALL VECTOR
N=20
CALL VECTOR
N=21
CALL VECTOR
N=22
CALL VECTOR
N=23
CALL VECTOR
N=24
CALL VECTOR
N=25
CALL VECTOR
N=26
CALL VECTOR
N=27
CALL VECTOR
N=28
CALL VECTOR
N=29
CALL VECTOR
N=30
CALL VECTOR
N=31
CALL VECTOR
N=32
CALL VECTOR
N=33
CALL VECTOR

NOW CALCULATE THE X, Y, Z COMPONENTS OF REACTIONS R2X, R2Y, R2Z, R4X, AND R4Y FOR JOINTS TWO AND FOUR, FOR LOADINGS (1) UPPER LEG STROKE, (2) UPPER LEG BEAM ACTION, (3) LOWER LEG STROKES, (4) LOWER LEG BEAM ACTION, AND (5) ECCENTRIC IMPACT, AND THEN ENTER SUBROUTINE EQBCD TO CALCULATE SUM OF FORCES.

510 R2X=0.0
R2Y=0.0
R2Z=0.0

Figure 3-9. Main Program (Continued)
R4X = -F(2)*COS(PI*TR)  
R4Y = F(2)*SIN(PI*TR)  
CALL EOF

515 R2X = 0.0  
R2Y = 0.0  
R2Z = 0.0  
R4X = -XM(2)*XL(2)*COS(ET)*SIN(BET)*RX/3.0*(A1/2.0+A4)  
R4Y = -XM(2)*XL(2)*COS(BETR)*RX*2*RX/3.0*(A1/2.0+A4)  
CALL EOF

520 R4X = 0.0  
R4Y = 0.0  
R2X = -F(1)*COS(ALPRD)*COS(GAMMA)  
R2Y = F(1)*COS(GAMMA)*SIN(ALPRD)  
R2Z = F(1)*COS(ALPRD)*SIN(GAMMA)  
CALL EOF

525 R4X = 0.0  
R4Y = 0.0  
R2X = -XM(1)*XL(1)*COS(ALPRD)*SIN(ALPRD)*RX/3.0*(A1/2.0+A2)  
R2Y = -XM(1)*XL(1)*COS(ALPRD)*RX/3.0*(A1/2.0+A2)  
CALL EOF

530 R4X = 0.0  
R4Y = 0.0  
R2X = BAR*RR/BAR/12.0*XL(1)*COS(ALPRD)*SIN(ALPRD)*(TAN(GAMMA)**2.0/TAN(GAMMA)**2.0)*XL(1)**2  
R2Y = BAR*RR/BAR/12.0*XL(1)*COS(ALPRD)*SIN(ALPRD)*(TAN(GAMMA)**2.0/TAN(GAMMA)**2.0)*XL(1)**2  
CALL EOF

540 DO 575 I = 1, 11, 1  
J = I+5  
RTMIN = 1.5*SF(I)/((2.0*3.1416*FU(J)))  
TMIN = SQRT(1.5*SF(I)/((3.0*3.1416*E)))  
545 IF (TMIN < 0.25) 550, 555, 555  
550 TMIN = 0.25  
555 RR = ((1.5*SF(J)**(1.0/2.0)*XL(1)**2)/(E*TMIN**3.1416)**3)**(1.0/3.0)  
560 IF (RTMIN < RTMIN) 565, 570, 570  
565 TPRIME = RTMIN/RR  
TPRIME = TPRIME

NOW, CALCULATE AND PRINT WEIGHT OF MEMBER FORCE IN MEMBER, LENGTH, RADIUS, AND WALL THICKNESS FOR EACH OF THE ELEVEN TIMES THRU THIS DO LOOP.

CALCULATE AND PRINT TOTAL LEG WEIGHT OF ALL FOUR LEGS. THEN, PRINT FINAL WEIGHT INCLUDING TOTAL LEG WEIGHT, TOTAL HONEYCOMB WEIGHT, AND TOTAL FOOTPAD WEIGHT.

570 WTTT(I) = DEN**2.0*3.1416*RR*TMIN**12.0*XL(1)  
PRINT 930, XL(1)  
575 PRINT 906, WTTT(I), SF(J), XL(1), RR, TMIN  
580 WTTTOT = WTTT(I)**2.0**2.0**2.0**2.0**2.0  
PRINT 907, WTTTOT

Figure 3-9. Main Program (Continued)
FINWT+WT=WT
PRINT 906,FINWT,WF,WPHC(1),WPHC(2)
PPINT 1500
590 CC TO 100
1971 FORMAT(6F20.1C)
901 FORMAT(5H R=F9.5,4X,2HM=F8.4,4X,6HALPHA=F7.3,4X,5HΒETA=F7.3,
14X,4HF22=F7.1,4X,4HF11=F7.1,4X,6HFLAGB=F4.1,4X,8HDENSITY=F8.4/2)
902 FORMAT(44H R1 T RT P/) 
903 FORMAT(F12.5,F11.5,F13.5,F12.5)
904 FORMAT(10H WEIGHT=F10.5,12H LENGTH=F9.5,10H R1PREV=F9.5,
11H TPREV=F9.5,7H P=F9.5/)
905 FORMAT(10H WEIGHT=F10.5,11H FORCE=F8.1,12H LENGTH=F9.5,
15H R=F9.5,10H TMIN=F9.5)
909 FORMAT(5X,8HFUN,2X,3HNEG,2X,8HNEG,F2X,3HYET)
910 FORMAT(13H SERIAL NO =,A6,A4/)
915 FORMAT(6H LEG,2X,A6)
END

Figure 3-9. Main Program (Concluded)
TITLE OF SUBROUTINE: INPUT

AUTHOR: H. C. CARR

DATE: 05-1-64

CALL: INPUT

PURPOSE: TO READ ONE DATA CARD AND THEN INITIALLY MATERIAL CONSTANTS FTU(1), FTU(2), FTU(3), E, AND DENSITY (THESE FIVE CONSTANTS MUST BE CHANGED FOR DIFFERENT MATERIALS SUCH AS ALUMINUM, BERYLLIUM, ETC.) AND THEN PRINT THE DATA CARD

SYMBOLS

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>DEFINITION OF INPUT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERIAL(1)</td>
<td>SERIAL NO OF DATA</td>
</tr>
<tr>
<td>SERIAL(2)</td>
<td>SAME</td>
</tr>
<tr>
<td>R</td>
<td>GEOMETRIC PARAMETER</td>
</tr>
<tr>
<td>H</td>
<td>SAME</td>
</tr>
<tr>
<td>ALPHA</td>
<td>SAME (IN DEGREES)</td>
</tr>
<tr>
<td>BETA</td>
<td>SAME (IN DEGREES)</td>
</tr>
<tr>
<td>F(1)</td>
<td>F22= STROKE LOAD IN MEMBER 1-4</td>
</tr>
<tr>
<td>F(2)</td>
<td>F11= STROKE LOAD IN MEMBER 1-2 AND 1-3</td>
</tr>
<tr>
<td>FLAG</td>
<td>FLAG B DEFINES THE LEG CONFIGURATION IN QUESTION. FOR CASES B, C, D FLAG B = +1.0</td>
</tr>
<tr>
<td></td>
<td>FOR CASE A FLAG B = +2.0</td>
</tr>
<tr>
<td></td>
<td>FOR CASE E FLAG B = -1.0</td>
</tr>
</tbody>
</table>

OTHER SYMBOLS

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTU(1)</td>
<td>ULTIMATE TENSILE STRENGTH IN MEMBER 1-2</td>
</tr>
<tr>
<td>FTU(2)</td>
<td>SAME IN MEMBER 1-4</td>
</tr>
<tr>
<td>FTU(3)</td>
<td>REPEAT OF MEMBER 1-2</td>
</tr>
<tr>
<td>DENS</td>
<td>DENSITY OF MATERIAL</td>
</tr>
<tr>
<td>E</td>
<td>MODULUS OF ELASTICITY</td>
</tr>
</tbody>
</table>

SUBROUTINE INPUT

COMMON/S1/ SERIAL(2), R, H, ALPHA, BETA, F(2), FLAGB, FTU(3), E, DENS
COMMON/S2/X(30), Y(30), Z(30), PHI, RONE, RTWO, A1, A2, A4, GAMMA, XL1, XL2,
       PRBAR, XM(2), RBAR, TM(2), UM1(2), UM2(2), X1, X2, F.
COMMON/S3/1, R1 PREV, TPREV, TDEL, XL1, R1, XM1, A, P, RT, TRMAX,
       RTRPREV, J1, J2, L, M, COSANG(30)
COMMON/S4/R4Y, R4X, R2Y, R2Z, R2X, F42, F48, F26, F25, F27, F35, F85,
       F610, F814, F813, F613, F59, SF1(30), I1
COMMON/S5/MM(14), MM(11), WM(2), WM(2), HC(2), DL(14), XLEN(11)
COMMON/S7/ALPRD, BETRD
101 READ 900, SERIAL(1), SERIAL(2), R, H, ALPHA, BETA, F(1), I=1,2, FLAGB
       FTU(1) = 61500.0
       FTU(2) = 53000.0
       FTU(3) = 61500.0
       E = 1000000.0

Figure 3-10. Subroutine INPUT

BENDIX PRODUCTS AEROSPACE DIVISION
DENS = .102

PRINT 910, SERIAL(1), SERIAL(2)
PRINT 901, R, H, ALPHA, BETA, (F(I), I=1, 2), FLAG, DENS
900 FORMAT (A6, A4, 4F9.5, F9.1, F9.1, F7.1)
901 FORMAT (5H R = , F9.5, 4X, 2H = , F8.4, 4X, 6H, ALPHA = , F7.3, 4X, BETA = , F7.3, 14X, 4HF22 = , F7.1, 4X, 4HF11 = , F7.1, 4X, 6HFLAG5 = , F4.1, 4X, 8H DENSITY = , F8.4/2/)
910 FORMAT (13H SERIAL NO = , A6, A4/)
RETURN
END

Figure 3-10. Subroutine INPUT (Concluded)
C TITLE OF SUBROUTINE  GEOM
C AUTHOR       H. O. CARR
C CALL         GEOM
C
C PURPOSE      TO INITIALIZE GEOMETRIC CONSTANTS FOR CASE A OR B
C            OR C OR D OR E.
C            TO CALCULATE LENGTH OF MEMBERS 1-2 AND 1-4
C            TO CALCULATE WEIGHT OF ONE FOOTPAD
C
C SYMBOLS
C   X(N),Y(N),Z(N)   COORDINATE LOCATIONS OF LEG JOINTS
C   FOR N=1 THRU 14
C   ALPRD        ALPHA IN RADIANS
C   BETRD        BETA IN RADIANS
C   PHI          TRUSS ANGLE - SEE DRAWING FOR CASE NO.
C   RUNE         GEOMETRIC DIMENSION - SEE DRAWING
C   RTWO         SAME
C   GAMMA        SAME
C   XL(1)        LENGTH OF MEMBER 1-2
C   XL(2)        LENGTH OF MEMBER 1-4
C   PRAR         VERTICAL REACTION ON THE FOOTPAD
C   A1           DECELERATION FACTOR
C   A2           SAME
C   A4           SAME
C   RRAR         RADIUS OF FOOTPAD
C   XM(1)        BEAM MASS
C   XM(2)        SAME
C   TMIN         MINIMUM WALL THICKNESS OF LEGS
C   UM(1)        ALPRD
C   UM(2)        BETRD
C   UM2(1)       ALPRD
C   UM2(2)       PI/2 RADIANS
C   XK1          DYNAMIC TRANSIENT FACTOR FOR MEMBER 1-2
C   XK2          SAME FOR MEMBER 1-4
C   WFP          WEIGHT OF FOOTPAD
C
C SUBROUTINE GEOM
COMMON/S1/SERIAL(2),R,H ALPHA,BETA,F(2),FLAG8,FTU(3),E,DENS
COMMON/S2/X(30),Y(30),Z(30),PHI,RONE,RTWO,A1,A2,A4,GAMMA,XL1,XL2,
1PRAR, XM(12), RRAR, TMIN, UM(12), UM2(2), XK1, XK2, WFP
COMMON/S3/R1 PREV, TPREV, T, TDEL, XL(2), R1, XMTOT, A, P, RT, TRMAX,

Figure 3-11. Subroutine GEOM
THE FOLLOWING GEOMETRY IS THE SAME FOR CASES A, B, C, D, E

103 X(1) = R
    Y(1) = 0.1250-H
    Z(1) = 0.0
    ALPRD = ALPHA*3.1416/180.0
    BETRD = BETA*3.1416/180.0
    PHI = ATAN(5.00/(R-COS(ALPRD)/SIN(ALPRD)*(H-7.575)))
    RONE = R-(H-2.165)*COS(ALPRD)/SIN(PHI)
    RTWO = (R-(H-7.575)*COS(ALPRD)/SIN(ALPRD))/COS(PHI)
    X(2) = RTWO*COS(PHI)
    Y(2) = -7.325
    Z(2) = 5.0
    X(3) = RTWO*COS(PHI)
    Y(3) = 7.325
    Z(3) = -5.0
    X(4) = RONE
    Y(4) = -1.915
    Z(4) = 0.0
    X(6) = 7.08
    Y(6) = -7.325
    Z(6) = 0.0

104 IF(FLAGB = 1.0) 107, 105, 106

THESE POINTS ARE FOR CASES B, C, D ONLY

105 X(5) = 5.0
    Y(5) = -7.325
    Z(5) = 9.0
    X(7) = 5.0
    Y(7) = -7.325
    Z(7) = -5.0
    X(8) = 7.08
    Y(8) = -1.915
    Z(8) = 0.0
    A1 = 225.4
    A2 = 115.92
    A4 = 115.92
    GO TO 108

THESE POINTS ARE FOR CASE A ONLY.

106 X(5) = 5.0
    Y(5) = -7.325
    Z(5) = 9.0
    X(7) = 5.0
    Y(7) = -7.325
    Z(7) = -5.0
    X(8) = 7.08
    Y(8) = -1.915
    Z(8) = 0.0
    A1 = 225.4
    A2 = 70.84
    A4 = 70.84
    GO TO 108

Figure 3-11. Subroutine GEOM (Continued)
THESE POINTS ARE FOR CASE E ONLY.

107 X(5) = 14.58
Y(5) = -7.325
Z(5) = 4.61
X(7) = 14.58
Y(7) = -7.325
Z(7) = 4.81
X(8) = 14.58
Y(8) = -1.915
Z(8) = 0.0
X(9) = 5.0
Y(9) = -7.325
Z(9) = 5.0
X(10) = 7.08
Y(10) = -7.325
Z(10) = 0.0
X(11) = 5.0
Y(11) = -7.325
Z(11) = -5.0
X(12) = 5.0
Y(12) = -1.915
Z(12) = 5.0
X(13) = 7.08
Y(13) = -1.915
Z(13) = 0.0
X(14) = 5.0
Y(14) = -1.915
Z(14) = -5.0
A1 = 225.4
A2 = 144.9
A4 = 144.9
GO TO 108

CONTINUE GEOMETRY CALCULATIONS COMMON TO ALL FIVE CASES.

108 GAMMA = ATAN(ABS(5.0/((X(1)-X(2)))
XL(1)=SQR((X(1)-X(2))^2+(Y(1)-Y(2))^2+(Z(1)-Z(2))^2)
XL(2)=SQR((X(1)-X(4))^2+(Y(1)-Y(4))^2+(Z(1)-Z(4))^2)
RBAR = F(2)*SIN(ALPRD)+2.0*F(1)*COS(GAMMA)*SIN(ALPRD)
X(1) = .0930
X(2) = .0930
RBAR = 18.0
TMIN = .025
UM1(1) = ALPRD
UM1(2) = BETRD
UM2(1) = ALPRD
UM2(2) = 1.14159265/2.0
XK1 = .65
XK2 = 2.0

CALCULATE WEIGHT OF ONE FOOTPAD WFP

WFP = 1.0/1728.0*3.1416*RBAR**2*6.0/127.3*(1.20*DBAR/(1.0*DBAR**2
*3.1416) + 300.0)
RETURN
END

Figure 3-11. Subroutine GEOM (Concluded)
Title of Subroutine CURVE

Author: H. O. Carr

Purpose: To determine the optimum leg cross section configuration of wall thickness T and radius R1 to give the minimum weight.

Method: This is found by equating ultimate stress to the critical allowable stress and then by incrementing wall thickness T until the minimum point is found on the curve T vs. R1 where R1 is the product of T and R1 the leg radius.

This function of R1 is best solved by the normal iteration process: increase R1 until the value of the function changes sign and then half the interval until the value of the function approaches zero. Convergence is satisfied when two successive values of the variable meet a percent error test.

Output: Values of R1 vs. T are printed until the minimum point on the T vs. R1 curve is found.

Symbols:

- R1: Current leg radius
- T: Current wall thickness
- RIPREV: Previous leg radius
- RIPR: Second previous leg radius
- TPREV: Previous wall thickness
- TPR: Second previous wall thickness
- XL(I): Length of member 1=2 for I=1 and 1=4 for I=2
- XK1: Dynamic transient factor
- XMTCT: Total strut moment
- R2: Last value of variable that gave a positive value to function being solved by iteration.
- R3: Same as above except the function value is negative
- RT: Current product of R1 and T
- RTPREV: Previous product

Figure 3-12. Subroutine CURVE
Figure 3-12. Subroutine CURVE (Continued)
Figure 3-12. Subroutine CURVE (Continued)
COUNT=0.0
R1PRE=0.0
RIPREV=0.0
R1=R2+R3
R1DEL=R1**.5
195 A = 2.0*3.1416*R1**2*TRMAX
P = 1.5*F(I)/(12.0*XL(I))**2/(3.1416**3*E*K1*TRMAX*R1**4)
F2=-F(I)+1.5/S(F(I)+2.0/R1*MTOT*(1.0/(1.0-F(I)))
IF (F2 > 205.0) 195,202,202
200 R2=R1
R1F1=R1DEL/2.0
R1=R1+R1DEL
COUNT=COUNT+1.0
IF (COUNT=40.0) 195,202,202
202 PRINT 900
PRINT 1500
JJ = 1
RETURN
205 R3=R1
R1=(R2+R3)/2.0
210 A = 2.0*3.1416*R1**2*TRMAX
P = 1.5*F(I)/(12.0*XL(I))**2/(3.1416**3*E*K1*TRMAX*R1**4)
F2=-F(I)+1.5/S(F(I)+2.0/R1*MTOT*(1.0/(1.0-F(I)))
IF (F2 > 225.0) 225,225,215
215 R2=R1
220 R1PRE=R1
R1=(R2+R3)/2.0
GO TO 230
225 R3=R1
GO TO 220
230 RFIN=ABS(R1-R1PRE)/R1)-0.0001
IF (RFIN > 235.225,210
235 T=R1*TRMAX
RT=R1*T
PRINT 903,R1,T,RT,P
240 RIPREV=R1
RTPREV=RT
TPREV=T
C
C SINCE MINIMUM POINT ON T V.S. RT CURVE IS REACHED AT NEXT
C TO LAST VALUES OF R1 AND T, THEN FINAL VALUES NEEDED FOR
C CALCULATING LEC WEIGHT ARE THE PREVIOUS SET RIPREV AND TPREV
245 FLASC = -1.0
903 FORMAT(F12.5,F11.5,F13.5,F12.5)
909 FORMAT(F5X,8HFUNCTION,2X,3HNOT,2X,8HNEGATIVE,2X,3HYET)
1500 FORMAT(1X)
JJ = 2
RETURN
END

Figure 3-12. Subroutine CURVE (Concluded)
TITLE OF SUBROUTINE  VECTOR

AUTHOR  H. C. CARR

CALL  VECTOR

PURPOSE  TO FIND THE COSINE OF THE ANGLE BETWEEN ANY TWO LEG MEMBERS CALLED COSANG

METHOD  BY TAKING THE DOT PRODUCT BETWEEN TWO VECTORS IN THREE DIMENSIONAL SPACE

BEFORE CALLING THIS SUBROUTINE N MUST BE SET BETWEEN THE NUMBERS 1 AND 33 AS SO AS TO PICK UP THE PROPER SET OF FIVE NUMBERS IN THE DATA TABLE FOR THE TWO LEG MEMBERS IN QUESTION. THIS DATA STATEMENT OCCURS AT THE BEGINNING OF THE MAIN PROGRAM.

SYMBOLS  DEFINITION

X(N),Y(N),Z(N)  COORDINATES IN SPACE
WHERE N CAN EQUAL I OR J OR K OR L
N = 1 THRU 30

COSANG(M)  COSINE OF THE ANGLE BETWEEN TWO VECTORS IN SPACE

SUBROUTINE VECTOR
DIMENSION II(33,5)
COMMON/S1/ SERIAL(2),R,H,ALPHA,BETA,F(2),FLAGD,FTU(3),C,DENS
COMMON/S2/X(30),Y(30),Z(30),PHI,ROKE,TWO,A1,A2,A4,GAMMA,XL1,XL2,
1PPAR,XM(2),RRAR,MINA,UM(2),UM2(2),XX1,XX2,XXP
COMMON/S3/I,R1 PREV,TPREV,T,TDEL,XL(2),R1,XMTOT,AP,RT,TRMAX,
13TPREV,J,J,K,L,Y,COSANG(30)
1F810,F814,F813,F810,F859,SF(30),II
COMMON/S5/WTT(14),WTT(11),WT(2),WHC(2),OL(14),XLEN(11)
COMMON/S6/N;
COMMON/S7/ALPRED,BETRD

I = II(N,1)
J = II(N,2)
K = II(N,3)
L = II(N,4)
M = II(N,5)

AAI = X(I) - X(J)
RPI = Y(I) - Y(J)
CCI = Z(I) - Z(J)

Figure 3-13. Subroutine VECTOR
AA2 = X(K) - X(L)
BB2 = Y(K) - Y(L)
CC2 = Z(K) - Z(L)
AR = AA1*AA2 + BB1*BB2 + CC1*CC2
AABS = SQRT(AA1**2 + BB1**2 + CC1**2)
BABS = SQRT(AA2**2 + BB2**2 + CC2**2)
COANG(R) = AABS/AR / (AABS * BABS)
RETURN
END

Figure 3-13. Subroutine VECTOR (Concluded)
TITLE OF SUBROUTINE    EQBCD

AUTHOR    H. O. CARR

DATE       09-1-64

CALL       EQBCD

PURPOSE    TO SOLVE A SET OF SIMULTANEOUS EQUILIBRIUM EQUATIONS FOR FORCES IN MEMBERS 4-2, 4-8, 2-8, 2-6, AND 2-5 FOR CASES B, C, D

METHOD    THE ABSOLUTE VALUE OF THE FORCES ARE SUMMED FOR EACH OF THE FIVE SEPARATE LOADINGS ON THE TRUSS

SYMBOLS

R2X, R2Y, R2Z, R4X, R4Y  THE X,Y,Z COMPONENTS AT JOINTS 2 + 4

COSANG(M)  THE COSINE OF THE ANGLE BETWEEN THE DESIRED TWO VECTORS IN SPACE

F42, F48, F28, F26, F25  FORCE IN MEMBER 4-2, 4-8, 2-8, 2-6, 2-5 FOR EACH SEPARATE LOADING

SF(1)  ABSOLUTE SUM OF FORCE F42 FOR FIVE LOADS

SF(2)  SAME FOR F48

SF(3)  SAME FOR F28

SF(4)  SAME FOR F26

SF(5)  SAME FOR F25

SUBROUTINE EQBCD

COMMON/S1/ SERIAL(2), R, H, ALPHA, BETA, F(2), FLAGB, FTU(3), E, DENS

COMMON/S2/X(30), Y(30), Z(30), PHI, RUNE, XTU(6), A1, A2, A4, GAMMA, XL1, XL2,

PFAR, XM(2), RBAR, TMIN, UMI(2), UM2(2), XK1, XK2, WFP

COMMON/S3/I, R1 PREV, TPREVF, TDEL, XL(2), K1, XMOT, A, P, R, TRMAX,

TRPREV, JJ, K1, L1, COSANG(30)


FB10, FBI4, F513, F510, F59, SF(30), I1

COMMON/S5/WTT(14), WTTT(11), T(2), WHC(12), DL(14), XLEN(11)

COMMON/S7/ALPRO, BETRD

F42 = R4Y/(2.0*COSANG(5))

SF(1) = SF(1)+ABS(F42)

F48 = R4X - 2.0*F42*COSANG(1)

SF(2) = SF(2)+ABS(F48)

F28 = (-R2Y - F42*COSANG(1))/COSANG(6)

SF(3) = SF(3)+ABS(F28)

F26 = (R2Z - F42*COSANG(7) - F28*COSANG(8))/COSANG(9)

SF(4) = SF(4)+ABS(F26)

F25 = R2X - F28*COSANG(2) - F26*COSANG(3) - F42*COSANG(4)

SF(5) = SF(5)+ABS(F25)

RETURN

END

Figure 3-14. Subroutine EQBCD  3-35

BENDIX PRODUCTS AEROSPACE DIVISION
Figure 3-15. Subroutine EQE
Figure 3-15. Subroutine EQE (Concluded)
Figure 3-16. Sample Output - Case A
Figure 3-17. Sample Output - Case B

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>101.6532</td>
<td>0.02500</td>
<td>2.54233</td>
<td>0.0002</td>
</tr>
<tr>
<td>24.21635</td>
<td>0.03500</td>
<td>3.47571</td>
<td>0.0065</td>
</tr>
<tr>
<td>12.06633</td>
<td>0.04500</td>
<td>5.42967</td>
<td>0.0035</td>
</tr>
<tr>
<td>7.24988</td>
<td>0.05500</td>
<td>7.42499</td>
<td>0.0182</td>
</tr>
<tr>
<td>5.26477</td>
<td>0.06500</td>
<td>9.42214</td>
<td>0.0448</td>
</tr>
<tr>
<td>4.20120</td>
<td>0.07500</td>
<td>10.41596</td>
<td>0.0866</td>
</tr>
<tr>
<td>3.27152</td>
<td>0.08500</td>
<td>12.27807</td>
<td>0.1421</td>
</tr>
<tr>
<td>2.49431</td>
<td>0.09500</td>
<td>14.56668</td>
<td>0.2187</td>
</tr>
<tr>
<td>2.46315</td>
<td>0.10500</td>
<td>16.25638</td>
<td>0.2695</td>
</tr>
<tr>
<td>3.15804</td>
<td>0.08700</td>
<td>17.48428</td>
<td>0.1943</td>
</tr>
<tr>
<td>3.05492</td>
<td>0.08900</td>
<td>18.51109</td>
<td>0.1683</td>
</tr>
<tr>
<td>2.95562</td>
<td>0.09100</td>
<td>19.60929</td>
<td>0.1792</td>
</tr>
<tr>
<td>2.87137</td>
<td>0.09300</td>
<td>20.70749</td>
<td>0.1920</td>
</tr>
<tr>
<td>2.79431</td>
<td>0.09500</td>
<td>21.50509</td>
<td>0.2050</td>
</tr>
<tr>
<td>2.71513</td>
<td>0.09700</td>
<td>22.30379</td>
<td>0.2179</td>
</tr>
<tr>
<td>2.64280</td>
<td>0.09900</td>
<td>23.10179</td>
<td>0.2309</td>
</tr>
<tr>
<td>2.58190</td>
<td>0.10100</td>
<td>24.00059</td>
<td>0.2439</td>
</tr>
<tr>
<td>2.52107</td>
<td>0.10300</td>
<td>24.90959</td>
<td>0.2568</td>
</tr>
<tr>
<td>2.46315</td>
<td>0.10500</td>
<td>25.80859</td>
<td>0.2695</td>
</tr>
<tr>
<td>2.51772</td>
<td>0.10314</td>
<td>25.90789</td>
<td>0.2753</td>
</tr>
</tbody>
</table>

LEG 1-2

WEIGHT= 23.90764 LENGTH= 11.95022 R1R= 2.51572 TPRL= 10314 P= 25793

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.34339</td>
<td>0.04500</td>
<td>2.64454</td>
<td>3.1597</td>
</tr>
<tr>
<td>3.75169</td>
<td>0.05500</td>
<td>2.06334</td>
<td>73535</td>
</tr>
<tr>
<td>3.36018</td>
<td>0.06500</td>
<td>2.20323</td>
<td>84600</td>
</tr>
<tr>
<td>4.66263</td>
<td>0.07500</td>
<td>2.16324</td>
<td>46603</td>
</tr>
<tr>
<td>4.24226</td>
<td>0.08500</td>
<td>2.07978</td>
<td>56985</td>
</tr>
<tr>
<td>4.20298</td>
<td>0.09500</td>
<td>2.02529</td>
<td>64253</td>
</tr>
<tr>
<td>3.86997</td>
<td>0.10500</td>
<td>2.05111</td>
<td>69497</td>
</tr>
<tr>
<td>3.75169</td>
<td>0.11500</td>
<td>2.06334</td>
<td>73535</td>
</tr>
</tbody>
</table>

LEG 1-4

WEIGHT= 21.83283 LENGTH= 13.64690 R1R= 3.66997 TPRL= 10314 P= 73535

LEG 4-2

WEIGHT= 3.81394 FORCE= 6314.9 LENGTH= 7.63388 R= 2.01735 TM= 0.05419

LEG 4-6
<table>
<thead>
<tr>
<th>LEG</th>
<th>HEIGHT</th>
<th>FORCE</th>
<th>LENGTH</th>
<th>R</th>
<th>TMIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-8</td>
<td>3.19589</td>
<td>8337.9</td>
<td>6.22001</td>
<td>1.63401</td>
<td>0.03643</td>
</tr>
<tr>
<td>2-6</td>
<td>7.23624</td>
<td>13052.0</td>
<td>8.48975</td>
<td>2.43168</td>
<td>0.04558</td>
</tr>
<tr>
<td>2-5</td>
<td>2.76172</td>
<td>5902.2</td>
<td>6.54277</td>
<td>1.79077</td>
<td>0.03665</td>
</tr>
<tr>
<td>2-5</td>
<td>3.64959</td>
<td>9653.4</td>
<td>6.29993</td>
<td>1.90189</td>
<td>0.03980</td>
</tr>
</tbody>
</table>

**TOTAL LEG FIGHT** = 431.05107

**FINST** = 537.72108  **WFP** = 8.72683  **PHC(1)** = 3.41731  **HC(2)** = 11.10506

---

*Figure 3-17. Sample Output - Case B (Concluded)*
**Figure 3-18. Sample Output - Case C**

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>144.4926</td>
<td>0.3250</td>
<td>3.61232</td>
<td>+000.61</td>
</tr>
<tr>
<td>309.9524</td>
<td>0.5500</td>
<td>1.08347</td>
<td>+000.658</td>
</tr>
<tr>
<td>154.17332</td>
<td>0.6450</td>
<td>0.68280</td>
<td>+000.647</td>
</tr>
<tr>
<td>92.3507</td>
<td>0.3550</td>
<td>0.51474</td>
<td>+000.355</td>
</tr>
<tr>
<td>6.254071</td>
<td>0.6250</td>
<td>0.42455</td>
<td>+000.621</td>
</tr>
<tr>
<td>4.00999</td>
<td>0.0050</td>
<td>0.3403</td>
<td>+000.494</td>
</tr>
<tr>
<td>3.9970</td>
<td>0.0950</td>
<td>0.32277</td>
<td>+000.300</td>
</tr>
<tr>
<td>2.9831</td>
<td>0.1023</td>
<td>0.31346</td>
<td>+000.295</td>
</tr>
<tr>
<td>2.69242</td>
<td>0.1150</td>
<td>0.30963</td>
<td>+000.295</td>
</tr>
<tr>
<td>3.36264</td>
<td>0.0970</td>
<td>0.32036</td>
<td>+000.290</td>
</tr>
<tr>
<td>3.21456</td>
<td>0.0990</td>
<td>0.31824</td>
<td>+000.290</td>
</tr>
<tr>
<td>3.13264</td>
<td>0.1010</td>
<td>0.31639</td>
<td>+000.289</td>
</tr>
<tr>
<td>3.05463</td>
<td>0.1030</td>
<td>0.31433</td>
<td>+000.289</td>
</tr>
<tr>
<td>2.96236</td>
<td>0.1050</td>
<td>0.31236</td>
<td>+000.289</td>
</tr>
<tr>
<td>2.87446</td>
<td>0.1070</td>
<td>0.31041</td>
<td>+000.289</td>
</tr>
<tr>
<td>2.78886</td>
<td>0.1100</td>
<td>0.30855</td>
<td>+000.289</td>
</tr>
<tr>
<td>2.74855</td>
<td>0.1130</td>
<td>0.30665</td>
<td>+000.289</td>
</tr>
<tr>
<td>2.75539</td>
<td>0.1147</td>
<td>0.30475</td>
<td>+000.289</td>
</tr>
</tbody>
</table>

**LEG 1-2**

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.57376</td>
<td>0.0400</td>
<td>3.40926</td>
<td>+01561.4</td>
</tr>
<tr>
<td>5.35926</td>
<td>0.0500</td>
<td>2.39767</td>
<td>+067.26</td>
</tr>
<tr>
<td>3.87530</td>
<td>0.0600</td>
<td>2.51835</td>
<td>+06079.1</td>
</tr>
<tr>
<td>2.81770</td>
<td>0.0700</td>
<td>2.73812</td>
<td>+03261.3</td>
</tr>
<tr>
<td>1.84144</td>
<td>0.0800</td>
<td>2.51932</td>
<td>+05956.4</td>
</tr>
<tr>
<td>4.717127</td>
<td>0.0900</td>
<td>2.83433</td>
<td>+05126.3</td>
</tr>
<tr>
<td>4.53189</td>
<td>0.0950</td>
<td>2.40192</td>
<td>+06186.1</td>
</tr>
<tr>
<td>4.35920</td>
<td>0.0990</td>
<td>2.39767</td>
<td>+06702.6</td>
</tr>
<tr>
<td>4.22541</td>
<td>0.1000</td>
<td>2.40935</td>
<td>+07098.9</td>
</tr>
</tbody>
</table>

**LEG 1-4**

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.58127</td>
<td>0.0050</td>
<td>4.58127</td>
<td>+053500.0</td>
</tr>
</tbody>
</table>

**LEG 4-2**

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.2854</td>
<td>0.0000</td>
<td>7.2854</td>
<td>+00000.0</td>
</tr>
<tr>
<td>LEG</td>
<td>WEIGHT</td>
<td>FORCE</td>
<td>LENGTH</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>4-B</td>
<td>6.03781</td>
<td>11356.5</td>
<td>8.0514</td>
</tr>
<tr>
<td>2-B</td>
<td>7.84139</td>
<td>13967.3</td>
<td>8.67061</td>
</tr>
<tr>
<td>2-6</td>
<td>2.99767</td>
<td>6115.4</td>
<td>6.77579</td>
</tr>
<tr>
<td>2-5</td>
<td>4.53175</td>
<td>11899.2</td>
<td>6.65289</td>
</tr>
</tbody>
</table>

**TOTAL LEG WEIGHT = 5897.302**

FINWT = 725219.5
BF = 8.75262
WMC(1) = 4.93319
WMC(2) = 15.24503

Figure 3-18. Sample Output - Case C (Concluded)
Figure 3-19. Sample Output - Case D
LEG 1-6
WEIGHT = 7.21968  FORCE = 6404.9  LENGTH = 6.94073  R = 1.68847  THIN = .03194

LEG 2-5
WEIGHT = 4.09455  FORCE = 936.7  LENGTH = 6.69391  R = 2.00265  THIN = .03861

TOTAL LEG WEIGHT = 361.74097

FINWT = 451.34036  WFP = 8.70421  WKC(1) = 2.55739  WKC(2) = 8.56584

Figure 3-19. Sample Output - Case D (Concluded)
### SERIAL NO = TEST E

<table>
<thead>
<tr>
<th>R1</th>
<th>T</th>
<th>RT</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>56</td>
<td>T</td>
<td>255</td>
</tr>
<tr>
<td>35</td>
<td>65</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>47</td>
<td>75</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>60</td>
<td>75</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>80</td>
<td>90</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>90</td>
<td>90</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>T</td>
<td>275</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>T</td>
<td>275</td>
</tr>
</tbody>
</table>

**Figure 3-20. Sample Output - Case E**
<table>
<thead>
<tr>
<th>LEG</th>
<th>2-8</th>
<th>WEIGHT= 11.65751</th>
<th>FORCE= 19407.9</th>
<th>LENGTH= 9.64142</th>
<th>R= 2.67784</th>
<th>THMIN= .05558</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEG</td>
<td>4-7</td>
<td>WEIGHT= 7.42650</td>
<td>FORCE= 6197.4</td>
<td>LENGTH= 11.61572</td>
<td>R= 2.64705</td>
<td>THMIN= .03141</td>
</tr>
<tr>
<td>LEG</td>
<td>2-5</td>
<td>WEIGHT= 5.01562</td>
<td>FORCE= 16373.7</td>
<td>LENGTH= 6.22293</td>
<td>R= 2.05298</td>
<td>THMIN= .05105</td>
</tr>
<tr>
<td>LEG</td>
<td>6-5</td>
<td>WEIGHT= 3.24162</td>
<td>FORCE= 5826.8</td>
<td>LENGTH= 7.23907</td>
<td>R= 1.91168</td>
<td>THMIN= .03446</td>
</tr>
<tr>
<td>LEG</td>
<td>6-10</td>
<td>WEIGHT= 11.93942</td>
<td>FORCE= 23336.2</td>
<td>LENGTH= 9.24760</td>
<td>R= 2.61552</td>
<td>THMIN= .05963</td>
</tr>
<tr>
<td>LEG</td>
<td>6-14</td>
<td>WEIGHT= 19.59881</td>
<td>FORCE= 31827.1</td>
<td>LENGTH= 10.80631</td>
<td>R= 3.31346</td>
<td>THMIN= .07117</td>
</tr>
<tr>
<td>LEG</td>
<td>5-13</td>
<td>WEIGHT= 7.58966</td>
<td>FORCE= 8393.1</td>
<td>LENGTH= 10.42373</td>
<td>R= 2.59040</td>
<td>THMIN= .03655</td>
</tr>
<tr>
<td>LEG</td>
<td>5-10</td>
<td>WEIGHT= 6.43272</td>
<td>FORCE= 9695.2</td>
<td>LENGTH= 8.90989</td>
<td>R= 2.36986</td>
<td>THMIN= .03928</td>
</tr>
<tr>
<td>LEG</td>
<td>5-9</td>
<td>WEIGHT= 15.40726</td>
<td>FORCE= 29964.5</td>
<td>LENGTH= 9.58188</td>
<td>R= 3.02760</td>
<td>THMIN= .06906</td>
</tr>
</tbody>
</table>

**TOTAL LEG WEIGHT = 1114.89930**

FINWT=1250.56380 WFP = 8.85384 WMC(1)= 4.54164 WMC(2)= 15.97903

Figure 3-20. Sample Output - Case E (Concluded)
SECTION IV
DIGITAL COMPUTER PROGRAMS FOR THE CONDENSED ANALYSIS
OF REAR LEG TOUCHDOWN AND FREE FLIGHT

DESCRIPTION

This computer program solves for the kinematical conditions involving the rear leg
 touchdown period and free flight to front leg touchdown.

The two main parameters in this study are the "coefficients of restitution" EX and EY.
The values of EX and EY that lead to a reasonable solution depend upon the other input
 parameters in a very complex way. To avoid this complexity, ranges of EX and EY are
 set up with eleven increments of each one. Then a solution is attempted for all the 121
 possible combinations of EX and EY. This program can be used to investigate the effect
 of EX and EY on the landing.

A valid solution may not exist for two reasons. First, XN(30) and XN(44) may not converge
 in the valid range of PS2 (-20° to 0°), and second, the function FF may have no solution
 within this range. In either case, a comment is made accordingly, the present values of
 EX and EY are printed, and then the problem continues with the next combination until
 all 121 combinations are exhausted.

COMPUTER INPUT DATA

The input data requires two cards. For identification of the input data, see Figure 4-7.

<table>
<thead>
<tr>
<th>INPUT DATA (CARD 1)</th>
<th>DEFINITIONS</th>
<th>COLUMNS USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNNO</td>
<td>See Figure 4-7</td>
<td>1 through 5</td>
</tr>
<tr>
<td>EXX</td>
<td>Ref. 1, Section 6, Eq. 33 (EX)_0</td>
<td>6 through 10</td>
</tr>
<tr>
<td>EXDEL</td>
<td>See Figure 4-7</td>
<td>11 through 15</td>
</tr>
<tr>
<td>EXMAX</td>
<td>See Figure 4-7</td>
<td>16 through 20</td>
</tr>
<tr>
<td>EYY</td>
<td>Ref. 1, Section 6, Eq. 36 (EY)_0</td>
<td>21 through 25</td>
</tr>
<tr>
<td>EYDEL</td>
<td>See Figure 4-7</td>
<td>26 through 30</td>
</tr>
<tr>
<td>EYMAX</td>
<td>See Figure 4-7</td>
<td>31 through 35</td>
</tr>
</tbody>
</table>
### Input Data (Card 2)

<table>
<thead>
<tr>
<th>Column</th>
<th>Definition</th>
<th>Columns Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNNO</td>
<td>See Figure 4-7</td>
<td>1 through 5</td>
</tr>
<tr>
<td>XDOT1</td>
<td>Ref. 1, Section 6, Eq. 1 $\dot{X}_1$</td>
<td>6 through 10</td>
</tr>
<tr>
<td>YDOT1</td>
<td>Ref. 1, Section 6, Eq. 1 $\dot{Y}_1$</td>
<td>11 through 15</td>
</tr>
<tr>
<td>WONE</td>
<td>Ref. 1, Section 6, Eq. 1 $\omega_1$</td>
<td>16 through 20</td>
</tr>
<tr>
<td>XKX</td>
<td>Ref. 1, Section 6, Eq. 30 $K_X$</td>
<td>21 through 25</td>
</tr>
<tr>
<td>KY</td>
<td>Ref. 1, Section 6, Eq. 39 $K_Y$</td>
<td>26 through 30</td>
</tr>
<tr>
<td>XKKP</td>
<td>Ref. 1, Section 6, Eq. 31 $K'_X$</td>
<td>31 through 35</td>
</tr>
<tr>
<td>XKYP</td>
<td>Ref. 1, Section 6, Eq. 40 $K'_Y$</td>
<td>36 through 40</td>
</tr>
<tr>
<td>XXKDP</td>
<td>Ref. 1, Section 6, Eq. 32 $K''_X$</td>
<td>41 through 45</td>
</tr>
<tr>
<td>XKYDP</td>
<td>Ref. 1, Section 6, Eq. 41 $K''_Y$</td>
<td>46 through 50</td>
</tr>
</tbody>
</table>

### Definitions

For identification of symbols, see Section VI of Reference 1 and computer program comments, Figure 4-7, as indicated above.

### Computer Output Data

1. The input data is printed first as part of the output record.

2. For every valid solution, the following is printed for that combination of EX and EY:
   a. $FF, XN(30), XN(44), PS_2, XN(42), EX, EY$ where $XN(30)$ and $XN(44)$ have converged and FF is close to zero.
   b. The function $FFF = f(T)$ is now solved by iteration and the values of $FFF$ and $T$ are printed for every 10th iteration. The final line is the last value of $FFF$ and $T$ which meets the percent error test.
   c. Now solve and print:
      $XN(30), XN(31), XN(32), XN(33), XN(34)$
      $XN(35), XN(44), \Delta T, X_3, Y_3, (PSI)_3, X_3, Y_3, \beta_3$.

3. For every non-valid combination of EX and EY one of two error statements is printed and then EX is incremented and program continues with the next combination of EX and EY.

### Note

The sample computer solution that follows gives the solution for one set of input parameters. For this example, there are only 18 valid solutions out of the 121 possible combinations of EX and EY. Only the first 15 solutions are shown in Figure 4-12, although a complete output would include all 121 trials.
Since the EX, EY combination which corresponds to the detailed digital computer solution cannot be analytically determined, the solution which compares most favorably is selected by visual inspection.

The first four lines of printed output consist of input data. The next 7 lines are invalid cases and then three valid cases occur. The last four lines are also invalid cases. This output is typical of the printout for the remaining 106 cases.
This program solves for the kinematical conditions involving the rear leg touchdown period, and free flight to front leg touchdown, as described in Reference 1.

**Figure 4-2. Computer Program - Condensed Analysis for Downhill Landing Dynamics**
Set $T = 0$, $T_{DEL} = .05$
calculate $FFF = f(T)$ for $T = 0$

Call: FTHREE. This subroutine solves the function $FFF$
by iteration

Iteration succeeds; calculate
and print $XN(30), XN(31), XN(32), XN(33), XN(34), XN(35), XN(44), X3DOT,$
$Y3DOT, PS3, X3, Y3, BETA3$

iteration fails for this EX value
increment EX
reset PS2 = $-20^\circ$
This subroutine reads two data cards, prints this data, initializes and sets basic constants.

Figure 4-3. Subroutine INCONT
This subroutine solves the equation \( FF = f(PS2) \) by iteration. \( PS2 \) is set initially to -20 degrees before entry.

---

**Figure 4-4. Subroutine FFIT**

---

**BENDIX PRODUCTS AEROSPACE DIVISION**
Figure 4-4. Subroutine FFIT (Continued)
Figure 4-4. Subroutine FFIT (Concluded)
This subroutine solves the equation \( F = f(T) \) by iteration. \( F = f(T_0) \) is calculated before entry.

Figure 4-5. Subroutine FTHREE
This subroutine calculates the value of FF which equals \( XN(30) - XN(44) \) where FF is a function of PS2 and WTWO.

Figure 4-6. Subroutine N30N44
- RUN CARR ZZ61C 3.40
- FOR LEM

**MAIN PROGRAM TITLE**

CONDENSED ANALYSIS FOR DOWNHILL LANDING DYNAMICS

**AUTHOR**

H. O. CARR

**DATE**

09-1-64

**CALL**

LEM

**PURPOSE**

TO SOLVE FOR THE KINEMATICAL CONDITIONS INVOLVING REAR LEG TOUCHDOWN PERIOD, FREE FLIGHT TO FRONT LEG TOUCHDOWN.

SEE FIGURES 6-3, 6-4, 6-5, AND 6-6 IN MAIN REPORT FOR VEHICLE GEOMETRY.

**BENDIX REPORT MP- 8-64**

**INPUT DATA**

**COLUMNS USED**

CARD 1.

| RUNNO | 1  | THRU 5 |
| FX | 6  | THRU 10 |
| FXDEL | 11 | THRU 15 |
| EYMAX | 16 | THRU 20 |
| EY | 21 | THRU 25 |
| EYDEL | 26 | THRU 30 |
| EYMAX | 31 | THRU 35 |

CARD 2.

| RUNNO | 1  | THRU 5 |
| XDOT1 | 6  | THRU 10 |
| YDOT1 | 11 | THRU 15 |
| WONE | 16 | THRU 20 |
| KXX | 21 | THRU 25 |
| KXY | 26 | THRU 30 |
| KXXP | 31 | THRU 35 |
| KXYP | 36 | THRU 40 |
| KXXDP | 41 | THRU 45 |
| KYDP | 46 | THRU 50 |

**SYMBOL**

**DEFINITION**

| EX | COEFFICIENT OF RESTITUTION IN X DIRECTION |
| EY | SAME BUT IN Y DIRECTION |
| EXX | INITIAL VALUE OF EX (INPUT DATA) |
| EYY | INITIAL VALUE OF EY (INPUT DATA) |

Figure 4-7. Main Program

BENDIX PRODUCTS AEROSPACE DIVISION
C EXDEL EX RANGE (INPUT DATA)
C EYDEL EY RANGE (INPUT DATA)
C FM MAX FX MAXIMUM (INPUT DATA)
C EYMAX EY MAXIMUM (INPUT DATA)
C XDOT1 (INPUT) INITIAL VELOCITY COMPONENT OF REAR FOOTPAD
C YDOT1 (INPUT) INITIAL VELOCITY COMPONENT OF REAR FOOTPAD
C WONE (INPUT) INITIAL ANGULAR VELOCITY OF REAR FOOTPAD
C XX X THESE SIX VARIABLES ARE FUNCTIONS
C XXY OF MASS
C XXYP AND VELOCITIES
C XXXD P (SEE EGS. 30-31-32-39-40-41)
C PFI VEHICLE INITIAL PITCH ANGLE (FIGURE 6-6) IN RADIANS
C XI LEG GEOMETRY ANGLE (FIGURE 6-5) IN RADIANS
C ZETA SLOPE OF LUNAR SURFACE (FIG. 6-6) IN RADIANS
C XM MASS OF FOOTPADS
C XMM MASS OF VEHICLE EXCLUDING FOOTPADS
C XI1 MOMENT OF INERTIA
C XG GRAVITY FORCE
C XLI LENGTH OF LEG ONE (FIGURE 6-5)
C XL2
C EPTONE CONVERGENCE ERROR TEST FOR XN(30) AND XN(44)
C EPTWO CONVERGENCE ERROR TEST FOR T AND TPREV
C FF = XN(30)-XN(44)
C FFZERO FFZERO EQUALS INITIAL VALUE OF FF
C TPREV PREV VALUE OF TIME
C FFF FUNCTION OF T (SOLVED BY ITERATION)
C FZERO INITIAL VALUE OF FFF
C PS2 VEHICLE FINAL PITCH ANGLE (FIGURE 6-6) IN RADIANS
C PS2DFG SAME AS PS2 BUT IN DEGREES
C PS2DFL INCREMENT OF PS2
C FLAGD TO PRINT OR NOT PRINT CONVERGENCE STEPS OF XN(30)
C JJ CONTROLS LOGICAL EXIT FROM SUBROUTINE N30N44
C KK CONTROLS LOGICAL EXIT FROM SUBROUTINE FFI
C LL CONTROLS LOGICAL EXIT FROM SUBROUTINE FTHREE
C RUNNO IDENTIFICATION OF RUN NUMBER
C PS5 EQUALS PS2 VALUE WHEN FF OF PS2 HAS SAME SIGN AS FFZERO.
C PS6 EQUALS PS2 WHEN FF OF PS2 HAS OPPOSITE SIGN FFZERO
C AA TO PRINT EVERY FIFTH VALUE OF FF, XN(30), XN(44), PS1
C AA2 IF CONVERGENCE STEPS ARE DESIRED
C A FUNCTION OF PS1, PS2, XI, ZETA (SEE EQ.46)
C ALPHA1 SAME AS ABOVE (SEE EQ.26)
C ALPHA2 SAME AS ABOVE (SEE EQ.47)
C BETA1 SAME AS ABOVE (SEE EQ.88)
C GAMMA
C A FUNCTION OF INITIAL AND FINAL CONDITIONS.
C FOR A SEE EQ. 34, FOR B SEE EQ. 35, FOR C SEE EQ.
C 37, FOR D SEE EQ. 38.
C E A FUNCTION OF THE INITIAL CONDITIONS

Figure 4-7. Main Program (Continued)
XN(1) THRU XN(44) ARE FUNCTIONS OF VEHICLE MASS,
INITIAL AND FINAL VELOCITIES, ETC. (SEE Eqs. 51-55)
VELOCITY COMPONENT WHEN FRONT FOOTPAD TOUCHES
 VELOCITY COMPONENT WHEN FRONT FOOTPAD TOUCHES
ATTITUDE WHEN FRONT FOOTPAD TOUCHES -EQ. 110
ANGULAR VELOCITY WHEN FRONT FOOTPAD TOUCHES -EQ. 109
X DISTANCE FROM AXES AT C.G. AS FOOTPAD TOUCHES
Y DISTANCE FROM AXES AT C.G. AS FOOTPAD TOUCHES
FUNCTION OF PS3

PS1D
SAME AS PS1 BUT IN DEGREES

XI
SAME AS XI BUT IN DEGREES

ZETA
SAME AS ZETA BUT IN DEGREES

PS2D
SAME AS PS2 BUT IN DEGREES

PS2DEG
SAME AS PS2D

T
TIME INTERVAL FOR REAR LEG TOUCHDOWN (SEE EQ. 69)
TEDEL
INCREMENTS OF T. (T=DELTA T SUB 2)
T2
EQUALS T WHEN FFF OF T HAS SAME SIGN AS FZERO

INITIAL VALUE OF FFF

CALL SUBROUTINE INCONT WHICH READS TWO DATA
CARDS AND SETS UP INITIAL CONDITIONS AND BASIC
GEOMETRY.

CALL SUBROUTINE FFIT WHICH SOLVES THE FUNCTION
FF BY ITERATION. FF=XN(30)-XN(44).

A VALID SOLUTION IS REACHED FOR FF SO FINAL
VALUES OF FF, XN(30),XN(44),ETC. ARE PRINTED

Figure 4-7. Main Program (Continued)
THIS IS THE START OF THE SECOND PART OF THE PROGRAM THAT SOLVES A SECOND FUNCTION FFF BY ITERATION. AN INITIAL VALUE IS CALCULATED FOR AN INITIAL VALUE OF T.

320 T = 0.0
BE = 10.0
TDEL = 0.05
PRINT 913
PRINT 903
FFF = -XL2*SIN(PS2) + XN(31)*T - XG/2.0*T*T*COS(ZETA) - XL1*SIN(XI + PS2) + XN(30)*T + XL1*SIN(XI + PS2)
FZERO = FFF
PRINT 902, FFF, T

CALL SUBROUTINE FTHREE WHICH THEN SOLVES THE FUNCTION FFF BY ITERATION. THE ONLY VALID SOLUTIONS ARE FOR T BETWEEN ZERO AND 1.2

375 CALL FTHREE
GO TO (480, 455), LL
455 PRINT 902, FFF, T
PRINT 913

CALCULATE FINAL VELOCITIES AND LENGTHS AND PRINT THEM FOR THIS SET OF VALUES OF THE TWO PARAMETERS EX AND EY.

460 X3DOT = XN(31) - XG*T*COS(ZETA)
Y3DOT = XN(32) + XG*T*SIN(ZETA)
PS3 = PS2 + XN(30)*T
w3 = XN(30)
x3 = -XL1*SIN(XI + PS3)
y3 = XL1*COS(XI + PS3)
BETA3 = -3.14159265/2.0 + XI + PS3 + ZETA
PRINT 909, XN(30), XN(31), XN(32), XN(33), XN(34)
PRINT 910, XN(35), XN(44), T, X3DOT, Y3DOT
PRINT 911, PS3, X3, Y3, BETA3
470 PRINT 913
PRINT 913
GO TO 480

INCREMENT EX TO NEXT VALUE AND RESET PS2D TO -20 DEGREES AND LOOP BACK TO STATEMENT 135.

480 EX = EX + EXDEL
PS2D = -20.0
PS2 = PS2D*3.14159265/180.0
PS2DEG = PS2D
PRINT 913
GO TO 135
901 FORMAT(5H FF=, F8.4, 11H XN(30)=, F8.4, 11H XN(44)=, F7.4, 8H 1PS2=, F8.2, 11H XN(42)=, F16.2, 7H EX=, F6.3, 7H EY=, F6.3)
902 FORMAT(F15.4, F15.4)
903 FORMAT(29H FFF T //)

Figure 4-7. Main Program (Continued)
END

Figure 4-7. Main Program (Concluded)
FOR INCONT

<table>
<thead>
<tr>
<th>SUBROUTINE TITLE</th>
<th>INCONT</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHOR</td>
<td>H. O. CARR</td>
</tr>
<tr>
<td>DATE</td>
<td>09-1-64</td>
</tr>
<tr>
<td>CALL</td>
<td>INCONT</td>
</tr>
</tbody>
</table>

PURPOSE TO READ TWO DATA CARDS AND PRINT THIS DATA AND THEN TO INITIALIZE AND SET BASIC CONSTANTS

NOTE FLAG D IS SET TO MINUS ONE HERE. CHANGE THIS TO +1 IF INTERMEDIATE CONVERGENCE STEPS OF XN(30) AND XN(44) ARE DESIRED

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>EX</td>
<td>COEFFICIENT OF RESTITUTION IN X DIRECTION</td>
</tr>
<tr>
<td>EY</td>
<td>SAME BUT IN Y DIRECTION</td>
</tr>
<tr>
<td>EXX</td>
<td>INITIAL VALUE OF EX (INPUT DATA)</td>
</tr>
<tr>
<td>EYY</td>
<td>INITIAL VALUE OF EX (INPUT DATA)</td>
</tr>
<tr>
<td>EXDEL</td>
<td>EX RANGE (INPUT DATA)</td>
</tr>
<tr>
<td>EYDEL</td>
<td>EY RANGE (INPUT DATA)</td>
</tr>
<tr>
<td>EXMAX</td>
<td>EX MAXIMUM (INPUT DATA)</td>
</tr>
<tr>
<td>EYMAX</td>
<td>EY MAXIMUM (INPUT DATA)</td>
</tr>
<tr>
<td>XDOT1</td>
<td>INITIAL VELOCITY COMPONENT OF REAR FOOTPAD</td>
</tr>
<tr>
<td>YDOT1</td>
<td>INITIAL VELOCITY COMPONENT OF REAR FOOTPAD</td>
</tr>
<tr>
<td>WONE</td>
<td>INITIAL ANGULAR VELOCITY OF REAR FOOTPAD</td>
</tr>
<tr>
<td>KKX</td>
<td>THESE SIX VARIABLES</td>
</tr>
<tr>
<td>KKY</td>
<td>ARE FUNCTIONS</td>
</tr>
<tr>
<td>XKXP</td>
<td>OF MASS</td>
</tr>
<tr>
<td>XKYP</td>
<td>AND VELOCITIES</td>
</tr>
<tr>
<td>XKXDP</td>
<td>(SEE EQUATIONS 30-31-32-39-40-41)</td>
</tr>
<tr>
<td>XKYP</td>
<td></td>
</tr>
<tr>
<td>PSI</td>
<td>VEHICLE INITIAL PITCH ANGLE (FIGURE 6-6) IN RADIANS</td>
</tr>
<tr>
<td>XI</td>
<td>LEG GEOMETRY ANGLE (FIGURE 6-5) IN RADIANS</td>
</tr>
<tr>
<td>ZETA</td>
<td>SLOPE OF LUNAR SURFACE (FIG. 6-6) IN RADIANS</td>
</tr>
<tr>
<td>XMM</td>
<td>MASS OF FOOTPADS</td>
</tr>
<tr>
<td>XMM</td>
<td>MASS OF VEHICLE EXCLUDING FOOTPADS</td>
</tr>
<tr>
<td>XIU</td>
<td>MOMENT OF INERTIA</td>
</tr>
<tr>
<td>XC</td>
<td>GRAVITY FORCE</td>
</tr>
<tr>
<td>XL1</td>
<td>LENGTH OF LEG ONE (FIGURE 6-5)</td>
</tr>
<tr>
<td>XL2</td>
<td></td>
</tr>
<tr>
<td>EPONE</td>
<td>CONVERGENCE ERROR TEST FOR XN(30) AND XN(44)</td>
</tr>
<tr>
<td>EPTWO</td>
<td>CONVERGENCE ERROR TEST FOR T AND TPREV.</td>
</tr>
<tr>
<td>PS2DEG</td>
<td>SAME AS PS2 BUT IN DEGREES</td>
</tr>
<tr>
<td>PS2DEL</td>
<td>INCREMENT OF PS2</td>
</tr>
<tr>
<td>FLGID</td>
<td>TO PRINT OR NOT PRINT CONVERGENCE STEPS OF XN(30)</td>
</tr>
<tr>
<td>RUNNC</td>
<td>IDENTIFICATION OF RUN NUMBER</td>
</tr>
<tr>
<td>PS1D</td>
<td>SAME AS PS1 BUT IN DEGREES</td>
</tr>
<tr>
<td>XI0</td>
<td>SAME AS XI BUT IN DEGREES</td>
</tr>
<tr>
<td>PS2D</td>
<td>SAME AS PS2 BUT IN DEGREES</td>
</tr>
<tr>
<td>PS2DEG</td>
<td>SAME AS PS2D</td>
</tr>
</tbody>
</table>

Figure 4-8. Subroutine INCONT
SUBROUTINE INCONT
COMMON/C1/RUNNO,EXX,EXDEL,EXMAX,EYY,EYDEL,EMYMAX,EX,EY,XN
COMMON/C2/XX,XKX,XKY,XXP,XXY,XXDP,XXYDP
COMMON/C3/XX,XKX,XKY,XXP,XXY,XXDP,XXYDP
COMMON/C4/EY,EYDEL,EYMAX,EXX,EXDEL,EXMAX,EYY,EYDEL,EMYMAX,EX,EY,XN
COMMON/C5/FLAGD,JJ,LL,XX,YY
READ 920,RUNNO,EXX,EXDEL,EXMAX,EYY,EYDEL,EMYMAX,EX,EY,XN
PRINT 906,RUNNO,EXX,EXDEL,EXMAX,EYY,EYDEL,EMYMAX,EX,EY,XN
READ 921,RUNNO,XX,XKX,XKY,XXP,XXY,XXDP,XXYDP
PRINT 906,RUNNO,EXX,EXDEL,EXMAX,EYY,EYDEL,EMYMAX,EX,EY,XN
READ 921,RUNNO,XX,XKX,XKY,XXP,XXY,XXDP,XXYDP

100 PSID = 20.0
    PSI = PSID * 3.14159265 / 180.0
    XI = XI * 3.14159265 / 180.0
    ZETA = ZETA * 3.14159265 / 180.0
    PS2DEL = 1.0 * ZETA * 3.14159265 / 180.0

110 X = 8.0
    Y = 4.0
    XI = 90.0
    XI = XI * 3.14159265 / 180.0
    X = X * 3.14159265 / 180.0
    PS2DEG = 1.0 * ZETA * 3.14159265 / 180.0

120 PS2 = 0.0001
    EPONE = 0.0001
    EPTWO = 0.0001
    PS2 = PS2 * 3.14159265 / 180.0
    PS2 = PS2 * 3.14159265 / 180.0
    EY = EY
    FLAGD = -1.0

906 FORMAT (17HO RUN NUMBER = ,F5.0//)
920 FORMAT (F5.0,F5.3)
921 FORMAT (F5.0,F5.3)
924 FORMAT (7HO EX = ,F6.3,9H EXDEL = ,F6.2,9H EXMAX = ,F6.3//)
925 FORMAT (7HO EX = ,F6.3,9H EXDEL = ,F6.2,9H EXMAX = ,F6.3//)
926 FORMAT (19HO XDOT1 = ,F5.2,9H YDOT1 = ,F5.2,9H X = ,F4.2,7H XXY = ,F4.2,7H XXYDP = ,F4.2,7H X = ,F4.2,7H XXYDP = ,F4.2,7H)
    RETURN
END

Figure 4-8. Subroutine INCONT (Concluded)
SUBROUTINE TITLE: FFIT

AUTHOR: H. O. CARR

DATE: 05-1-64

CALL: FFIT

PURPOSE:
SOLVE EQUATION FF BY ITERATION IN TERMS OF THE VARIABLE PS2. FF IS SATISFIED WHEN XN(30) DIFFERS FROM XN(44) BY A SMALL PERCENT DIFFERENCE.

METHOD:
INCREMENT PS2 UNTIL VALUE OF FF CHANGES SIGN AND THEN HALVE INTERVAL UNTIL CONVERGENCE OCCURS.

NOTE 1:
IF THE CONVERGENCE TEST FAILS AFTER HALVING THE INTERVAL 20 TIMES, THEN INCREMENT EX AND EXIT FROM SUBROUTINE.

NOTE 2:
IF CONVERGENCE DOES NOT OCCUR FOR VALUES OF PS2 BETWEEN -20 DEGREES AND ZERO DEGREES, THEN INCREMENT EX AND EXIT FROM SUBROUTINE.

SYMBOL

EX
COEFFICIENT OF RESTITUTION IN X DIRECTION

EY
SAME BUT IN Y DIRECTION

EXDEL
EX RANGE (INPUT DATA)

FFCONV
CONVERGENCE ERROR TEST FOR XN(30) AND XN(44)

FF = XN(30) - XN(44)

FFZERO
FFZERO EQUALS INITIAL VALUE OF FF

PS2
VEHICLE FINAL PITCH ANGLE (FIGURE 6-6) IN RADIANS

PS2DEG
SAME AS PS2 BUT IN DEGREES

PS2DEL
INCREMENT OF PS2

FLAGD
TO PRINT OR NOT PRINT CONVERGENCE STEPS OF XN(30)

JJ
CONTROLS LOGICAL EXIT FROM SUBROUTINE XN30N44

KK
CONTROLS LOGICAL EXIT FROM SUBROUTINE FFIT

PS5
EQUALS PS2 VALUE WHEN FF OF PS2 HAS SAME SIGN AS FFZERO

PS6
EQUALS PS2 WHEN FF OF PS2 HAS OPPOSITE SIGN FFZERO

AA
TO PRINT EVERY FIFTH VALUE OF FF. XN(30), XN(44), PS1

AA2
IF CONVERGENCE STEPS ARE DESIRED

AA2
IF CONVERGENCE OF FF DOESN'T SUCCED WITHIN 20 ITERATIONS, THEN ON TO NEXT EX VALUE

XN(1) THRU XN(44)
XN(1) THRU XN(44) ARE FUNCTIONS OF VEHICLE MASS, INITIAL AND FINAL VELOCITIES, ETC. (SEE EQU. 51-49)

PS2D
SAME AS PS2 BUT IN DEGREES

PS2DEG
SAME AS PS2D

Figure 4-9. Subroutine FFIT
SUBROUTINE FFIT
COMMON/C1/RUNNG,EXX,EXDEL,EXYMAX,LY,LYDEL,LYMAX,LY,LY(150)
COMMON/C2/YOUT1,YOUT1,XXYY,XXYY,XXYY,XXYY,XXYY,XXYY,XXYY,XXYY
COMMON/C3/P5I,XXI,ETA,PS2DEL,XXI,XXI,XXI,XX1,XX1,XX2,XX2,XX2,XX2
COMMON/C4/FF,FFZERO,T,TDL,FFF,FZERO,TPREV,PS2,PS2DEG
COMMON/C5/FLAGD,JJ,LL,XX,

CALL SUBROUTINE N3ON44 AND CALCULATE XN(30) AND XN(44) AND FF
SUBROUTINE SETS JJ=2 FOR NORMAL EXIT
AND IT SETS JJ=1 FOR ABNORMAL EXIT WHEN
XN(42) IS NEGATIVE

150 CALL N3ON44
GO TO (150,170),JJ
155 PS2=PS2+PS2DEL
PS2DEG=PS2*180.0/3.14159265
160 IF (PS2) 150,150,155
155 PRINT 927,EX,EY
GO TO 167
167 EX = EX +EXDEL
PS2DEG=PS2*180.0/3.14159265
IF (PS2) 180,180,165
180 EX = EX +EXDEL
PS2DEG=PS2*180.0/3.14159265
IF (PS2) 180,180,165
165 PRINT 913

SUBROUTINE SETS KK=2 FOR NORMAL EXIT
AND SETS KK=1 WHEN ITERATION FAILS OR CONVERGENCE
DOESN'T OCCUR WITHIN PS2 RANGE

KK = 1
RETURN
170 FFZERO=FF
PS5 = PS2

AA IS A COUNTER THAT ALLOWS A TEMP PRINT EVERY
FIVE TIMES THRU THIS LOOP (NOTE: THIS PRINT ONLY
OCURRS WHEN FLAGD IS SET TO +1.0)

AA=0.0
IF (FLAGD) 173,173,173
173 PRINT 901,FF,XN(301),XN(441),PS2DEG,XN(421),EX,EY
175 PS2=PS2+PS2DEL
PS2DEG=PS2*180.0/3.14159265
176 IF (PS2) 180,180,165
165 CALL N3ON44
GO TO (185,195),JJ
185 PS2=PS2+PS2DEL
PS2DEG=PS2*180.0/3.14159265
190 IF (PS2) 180,180,165
195 AA=AA+1.0
196 IF (AA=4.0) 200,200,197
197 AA=0.0
IF (FLAGD) 197,198,198
196 PRINT 901,FF,XN(301),XN(441),PS2DEG,XN(421),EX,EY
GO TO 200

Figure 4-9. Subroutine FFIT (Continued)
IF FF HAS CONV. SIGN AS -FZERO THEN VALUE OF 
FUNCTION HAS NOT CROSSED OVER ZERO POINT. 
IF SIGNS ARE DIFFERENT, THEN HALF INTERVAL UNTIL 
CONVERGENCE OCCURS.

270 IF(FFZZERO)275,210,210
275 IF(FF)215,217,217
210 IF(FF)217,215,213
213 PS1=PS2
GO TO 175
217 N=0
215 GO TO 220
220 PS6=PS2
225 PS2=(PS5+PS6)/2.0
PS2DEG=PS2*180.0/3.14159265
230 CALL N3CN44
GO TO (235,240),JJ
235 GO TO 165
240 GO TO 245
245 AA2=AA2+.10
246 IF(AA2-20.0)250,255,247
247 PRINT 914
GO TO 167

PERCENT CONVERGENCE TEST

250 IF(AA31(XN(30)-XN(44))/XN(30))EPONE)255,255,280
280 IF(FFZZERO)290,300,300
290 IF(FF)310,220,220
310 PS5=PS2
GO TO 225
255 XX = 2
901 FORMAT(15H FF=F,16,4,11H XN(30)=,F8.4,11H XN(44)=,F7.4,8H 
1P=F2,F8.2,11H XN(42)=,F16,7,7H EX=F6.3,7H FY=F6.3)
912 FORMAT(17H )
914 FORMAT(51H FF ITERATION FAILS. INCREMENT EX AND CONTINUE.//)
927 FORMAT(99H XN(30) + XN(44) DO NOT CONV. IN VALID PS2 INTERVAL (-20 
1 TO 0 DEGREES)---THIS VOIDED CASE HAS EX=F6.3,7H FY=F6.3) 
RETURN
END

Figure 4-9. Subroutine FFIT (Concluded)
SUBROUTINE TITLE FTHREE

AUTHOR H. O. CARR

DATE 09-1-64

CALL FTHREE

PURPOSE

SOLVE EQUATION FFF BY ITERATION IN TERMS OF THE VARIABLE T = DELTA T SUB 2.

METHOD

INCREMENT T UNTIL VALUE OF FFF CHANGES SIGN AND THEN HALF INTERVAL UNTIL CONVERGENCE OCCURS.

NOTE 1.

MAXIMUM T VALUE IS 1.20.

IF CONVERGENCE DOES NOT OCCUR BEFORE THIS MAXIMUM T VALUE THEN EXIT FROM SUBROUTINE.

SYMBOL

DEFINITION

ZETA
SLOPE OF LUNAR SURFACE (FIG. 6-6) IN RADIANS

XL1
LENGTH OF LEG ONE (FIGURE 6-5)

XL2

FPTWO
CONVERGENCE ERROR TEST FOR T AND TPREV.

TPREV
PREV VALUE OF TIME

FZERO
INITIAL VALUE OF FFF

PS2
VEHICLE FINAL PITCH ANGLE (FIGURE 6-6) IN RADIANS

LL
CONTROLS LOGICAL EXIT FROM SUBROUTINE FTHREE

XN(1) THRU XN(44)
XN(1) THRU XN(44) ARE FUNCTIONS OF VEHICLE MASS, INITIAL AND FINAL VELOCITIES, ETC. (SEE EQUATIONS 51-99)

ZETAD
SAME AS ZETA BUT IN DEGREES

PS2D
SAME AS PS2 BUT IN DEGREES

PS2DEG
SAME AS PS2D

T
TIME INTERVAL FOR REAR LEG TOUCHDOWN (SEE EQUATION 69)

TDEL
INCREMENT OF T. (T = DELTA T SUB 2)

T1
EQUALS T WHEN FFF OF T HAS SAME SIGN AS FZERO

FZFFRO
INITIAL VALUE OF FFF

COMMON/C1/RUNNO,EXX,EXDEL,EXMAX,EYY,EYDEL,EYMAX,EX,EY,XN(50)
COMMON/C2/XDOT1,YDOT1,WUNE,XXK,XXXP,XXYP,XXKDP,XXYDP
COMMON/C3/PS1,XI,ZETA,PS2DEL,XM,XMM,XII,XG,XL1,XL2,EPTWO,FPTWO
COMMON/C4/FF,FFZERO,T,TDEL,FFF,FZERO,TPREV,PS2,PS2DEG
COMMON/C5/FLAGD,JJ,LL,KK,BB

JUST BEFORE ENTERING THIS SUBROUTINE, THE MAIN PROGRAM SET T = 0 AND CALCULATED FFF FOR T = 0.

NOW T IS INCREMENTED AND THE FUNCTION FFF IS SOLVED BY ITERATION.

Figure 4-10. Subroutine FTHREE
Figure 4-10. Subroutine FTHREE (Concluded)
Subroutine Title: N30N44
Author: H. O. Cark
Date: 09-1-64
Call: N30N44

Purpose: Calculate the value of FF which equals XN(30) - XN(44). FF is a function of PS2 and Wtwo.

Method: PS2 is the variable in this set of equations. The only valid values of PS2 are from -20 degrees to zero degrees.

For some values of PS2 the value of XN(42) is negative, which occurs exit from the subroutine, get a new value of PS2 and re-enter subroutine.

Symbol | Definition
--- | ---
EX | COEFFICIENT OF RESTITUTION IN X DIRECTION
EY | COEFFICIENT OF RESTITUTION IN Y DIRECTION
 XDCT1 | INITIAL VELOCITY COMPONENT OF REAR FOOTPAD
 YDCT1 | INITIAL VELOCITY COMPONENT OF REAR FOOTPAD
 WONE | INITIAL ANGULAR VELOCITY OF REAR FOOTPAD
 XXX | THESE SIX VARIABLES ARE FUNCTIONS OF MASS AND VELOCITIES
 XXXP | (SEE Eqs. 30-31-32-33-40-41)
 XXXDP | PSI | VEHICLE INITIAL PITCH ANGLE (FIGURE 6-6) IN RADIANS
 X | LEG GEOMETRY ANGLE (FIGURE 6-5) IN RADIANS
 Y | MAFF OF FOOTPADS
 XMM | MASS OF VEHICLE EXCLUDING FOOTPADS
 XII | MOMENT OF INERTIA
 X9 | GRAVITY FORCE
 XL1 | LENGTH OF LEG ONE (FIGURE 6-5)
 XL2 | FF | FFZERO | INITIAL VALUE OF FF FFZERO EQUALS INITIAL VALUE OF FF
 PS2 | VEHICLE FINAL PITCH ANGLE (FIGURE 6-6) IN RADIANS
 JJ | CONTROLS LOGICAL EXIT FROM SUBROUTINE N30N44
 ALPHA1 | A FUNCTION OF PS1,PS2,XI,ZETA (SEE EQ.46)
 ALPHA2 | SAME AS ABOVE (SEE EQ.47)
 PETA1 | SAME AS ABOVE (SEE EQ.88)
 GAMMA | A FUNCTION OF INITIAL AND FINAL CONDITIONS
 A | FOR A SEE EQ. 34, FOR B SEE EQ. 35, FOR C SEE EQ. 37, FOR D SEE EQ. 38

Figure 4-11. Subroutine N30N44
A FUNCTION OF THE INITIAL CONDITIONS

FOR E SEE EQ. 12-FOR F SEE EQ. 13.

AND FOR G SEE EQ. 14.

XN(11) THRU XN(44) ARE FUNCTIONS OF VEHICLE MASS, INITIAL AND FINAL VELOCITIES, ETC. (SEE Eqs. 51-99).

SUBROUTINE N3044
COMMON/C1/RUNIN,EXX,EXDEL,EXMAX,EYY,EXDEL,EXMAX,LY,EN,ENX(50)
COMMON/C2/XOUT1,YOUT1,UNX,UNY,XXXP,XXYP,XXDP,XXXD
COMMON/C3/RI,IX,EYDEL,YXDEL,XY,IXX,IXL,IXLZ,EPUNX,EPXU
COMMON/C4/FF,F2,ER,TT,F2ER,T,PREV,PSZ,PSZD
COMMON/C5/FLAG0,JU,FLX,FL
1000 DO1=1,3,14-PS1-P52)/2.0
1050 X=0L=1,2,3,4,5,6,7
SUBROUTINE N3044
COMMON/C1/RUNIN,EXX,EXDEL,EXMAX,EYY,EXDEL,EXMAX,LY,EN,ENX(50)
COMMON/C2/XOUT1,YOUT1,UNX,UNY,XXXP,XXYP,XXDP,XXXD
COMMON/C3/RI,IX,EYDEL,YXDEL,XY,IXX,IXL,IXLZ,EPUNX,EPXU
COMMON/C4/FF,F2,ER,TT,F2ER,T,PREV,PSZ,PSZD
COMMON/C5/FLAG0,JU,FLX,FL
1000 DO1=1,3,14-PS1-P52)/2.0
1050 X=0L=1,2,3,4,5,6,7

SUBROUTINE N3044
COMMON/C1/RUNIN,EXX,EXDEL,EXMAX,EYY,EXDEL,EXMAX,LY,EN,ENX(50)
COMMON/C2/XOUT1,YOUT1,UNX,UNY,XXXP,XXYP,XXDP,XXXD
COMMON/C3/RI,IX,EYDEL,YXDEL,XY,IXX,IXL,IXLZ,EPUNX,EPXU
COMMON/C4/FF,F2,ER,TT,F2ER,T,PREV,PSZ,PSZD
COMMON/C5/FLAG0,JU,FLX,FL
1000 DO1=1,3,14-PS1-P52)/2.0
1050 X=0L=1,2,3,4,5,6,7

Figure 4-11. Subroutine N3044 (Continued)

4-26

BENDIX PRODUCTS AEROSPACE DIVISION
Figure 4-11. Subroutine N30N44 (Concluded)
**Figure 4-12. Sample Output**
Figure 4-12. Sample Output (Concluded)
SECTION V

PROGRAM FOR COMPUTER SOLUTION OF EQUATIONS FROM CONDENSED ANALYSIS OF FRONT LEG IMPACT

A set of non-linear, algebraic equations has been derived in the Final Report. These equations, based upon rational analysis together with certain reasonable, simplifying assumptions, describe the behavior of the vehicle during stroking of the landing gear. This condensed analysis technique has been applied to examine vehicle motion during the first forward leg impact for a downhill landing, 2-2 footpad orientation, with the rear legs sustaining the initial ground contact. The applicability of the technique is not limited to this specific situation, but the 2-2, back-pitched, downhill landing is likely to be the critical landing configuration.

Due to the complexity of the set of non-linear equations describing the front leg impact, a numerical method has been utilized for simultaneous solution of the equations. The set of equations comprises three primary equations which must be solved simultaneously, and a number of auxiliary equations which can be substituted directly in the primary equations. The primary equations, derived in the previous section, are:

\[
\begin{align*}
\dot{\beta}_f &= \frac{1}{I+\epsilon_1^2} \left\{ \gamma \epsilon_0 \psi_0 t + I \psi_0 \left( \epsilon_0 \psi_1 \sin \alpha + \frac{(7 \epsilon \omega + 3 \psi \alpha) t^2}{2 \alpha} \right) + \frac{(3 \epsilon \omega - 2 \epsilon \alpha) t \epsilon \alpha}{60} \right\} \\
\dot{\theta}_f &= \epsilon_0 + \frac{\epsilon_0 + \epsilon_1 \psi_1 t^2}{2} + \frac{\epsilon_0 - \epsilon_1 \psi_1 t^2}{12} \\
\frac{1}{2}(I \psi_0^2 + \psi_0^2) - \frac{1}{2}(I + \epsilon_1^2 \psi_1^2) \epsilon_1^2 + (\epsilon_0 \omega \epsilon \psi_0 - \epsilon_0 \omega \cos \alpha) &= \sum \epsilon_1 \delta_1 \end{align*}
\]

The unknowns in the primary equations, describing vehicle orientation and motion at the completion of front leg stroking, are the stability angle (\( \beta_f \)), time rate of change of stability angle (\( \dot{\beta}_f \)) and the distance in the plane of motion from the vehicle c.g. to the forward footpads (\( I \psi_1 \)). Parameters used in these equations are defined in Table 5-1.

Once the set of equations is solved, the ultimate stability of the vehicle is predicted based upon considerations of potential and kinetic energies.

The numerical procedure used is an adaptation of Newton's method for solving a set of non-linear algebraic equations. The procedure is an automatized trial and error method in which values of the unknowns are sought which simultaneously satisfy the three primary equations. To facilitate the search for a solution, Equations (1), (2), and (3) are each rewritten in terms of a difference function:
Table 5-1. Summary of Important Parameters Used In Condensed Analysis of Front Leg Impact

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Mass (Vehicle and Pads)</td>
</tr>
<tr>
<td>I</td>
<td>Mass Moment of Inertia (Vehicle and Pads)</td>
</tr>
<tr>
<td>g</td>
<td>Gravitational Acceleration</td>
</tr>
<tr>
<td>F&lt;sub&gt;i&lt;/sub&gt; (i=1, 2, 3)</td>
<td>Crushing Force in i&lt;sup&gt;th&lt;/sup&gt; Strut</td>
</tr>
<tr>
<td>δ&lt;sub&gt;i&lt;/sub&gt; (i=1, 2, 3)</td>
<td>Total Stroking of i&lt;sup&gt;th&lt;/sup&gt; Strut</td>
</tr>
<tr>
<td>A, B, C, D, E</td>
<td>Hardpoint Coordinates Relative to C.G.</td>
</tr>
<tr>
<td>α</td>
<td>Ground Slope</td>
</tr>
<tr>
<td>x&lt;sub&gt;i&lt;/sub&gt;, y&lt;sub&gt;i&lt;/sub&gt;, z&lt;sub&gt;i&lt;/sub&gt;</td>
<td>Initial Vehicle C.G. Coordinates (Ground Coordinate Axes)</td>
</tr>
<tr>
<td>X&lt;sub&gt;0&lt;/sub&gt;, Y&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Vehicle C.G. Velocities (Ground Coordinate Axes)</td>
</tr>
<tr>
<td>X&lt;sub&gt;p&lt;/sub&gt;, Y&lt;sub&gt;p&lt;/sub&gt;, Z&lt;sub&gt;p&lt;/sub&gt;</td>
<td>Initial Pad Coordinates (Vehicle Coordinate Axes)</td>
</tr>
<tr>
<td>ψ&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Pitch Attitude</td>
</tr>
<tr>
<td>ψ&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Pitch Attitude</td>
</tr>
<tr>
<td>ψ&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Pitch Rate</td>
</tr>
<tr>
<td>ψ&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Initial Pitch Acceleration</td>
</tr>
<tr>
<td>β&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Stability Angle</td>
</tr>
<tr>
<td>β&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Stability Angle</td>
</tr>
<tr>
<td>β&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Stability Angular Rate</td>
</tr>
<tr>
<td>β&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Stability Angular Rate</td>
</tr>
<tr>
<td>g&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Stability Angular Acceleration</td>
</tr>
<tr>
<td>g&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Stability Angular Acceleration</td>
</tr>
<tr>
<td>L&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Effective Leg Length</td>
</tr>
<tr>
<td>L&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Effective Leg Length</td>
</tr>
<tr>
<td>t&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Duration of Front Leg Stroking</td>
</tr>
<tr>
<td>V&lt;sub&gt;oh&lt;/sub&gt;</td>
<td>Initial Horizontal Velocity of C.G.</td>
</tr>
<tr>
<td>V&lt;sub&gt;hf&lt;/sub&gt;</td>
<td>Final Horizontal Velocity of C.G.</td>
</tr>
<tr>
<td>η&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Horizontal Acceleration of C.G.</td>
</tr>
<tr>
<td>η&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Horizontal Acceleration of C.G.</td>
</tr>
<tr>
<td>V&lt;sub&gt;ot&lt;/sub&gt;</td>
<td>Initial Tangential Velocity of C.G.</td>
</tr>
<tr>
<td>a&lt;sub&gt;ot&lt;/sub&gt;</td>
<td>Initial Tangential Acceleration of C.G.</td>
</tr>
<tr>
<td>F&lt;sub&gt;ot&lt;/sub&gt;</td>
<td>Initial Tangential Force on C.G.</td>
</tr>
<tr>
<td>m&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Torque About C.G.</td>
</tr>
<tr>
<td>F&lt;sub&gt;oh&lt;/sub&gt;</td>
<td>Initial Horizontal Force on C.G.</td>
</tr>
<tr>
<td>γ&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Initial Angle (in plane of motion) Included Between Footpad, C.G., and Vehicle Longitudinal Axis</td>
</tr>
<tr>
<td>γ&lt;sub&gt;f&lt;/sub&gt;</td>
<td>Final Angle (in plane of motion) Included Between Footpad, C.G., and Vehicle Longitudinal Axis</td>
</tr>
</tbody>
</table>

"Initial" and "Final" Respectively Refer to Conditions at Front Leg Impact and at the Completion of Front Leg Stroking.
The work done to crush the honeycomb is written
\[ \sum_{i=1}^{2} F_{ii} \delta_i \] to take advantage of the
symmetry of a 2-2 landing.

\( \beta_{ft}, \dot{\beta}_{ft}, \) and \( \ell_{ft} \) are trial values of the unknowns. Successive sets of trial values are
substituted until \( F, G, \) and \( H \) are sufficiently small.

Since \( F, G, \) and \( H \) are complicated functions, it is preferred to initiate the search for a
solution by choosing a set of trial values close to the trial solution. An acceptable initial
set of trial values is sought by choosing the "best" set of trial values for a large number
of combinations of arbitrary values of \( \beta_{ft}, \dot{\beta}_{ft}, \) and \( \ell_{ft} \). These arbitrary values are:

\[ (\beta_{ft})_j = a_j \beta_0 \quad (a_j = .6, .75, .8, .85, .9, .95) \]
\[ (\dot{\beta}_{ft})_k = b_k \dot{\beta}_0 \quad (b_k = .6, .7, .8, .9, .95, 1.0, 1.05, 1.1, 1.15, 1.20, 1.30, 1.40) \]
\[ (\ell_{ft})_n = c_n \ell_0 \quad (c_n = .75, .775, .8, .825, .875, .95) \]

where \( \beta_0, \dot{\beta}_0, \) and \( \ell_0 \) are the known values of \( \beta, \dot{\beta}, \) and \( \ell \) at the time of front leg
impact. A total of 432 combinations are thus defined. The "best" combination is chosen
by defining:

\[ \text{Index} = F^o + G^o + H^o \]  \hspace{1cm} (7)

Equation (7) is evaluated for each of the 432 combinations, and the set of trial values
which results in the minimum value of Index is used as the starting point in the search for
a solution.

To obtain subsequent sets of trial solutions, an extrapolation technique is used. Ignoring
terms of second order and above, \( F, G, \) and \( H \) can each be expanded in a truncated Taylor's
Series:

\begin{align*}
v(\rho_c, \dot{\rho}_c, \ddot{\rho}_c) &= c + \frac{\dot{\rho}_c}{2} c + \frac{\ddot{\rho}_c}{12} c + c - \beta c_t \\
v(\rho_c, \dot{\rho}_c, \ddot{\rho}_c) &= \frac{1}{2} \left( \dot{\rho}_c \dot{\rho}_c + \dot{\rho}_c \dot{\rho}_c \right) + \frac{1}{2} \left( \ddot{\rho}_c \ddot{\rho}_c + \ddot{\rho}_c \ddot{\rho}_c \right) + \frac{1}{2} \left( \dddot{\rho}_c \dddot{\rho}_c + \dddot{\rho}_c \dddot{\rho}_c \right) + \frac{1}{2} \left( \ddddot{\rho}_c \ddddot{\rho}_c + \ddddot{\rho}_c \ddddot{\rho}_c \right)
\end{align*}
where the subscript "t" denotes that functions and partial derivatives are evaluated using the trial solution. It should be noted that the partial derivatives as well as the functions are available in analytical form.

Since the objective of the extrapolation is to determine values of \( \beta_f, \delta \beta_f, \) and \( \ell_f \) which drive \( F, G, \) and \( H \) simultaneously toward zero, Equations (8), (9), and (10) are solved for \( \delta \beta_f, \delta \beta_f, \) and \( \ell_f \) by setting \( F, G, \) and \( H \) all equal to zero. This result is

\[
\Delta = \left[ \frac{\partial F}{\partial \beta_f} \left( \frac{\partial \beta}{\partial \beta_f} \frac{\partial F}{\partial \beta_f} - \frac{\partial G}{\partial \beta_f} \frac{\partial H}{\partial \beta_f} - \frac{\partial H}{\partial \beta_f} \frac{\partial F}{\partial \beta_f} \right) + \frac{\partial F}{\partial \beta_f} \left( \frac{\partial \beta}{\partial \beta_f} \frac{\partial G}{\partial \beta_f} - \frac{\partial G}{\partial \beta_f} \frac{\partial F}{\partial \beta_f} \right) + \frac{\partial F}{\partial \beta_f} \left( \frac{\partial \beta}{\partial \beta_f} \frac{\partial H}{\partial \beta_f} - \frac{\partial H}{\partial \beta_f} \frac{\partial F}{\partial \beta_f} \right) \right]
\]
The next set of trial solutions is obtained from:

\[ \begin{align*}
\delta f_{t+1} &= \delta f_{t} + 5^\circ f \\
\delta \alpha_{t+1} &= \delta \alpha_{t} + 5^\circ \alpha \\
\delta \beta_{t+1} &= \delta \beta_{t} + 5^\circ \beta
\end{align*} \tag{14} \tag{15} \tag{16} \]

The process is repeated until \( \delta f, \delta \beta, \) and \( \delta \alpha \) simultaneously satisfy some predetermined error criteria.

Details of the computer program are presented in Figures 5-1 through 5-14. Figure 5-1 is a sample of the input data used in this program. The FORTRAN nomenclature is defined in the program and subroutine listings, Figure 5-8. Flow diagrams for the program are presented in the following order:

- Figure 5-2: Flow Diagram of Main Program
- Figure 5-3: Flow Diagram of Subroutine INCOND
- Figure 5-4: Flow Diagram of Subroutine GUESS
- Figure 5-5: Flow Diagram of Subroutine FEVAL
- Figure 5-6: Flow Diagram of Subroutine SOLN
- Figure 5-7: Flow Diagram of Subroutine DEVAL

INCOND calculate various initial conditions of geometry and motion from the input data. GUESS computes arbitrary trial solutions, from which will be determined a "best" set from which to initiate the extrapolation procedure. FEVAL evaluates the difference functions for a particular set of trial solutions. SOLN conducts the extrapolation procedure which should result in the final solution. DEVAL computes derivatives of the difference functions with respect to the independent variables.

The program listings are presented in the following order:

- Figure 5-8: Listing of Main Program
- Figure 5-9: Listing of Subroutine INCOND
- Figure 5-10: Listing of Subroutine GUESS
- Figure 5-11: Listing of Subroutine FEVAL
- Figure 5-12: Listing of Subroutine SOLN
- Figure 5-13: Listing of Subroutine DEVAL

A sample of the output data is presented in Figure 5-14.
<table>
<thead>
<tr>
<th>IDENTIFICATION</th>
<th>1</th>
<th>I</th>
<th>M</th>
<th>GRAV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
<td>F11</td>
<td>F22</td>
<td>F33</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>XPAD</td>
<td>YPAD</td>
<td>ZPAD</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>PSO</td>
<td>PSOVEL</td>
<td>BETO</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>ERRBF</td>
<td>ERRBFV</td>
<td>ERRLF</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>ALPHA</td>
<td>R</td>
<td>RUN</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>D</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>XVEL</td>
<td>YVEL</td>
<td></td>
</tr>
</tbody>
</table>

Figure 5-1. Sample Input Data
MAIN

LUNAR LANDING DYNAMICS STUDY
CONDENSED ANALYSIS OF FRONT LEG IMPACT

Read Input Data From Cards

Print Input Data

Call Subroutine INCOND

Compute Initial Conditions (Vehicle Motions and Configuration at Front Leg Impact)

Call Subroutine GUESS

Determine Arbitrary Trial Solutions:
BBF(J), 1 ≤ J ≤ 6 - Final Stability Angle (At Completion of Front Leg Stroking)
BBFV(K), 1 ≤ K ≤ 12 - Final Stability Angular Rate
LLF(N), 1 ≤ N ≤ 6 - Final Value of Distance From C.G. to Vehicle Footpad, Projected In Plane of Motion

DO 1 J = 1, 6
DO 1 K = 1, 12
DO 1 N = 1, 6

Call Subroutine FEVAL

Evaluate Difference Functions F, G, H For All Combinations of Trial Solutions

Figure 5-2. Flow Diagram of Main Program
Index is a Measure of the Degree with which a set of Trial Solutions Satisfies the Nonlinear Equations. The smallest Value of Index Available is the Most Desirable.

Index < MINIDX

Test Index Against Minimum Previous Value

Define Best Set of Trial Solutions Found so far:
BETF = BBF(J)
BETFVT = BBFV(K)
LFT = LLF(N)

CONTINUE

Call Subroutine SOLN

Extrapolate Best Set of Trial Solutions in Search of True Solution to Nonlinear Equations

Yes

Test if Extrapolation Converged within 25 Iterations

No

Figure 5-2. Flow Diagram of Main Program (Continued)
Figure 5-2. Flow Diagram of Main Program (Continued)
Figure 5-3. Flow Diagram of Subroutine INCOND

Figure 5-4. Flow Diagram of Subroutine GUESS
Calculate CORK based on set of trial solutions to nonlinear equations. CORK represents the quantity under a radical which should be tested to prevent SQRT subroutine error.

CORK < 0

Define difference functions F, G, H = 0

CORK ≥ 0

Calculate final values (after stroking) of vehicle motion and geometry, and calculate difference functions F, G, H.

RETURN

Figure 5-5. Flow Diagram of Subroutine FEVAL
Call Subroutine FEVAL
Evaluate Difference Functions F, G, H Using Latest Set of Trial Solutions

\[ F + G + H = 0 \]

Arbitrarily Define \( G = \frac{1}{F} \)

Call Subroutine DEVAL
Determine Partial Derivatives of Difference Functions F, G, H with Respect to Independent Variables

Calculate Extrapolation Increments Based On Truncated Taylor Series Expansion of Difference Functions

Print Latest Set of Trial Solutions and Extrapolation Increments

Update Trial Solutions

Count CC (The Number of Extrapolations Performed So Far)

\[ CC < 25 \]

Test Extrapolation Increments

\[ \text{Increments Do Not All Satisfy Error Criteria} \]

\[ \text{Increments All Satisfy Error Criteria} \]

RETURN

\[ \text{CC} \geq 25 \]

Figure 5-6. Flow Diagram of Subroutine SOLN
Figure 5-7. Flow Diagram of Subroutine DEVAL
LUNAR LANDING DYNAMICS STUDY
CONDENSED ANALYSIS OF FRONT LEG IMPACT

R.G. ALDERSON AND L.L. ANDERSON
BENDIX MISHAWAKA DIVISION

10-31-64

TO PREDICT THE STABILITY AND MOTION OF THE LUNAR VEHICLE
AS A FUNCTION OF CONDITIONS AT FRONT LEG IMPACT

FROM PUNCHED CARDS:

FOR DEFINITION OF THE GEOMETRY USED IN THIS PROGRAM,
REFER TO SECTION VI-P OF THE XALRU FINAL REPORT

SYMBOL DEFINITION
I MASS MOMENT OF INERTIA (FT*Lb*SEC*SEC)
M MASS (Lb*SEC*SEC/FT)
GRAV GRAVITATIONAL ACCELERATION (FT/SEC/SEC)
F1 STROKE FORCE IN MAIN STRUT (Lb)
F22,F33 STROKE FORCE IN LOWER STRUTS (Lb)
A THESE QUANTITIES LOCATE THE UPPER AND LOWER
B HARD POINTS RELATIVE TO THE VEHICLE C.G.
C A IS POSITIVE IF UPPER HARD POINT IS ABOVE THE
D C.G. - B IS POSITIVE IF LOWER HARD POINTS ARE
E BELOW THE C.G. - C,D,E ARE POSITIVE UNITS (FT)
ALPHA GROUND SLOPE (RAD)
X X-COMP. OF C.G. INITIALLY (GROUND AXES) (FT)
Y Y-COMP. OF C.G. INITIALLY (GROUND AXES) (FT)
Z Z-COMP. OF C.G. INITIALLY (GROUND AXES) (FT)
XVEL X-COMP. OF C.G. INITIAL VELOCITY (GROUND)
YVEL Y-COMP. OF C.G. INITIAL VELOCITY (GROUND)
XVEL AXES) (FT/SEC)
YPAD X-COMP. OF FOOTPAD (GROUND AXES) (FT)
ZPAD Y-COMP. OF FOOTPAD (GROUND AXES) (FT)
ZPAD Z-COMP. OF FOOTPAD (GROUND AXES) (FT)
PSO INITIAL PITCH ATTITUDE (RAD)
PSOVEL INITIAL PITCH RATE (RAD/SEC)
PSOVV INITIAL PITCH ACCELERATION (RAD/SEC/SEC)
PSF FINAL PITCH ATTITUDE (RAD)
BETO INITIAL STABILITY ANGLE (RAD)
BETOV INITIAL STABILITY ANGULAR RATE (RAD/SEC)
BETOVV INITIAL STABILITY ANGULAR ACC. (RAD/SEC/SEC)
BETA F FINAL VALUE OF STABILITY ANGLE (RAD)
BETAFV FINAL VALUE OF STABILITY ANGULAR RATE (RAD/SEC)

INITIAL AND FINAL REFER TO STATES BEFORE AND AFTER
STROKING OF THE FRONT LEGS

GAMMO INITIAL ANGLE INCLUDED BETWEEN FOOTPAD*C.G.
AND VEHICLE LONGITUDINAL AXIS (RAD)
GAMMF FINAL VALUE OF GAMMO
VOH HORIZONTAL COMP. OF C.G. INITIAL VEL. (FT/SEC)
VOT TANGENTIAL COMP. OF C.G. INITIAL VEL. (FT/SEC)
VFH HORIZONTAL COMP. OF C.G. FINAL VEL. (FT/SEC)
AOT TANGENTIAL COMP. OF C.G. INIT. ACC. (FT/SEC/SEC)
VO ABSOLUTE VELOCITY OF C.G. INITIALLY (FT/SEC)
XDOP X-COMP. OF FOOTPAD RELATIVE TO C.G.
(GROUND AXES) (FT)

Figure 5-8. Main Program Listing
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>YDOP</td>
<td>Y-COMP. OF FOOTPAD RELATIVE TO C.G. (GROUND AXES) (FT)</td>
</tr>
<tr>
<td>XDO</td>
<td>X-COMP. OF FOOTPAD RELATIVE TO C.G. (VEHICLE AXES) (FT)</td>
</tr>
<tr>
<td>YDO</td>
<td>Y-COMP. OF FOOTPAD RELATIVE TO C.G. (VEHICLE AXES) (FT)</td>
</tr>
<tr>
<td>ZDO</td>
<td>Z-COMP. OF FOOTPAD RELATIVE TO C.G. (EITHER VEHICLE OR GROUND AXES) (FT)</td>
</tr>
<tr>
<td>L1</td>
<td>LENGTH OF NO.1 STRUT (FT)</td>
</tr>
<tr>
<td>L2</td>
<td>LENGTH OF NO.2 STRUT (FT)</td>
</tr>
<tr>
<td>L3</td>
<td>LENGTH OF NO.3 STRUT (FT)</td>
</tr>
<tr>
<td>L0</td>
<td>INITIAL DISTANCE FROM FOOTPAD TO C.G. (MEASURED IN PLANE OF MOTION) (FT)</td>
</tr>
<tr>
<td>LF</td>
<td>FINAL DISTANCE FROM FOOTPAD TO C.G. (MEASURED IN PLANE OF MOTION) (FT)</td>
</tr>
<tr>
<td>LOVEL</td>
<td>TIME RATE OF CHANGE OF LG (FT/SEC)</td>
</tr>
<tr>
<td>FX</td>
<td>X-COMP. OF INITIAL STROKING FORCES (VEHICLE AXES) (LB)</td>
</tr>
<tr>
<td>FY</td>
<td>Y-COMP. OF INITIAL STROKING FORCES (VEHICLE AXES) (LB)</td>
</tr>
<tr>
<td>TG</td>
<td>TORQUE INDUCED BY INITIAL STROKE FORCES (FT*LB)</td>
</tr>
<tr>
<td>FOH</td>
<td>HORIZONTAL COMP. OF INITIAL STROKE FORCES (LB)</td>
</tr>
<tr>
<td>FOT</td>
<td>TANGENTIAL COMP. OF INITIAL STROKE FORCES (LB)</td>
</tr>
<tr>
<td>TF</td>
<td>DURATION OF STROKING (SEC)</td>
</tr>
<tr>
<td>DEL1</td>
<td>STROKING OF NO.1 STRUT (FT)</td>
</tr>
<tr>
<td>DEL2</td>
<td>STROKING OF NO.2 STRUT (FT)</td>
</tr>
<tr>
<td>DEL3</td>
<td>STROKING OF NO.3 STRUT (FT)</td>
</tr>
<tr>
<td>W</td>
<td>WORK DONE AGAINST STROKING FORCES (FT*LB)</td>
</tr>
<tr>
<td>BFVCR</td>
<td>CRITICAL VALUE OF FINAL STABILITY ANGULAR RATE (RAD/SEC)</td>
</tr>
<tr>
<td>RFMIN</td>
<td>STABILITY INDEX FOR STABLE VEHICLE; THIS IS THE MINIMUM STABILITY ANGLE</td>
</tr>
<tr>
<td>RFVMIN</td>
<td>STABILITY INDEX FOR UNSTABLE VEHICLE; THIS IS THE STABILITY ANGULAR RATE</td>
</tr>
<tr>
<td>ERRBF</td>
<td>ERROR STANDARD FOR BETAF</td>
</tr>
<tr>
<td>ERRBFV</td>
<td>ERROR STANDARD FOR BETAFV</td>
</tr>
<tr>
<td>ERRLF</td>
<td>ERROR STANDARD FOR LF</td>
</tr>
<tr>
<td>RUN</td>
<td>RUN NUMBER</td>
</tr>
<tr>
<td>R</td>
<td>MULTIPLIER FOR NORMAL FORCE, USUALLY SET=2</td>
</tr>
<tr>
<td>BETFT</td>
<td>TRIAL SOLUTION FOR FINAL STABILITY ANGLE (RAD)</td>
</tr>
<tr>
<td>BETFVT</td>
<td>TRIAL SOLUTION FOR FINAL STABILITY ANGULAR RATE (RAD/SEC)</td>
</tr>
<tr>
<td>LFT</td>
<td>TRIAL SOLUTION FOR FINAL DISTANCE FROM FOOTPAD TO CO (PROJECTED IN PLANE</td>
</tr>
<tr>
<td>F,G,H</td>
<td>DIFFERENCE FUNCTIONS DENOTING DIFFERENCES BETWEEN LEFT AND RIGHT SIDES OF</td>
</tr>
<tr>
<td>INDEX</td>
<td>DIFFERENCE FUNCTION DENOTING QUALITY OF SET OF TRIAL SOLUTIONS</td>
</tr>
<tr>
<td>MINIDX</td>
<td>MINIMUM PREVIOUS VALUE OF INDEX</td>
</tr>
</tbody>
</table>

**Output**

PRINT SUMMARY OF INITIAL AND FINAL VALUES OF PARAMETERS

**Method**

SOLUTION OF THREE SIMULTANEOUS NONLINEAR ALGEBRAIC EQUATIONS IS ACCOMPLISHED BY A MODIFICATION OF NEWTON'S METHOD.

REAL LLF, I, M, L0, LFT, L1, L2, L3, LOVEL, LF, MINIDX, LFQ, INDEX

Figure 5-8. Main Program Listing (Continued)
PART 1  CALCULATION OF INITIAL CONDITIONS

3 READ 1000,1,M,GRAV,F11,F22,F33,X,Y,Z,XPAD,YPAD,ZPAD,PSO,PSOVEL,
   BETO,ERREF,ERLRFV,ERLRF,ALPHA,R,R,F
1000 FORMAT(20X,3F17.5)
RLAD 1002,D,E,XVEL,YVEL
1002 FORMAT(20X,2F17.5)
PRINT 951
PRINT 3059
3059 FORMAT(11H INPUT DATA)
PRINT 3060,I,M,GRAV,F11,F22,F33
3060 FORMAT(8H 1=F11,2=F11,3=F11,4=F11,5=F11,6=F11,7=F11,8=F11)
PRINT 3061,X,Y,Z,XPAD,YPAD,ZPAD
3061 FORMAT(8H X=,E11.5,9H Y=,E11.5,9H Z=,E11.5,9H
   XPAD=,E11.5,9H YPAD=,E11.5,9H ZPAD=,E11.5,9H)
PRINT 3062,PSO,PSOVEL,BETO,ERREF,ERLRFV,ERLRF
3062 FORMAT(8H PSO=,E11.5,9H PSOVEL=,E11.5,9H BETO=,E11.5,9H
   ERREF=,E11.5,9H ERLRFV=,E11.5,9H ERLRF=,E11.5,9H)
PRINT 3063,A,F,C,D,E,R
PRINT 3064,ALPHA,XVEL,YVEL,RUN
3064 FORMAT(8H ALPHA=,E11.5,9H XVEL=,E11.5,9H YVEL=,E11.5,9H
   1 RUN=,E11.5,9H)
CALL INCOND
CALL G0S

PART 2  OBTAIN STARTING POINT FOR EXTRAPOLATION

CHOICE OF BEST GUESS SET OF TRIAL SOLUTIONS. DECISION BASED ON
MINIMIZATION OF INDEX, A DIFFERENCE FUNCTION. INDEX=F*F+G*G+H*H
WHERE F,G,H ARE DIFFERENCES BETWEEN RIGHT AND LEFT SIDES OF
PRIMARY EQUATIONS (AFTER SUBSTITUTION OF TRIAL SOLUTIONS).

INITIALIZE MINIDX (MINIMUM PREVIOUS VALUE OF INDEX)

MINIDX=IE20
DO 1 J=1,6
DO 1 K=1,12
DO 1 N=1,6
BETO=BBF(J)
BETFQ=BBFV(K)
LFG=LLF(N)
CALL FEVAL
INDEX=F*F+G*G+H*H
IF(INDEX) 10,1,10

NOTE  THE ABOVE TEST IS NECESSITATED BY A FEATURE OF THE FEVAL
SUBROUTINE. THIS FEATURE PREVENTS PROGRESS STOPPAGED DUE TO SQUARE
ROOT SUBROUTINE ERROR DURING THE SEARCH FOR THE BEST SET OF TRIAL
SOLUTIONS. WHEN THIS FEATURE FUNCTIONS F,G,H ARE ARBITRARILY SET
EQUAL TO ZERO. THE RESULTING ZERO VALUE OF INDEX MUST THEN BE

Figure 5-8. Main Program Listing (Continued)
EXCLUDED FROM CONSIDERATION.

CHECK INDEX AGAINST MINIMUM PREVIOUS VALUE

10 IF (INDEX-MINIDX) 2.1.1

   IF (INDEX-MINIDX) < 2.1.1
   
   RESTART MINIDX AND DEFINE BEST SET OF TRIAL SOLUTIONS FOUND SO FAR.

2 MINIDX=INDEX
   BETFT=BBF(J)
   BETFV=BBF(V(K)
   LFT=LLF(N)

1 CONTINUE
   BETFQ=BETFT
   BETFV=BBF
   LQ=LFT

PART 3 SOLVING EQUATIONS

CALL SOLN
IF (CC-25) 6,7,7

THIS IS A TEST TO DETERMINE IF 25 ITERATIONS HAVE OCCURRED. IF YES, THE METHOD IS NOT CONVERGING AND PROGRAM IS TERMINATED. IT IS POSSIBLE THAT CONVERGENCE CAN BE OBTAINED BY GUESSING A DIFFERENT SET OF TRIAL SOLUTIONS

7 PRINT 500
500 FORMAT(31H ITERATION ASTRAY - GUESS AGAIN )
STOP

PART 4 OUTPUT RESULTS

6 PRINT 950
950 FORMAT(1H1)
   PRINT 100
100 FORMAT(19H INITIAL CONDITIONS )
   PRINT 951
   PRINT 2000, VOH, VOT, SUBX, SUBY, FOH, FOT
2000 FORMAT(8H VOH = ,E11.5,9H VOT = ,E11.5,9H SUBX = ,E11.5,9H F)
   PRINT 2001, L1, SINA, COSA, GAMMA, PGO, XDO
2001 FORMAT(8H L1 = ,E11.5,9H SINA = ,E11.5,9H COSA = ,E11.5,9H)
   PRINT 2002, YDOP, ZDOP, COSPO, SNBETO, XDO, YDO
2002 FORMAT(8H YDOP = ,E11.5,9H ZDOP = ,E11.5,9H COSPO = ,E11.5,9H S)
   PRINT 2003, L1, L2, L3, PSOV, PAO, SNBETO
2003 FORMAT(8H L1 = ,E11.5,9H L2 = ,E11.5,9H L3 = ,E11.5,9H P)
   PRINT 2004, CSBETO, AOT, ETAO, BETO, LOVEL, BETOV
2004 FORMAT(8H CSBETO = ,E11.5,9H AOT = ,E11.5,9H ETAO = ,E11.5,9H B)
   PRINT 951
   PRINT 951
   PRINT 200
200 FORMAT(17H FINAL CONDITIONS )
   PRINT 951
   PRINT 951

   CALCULATE CRITICAL STABILITY ANGULAR RATE

Figure 5-8. Main Program Listing (Continued)

BENDIX PRODUCTS AEROSPACE DIVISION
BFVCR = SQRT((2 * M * GRAV * LF * (1 - COS(BETAF))) / (1 + M * LF * LF))
BETFQ = BETAF
BETFVQ = BETAFV
LFQ = LF
CALL FEVAL
PRINT 2005, BETFT, BETFTV, LFT, BETAF, BETAFV, LF
2005 FORMAT(8H BETFT = EI1.5,9H BETFTV = EI1.5,9H LFT = EI1.5,9H)
1 BETAF = EI1.5,9H BETAFV = EI1.5,9H LF = EI1.5)
PRINT 2006, F, G, TF, VFH, PSF
2006 FORMAT(8H F = EI1.5,9H G = EI1.5,9H H = EI1.5,9H
1 TF = EI1.5,9H VFH = EI1.5,9H PSF = EI1.5)
PRINT 2007, DEL1, DEL2, DEL3, W, VO, BFVCR
2007 FORMAT(8H DEL1 = EI1.5,9H DEL2 = EI1.5,9H DEL3 = EI1.5,9H
1 W = EI1.5,9H VO = EI1.5,9H BFVCR = EI1.5)
PRINT 951
PRINT 951
951 FORMAT(1HD)

C TEST VEHICLE STABILITY BY COMPARING STABILITY ANGULAR RATE WITH

C CRITICAL STABILITY ANGULAR RATE

C IF (BETAFV - BFVCR) > 5

C DETERMINE MINIMUM STABILITY ANGLE IF VEHICLE STABLE

C 4 BFMIN = 1.5707963 + ASIN((1 + M * LF * LF) * BETAFV * BETAFV / (2 * M * GRAV * LF)

C COS(BETAF))

C 400 FORMAT(30H VEHICLE STABLE BFMIN = F10.7)

C DETERMINE MINIMUM STABILITY ANGULAR RATE IF VEHICLE UNSTABLE

C 5 BFVMIN = SQRT(BETAFV * BETAFV - 2.0 * M * GRAV * LF * (1 - COS(BETAF)) / (1 + M * LF * LF))

C 400 FORMAT(30H VEHICLE UNSTABLE BFVMIN = F10.7)

C GO TO 3

END
TO COMPUTE INITIAL CONDITIONS, THESE CONDITIONS ARE THE VEHICLE MOTIONS AND CONFIGURATION AT FRONT LEG IMPACT

FROM INPUT DATA TO MAIN PROGRAM

VH
INITIAL HORIZONTAL CG VELOCITY (FT/SEC)

GAMMO
INITIAL ANGLE (PROJECTED IN PLANE OF MOTION)
INCLUDED BETWEEN FOOTPAD, CG AND VEHICLE LONGITUDINAL AXIS (RAD)

VOT
INITIAL TANGENTIAL CG VELOCITY (FT/SEC)

XDOP
X-COMPONENT OF FOOTPAD RELATIVE TO CG (GROUND AXES) (FT)

YDOP
Y-COMPONENT OF FOOTPAD RELATIVE TO CG (GROUND AXES) (FT)

XDO
X-COMPONENT OF FOOTPAD RELATIVE TO CG (VEHICLE AXES) (FT)

YDO
Y-COMPONENT OF FOOTPAD RELATIVE TO CG (VEHICLE AXES) (FT)

ZDO
Z-COMPONENT OF FOOTPAD RELATIVE TO CG (VEHICLE OR GROUND AXES) (FT)

L1
LENGTH OF NO.1 STRUT (FT)

L2
LENGTH OF NO.2 STRUT (FT)

L3
LENGTH OF NO.3 STRUT (FT)

FSUBX
X-COMPONENT OF INITIAL STROKING FORCES (VEHICLE AXES) (L_B)

FSUBY
Y-COMPONENT OF INITIAL STROKING FORCES (VEHICLE AXES) (L_B)

TQ
TORQUE ABOUT Z-AXIS INDUCED BY INITIAL STROKING FORCES (FT-LB)

PSOVV
INITIAL PITCH ANGULAR ACCELERATION (RAD/SEC/SEC)

FOH
HORIZONTAL COMPONENT OF INITIAL STROKING FORCES (L_B)

FOT
TANGENTIAL COMPONENT OF INITIAL STROKING FORCES (L_B)

AOT
INITIAL TANGENTIAL COMPONENT OF CG ACCELERATION (FT/SEC/SEC)

ETAOVV
INITIAL HORIZONTAL COMPONENT OF CG ACCELERATION (FT/SEC/SEC)

LO
INITIAL LENGTH FROM FOOTPAD TO CG (PROJECTED IN PLANE OF MOTION) (FT)

BETOV
INITIAL STABILITY ANGULAR VELOCITY (RAD/SEC)

LOVEL
INITIAL RATE OF CHANGE OF LENGTH FROM FOOTPAD TO CG (PROJECTED IN PLANE OF MOTION) (FT/SEC)

BETOVV
INITIAL STABILITY ANGULAR ACCELERATION (RAD/SEC/SEC)

VO
INITIAL MAGNITUDE OF CG VELOCITY (FT/SEC)

GRAV
GRAVITATIONAL ACCELERATION (FT/SEC/SEC)

输出
THESE PARAMETERS USED FOR CALCULATIONS ELSEWHERE IN PROGRAM, VALUES OF SELECTED PARAMETERS ARE PRINTED VIA PRINT STATEMENT IN MAIN PROGRAM

Figure 5-9. Subroutine INCOND
SUBROUTINE INCOND
REAL LLF*, I*, V*, LC*, LF*, I NDX*, LF*, INDEX
3ETAOVV*, LO*ETOVV*, LOVEL*, SETOVV*, VO*, FSOB*, FSOV*, BETVO*, TFVO*, LFO*V*,
7FBV*, FLGB*, GVG*, GL*, HR*, HBV*, HL*
SINA=SIN(ALPHA*)
COSA=COS(ALPHA*)
VOH=XVEL*SINA+YVEL*COSA
GAMMO=1.5708-ALPHA+RFTO-PSO
PGO=PSO+GAMMO
CO_SGO=COS(PGO)
SINGO=SIN(PGO)
VOT=XVEL*CO_SGO+YVEL*SINGO
XDOP=XPAD-X
YDOP=YPAD-Y
ZDO=ZPAD-Z
CO_SPO=CO_SGO+YDOP*SINGO
SINGO=SIN(PGO)
XDO=XDOP*CO_SPO+YDOP*SINGO
YDO=YDOP*CO_SPO-XDOP*SINGO
L1=SQRT((A-XDC)*(A-XDO)+(E-YDO)*(E-YDO)+(E-ZDO)*(E-ZDO))
L2=SQRT((-B-XDO)*(-B-XDO)+(D-YDO)*(D-YDO)+(C-ZDO)*(C-ZDO))
L3=SQRT((-B-XDO)*(-B-XDO)+(C-YDO)*(C-YDO)+(D-ZDO)*(D-ZDO))
FSUBX=2.0*FII/L1*X-XDO)+2.0*F22/L2*I-B-XDO)+2.0*F33/L3*(-B-XDO)
FSUBY=2.0*FII/L1*IE-YDO)+2.0*F22/L2*IC-YDO)+2.0*F33/L3*(D-YDO)
TO=XDO*FSUBY-YDO*FSUBX
PSOVV:TO/I
PAO=PSO+ALPHA
SINGO=SIN(PAO)
CO_SPO=CO_SAO+FSUBX*SINGO
SINGO=SIN(GAMMO)
CO_SGO=CO_SGO+FSUBX*CO_SGO
SINGO=SIN(PAO)
CSETOG=SIN(FETO)
CSETOG=SIN(BETO)
FOT=FSUBX*CO_SGO+FSUBX*SINGO+GRAV*SNBETO
AOT=FOT/M
ETAOVV=FOH/M
LO=SQRT((XPAD-X)*(XPAD-X)+(YPAD-Y)*(YPAD-Y))
ETOVV=VOT/LO
LOVEL=XVEL*SINGO-YVEL*CO_SGO
RETOVV=FOT-2.*LOVEL*ETOVV/LO
VO=SQRT(XVEL*XVEL+YVEL*YVEL)
RETURN
END

Figure 5-9. Subroutine INCOND (Concluded)
TO GUESS TRIAL SOLUTIONS TO THE NONLINEAR EQUATIONS SO THAT THE EXTRAPOLATION WILL COMMENCE IN THE VICINITY OF THE TRUE SOLUTION

SUBROUTINE GUESS
REAL LLF, I, Y, LO, LFT, L1, L2, L3, LOVEL, LF, MINIDX, LFG, INDEX
COMMON ALPHA, XVEL, YVEL, ZETAO, PSO, X, Y, Z, XPAD, YPAD, ZPAD, I, M, A, B, C, D,
1E, F11, F22, F33, FSOVL, GRAV, SIN, COS, VOH, GAMMO, PGO, COSPGO, SINPGO,
2VOT, ZD0, COSPO, SINPO, L1, L2, L3, PSO, VOH, SNBETO, CWEBETO, F0H, AOT,
3ETAQV, LO, BETOV, LOVEL, BETOVV, VO, FSOV, FSOBY, BETFQ, BETFVQ, LFQ, R,
4ETAVV, TERM, TERM, SIND, COSB, VFF, TF, BETFFQ, BETAFV, PSF, SINGF,
5COSG, C0, C1, C2, W, CC, FG, H, FX, FY, FZ, GX, GY, GZ, HX, HY, HZ, PAO, LF,
6ERBFV, ERBFV, ERBFV, BBF(6), BBFV(12), LLF(6), XDO, YDO, XDOP, YDOP, FB,
7FBF, FL, GB, GBV, GL, HB, HSV, HL
BBF(1) = .6 * BETO
BBF(2) = .75 * BETO
BBF(3) = .8 * BETO
BBF(4) = .85 * BETO
BBF(5) = .9 * BETO
BBF(6) = .95 * BETO
BBFV(1) = .6 * BETOV
BBFV(2) = .7 * BETOV
BBFV(3) = .8 * BETOV
BBFV(4) = .9 * BETOV
BBFV(5) = .95 * BETOV
BBFV(6) = 1 * BETOV
BBFV(7) = 1.05 * BETOV
BBFV(8) = 1.1 * BETOV
BBFV(9) = 1.15 * BETOV
BBFV(10) = 1.2 * BETOV
BBFV(11) = 1.3 * BETOV
BBFV(12) = 1.4 * BETOV
LLF(1) = .75 * LO

Figure 5-10. Subroutine GUESS
Figure 5-10. Subroutine - GUESS (Concluded)

BENDIX PRODUCTS AEROSPACE DIVISION
TO DETERMINE VALUES OF THE DIFFERENCE FUNCTIONS FOR VARIOUS SETS OF TRIAL SOLUTIONS

Either from trial set of solutions (subroutine GUESS or SOLN) or from final solution

Symbol                     Definition

RETFT                      TRIAL SOLUTION FOR FINAL STABILITY ANGLE (RAD)
BETFVT                     TRIAL SOLUTION FOR FINAL STABILITY ANGULAR RATE (RAD/SEC)
LFT                        TRIAL SOLUTION FOR FINAL DISTANCE FROM FOOTPAD TO CG (PROJECTED IN PLANE OF MOTION) (FT)
BETFQ                      THESE QUANTITIES ARE DUMMY VARIABLES TO ALLOW ENTRY TO SUBROUTINE FROM DIFFERENT POINTS
BETFVQ                     THESE QUANTITIES ARE CONVENIENT GROUPINGS OF PARAMETERS
TERMA                      THESE QUANTITIES ARE CONVENIENT GROUPINGS OF PARAMETERS
ETAFVV                     FINAL HORIZONTAL ACCELERATION OF CG (FT/SEC/SEC)
R                          MULTIPLIER FOR NORMAL FORCE, USUALLY = 2
VFH                        FINAL HORIZONTAL VELOCITY OF CG (FT/SEC)
CORK                       ARTIFICIAL PARAMETER DEFINED TO ALLOW A TEST OF A QUANTITY UNDER A RADICAL
F,G,H                      DIFFERENCE FUNCTIONS DENOTING DIFFERENCES BETWEEN LEFT AND RIGHT SIDES OF PRIMARY EQUATIONS AFTER SUBSTITUTION OF TRIAL SOLUTIONS
BETFVV                     FINAL STABILITY ANGULAR ACCELERATION (RAD/SEC/SEC)
TF                         DURATION OF STROKING PERIOD (SEC)
BETA                       FINAL STABILITY ANGLE (RAD)
BETAFF                     FINAL STABILITY ANGULAR RATE (RAD/SEC)
PSF                        FINAL PITCH ANGLE (RAD)
GAMMF                      FINAL VALUE OF ANGLE (PROJECTED IN PLANE OF MOTION) OF ANGLE INCLUDED BETWEEN FOOTPAD, CG AND VEHICLE LONGITUDINAL AXIS (RAD)
XD                         X-COMPONENT OF FOOTPAD RELATIVE TO CG (VEHICLE AXES) (FT)
YD                         Y-COMPONENT OF FOOTPAD RELATIVE TO CG (VEHICLE AXES) (FT)
DEL1                       SHORTENING OF NOS 1, 2, 3 STRUTS RESPECTIVELY (FT)
W                           WORK DONE IN STROKING (FT-LB)

F,G,H ARE USED IN THE EXTRAPOLATION PROCESS. VALUES OF SELECTED PARAMETERS ARE PRINTED VIA PRINT STATEMENT IN MAIN PROGRAM.

SUBROUTINE FEVAL
REAL LLF, L1, L2, L3, LOVEL, LF, MINIDX, LFQ, INDEX
COMMON ALPHA, XVEL, YVEL, BETO, PSO, X, Y, Z, XPAD, YPAD, ZPAD, I, M, A, B, C, D, E, F11, F22, F33, PSOVEL, GRAV, SINA, COSA, VOH, GAMMO, PGO, COSPGO, SINPGO

Figure 5-11. Subroutine FEVAL
DEFINE AND TEST COPK. THIS IS A TEST OF THE QUANTITY WHICH APPEARS
UNDER THE RADICAL IN THE CALCULATION OF TF. THIS TEST HAS BEEN
INCLUDED TO ASSURE PROGRAM CONTINUITY IN THE EVENT THAT ONE OF THE
SETS OF TRIAL SOLUTIONS (SUBROUTINE GUESS) SHOULD PRODUCE A
NEGATIVE QUANTITY AT THIS POINT (THIS IS NOT UNUSUAL). THIS
FEATURE PERMITS AUTOMATIC REJECTION OF THAT SET OF TRIAL SOLUTIONS
BEFORE A SORT SUBROUTINE ERROR IS ENCOUNTERED. HOWEVER, IF THE
QUANTITY UNDER THE RADICAL SHOULD BECOME NEGATIVE DURING THE
EXTRAPOLATION (AS CAN HAPPEN IF THE PROCESS IS NOT CONVERGENT)
THEN THE PROCESS WILL CAUSE OVERFLOW.

\[
\text{CORK} = \text{TERM} - \text{TERM} - 12.0 \times \left( \frac{\text{LO} \times \text{SNBETO} - \text{LFT} \times \text{SNB}}{\text{ETAOVV} - \text{ETAFFVF}} \right)
\]

\text{IF(CORK) 1, 2, 2}

1 \text{ F=0}
\text{ G=0}
\text{ H=0}
\text{ GO TO 3}

2 \text{ TF=-TERM-SQRT(CORK)}
\text{ BETAFVF=TERM*grav*LFT*SNB}
\text{ BETAF=BETO+BETOV+RETFFVT}/2.0*TF
\text{ +L*ETAFVF+30.0*TF*TF}
\text{ BETAFV=TERM*grav*LO*TF*SNBETO}
\text{ +ETAOVV-2.0*ETAFVF} / 20.0*TF*TF / (3.0*ETAOVV-2.0*ETAFVF)

1 \text{ F=ETAF-BETFT}
\text{ G=ETAF-BETFT}
\text{ PSF=PSO+(PSOVEL+BETFVT)/2.0*TF}
\text{ +ETAOVV-ETAFFVF} / 12.0*TF*TF
\text{ GAMMF=1.5708-ALPHA+BETFT-PSF}
\text{ SINGF=SIN(GAMMF)}
\text{ COSGF=COS(GAMMF)}
\text{ XD=-LFT*SINGF}
\text{ YD=LFT*COSGF}
\text{ DEL1=L1-SQRT((XD-A)*(XD-A)+(YD-E)*(YD-E)+(ZDO-E)*(ZDO-E))}
\text{ DEL2=L2-SQRT((XO+B)*(XO+B)+(YD-D)*(YD-D)+(ZDO-C)*(ZDO-C))}
\text{ DEL3=L3-SQRT((XO+B)*(XO+B)+(YD-C)*(YD-C)+(ZDO-D)*(ZDO-D))}
\text{ W=+0.0*F11*DEL1*F22*DEL2+F33*DEL3}
\text{ H=1/2.0*PSOVEL+PSOVEL+H*VVO+VFO+VTO+ETAFVF/2.0*TF}
\text{ 1+TERM1*grav*LO*CSBETO-LFT*COSB}\); w

3 RETURN

END

Figure 5-11. Subroutine FEVAL (Concluded)
SUBROUTINE SOLN
REAL LLF,LF,LO,L1,L2,L3,LOVEL,LFQ,INDEX
COMMON ALPHA,XVEL,YVEL,BETO,PSO,X,Y,Z,XPAD,YPAD,ZPAD,1,S,M,A,B,C,D,
1E,F11,F22,F33,P5OVEL,GRAV,SINA,COSA,VOH,GAMMO,PGO,COSPGO,SINPGO,
2V0,TZ0,COSPGO,SINPGO,L12,L3,P3OVEL,FOH,XEETO,CSETO,F0,AT,
3ETAVV,LC,BETVV,LOVEL,BETOVV,VO,FSUBX,FSUBY,BETFQ,BETFVQ,LFQ,R,
4LTAVV,TERMA,TERMB,SINB,COSB,VFH,TF,BETFVV,BETAF,BETAVF,PSF,SINGF,
5COSGF,DEL1,DEL2,DEL3,CC,F,G,H,F,Y,FZ,GX,GY,GZ,HX,HY,HZ,PAG,LF,
6ERRBF,ERRBFV,ERRLF,BETF,V,BETFVF,LLF,LLF,LLF,LLF,LLF,LLF,LLF,LLF,
7FV,FL,GR,GRV,GL,HR,HRV,HL

INITIALIZE CC, THE NUMBER OF EXTRAPOLATIONS WHICH HAVE BEEN
PERFORMED
CC = 0

Figure 5-12. Subroutine SOLN
REDEFINE PARAMETERS FOR CONVENIENCE IN PRINTING

\[
\begin{align*}
XF &= XEFQ \\
YF &= YEFVQ \\
ZF &= LFQ \\
\end{align*}
\]

PRINT 951
PRINT 951
PRINT 77

77 FORMAT(15H ITERATION DATA )

GO TO 8
1 BETFQ=XF
BETFVQ=YG
LFQ=2F
8 CALL FEVAL

TEST IF \((F+G+H)\) TO PREVENT OVERFLOW IF THIS QUANTITY \(=0\) EXACTLY

IF \((F+G+H)\) 5,6,5
6 G=1.0/F
5 CALL DEVAL

DETERMINE EXTRAPOLATION INCREMENTS

\[
\begin{align*}
DEL &= FX*GY*HZ+FY*GZ*HX+FZ*GX*HY-FZ*GY*HX-FY*GX*HZ-FX*GZ*HY \\
DX &= (-F*GY*HZ-FG*GZ*FY*GZ*HY+FY*GZ*HZ+F*GZ*HY)/DEL \\
DY &= (-FX*GY*HZ-FG*GZ*FY*GZ*HX+F*GZ*HX+FX*GZ*HY)/DEL \\
DZ &= (-FX*GY*HZ-FG*GZ*FY*GZ*HX+F*GZ*HX+FX*GZ*HY)/DEL \\
PRINT 951
951 FORMAT(1Hn)
PRINT 2008, XF,YF,ZF,DX,DY,DZ
2008 FORMAT(8H XF=,E11.5,YF=,E11.5)ZF=,E11.5
1 DX=,E11.5,YF=,E11.5,DY=,E11.5,DZ=,E11.5
PRINT 2009,F,G,H
2009 FORMAT(8H F=,E11.5,G=,E11.5,H=,E11.5)

DETERMINE NEW VALUE OF VARIABLES BASED ON HALF THE INCREMENT. THE
FACTOR OF A HALF IS INTRODUCED ARBITRARILY FOR STABILIZATION OF
THE EXTRAPOLATION PROCEDURE

\[
\begin{align*}
XF &= XF+DX*.5 \\
YF &= YF+DY*.5 \\
ZF &= ZF+DZ*.5 \\
\end{align*}
\]

CALL DEVAL

COUNT NUMBER OF EXTRAPOLATIONS AND STOP PROGRAM IF CONVERGENCE
DOES NOT OCCUR IN 25 STEPS

CC=CC+1
IF (CC=25) 7,4,4
7 IF (ABS(DX)-ERRBF) 2,2,1
2 IF (ABS(DY)-ERRBFV) 3,3,1
3 IF (ABS(DZ)-ERRLF) 4,4,1
4 RETURN
END

Figure 5-12. Subroutine SOLN (Concluded)
DEVAL

R.G.ALDERSON AND L.E.ANDERSON

10-31-64

TO DETERMINE VARIOUS PARTIAL DERIVATIVES OF THE THREE
DIFFERENCE FUNCTIONS WITH RESPECT TO INDEPENDENT
VARIABLES

DEVAL

FROM BEST SET OF TRIAL SOLUTIONS AND FROM QUANTITIES
EVALUATED IN MAIN PROGRAM

ONLY PREVIOUSLY UNDEFINED TERMS WILL BE DEFINED HERE

PARTIAL TF WITH RESPECT TO BETFT
PARTIAL TF WITH RESPECT TO BETFT

BETFT
BETFT

3-13. Subroutine DEVAL

5-27
THE FOLLOWING DETERMINATION OF GL HAS BEEN SPLIT DUE TO COMPUTER LIMITATIONS. HODGE AND PUDGE ARE DUMMY VARIABLES.

HODGE = M * GRAV * TERMA * ((LO * SNBETO + (7.0 * VOH + 3.0 * VFH) / 10.0) * TFF + (3.0 * ETAQVV - 2.0 * ETAQVV) / 20.0) / TF * TF / 20.0 * TF * TF / 12.0 * VFHL

PODGE = 2.0 * M * LFT * TERMA * ((LO * SNBETO + (7.0 * VOH + 3.0 * VFH) / 10.0) * TFF + (3.0 * ETAQVV - 2.0 * ETAQVV) / 20.0) / TF * TF * TF / 12.0 * VFHL

GL = HODGE - PODGE

TERMG = (PSOVEL + BETFVT) / 2.0 + (PSOVEL + BETFVT) / 6.0 * TF

GAMBV = -TERMG * TFB + TF * TF / 12.0 * VFHL

GAMB = 1.0 + M * GRAV * TERMA * ((LO * SNBETO + (7.0 * VOH + 3.0 * VFH) / 10.0) * TFF + (3.0 * ETAQVV - 2.0 * ETAQVV) / 20.0) / TF * TF * TF / 12.0 * VFHL

GL = HODGE - PODGE

TERMG = (PSOVEL + BETFVT) / 2.0 + (PSOVEL + BETFVT) / 6.0 * TF

GAMBV = -TERMG * TFB + TF * TF / 12.0 * VFHL

GAMB = 1.0 + M * GRAV * TERMA * ((LO * SNBETO + (7.0 * VOH + 3.0 * VFH) / 10.0) * TFF + (3.0 * ETAQVV - 2.0 * ETAQVV) / 20.0) / TF * TF * TF / 12.0 * VFHL

Figure 5-13. Subroutine DEVAL (Continued)
GAML = TERN*TF*TFF / 12.0*AVVL
TERN3 = Sqr((LFT*SINGF - 6)**2 + (LFT*COSGF - C)**2 + (ZDO - D)**2)
TERM1 = Sqr((LFT*SINGF - D)**2 + (LFT*COSGF - D)**2 + (ZDO - C)**2)
TERM2 = Sqr((LFT*SINGF - A)**2 + (LFT*COSGF - E)**2 + (ZDO - E)**2)

DELBV = (A*LFT*COSGF + LFT*SINGF)*GAMBV/TERM1
DEL2BV = (B*LFT*COSGF + D*LFT*SINGF)*GAMBV/TERM2
DEL3BV = (B*LFT*COSGF + C*LFT*SINGF)*GAMBV/TERM3

HVF = FTVFT/TERM - 2.0*(F11*DEL1BV + F22*DEL2BV + F23*DEL3BV)

DEL1V = -(LFT*(A*COSGF + E*SINGF)*GAMBV/TERM1
DEL2V = -(LFT*(-B*COSGF + D*SINGF)*GAMBV/TERM2
DEL3V = -(LFT*(-B*COSGF + C*SINGF)*GAMBV/TERM3

H3 = M*GRAV*LFT*SINB - 2.0*(F11*DEL1B + F22*DEL2B + F33*DEL3B)

DEL1L = -(LFT*(A*COSGF + E*SINGF)*GAML + LFT*A*SINGF -
1E*COSGF)/TERM1
DEL2L = -(LFT*(-B*COSGF + D*SINGF)*GAML + LFT*B*SINGF -
1D*COSGF)/TERM2
DEL3L = -(LFT*(-B*COSGF + C*SINGF)*GAML + LFT*B*SINGF -
1C*COSGF)/TERM3

HL = -M*LFT*RETFTV*3*FTVT - M*GRAV*COSL - 2.0*(F11*DEL1L +
1F22*DEL2L + F33*DEL3L)

FX = FB
FY = FB
FZ = FL
GX = GR
GY = GRV
GZ = GL
HX = HB
HY = HRV
HZ = HL
RETURN
END

Figure 5-13. Subroutine DEVAL (Concluded)
**INPUT DATA**

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>UX</th>
<th>UY</th>
<th>UZ</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
</tr>
</thead>
<tbody>
<tr>
<td>-11.5487</td>
<td>-1.8453</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**ITERATIVE DATA**

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>UX</th>
<th>UY</th>
<th>UZ</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>-1.11639</td>
<td>-2.11639</td>
<td>-0.3059</td>
<td>0.4059</td>
<td>-0.4509</td>
<td>-0.0060</td>
<td>1.0500</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**Figure 5-14. Output Data**
### Initial Conditions

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCH</td>
<td>1.2457+1</td>
</tr>
<tr>
<td>VAT</td>
<td>4.6244+1</td>
</tr>
<tr>
<td>FSWAY</td>
<td>2.8457+6</td>
</tr>
<tr>
<td>FSUR Y</td>
<td>-3.3184+0</td>
</tr>
<tr>
<td>FOM</td>
<td>-3.6976+0</td>
</tr>
<tr>
<td>FCT</td>
<td>3.9242+0</td>
</tr>
<tr>
<td>LOC</td>
<td>1.8456+2</td>
</tr>
<tr>
<td>SIA</td>
<td>2.5057+2</td>
</tr>
<tr>
<td>CUSAS</td>
<td>9.6593+0</td>
</tr>
<tr>
<td>GAMMO</td>
<td>7.7400+0</td>
</tr>
<tr>
<td>PGM</td>
<td>8.6500+0</td>
</tr>
<tr>
<td>XDDP</td>
<td>-1.2194+0</td>
</tr>
<tr>
<td>YDD</td>
<td>1.4064+0</td>
</tr>
<tr>
<td>LCE</td>
<td>1.1970+2</td>
</tr>
<tr>
<td>LCM</td>
<td>1.1615+2</td>
</tr>
<tr>
<td>MSWV</td>
<td>-1.4855+6</td>
</tr>
<tr>
<td>PSNW</td>
<td>-3.7280+0</td>
</tr>
<tr>
<td>SBET0</td>
<td>-4.1141+0</td>
</tr>
</tbody>
</table>

### Final Conditions

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETFT</td>
<td>1.2906+0</td>
</tr>
<tr>
<td>BETVFE</td>
<td>1.2232+0</td>
</tr>
<tr>
<td>LFT</td>
<td>1.6683+2</td>
</tr>
<tr>
<td>BETAFT</td>
<td>-3.6626+0</td>
</tr>
<tr>
<td>BLTAFV</td>
<td>2.5579+0</td>
</tr>
<tr>
<td>LF</td>
<td>1.1623+0</td>
</tr>
<tr>
<td>UVC</td>
<td>0.01192-6</td>
</tr>
<tr>
<td>LLL</td>
<td>-1.1230+0</td>
</tr>
<tr>
<td>ELL</td>
<td>-1.0556+0</td>
</tr>
<tr>
<td>USW</td>
<td>-1.0764+0</td>
</tr>
<tr>
<td>LSL</td>
<td>-1.0500+0</td>
</tr>
</tbody>
</table>

### Vehicle Installed

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>UVW1</td>
<td>1.74677</td>
</tr>
</tbody>
</table>

---

**Figure 5-14. Output Data (Concluded)**
SECTION VI

LANDING DYNAMICS COMPUTER PROGRAM FOR NON-PLANAR LANDINGS
WITH INFINITE GROUND COEFFICIENT OF FRICTION AND ZERO FOOTPAD MASS

THEORETICAL ANALYSIS OF NON-PLANAR MOTION

This section describes a six degree of freedom mathematical model for prediction of motion and stability for the case of infinite friction between the feet and the lunar surface. Sufficient runs have been made with this program to provide a comparison between a theoretical and experimental stability profile. It is emphasized, however, that detailed comparative studies of vehicle motion during the landing have not been carried out at this time. Several areas in the program such as strut re-extension and foot contact conditions are open to question and will have to be more thoroughly investigated before this program could be used extensively. Results to date do indicate a reasonable correlation with drop test results from the standpoint of gross stability predictions.

Discussion of Mathematical Model

The following is a brief discussion of the six degree of freedom computer program which is fully discussed along with definition of terms, flow diagram, and sample input and output in the documentation of computer programs report.

The moving (vehicle) coordinate system and leg numbering system is as shown in Figure 6-1.

The x axis is a line from the c.g. on the vehicle centerline. The y axis is a line through the c.g., perpendicular to the x axis and in the direction and plane of the number one leg of the number one leg set. The z axis passes through the c.g. and is perpendicular to both the x and y axis and in the direction of the number 4 leg set. The struts are numbered clockwise from the y axis as shown in Figure 6-1.

The fixed (ground) coordinate system and output variables are shown in Figure 6-2. The X axis is perpendicular to the slope, the Y axis is parallel to the slope and in the direction of the principal slope. The Z axis is perpendicular to both the X and Y axis (i.e., across the slope). The variables are defined as follows:

Input Variables

\[ X_B(i, j) = x \text{ of the } i\text{th strut of the } j\text{th leg set body attachment (hardpoint) in feet} \]
\[ Y_B(i, j) = y \text{ of the } i\text{th strut of the } j\text{th leg set body attachment (hardpoint) in feet} \]
Figure 6-1. Moving (Vehicle) Coordinate System and Leg Numbering System

Z B (i, j) = z of the ith strut of the jth leg set body attachment (hardpoint) in feet. x, y, and z are in terms of the vehicle coordinate system and relative to this system.

Numbering of Leg Sets, Hardpoints, and Struts - See Figure 6-1

TYPE (i) = Type of strut i (e.g. Strut No. 1 is of Type 1, Strut Nos. 2 and 3 are Type 2 on each leg set).
VASLUG = Mass of Vehicle in slugs.
U1 = Mass moment of inertia of vehicle about vertical, x-axis (yaw), vehicle coordinate system, in slug ft².
U2 = Mass moment of inertia of vehicle about the y-axis (roll), vehicle coordinate system, in slug ft².
U3 = Mass moment of inertia of vehicle about the z-axis (pitch), vehicle coordinate system, in slug ft².
Figure 6-2. Fixed (Ground) Coordinate System and Output Variables
FOTAC or \_ = Mode of action of the \( i \)th footpad where value

*FOTAC (1(i))*

1 Corresponds to sliding on surface
2 Corresponds to stationary on surface
3 Corresponds to off surface and thus moving with the vehicle.

\( \text{V FOT}(j, i) \) = \( i \)th component of the vector position of the \( j \)th footpad in terms of


vehicle coordinates and relative to this system.

\( i = 1 \) is \( x \) component (in feet).
\( i = 2 \) is \( y \) component (in feet).
\( i = 3 \) is \( z \) component (in feet).

\( \text{XCG or XCG1} \) = Position of vehicle center of gravity in fixed (ground) coordinate


system and relative to this system, in feet.

\( \text{YCG or YCG1} \) =

\( \text{ZCG or ZCG1} \) =

\( \text{V VEL} \) = Initial vertical velocity of vehicle center of gravity parallel to


gravity vector, in ft./sec.

\( \text{H VEL} \) = Initial horizontal velocity of vehicle center of gravity, perpendicular
to gravity vector and in the direction of the principal slope in ft./sec.

\( \text{ZDCG or ZDCG1} \) = Horizontal velocity of the vehicle center of gravity across the


slope, in ft./sec.

\( \text{W1E or W1E1} \) = Angular velocity components about the \( x \), \( y \), and \( z \) vehicle axes respec-
tively and expressed in terms of vehicle coordinate system in


Rads/sec.

\( \text{W2E or W2E1} \) =

\( \text{W3E or W3E1} \) =

\( \text{PITCH or PITCH 1} \) = Initial vehicle orientation, in radians. See text on initial orientation


of vehicle.

\( \text{YAW or YAW 1} \) =

\( \text{ROLL or ROLL 1} \) =

\( \text{ETA or ETA1} \) = Ground slope, positive for downhill in positive \( X \) direction (fixed co-


ordinate system) in radians.

Output Variables

\( \text{TIME} \) = Time after touchdown in seconds.

\( \text{XCG} \) = Position of vehicle center of gravity in fixed ground coordinates, in ft.

\( \text{YCG} \) =

\( \text{ZCG} \) =

\( \text{PHI} \) = Angular orientation of the vehicle relative to the fixed coordinates


\( \text{PSI} \) = system, in terms of the projections of the \( x \) and \( y \) vehicle axis on the


XY, XZ, and YZ fixed (ground) coordinate system planes, in radians

(see Figure 6-2 and sketch on the following page).

* Appearance of "1" after any variable indicates the initial value of the


variable. Absence of the "1" indicates instantaneous values.
Analytically

\[ \psi = \tan^{-1} \left( \frac{f_{21}}{f_{11}} \right) \]

\[ \phi = \tan^{-1} \left( \frac{-f_{31}}{f_{11}} \right) \]

\[ \theta = \tan^{-1} \left( -\frac{f_{32}}{f_{22}} \right) \]

**XDCG** = Velocity of vehicle center of gravity in terms of the fixed coordinate system, feet/second.

**YDCG**

**ZDCG**

**FORCE X** = Net force acting on vehicle center of gravity in direction of X of the fixed coordinate system, in pounds. (Includes gravity and footpad action.)

**TORQZ** = Component, parallel to fixed coordinate Z-axis, of torque acting about c.g.

**BETA** = Stability angle. See text on stability angle.

**PX1**

**PX2**

**PX3**

**PX4** = Height above surface of footpads one through four respectively, in feet.
FX1, FX2, FX3, FX4

- Force acting perpendicular to the surface on footpads one through four, respectively, in pounds.

XBFT1, XBFT2, XBFT3, XBFT4

- x component of VFOT for footpads one through four, respectively, in feet. Or the component of the distance from the vehicle center of gravity to four footpads in the direction parallel to the vehicle axis of symmetry.

ON MOON

- Four digits representing FOTAC of footpads one through four, respectively.

THEDT

- Components of vehicle angular velocity in the X, Y, and Z directions for the fixed coordinate system. NOTE: These are NOT the rate of change of THETA, PHI, and PSI previously defined except in the 1-2-1 landing configuration.

PHIDOT

- Rate of change of BETA.

PSIDOT

- Twelve digits representing the force causing mechanism in the twelve leg struts in the order

<table>
<thead>
<tr>
<th>Leg Set No.1</th>
<th>Leg Set No.2</th>
<th>Leg Set No.3</th>
<th>Leg Set No.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strut No.1, No.2, No.3</td>
<td>Strut No.1, No.2, No.3</td>
<td>Strut No.1, No.2, No.3</td>
<td>Strut No.1, No.2, No.3</td>
</tr>
</tbody>
</table>

The digit code is as follows:

0 = No force because the footpad is off the surface.
1 = Elastic compression in strut.
2 = Plastic compression in strut.
3 = Strut re-extended from previously crushed position but is still shorter than the original length.
8 = Strut stretched beyond original length, but less than 10,000 lb. force.
9 = Strut in tension more than 10,000 lbs.

The basic computational flow required to determine the trajectory of a non-planar lunar landing is as follows:

1. From the geometry of the leg sets, calculate the forces in each leg set (see subroutine FORCE) and lift from the surface any footpad with negative ground force.

2. From these forces, at the footpads and center of gravity, calculate the forces and torques acting on the vehicle and convert to moving (vehicle) coordinate system.

3. Integrate (see subroutine INTEQ) the Euler and Newton equations to find the new vehicle positions expressed in terms of center of gravity translation and vehicle orientation, by direction cosines. (\( \Delta t \) is determined by the rate of change of the torques and later checked in INTEQ.)
4. Check to see if any footpads have just contacted the surface and compute geometry of all the leg sets in contact with the surface.

5. Check stability and print sufficient information to reveal trajectory details for analysis.

6. Return to 1.

The infinite surface - footpad coefficient of friction used in the calculations significantly simplified the process. No slip of any footpad on the surface is allowed. Footpads are removed from the surface if the contact force between the surface and the footpad becomes negative. Any footpad not on the surface is assumed to move in rigid body rotation with the vehicle.

Finite friction coefficient simulations have not been conducted as yet, because of the difficulty of finding the pad motions satisfying the requirements or the constraints:

1. \( F_{\text{friction}} = \mu F_{\text{normal}} \)

2. Direction of friction force = opposite to direction of motion.

Infinite surface friction, generally, is the most stringent (de-stabilizing) case and best simulates the spike footpad experimental work.

Coordinate Transformations

Using the direction cosines which describe the relative position of the two coordinate systems, \( X, Y, \) and \( Z \) fixed (ground) coordinates and \( x, y, \) and \( z \) moving (fixed in vehicle) coordinates as shown below (see also Figures 6-2 and 6-1, respectively), we can convert quantities expressed in terms of one coordinate system to the other coordinate system.

![Fixed (Ground) and Moving (Vehicle) Coordinate Systems](image-url)
The equations are in matrix form
\[
\begin{align*}
\{X_i\} &= \left[ \mathbf{L}_{ij} \right] \{x_{ij}\} + \{v_i, \text{ c.g.}\} \tag{1}
\end{align*}
\]
for conversion from vehicle to ground coordinate systems

and
\[
\begin{align*}
\{x_j\} &= \left[ \mathbf{L}_{ij} \right]^T \{X_i - v_i, \text{ c.g.}\} \tag{2}
\end{align*}
\]
for conversion from ground to vehicle coordinate systems. The first equation is used to find hardpoint and footpad positions in the ground coordinate system and the second to convert footpad positions and torques into the vehicle coordinate system.

**Initial Orientation of the Vehicle**

Although direction cosines are used to define vehicle orientation throughout the computer program, it was felt that a simpler input form would be preferable. Accordingly, the vehicle initial orientation is specified by starting with the vehicle in a 1-2-1 position, resting on the surface and with the number one leg set in the direction of the principal slope. The vehicle is then rotated as follows:

1. A pitch rotation from \(X_F, Y_F\) (fixed coordinate system), to a \(x_1, y_1\) position, as shown below.

2. A yaw from \(x_1, y_1, z_1\), to \(x_2, y_2, z_2\)
3. A roll from $x_2, y_2, z_2$ to vehicle coordinate final positions of $x_B, y_B, z_B$.

Roll Rotation

The preceding specification of the initial position is easier to visualize than the Euler angles would be. The calculation of the initial direction cosines from the pitch, roll, and yaw angles is easily carried out. The resulting formulas appear in the computer program.

Description of Subroutine INTEQM

This subroutine basically integrates the equations for motion of a six degree of freedom (3 translational, 3 rotational) system to find the $x_K, y_K, z_K, \theta_K, \phi_K, \psi_K$ position from the previous position and the forces and torques acting during a time increment $dt$. The angular position is actually stated in terms of the direction cosines between a triad fixed in the moving system and a triad fixed in the "ground" rather than in terms of the Euler angles.

The following figure depicts the fixed coordinate system $(X_1, X_2, X_3)$ and the "vehicle" coordinate system $(x_1, x_2, x_3)$. Both are right-handed triads and unit vectors ($\mathbf{i}, \mathbf{j}, \mathbf{k}$) and ($\mathbf{\hat{i}}, \mathbf{\hat{j}}, \mathbf{\hat{k}}$) will be used to indicate direction in the vehicle and fixed systems, respectively.
For computing translation of the vehicle, the forces will be expressed in terms of the fixed coordinate system and the equations which are integrated are simply

\[ \ddot{x}_{CG} = \frac{\sum F_x}{m} \]  
\[ \ddot{y}_{CG} = \frac{\sum F_y}{m} \]  
\[ \ddot{z}_{CG} = \frac{\sum F_z}{m} \]  

(3) (4) (5)

For rotation, the equations are more complex. First, note that it is desirable to find torques and rotations in terms of the vehicle coordinate system so that the moments of inertia remain constant in the direction of the applied torques.

With this done these torques \( G_1, G_2, G_3 \) which are the torques about the \( x_1, x_2, \) and \( x_3 \) axes (positive clockwise looking at the origin) respectively, give rise to angular velocity changes according to the Euler equations

\[ \dot{\omega}_1 = \frac{G_1}{U_1} + \frac{U_2 - U_3}{U_1} (\omega_2 \omega_3) \]  
\[ \dot{\omega}_2 = \frac{1}{U_2} (G_2 + (U_3 - U_1) \omega_1 \omega_3) \]  
\[ \dot{\omega}_3 = \frac{1}{U_3} (G_3 + (U_1 - U_2) \omega_1 \omega_2) \]  

(6) (7) (8)

where \( U_1, U_2, \) and \( U_3 \) are the moments of inertia of the vehicle body about its \( x_1, x_2, \) and \( x_3 \) axes and \( \omega_1, \omega_2, \) and \( \omega_3 \) are the angular velocities around these axes (positive clockwise looking at the origin).

In Equation (6) the term \( \frac{U_2 - U_3}{U_1} (\omega_2 \omega_3) \) is zero if \( x_1 \) is an axis of symmetry. The difficulty in transforming the torques to the vehicle coordinate system and the new position resulting from the integration back into terms of the fixed coordinate system is resolved by the use of the direction cosines, \( \ell(i, j) \). Here, \( \ell(i, j) \) is the cosine of the angle between the positive \( i \)th axis of the fixed coordinate system and the positive \( j \)th axis of the vehicle coordinate system when the vehicle system is translated so that the origins of the two systems coincide.

If the \( \{\hat{i}, \hat{j}, \hat{k}\} \) triad is expressed in terms of the fixed triad \( \{\hat{I}, \hat{J}, \hat{K}\} \) with components expressed as \( i(\hat{I}), i(\hat{J}), \) etc., where \( i(\hat{i}) \) is the \( \hat{i} \) - component of \( \hat{i} \), etc., then for \( \ell(i, j) \):

\[ \ell(1,1) = i(\hat{I}) \]  
\[ \ell(2,1) = i(\hat{J}) \]  
\[ \ell(3,1) = i(\hat{K}) \]  
\[ \ell(1,2) = j(\hat{I}) \]  
\[ \ell(2,2) = j(\hat{J}) \]  
\[ \ell(3,2) = j(\hat{K}) \]  
\[ \ell(1,3) = k(\hat{I}) \]  
\[ \ell(2,3) = k(\hat{J}) \]  
\[ \ell(3,3) = k(\hat{K}) \]
In the program the motion of the vehicle body during one time step is expressed in terms of the \((i, j, k)\) triad as oriented at the beginning of the time step. Then the acceleration of any point in the vehicle body is given by:

\[
\ddot{\vec{r}} = \frac{\vec{\omega} \times (\vec{\omega} \times \vec{r})}{\text{Radial Component}} + \frac{\vec{\omega} \times \vec{r}}{\text{Tangential Component}}
\]

\(\vec{r}\) is where a vector originates at the C.G.

If we let \(\vec{r}\) be, in turn, the \(i, j, k\) vectors we can, by integrating, find the position of the \(i, j, k\) vectors at the end of the iteration interval. This is an especially convenient choice of \(\vec{r}\) because we can, from the new \((\vec{i}, \vec{j}, \vec{k})\), easily find the new \(J(i, j)\) and thus any point on the body can be located in the fixed coordinate system from its known position on the vehicle body. (For example, the hardpoints can be found quickly.)

Considering that \(\vec{r}\) is any vector expressed in terms of the vehicle coordinates, we have

\[
\vec{r}_{\text{new}} = \left\{ \left( \vec{\omega}_{\text{avg}} \times (\vec{\omega}_{\text{avg}} \times \vec{r}_{\text{old}}) + \vec{\omega} \times \vec{r}_{\text{old}} \right) \right\} \frac{(\Delta t)^2}{2} + \vec{r}_{\text{old}} \Delta t + \vec{r}_{\text{old}}
\]

The symbol \(V(m, n)\) is introduced to denote the nth component in the \((i, j, k)\) triad \((i = 1, j = 2, k = 3)\) of the new position of the mth vector where \(m = 1\) denotes the \(i\) vector, etc.

For \(V(1, n)\):

\[
\vec{r} \times (\vec{\omega} \times \vec{i}) = \vec{\omega} \times (\omega_2 j + \omega_3 k) \times \vec{i}
\]

\[
= \vec{\omega} \times (-\omega_2 k + \omega_3 j) = (\omega_1 \vec{i} + \omega_2 \vec{j} + \omega_3 \vec{k}) \times (-\omega_2 \vec{k} + \omega_3 \vec{j})
\]

\[
= \omega_1 \omega_2 j + \omega_1 \omega_3 k - (\omega_2^2 + \omega_3^2) \vec{i}
\]

Similarly, for \(V(2, n)\):

\[
\vec{r} \times (\vec{\omega} \times \vec{j}) = -\left(\omega_1^2 + \omega_3^2\right) \vec{j} + \omega_2 \omega_3 \vec{k} + \omega_1 \omega_2 \vec{i}
\]

and

for \(V(3, n)\):

\[
\vec{r} \times (\vec{\omega} \times \vec{k}) = \omega_2 \omega_3 j - (\omega_1^2 + \omega_2^2) \vec{k} + \omega_1 \omega_3 \vec{i}
\]

for \(V(1, n)\):

\[
\vec{r} \times \vec{i} = - \omega_2 \vec{k} + \omega_3 \vec{j}
\]

for \(V(2, n)\):

\[
\vec{r} \times \vec{j} = \omega_1 \vec{k} - \omega_3 \vec{i}
\]
for $V(3, n)$:
\[ \vec{\omega} \times \vec{k} = -\omega_1 \hat{j} + \omega_2 \hat{i} \]

Also for $V(1, n)$:
\[ \vec{i}_{\text{old}} = \vec{i}_{\text{old}} = -\omega_3 \hat{j} + \omega_2 \hat{k} \]

and for $V(2, n)$:
\[ \vec{j}_{\text{old}} = \omega_3 \hat{i} - \omega_1 \hat{k} \]

for $V(3, n)$:
\[ \vec{k}_{\text{old}} = -\omega_2 \hat{i} + \omega_1 \hat{j} \]

All of these quantities are combined to give the values of $\vec{r}_{\text{new}}$ (denoted by $V(m, n)$) using Equation (9).

The new $\vec{r}$ (that is, new $\vec{i}$, $\vec{j}$, and $\vec{k}$) are expressed in terms of the old $(\vec{i}, \vec{j}, \vec{k})$ triad and, thus, can be expressed in terms of the fixed coordinate system by means of the old direction cosines, $\mathbf{\ell}(i, j)$. First, it should be noted that the components of the new $\vec{i}$, $\vec{j}$, and $\vec{k}$ in terms of the fixed coordinate system are exactly the direction cosines relating the two systems when the origins are brought together by translation without rotation. Thus

\[ \mathbf{\ell}(i, j)_{\text{new}} = \sum_{k=1}^{3} \mathbf{\ell}(i, k)_{\text{old}} \cdot V(j, k) \quad (10) \]

This equation states that the direction cosine between the $i$th fixed axis and the $j$th vehicle body axis after an iteration interval, that is the new $j$th axis, is the sum of three terms which are old direction cosines and the new $i$, $j$, or $k$ vector, which is the desired result. The direction cosines relating the "new" vehicle position, the fixed axes, and the new position of the center of gravity are based on integration of $F = ma$ by the equations:

\[ X_{\text{CG}} = X_{\text{CG (prev)}} + \dot{X}_{\text{CG}} \Delta t + \frac{F_X \Delta t^2}{2 \text{ VASLUG}} \quad (11) \]

\[ Y_{\text{CG}} = Y_{\text{CG (prev)}} + \dot{Y}_{\text{CG}} \Delta t + \frac{F_Y \Delta t^2}{2 \text{ VASLUG}} \quad (12) \]

\[ Z_{\text{CG}} = Z_{\text{CG (prev)}} + \dot{Z}_{\text{CG}} \Delta t + \frac{F_Z \Delta t^2}{2 \text{ VASLUG}} \quad (13) \]

These constitute the INTEQM Subroutine Outputs.
An error check on the 3rd term in the Taylor series for the position change from the previous position is made based on the torque equations.

\[
\vec{v}_{\text{new}} = \vec{v}_{\text{old}} + \frac{\partial\vec{v}}{\partial t} \bigg|_{\text{old}} \Delta t + \frac{\partial^2 v}{\partial t^2} \bigg|_{\text{old}} \Delta t^2 + \frac{\partial^3 v}{\partial t^3} \bigg|_{\text{old}} \Delta t^3 + \cdots \quad (14)
\]

terms of the form \( \vec{f} \)

terms of the form \( \vec{\omega} \times \vec{\omega} \times \vec{r} \)

differences in \( \frac{\partial^2 v}{\partial t^2} \) from present and previous time increments divided by \( \Delta t \).

The error check was set up so that a total error of \( .1^\circ \) per second was allowed.

**Description of Subroutine FORCE**

This subroutine requires as input the positions of the vehicle hardpoints in terms of the ground (fixed) coordinate system, the footpad position, the just previous length, and previous shortest length, of all the struts. It provides, as output, the net force acting between the footpad and the surface in terms of the fixed (ground) coordinate system and the strut lengths.

For each strut a static force vs. length relationship, as shown below, is assumed. To this force is added a friction (Coulomb) force depending on the direction of the strut length change in the previous interval.
The subroutine calculates forces in only one leg set at a time. The first step in the program is the calculation of the three strut lengths and direction cosines for the leg set under consideration. Then the preceding curve is used to determine the strut forces. Finally, these forces are summed to find the vertical, horizontal, and lateral components in the fixed coordinate system.

Stability Determination

The basic idea of the instability check is simply that, if there are no footpads "in front of" the vehicle center of gravity, then the motion is unstable. "In front of" means ahead of the vehicle center of gravity in the direction of its motion. As presently written, only instability in the generally downhill direction is provided for an all-directional instability check could be provided by simply adding an uphill stability criteria essentially identical to the present downhill case described below.

The BETA value is derived as follows: (See Figure 6-3 for nomenclature.)

\[
\begin{align*}
\text{Vehicle C.G.} & \quad \text{ZCG} \\
\text{ZCG} & \quad \text{YCG} \\
\text{YCG} & \quad \text{XCG} \tan \eta \\
\end{align*}
\]

\[
\hat{V}_{CG} (\hat{Z} & \quad \hat{Y} \text{Velocity Components})
\]

View Looking Down on X Axis - Fixed (Ground) Coordinate System
B-B is a line perpendicular to the vector \( \vec{V}_{CG} \) and through the vehicle center of gravity. C-C is the projection of the line B-B onto the YZ plane in the gravity direction.

AA is defined as

\[
AA = \arctan \left( \frac{Z_{DCG}}{Y_{DCG}} \right)
\]  

Now we can write an equation for the line B-B

\[
Y = C_1 + C_2 Z
\]  
\[
C_2 = - \tan (AA)  
\]  
\[
Y_{CG} = C_1 - \tan (AA) \cdot Z_{CG}
\]  
\[
\therefore C_1 = Y_{CG} + \tan (AA) Z_{CG}
\]  
\[
Y = Y_{CG} + Z_{CG} \tan (AA) - Z \tan (AA)
\]  

Using \( AAA = \left( \frac{Z_{DCG}}{Y_{DCG}} \right) \) = \( \tan (AA) \) and rearranging, we have the Equation of B-B;

\[
Y = Y_{CG} + (Z_{CG}-Z) \cdot AAA
\]  

Then the amount by which the Y-coordinate of a footpad with coordinates VFP and ZFP exceeds the Y-coordinate of the plane through line B-B and perpendicular to plane YZ is given by

\[
Y_{FP} - Y = Y_{FP} - [ Y_{CG} + (Z_{CG} - Z_{FP}) AAA ]
\]  
\[
Y_{FP} - Y = Y_{FP} + Z_{FP} \cdot AAA - (Y_{CG} + Z_{CG} \cdot AAA)
\]

If gravity were acting perpendicular to the YZ plane, then this quantity would be the stability test. However, because the YZ plane is tilted with respect to gravity, we must make a correction.
\[
D = \left[ YFP + ZFP \cdot AAA - YCG - ZCG \cdot AAA \right] \cos (AA) \quad (24)
\]

\[
D = FPTEST \cdot \cos (AA) \quad (25)
\]

\[
BETA = \beta = \arctan \left[ \frac{D}{XCG} \right] - \arctan \left[ \tan \gamma \cos (AA) \right] \quad (26)
\]

If BETA becomes negative the calculation of vehicle motion is stopped, instability being evident.

Figure 6-4 illustrates the input data format required for use of the program. The input quantities are defined in Figure 6-14 under "input definitions."

Figures 6-4 through 6-13 are flow diagrams of the main program and its subroutines. Figures 6-14 through 6-22 are complete listings of the program and its subroutines.

The output data for a sample run is shown in Figure 6-24. The input data is printed as part of the output as shown in Figure 6-23. This is followed by "on line" printing of pertinent information. At the completion of the run, additional stored output is printed as illustrated in Figure 6-25.
<table>
<thead>
<tr>
<th>STATEMENT NUMBER</th>
<th>FORTRAN STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>XB(1, 1)</td>
</tr>
<tr>
<td>2</td>
<td>XB(2, 1)</td>
</tr>
<tr>
<td>3</td>
<td>XB(3, 1)</td>
</tr>
<tr>
<td>4</td>
<td>XB(1, 2)</td>
</tr>
<tr>
<td>5</td>
<td>XB(2, 2)</td>
</tr>
<tr>
<td>6</td>
<td>XB(3, 2)</td>
</tr>
<tr>
<td>7</td>
<td>YB(1, 1)</td>
</tr>
<tr>
<td>8</td>
<td>YB(2, 1)</td>
</tr>
<tr>
<td>9</td>
<td>YB(3, 1)</td>
</tr>
<tr>
<td>10</td>
<td>YB(1, 2)</td>
</tr>
<tr>
<td>11</td>
<td>YB(2, 2)</td>
</tr>
<tr>
<td>12</td>
<td>YB(3, 2)</td>
</tr>
<tr>
<td>13</td>
<td>ZB(1, 1)</td>
</tr>
<tr>
<td>14</td>
<td>ZB(2, 1)</td>
</tr>
<tr>
<td>15</td>
<td>ZB(3, 1)</td>
</tr>
<tr>
<td>16</td>
<td>ZB(1, 2)</td>
</tr>
<tr>
<td>17</td>
<td>ZB(2, 2)</td>
</tr>
<tr>
<td>18</td>
<td>ZB(3, 2)</td>
</tr>
<tr>
<td>19</td>
<td>VFOT1(1, 1)</td>
</tr>
<tr>
<td>20</td>
<td>VFOT1(2, 1)</td>
</tr>
<tr>
<td>21</td>
<td>VFOT1(3, 1)</td>
</tr>
<tr>
<td>22</td>
<td>VFOT1(4, 1)</td>
</tr>
<tr>
<td>23</td>
<td>VFOT1(1, 2)</td>
</tr>
<tr>
<td>24</td>
<td>VFOT1(2, 2)</td>
</tr>
<tr>
<td>25</td>
<td>VFOT1(3, 2)</td>
</tr>
<tr>
<td>26</td>
<td>VFOT1(4, 2)</td>
</tr>
<tr>
<td>27</td>
<td>VFOT1(1, 3)</td>
</tr>
<tr>
<td>28</td>
<td>VFOT1(2, 3)</td>
</tr>
<tr>
<td>29</td>
<td>VFOT1(3, 3)</td>
</tr>
<tr>
<td>30</td>
<td>VFOT1(4, 3)</td>
</tr>
</tbody>
</table>

Figure 6-4. Input Data Format
This program determines the detailed vehicle motions in three dimensions for a lunar landing vehicle having four tripod type legs as described in Reference 1.

Figure 6-5. Three Dimensional Landing Dynamics
Figure 6-5. Three Dimensional Landing Dynamics (Continued)
Are all four footpads off the ground?

- NO
  - Set iteration interval $DT = 0.0005$.

- YES
  - Set iteration interval $DT = 0.05$ sec.

Is more than one foot off the ground?

- NO
  - Set iteration interval $DT = 0.01$.

- YES
  - Will any footpad strike surface during next $DT$ time interval?
    - NO
      - Adjust $DT$ to occur at exact print interval.
    - YES
      - Set iteration interval $DT = 0.002$ sec.

Figure 6-5. Three Dimensional Landing Dynamics (Continued)
Call "INTEQM"
This subroutine integrates Newton's equations to determine vehicle motion.

Index time to
$T = T + DT$

Is vehicle C.G. below surface of ground?

Is vehicle velocity in X-Y plane zero or negative?

Return to Program start and read new input data on terminate run series.

Has time $T$ reached maximum time?

Print time $PSIMAX$ and $BTAMIN$

Print stored output

Return to 98 for next iteration interval

Figure 6-5. Three Dimensional Landing Dynamics (Concluded)
This subroutine reads the input data cards and prints this data as part of the output record.

Figure 6-6. Subroutine - INPUT
This subroutine initializes the LAND 3D program and sets program constants.

[Diagram]

Figure 6-7. Subroutine - INIT
This subroutine sets crush stroke loads and springrates for the struts.

**Figure 6-8. Subroutine - LEGTYP**
This subroutine defines the stability angle BETA.

STAB

Determine vehicle orientation and angular relationships.

Determine stability angle BETA

Is PSI greater than has occurred previously during the run.

PSIMAX = PSI

Test BETA is vehicle stable?

Return to LAND 3D at 1102

RETURN

Return to LAND 3D at 127

RETURN

Figure 6-9. Subroutine - STAB
This subroutine stores output data for printing at the end of the run.

Figure 6-10. Subroutine - DATA
This subroutine determines the vehicle geometry in space.

**CONFIG**

- **Sliding**
  - Test footpad motion for each individual footpad (I)
  - Print leg number
  - This branch provides for a sliding mode if friction is not infinite. Not used at this time

- **off the ground**
  - stationary
    - Does footpad (I) hit surface during this interval DT
      - Yes: Set flags to indicate this information for output printing
      - No: Will footpad (I) strike surface during next interval
        - Yes: fix footpad to surface
        - No: Determine XH, YH, and ZH - the X, Y, and Z components of strut lengths for all 3 struts of all four legs

**Figure 6-11. Subroutine - CONFIG**
pick up strut lengths SSTRT and SPR by equivalence to common storage

Call FORCE
This subroutine determines leg strut forces as a function of strut length

No

Yes

Does the footpad (I) lift off the surface during this time interval DT

Reset strut length since it is stroking

Determines vector position of Ith footpad

find total forces and torques acting on vehicle C. G.

Translate torques to vehicle coordinate system

RETURN

Figure 6-11. Subroutine - CONFIG (Concluded)
This subroutine determines strut stroke forces as a function of strut length.

**FORCE**

find lengths and direction cosines (in fixed coordinates system) of the three struts of leg set "KRG"

Call LEGTYP

This subroutine sets plastic strut stroke forces and springrates for upper (primary) and lower struts

is strut (I) in tension?

No

10

Yes

Is tension load greater than 10,000 lbs?

No

Set tension load equal to 10,000 lbs.

Yes

2

Figure 6-12. Subroutine - FORCE
set "ratchet" so strut may not reextend and "uncrush" the honeycomb again

Determine frictional effects

Determine the sum of the forces acting on the vehicle in the X, Y, and Z directions (QFSUM, VFSUM and WFSUM)

RETURN
This subroutine integrates the equations of motion for the vehicle.

**Figure 6-13. Subroutine - INTEQM**
INTEQM

Figure 6-13. Subroutine - INTEQM (Concluded)

BENDIX PRODUCTS AEROSPACE DIVISION
Bendix

I FOR LAND3D

TITLE LAND3D THREE DIMENSIONAL LANDING DYNAMICS COMPUTER PROGRAM

AUTHOR DR. R. DIX THE BENDIX CORPORATION

DATE NOVEMBER, 1964

PURPOSE THIS PROGRAM DETERMINES THE DETAILED VEHICLE MOTIONS IN THREE DIMENSIONS FOR A LUNAR LANDING VEHICLE HAVING FOUR TRIPODAL TYPE LEGS AS ILLUSTRATED IN BENDIX REPORT NO. MM 64 - 9

METHOD THE PROGRAM DETERMINES LEG STRUT FORCES AS A FUNCTION OF STRUT LENGTH. THESE FORCES PLUS GRAVITY DETERMINE VEHICLE MOTIONS BY INTEGRATION USING VARIABLE TIME INCREMENTS DETERMINED BY ALLOWABLE ERROR TESTS

NOTES PROGRAM WRITTEN IN FORTRAN IV
EXECUTES ON UNIVAC 1107 COMPUTER

THIS PROGRAM ASSUMES ZERO FOOTPAD MASS AND AN INFINITE GROUND COEFFICIENT OF FRICTION

THE PROGRAM WILL NOT OPERATE IF VVEL AND HVEL ARE INITIALLY ZERO SIMULTANEOUSLY

THIS PROGRAM WILL NOT OPERATE FOR UPHILL LANDINGS

THIS PROGRAM HAS BEEN CHECKED OUT TO A LIMITED EXTENT ONLY. FURTHER CHECK OUT AND CORRELATION WITH A PHYSICAL DROP TEST MODEL ARE CURRENTLY IN PROGRESS. AS THE RESULT OF THIS LIMITED CHECKOUT, THE RESULTS OF THIS PROGRAM SHOULD BE USED WITH CAUTION

NOTE ALL UNITS ARE FEET, SECONDS, RADIANS AND SLUGS

(M) DENOTES COORDINATE SYSTEM MOVING WITH THE VEHICLE
(F) DENOTES COORDINATE SYSTEM FIXED IN SPACE

APPEARANCE OF 1 AFTER ANY VARIABLE INDICATES THE INITIAL VALUE OF THAT VARIABLE.
ABSENCE OF THE 1 INDICATES THE INSTANTANEOUS VALUE.

SUBROUTINES USED

INPUT - READS INPUT CARDS AND PRINTS DATA AS PART OF THE OUTPUT RECORD

INIT - INITIALIZES LAND3D AND SETS PROGRAM CONSTANTS

STAR - DETERMINES STABILITY ANGLE BETA FOR THREE DIMENSIONAL VEHICLE

DATA - STORES OUTPUT DATA FOR PRINTING AT THE END OF THE RUN

CONFIG - DETERMINES VEHICLE ORIENTATION IN THREE

Figure 6-14. Main Program
Bendix

DIMENSIONS

LEGTYP - SETS SPRING RATES AND PLASTIC STRUT FORCE
MAGNITUDE FOR ALL LEG STRUTS

FORCE - DETERMINES LEG STRUT FORCES AS FUNCTION OF STRUT
LENGTH AND SPRINGRATES, ETC.

INTFQM - INTEGRATES EQUATIONS OF MOTION OF VEHICLE

INPUT INPUT BY PUNCH CARD
INPUT FORMAT IS ILLUSTRATED IN BENDIX REPORT MM 64 - 9

INPUT DEFINITIONS

ETA1 (OR ETA) ANGLE OF SLOPE IN PRINCIPLE DIRECTION

FOTAC(I) (OR FOTAC(I)) MODE OF ACTION OF THE I TH FOOTPAD
I = 1 CORRESPONDS TO SLIDING ON THE SURFACE
I = 2 CORRESPONDS TO THE FOOTPAD STATIONARY ON THE SURFACE
I = 3 CORRESPONDS TO THE FOOTPAD OFF THE SURFACE

NOTE FOTAC(I) ARE PROGRAM INPUT CONSTANTS. SET THEM EQUAL
TO 3.0 FOR ALL RUNS

PITCH1 (OR PITCH) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9

YAW1 (OR YAW) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9

ROLL1 (OR ROLL) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9

HVEL HORIZONTAL VELOCITY OF THE VEHICLE C.G. (NORMAL TO
THE GRAVITY VECTOR) AND IN THE DIRECTION OF THE PRINCIPLE
SLOPE (IN THE Y DIRECTION IN (F) COORDINATE SYSTEM)

U1 MASS MOMENT OF INERTIA OF VEHICLE ABOUT X AXIS IN (M)
COORDINATE SYSTEM

U2 MASS MOMENT OF INERTIA OF VEHICLE ABOUT Y AXIS IN (M)
COORDINATE SYSTEM

U3 MASS MOMENT OF INERTIA OF VEHICLE ABOUT Z AXIS IN (M)
COORDINATE SYSTEM

VSLUG VEHICLE MASS

VVEL VERTICAL VELOCITY OF THE VEHICLE C.G. (PARALLEL TO THE
GRAVITY VECTOR)

VFOT(I,J) (OR VFOT(I,J)) I TH COMPONENT OF THE VECTOR POSITION
OF THE J TH FOOTPAD IN (M) COORDINATE SYSTEM
I = 1 IS X COMPONENT
I = 2 IS THE Y COMPONENT
I = 3 IS THE Z COMPONENT

WIE1 (OR WIE) ANGULAR VELOCITY OF VEHICLE ABOUT THE X AXIS IN THE
(M) COORDINATE SYSTEM

W2E1 (OR W2E) ANGULAR VELOCITY OF VEHICLE ABOUT THE Y AXIS IN THE
(M) COORDINATE SYSTEM

W3E1 (OR W3E) ANGULAR VELOCITY OF VEHICLE ABOUT THE Z AXIS IN THE
(M) COORDINATE SYSTEM

XBI(2,4) X COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

YBI(2,4) Y COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

ZBI(2,4) Z COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

XCG X POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

Figure 6-14. Main Program (Continued)
YCG  Y POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM
ZCG  Z POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM
ZDCG (GR ZDCG) HORIZONTAL VELOCITY OF VEHICLE C.G. (NORMAL TO
GR AVITY VECTOR) NORMAL TO THE PRINCIPLE DIRECTION OF
THE SLOPE ( IN Z DIRECTION IN (F) COORDINATE SYSTEM)

OUTPUT  PRINTED OUTPUT

OUTPUT DEFINITIONS

BETA  STABILITY ANGLE  SEE TEXT BENDIX REPORT MM-64-9
BETADT  RATE OF CHANGE OF THE STABILITY ANGLE BETA
BETAP  PREVIOUS VALUE OF BETA AT LAST ITERATION INTERVAL
BETAMIN  MINIMUM BETA ANGLE REACHED DURING THE COMPUTER RUN

CRUSH  OUTPUT INDICATOR  - TWELVE DIGITS, ONE FOR EACH STRUT, REPRESENTING THE FORCE CAUSING MECHANISM IN EACH STRUT
ORDER OF OUTPUT  1 2 3 4
STRUT  1 2 3 1 2 3 1 2 3
THE DIGITAL CODE IS AS FOLLOWS
0  NO FORCE BECAUSE FOOTPAD IS OFF SURFACE
1  ELASTIC COMPRESSION IN STRUT
2  PLASTIC COMPRESSION IN STRUT
3  STRUT RE-EXTENDED FROM PREVIOUSLY CRUSHED
   POSITION BUT IS STILL SHORTER THAN ORIGINAL
   LENGTH
8  STRUT STRETCHED BEYOND ORIGINAL LENGTH BUT
   LESS THAN 10000 LBS FORCE
9  STRUT IN TENSION MORE THAN 10000 LBS

DT  ITERATION TIME INTERVAL
E  OUTPUT INDICATOR  E=0 IF LEG IS NOT COMRESSING OR THE
   FORCE IN LEG IS LESS THAN 10000 LBS  E=1 IF AT LEAST
   ONE LEG IS IN TENSION GREATER THAN 10000 LBS

IE  SAME AS E
FC  STRUT PLASTIC STROKE FORCE
FSUMX  SAME AS FXP
FSUMY  SAME AS FYP
FSUMZ  SAME AS FZP
FYGRAV  GRAVITY FORCE IN Y DIRECTION IN (F) COORDINATE SYSTEM
FXP  FORCE IN THE X DIRECTION ON THE FOOTPAD
FYP  FORCE IN THE Y DIRECTION ON THE FOOTPAD
FZP  FORCE IN THE Z DIRECTION ON THE FOOTPAD
FORCEx  NET FORCE ACTING ON THE VEHICLE C.G. IN THE X DIRECTION
   IN THE (F) COORDINATE SYSTEM
FORCEY  NET FORCE ACTING ON THE VEHICLE C.G. IN THE Y DIRECTION
   IN THE (F) COORDINATE SYSTEM
FORCEZ  NET FORCE ACTING ON THE VEHICLE C.G. IN THE Z DIRECTION
   IN THE (F) COORDINATE SYSTEM
IFORCX  SAME AS FORCEX
IFORCY  SAME AS FORCEY
IFORCZ  SAME AS FORCEZ
TORQX  COMPONENT OF TORQUE ACTING ON THE VEHICLE C.G. IN THE
   X-Y PLANE
ITORG  SAME AS TORQX
FX1  GROUND REACTION FORCE PERPENDICULAR TO GROUND FOR FOOTPAD
   NO. 1
FX2  GROUND REACTION FORC1 PERPENDICULAR TO GROUND FOR FOOTPAD
   NO. 2
FX3  GROUND REACTION FORCE PERPENDICULAR TO GROUND FOR FOOTPAD

Figure 6-14. Main Program (Continued)
Figure 6-14. Main Program (Continued)

FX4
NO. 3
GROUND REACTION FORCE PERPENDICULAR TO GROUND FOR FOOTPAD

NO. 4
IX1
SAME AS FX1
IX2
SAME AS FX2
IX3
SAME AS FX3
IX4
SAME AS FX4

GSUM
SUM OF THE TORQUES. USED TO CONTROL DT (USED TO FIND IF
THE TORQUE IS CHANGING. IF THE DIFFERENCE BETWEEN GSUM
AND GSUMP IS LARGE, TAKE SMALL DT TIME INCREMENTS.
IF THE DIFFERENCE IS SMALL, TAKE LARGE DT TIME INCREMENTS.

GSUMP
PREVIOUS VALUE OF GSUM

GND
INDICATES IF ANY FOOTPAD IS ON THE GROUND
GND = 1 -- ONE OR MORE FEET ARE ON THE GROUND
GND = 0 --- ALL FEET ARE OFF THE GROUND

IFLAG
INDICATOR FOR CONDITIONAL RETURN TO MAIN PROGRAM
ITORE
STORAGE INDEX FOR FINAL OUTPUT PRINTING
LINE
LINE COUNT FOR OUTPUT PAGE ORDERING
LCOUNT
MAXIMUM ALLOWABLE LINES PER PAGE OF OUTPUT DATA

INMCN
DIGITAL REPRESENTATION OF FOTAC FOR FOOTPADS 1 -- 4
RESPECTIVELY (SEE INPUT DEFINITIONS FOR FOTAC)

PHI
ANGULAR ORIENTATION OF VEHICLE IN (F) COORDINATE SYSTEM
SEE TEXT BENDIX REPORT MM-64-9

PSI
ANGULAR ORIENTATION OF VEHICLE IN (F) COORDINATE SYSTEM
SEE TEXT BENDIX REPORT MM-64-9

THETA
ANGULAR ORIENTATION OF VEHICLE IN (F) COORDINATE SYSTEM
SEE TEXT BENDIX REPORT MM-64-9

PSIDOT
COMPONENT OF VEHICLE ANGULAR VELOCITY IN X-Y PLANE OF
(F) COORDINATE SYSTEM

PX1
HEIGHT OF FOOTPAD 1 ABOVE THE SURFACE AND NORMAL TO IT

PX2
HEIGHT OF FOOTPAD 2 ABOVE THE SURFACE AND NORMAL TO IT

PX3
HEIGHT OF FOOTPAD 3 ABOVE THE SURFACE AND NORMAL TO IT

PX4
HEIGHT OF FOOTPAD 4 ABOVE THE SURFACE AND NORMAL TO IT

PHIDOT
COMPONENT OF VEHICLE ANGULAR VELOCITY IN X-Z PLANE OF
(F) COORDINATE SYSTEM

PSIMAX
MAXIMUM PSI ANGLE REACHED DURING THE ENTIRE COMPUTER RUN

PRNI
PRINT TIME (0.000004, 0.000008, ETC)

SO
STRUT LENGTH

SSTRUT(I,J)
STRUT LENGTH OF THE J TH STRUT OF THE I TH LEG SET

T
SPRING RATE OF THE UPPER STRUT

T2
SPRING RATE OF THE LOWER STRUTS

THEOT
COMPONENT OF VEHICLE ANGULAR VELOCITY IN Y-Z PLANE OF
(F) COORDINATE SYSTEM

SCOS
DIRECTION COSINES RELATING FIXED (F) AND MOVING (M)
COORDINATE SYSTEMS

SPRR(I,J)
PREVIOUS SHORTEST LENGTH OF STRUT I OF THE J TH LEG SET

SNTRVL
PRINT FREQUENCY (EVERY 0.04 SEC.)

TIME
TIME AFTER TOUCHDOWN

TYPE(I)
TYPE OF STRUT -- STRUT NO. 1 IS TYPE 1
STRUTS NO. 2 AND 3 ARE TYPE 2

VFL
VELOCITY OF VEHICLE C.G. IN X-Y PLANE OF (F)
COORDINATE SYSTEM

XFIF(I)
POSITION OF THE FOOTPAD I IN THE J TH (X,Y,Z) DIRECTION
IN THE FIXED COORDINATE SYSTEM

XCG1 (OR XCG)
X POSITION OF VEHICLE C.G. IN (F) COORDINATE SYSTEM

YCG1 (OR YCG)
Y POSITION OF VEHICLE C.G. IN (F) COORDINATE SYSTEM

ZCG1 (OR ZCG)
Z POSITION OF VEHICLE C.G. IN (F) COORDINATE SYSTEM

XSFCTP
PARTICULAR VALUE OF XFIF(I,J)
DIMENSION XXFX(4)
DIMENSION CG(3)
COMMON/LS1/BETAP,BETADT,BETAMN,CRUSH(12),DT,E,ETA,FORCEZ,FOTAC(4),
1FPI(4),FORCEX,FORCEY,FC,FSUMZ,FSUMX,FSUMY,FYGRAV,FXGRAV,G1,G2,
3G3,GN,DNTF,1,ISTMAX,SCOS(3,3),SPR(4,3),8STRUT(4,3),
350,S01,S02,TIME,TYPE(3),TurK2,T12,U1,U2,U3,VASLUG,VFOT(4,3)*1E,
4*2FWE*XCG,XDCG,X(3),VFOT(4,3),XXFX(4),XYFOT(4),Y(3),YB(3,4),
5YCG,YDCG, Z13,Z23(3,4),ZCG,ZDCG
COMMON/LS2/BETADT(500),1CRSH1(500),1CRSH2(500),1CRSH3(500),
11CRSH4(500),1E,IFORCX,ITCR,1X1(1500),1X2(500),1X3(500),1X4(500),
20MONDON(500)+PH1+PH1DOR(500),PS1DOR(500),PX1,PX2,PX3,PX4,TMETA,
3THERD(500),TIME2(500),XBFIT(500),XBFIT2(500),XBFIT3(500),
4XBFT4(500),IFORCY
COMMON/LS3/ETA1,FOTAC1(4),HVEL,PITCH1,ROLL1,VFOT1(4,3),VVEL,
1W1EI,W2EI,3Z13,ZCG, ZCG1,ZCGG1
COMMON/LS4/GSUM,LCOUNT,LINE_PR2,SNTRVL
COMMON/LS6/BETA,IFLAG,PSI
EQUIVALENCE (VASLUG,VASS),(XH(1),X(1)),(YH(1),Y(1)),
1Z1H(1),Z(1)),(FSUMY,FSUM),(FSUMZ,VSUM),(FSUMX,WFSUM)

999 CALL INPUT
993 CALL INIT
140 FORMAT(19H) TIME XCG YCG ZCG PHI PSI THETA XDC
6 YDCG ZDCG FORCEX FORCEY TORO Z BETA PX1 PX2 PX3 PX4 E)
15 THIS STATEMENT PUTS THE LOWEST FOOT DOWN ON THE SURFACE
DO 105 J=1,4
105 XFX(J)=SCOS(1,1)*VFOT(J,1)+SCOS(1,2)*VFOT(J,2)+SCOS(1,3)*VFOT
1(J,1)+XCG
XCG=XCG-AMJN1*XXFX(1) *XXFX(2) *XXFX(3) *XXFX(4)=.001
CG(1)=XCG
CG(2)=YCG
CG(3)=ZCG
DO 106 J=1,4
DO 106 I=1,3
106 XFIF(J,1)=SCOS(1,1)*VFOT(J,1)+SCOS(1,2)*VFOT(J,2)

Figure 6-14. Main Program (Continued)

BENDIX PRODUCTS AEROSPACE DIVISION
THE PRIMARY INTEGRATION DO LOOP STARTS HERE
THE PROGRAM RETURNS HERE AT THE START OF EACH INTEGRATION TIME STEP

DO 223 I=1,4
UPDATE VFOT
FOTAL=FOTAC(I)
221 DO 222 J=1,3
222 VFOT(I,J)=SCOS(I,J)*(XFIF(I,I)-XCG)+SCOS(I,J)*(XFIF(I,2)-YCG)+
1 SCOS(I,J)*(XFIF(I,3)-ZCG)
223 CONTINUE
CALL STAB
GO TO (1102,127),IFLAG
IS THIS A PRINT INTERVAL
127 IF(PRNI-TIME)121,122,121
122 PRNI=PRNI+SNTRVL
132 CONTINUE
CALL DATA
141 FORMAT(F6.2,3F8.3,2F6.2,F7.2,3F7.2,3I8,F7.3,4F4.1,12)
PRINT 141,TIME,XCG,YCG,ZCG,PHI,PSI,THETA,XDCG,YDCG,ZDCG,
IFORC,IFORCY,ITOR,ITORF,IXI,IX2,IX3,IX4,IE
ITORE=ITORE+1
IF(LINE=LCOUNT)1003,1004,1004
1003 LINE=LINE+1
GO TO 121
1004 PRINT 950
PRINT 140
LINE=1
121 CONTINUE
E=0.0
CALL CONFIG
GND=0 IF ALL FEET OFF GROUND
33 DT=.0005
GNDF=0.0
GSUM=G1+G2+G3
IF(ABS((GSUM-GSUMP+1)/(GSUM+.1))-.1) 86,87,87
86 DT=.01
87 CONTINUE
GSUMP=GSUM
GO TO 34
32 DT=.05
GNDF=0 UNLESS A FOOT WILL STRIKE IN NEXT INTERVAL
IF (GNDF-.5) 88,89,89
89 DT=.002
88 CONTINUE
34 CONTINUE
GND=0.0
CAUSE CALCULATION TO OCCUR EXACTLY AT THE PRINT INTERVAL
IF (TIME+DT-PRNI) 111,111,112
112 DT=PRNI-TIME
111 CONTINUE
CALL INTFQM
TIME=TIME+DT
C_USE CALCULATION TO OCCUR EXACTLY AT THE PRINT INTERVAL
IE(TIME+DT-PRNI)
III,iii,i12
DT=PRNI-TIME
112
III CONTINUE
CALL INTFQM
TIME=TIME+DT
CHECK TO BE SURE MOTIONS ARE MEANINGFUL
IF (XCG) 97,97,96
97 STOP
CHECK TO BE SURE VEHICLE C.G. IS MOVING
96 VEL=SQRT(XDCG*XDCG+YDCG*YDCG)
IF (VEL) 95,95,94
95 PRINT 951,VEL
951 FORMAT(11H VEL =,F10.5)
GO TO 995
TEST TO SEE IF PROBLEM TIME EXCEEDS MAX. ALLOWABLE TIME
94 IF (TIME-2.0) 98,98,92
97 PRINT 952,TIME
952 FORMAT(10H TIME =,F10.5)
95 PRINT 953,PIMAX,BTAMIN,TIME
953 FORMAT(10H PSIMAX =,F10.5,10H BTAMIN =,F10.5,10H TIME =,F10.5)
1102 PRINT 950
IF RUN IS OVER, PRINT THE FINAL SUMMARY PRINT
PRINT 146
LINE=1
ITORE=ITORE-1
DO 1000 N=1,ITORE
PRINT 145,TIME2(N),IX1(N),IX2(N),IX3(N),IX4(N),XBFT1(N),XBFT2(N),
1XBFT3(N),XBFT4(N),ONMOON(N),THEDT(N),PHIDOT(N),BETADT(N),
2PSIDOT(N),ICRSH1(N),ICRSH2(N),ICRSH3(N),ICRSH4(N)
IF (LINE-LCOUNT) 1001,1002,1002
LINE=LINE+1
1001 LINE=LINE+1
GO TO 1000
1002 PRINT 950
PRINT 146
LINE=1
1000 CONTINUE
PRINT 950
GO TO 999
145 FORMAT(F6,2,417,SF7.2,2F8.3,F9.3,F8.3,15,3I4)
146 FORMAT(114H TIME FX1 FX2 FX3 FX4 XBFT1 XBFT2 XBFT3
1 XBFT4 ONMOON THEDT PHIDOT BETADT PSIDOT CRUSH)
END

Figure 6-14. Main Program (Concluded)
TITLE INPUT SUBROUTINE INPUT

AUTHOR J.C. GIBSON BENDIX CORPORATION

DATE NOVEMBER 1, 1964

PURPOSE THE SUBROUTINE READS THE INPUT DATA CARDS AND PRINTS THE INPUT DATA AS PART OF THE OUTPUT RECORD

CALL CALL INPUT

NOTE ALL UNITS ARE FEET, SECONDS, RADIANS AND SLUGS
(M) DENOTES COORDINATE SYSTEM MOVING WITH THE VEHICLE
(F) DENOTES COORDINATE SYSTEM FIXED IN SPACE

APPEARANCE OF 1 AFTER ANY VARIABLE INDICATES THE INITIAL VALUE OF THAT VARIABLE.
ABSENCE OF THE 1 INDICATES THE INSTANTANEOUS VALUE.

INPUT PUNCHED CARDS

INPUT DEFINITIONS

ETA1 (OR ETA) ANGLE OF SLOPE IN PRINCIPLE DIRECTION
FOTAC1(I) OR (FOTAC(1)) MODE OF ACTION OF THE I TH FOOTPAD
I = 1 CORRESPONDS TO SLIDING ON THE SURFACE
I = 2 CORRESPONDS TO THE FOOTPAD STATIONARY ON THE SURFACE
I = 3 CORRESPONDS TO THE FOOTPAD OFF THE SURFACE
NOTE FOTAC1(I) ARE PROGRAM INPUT CONSTANTS. SET THEM EQUAL TO 3.0 FOR ALL RUNS

PITCH1 (OR PITCH) INITIAL VEHICLE ORIENTATION - SEE BENDIX REPORT MM-64-9

YAW1 (OR YAW) INITIAL VEHICLE ORIENTATION - SEE BENDIX REPORT MM-64-9

ROLL1 (OR ROLL) INITIAL VEHICLE ORIENTATION - SEE BENDIX REPORT MM-64-9

H VEL HORIZONTAL VELOCITY OF THE VEHICLE C.G. (NORMAL TO THE GRAVITY VECTOR) AND IN THE DIRECTION OF THE PRINCIPLE SLOPE. (IN THE Y DIRECTION IN (F) COORDINATE SYSTEM)

U1 MASS MOMENT OF INERTIA OF VEHICLE ABOUT X AXIS IN (M) COORDINATE SYSTEM

U2 MASS MOMENT OF INERTIA OF VEHICLE ABOUT Y AXIS IN (M) COORDINATE SYSTEM

U3 MASS MOMENT OF INERTIA OF VEHICLE ABOUT Z AXIS IN (M) COORDINATE SYSTEM

VASLUG VEHICLE MASS

VVEL VERTICAL VELOCITY OF THE VEHICLE C.G. (PARALLEL TO THE GRAVITY VECTOR)

VFOT1(I,J) (OR VFOT(1,J)) I TH COMPONENT OF THE VECTOR POSITION OF THE J TH FOOTPAD IN (M) COORDINATE SYSTEM
I = 1 IS X COMPONENT
I = 2 IS THE Y COMPONENT
I = 3 IS THE Z COMPONENT

WIE1 (OR WIE) ANGULAR VELOCITY OF VEHICLE ABOUT THE X AXIS IN THE (M) COORDINATE SYSTEM

W2E1 (OR W2E) ANGULAR VELOCITY OF VEHICLE ABOUT THE Y AXIS IN THE

Figure 6-15. Subroutine - INPUT
(M) COORDINATE SYSTEM

W3E1 (OR W2E1) ANGULAR VELOCITY OF VEHICLE ABOUT THE Z AXIS IN THE
(V) COORDINATE SYSTEM

W1E1 (OR W1E1) ANGULAR VELOCITY OF VEHICLE ABOUT THE X AXIS IN THE
(V) COORDINATE SYSTEM

W2E1 (OR W2E1) ANGULAR VELOCITY OF VEHICLE ABOUT THE Y AXIS IN THE
(V) COORDINATE SYSTEM

W3E1 (OR W3E1) ANGULAR VELOCITY OF VEHICLE ABOUT THE Z AXIS IN THE
(V) COORDINATE SYSTEM

X^2 (2,4) X COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

Y^2 (2,4) Y COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

Z^2 (2,4) Z COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET
BODY ATTACH POINT (HARDPOINT)

XCG X POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

YCG Y POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

ZCG Z POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

ZDCG1 (OR ZDCG) HORIZONTAL VELOCITY OF VEHICLE C.G. (NORMAL TO
GRAVITY VECTOR) NORMAL TO THE PRINCIPLE DIRECTION OF
THE SLOPE ( IN Z DIRECTION IN (F) COORDINATE SYSTEM)

OUTPUT

1. PRINTED OUTPUT

2. EQUIVALENCE TO COMMON STORAGE

SUBROUTINE INPUT

COMMON/LSI/IETAP, RETDT, BTAMIN, CRUSH(12), DT, ETA, FORCEZ, FOTACI(4),
1, FXPI(4), FORCEx, OREY, FC, SMARTZ, FSUMy, FSUMX, FXGRAV, FYGRAV, G1, G2,
G3, GND, GDIF, ISTORE, J, PSIMPAX, SCOS(3, 3), SPRI(4, 3), STRUT(4, 3),
3, S0, S02, TIME, TYPFR(3), TORGZ, TORGX = u, U2, U3, VASLUG, VFOT(4, 3), W1E1,
W2E1, W3E1, XCG, XDCG, X(13), X(3, 4), X(3, 5), XF1(4, 3), XSFUTP(4, 3), Y(3), YB1(3, 4),
5, YCG, YDCG, Z(13), ZB(3, 4), ZCG, ZDCG

COMMON/LSI/EFA1, FOTACI(4), HVEL, PITCH1, ROLL1, VFOTI(4, 3), VVEL,
1, We1, W2E1, W3E1, XCG1, YAW1, YCG1, ZCG1, ZDCG1

READ INPUT CARDS BY FORMAT SHOWN IN BENDIX REPORT MM-64-9

1 READ 100, ((XB(II,JJ), II=1,3), JJ=1,4), ((YB(II,JJ), II=1,3), JJ=1,4),
1, ((ZB(II,JJ), II=1,3), JJ=1,4), (VFOTI(II,JJ), II=1,4), (VFOTI(II,JJ), II=1,3),
1 READ 100, (FOTACI(II), II=1,4), VVEL, HVEL

READ 100, UI, U2, U3, XCG1, YCG1, ZCG1, W1E1, W2E1, W3E1, PITCH1, YAW1,
1, ROLL1, (TYPFII), II=1,3), VASLUG, ZDCG1, ETA1

PRINT INPUT DATA AS PART OF OUTPUT RECORD

PRINT 200, (XR(II,JJ), II=1,3), JJ=1,4)
PRINT 201, (XB(II,JJ), II=1,3), JJ=1,4)
PRINT 202, (YB(II,JJ), II=1,3), JJ=1,4)
PRINT 203, (YB(II,JJ), II=1,3), JJ=1,4)
PRINT 204, (ZB(II,JJ), II=1,3), JJ=1,4)
PRINT 205, (ZB(II,JJ), II=1,3), JJ=1,4)
PRINT 206, (VFOTI(II,JJ), II=1,4), (VFOTI(II,JJ), II=1,4)
PRINT 207, (VFOTI(II,JJ), II=1,4)
PRINT 208, (FOTACI(II), II=1,4), VVEL, HVEL
PRINT 209, UI, U2, U3, XCG1, YCG1, ZCG1
PRINT 201, (TYPFII), II=1,3), VASLUG, ZDCG1, ETA1

100 FORMAT(10X,6F10.5)

200 FORMAT(10X,6F10.5)

Figure 6-15. Subroutine - INPUT (Continued)
Figure 6-15. Subroutine - INPUT (Concluded)
INIT SUBROUTINE INITIALIZER

AUTHOR J.C. GIPSON  BENDIX PRODUCTS AEROSPACE DIVISION

DATE NOVEMBER, 1964

PURPOSE THIS SUBROUTINE INITIALIZES THE CONSTANTS FOR THE LAND3D
PROGRAM

METHOD THE SUBROUTINE COMPUTES PROGRAM PARAMETERS FROM
GEOMETRIC RELATIONSHIPS

CALL CALL INIT

NOTES THIS PROGRAM WAS WRITTEN IN FORTRAN IV
THIS PROGRAM EXECUTES ON UNIVAC 1107 COMPUTER

NOTE ALL UNITS ARE FEET, SECONDS, RADIANS AND SLUGS
(N) DENOTES COORDINATE SYSTEM MOVING WITH THE VEHICLE
(F) DENOTES COORDINATE SYSTEM FIXED IN SPACE

APPEARANCE OF 1 AFTER ANY VARIABLE INDICATES THE
INITIAL VALUE OF THAT VARIABLE.
ABSENCE OF THE 1 INDICATES THE INSTANTANEOUS VALUE.

SUBROUTINES USED

LEGTRIP SETS SPRING RATES AND STRUT FORCES FOR ALL STRUTS

INPUT DEFINITIONS

ETA1 (OR ETA) ANGLE OF SLOPE IN PRINCIPLE DIRECTION
FC STRUT PLASTIC STROKE FORCE
FOTAC1(I) OR (FOTAC(II)) MODE OF ACTION OF THE I TH FOOTPAD
  I = 1 CORRESPONDS TO SLIDING ON THE SURFACE
  I = 2 CORRESPONDS TO THE FOOTPAD STATIONARY ON THE SURFACE
  I = 3 CORRESPONDS TO THE FOOTPAD OFF THE SURFACE
NOTE FOTAC1(I) ARE PROGRAM INPUT CONSTANTS. SET THEM EQUAL
TO 3.0 FOR ALL RUNS
H VEL HORIZONTAL VELOCITY OF THE VEHICLE C.G. (NORMAL TO
THE GRAVITY VECTOR) AND IN THE DIRECTION OF THE PRINCIPLE
SLOPE. (IN THE Y DIRECTION IN (F) COORDINATE SYSTEM)
PITCH1 (OR PITCH) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9
T SPRING RATE OF THE UPPER STRUT
T2 SPRING RATE OF THE LOWER STRUTS
YAW1 (OR YAW) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9
ROLL1 (OR ROLL) INITIAL VEHICLE ORIENTATION - SEE BENDIX
REPORT MM-64-9
TYPE(I) TYPE OF STRUT - STRUT NO. 1 IS TYPE 1
  STRUTS NOS 2 AND 3 ARE TYPE 2
VASLUG VEHICLE MASS
VVEL VERTICAL VELOCITY OF THE VEHICLE C.G. (PARALLEL TO THE
GRAVITY VECTOR)

Figure 6-16. Subroutine - INIT
VFCT(I,J) (OR VFOT(I,J)) I TH COMPONENT OF THE VECTOR POSITION OF THE J TH FOOTPAD IN (M) COORDINATE SYSTEM
I = 1 IS X COMPONENT
I = 2 IS THE Y COMPONENT
I = 3 IS THE Z COMPONENT

W1E1 (OR W1E) ANGULAR VELOCITY OF VEHICLE ABOUT THE X AXIS IN THE (M) COORDINATE SYSTEM
W2E1 (OR W2E) ANGULAR VELOCITY OF VEHICLE ABOUT THE Y AXIS IN THE (M) COORDINATE SYSTEM
W3E1 (OR W3E) ANGULAR VELOCITY OF VEHICLE ABOUT THE Z AXIS IN THE (M) COORDINATE SYSTEM

XCG X POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM
YCG Y POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM
ZCG Z POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM
XH12(4) X COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET BODY ATTACH POINT (HARDPOINT)
YH12(4) Y COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET BODY ATTACH POINT (HARDPOINT)
ZH12(4) Z COORDINATE (M) OF THE I TH STRUT OF THE J TH LEG SET BODY ATTACH POINT (HARDPOINT)

OUTPUT BY EQUIVALENCE TO COMMON STORAGE

OUTPUT DEFINITIONS

BETA STABILITY ANGLE SEE TEXT BENDIX REPORT MM-64-9
BETAP PREVIOUS VALUE OF BETA AT LAST ITERATION INTERVAL
BETAMIN MINIMUM BETA ANGLE REACHED DURING THE COMPUTER RUN

CRUSH OUTPUT INDICATOR - TWELVE DIGITS, ONE FOR EACH STRUT, REPRESENTING THE FORCE CAUSING MECHANISM IN EACH STRUT
ORDER OF OUTPUT 1 2 3 1 2 3 1 2 3 1 2 3
STRUT 1 2 3 1 2 3 1 2 3
THE DIGITAL CODE IS AS FOLLOWS
0 NO FORCE BECAUSE FOOTPAD IS OFF SURFACE
1 ELASTIC COMPRESSION IN STRUT
2 PLASTIC COMPRESSION IN STRUT
3 STRUT RE-EXTENDED FROM PREVIOUSLY CRUSHED POSITION BUT IS STILL SHORTER THAN ORIGINAL LENGTH
5 STRUT STRETCHED BEYOND ORIGINAL LENGTH BUT LESS THAN 10000 LBS FORCE
9 STRUT IN TENSION MORE THAN 10000 LBS

DT ITERATION TIME INTERVAL
E OUTPUT INDICATOR E=0 IF LEG IS NOT COMRESSING OR THE FORCE IN LEG IS LESS THAN 10000 LBS. E=1 IF AT LEAST ONE LEG IS IN TENSION GREATER THAN 10000 LBS

IE SAME AS E

ETA1 (OR ETA) ANGLE OF SLOPE IN PRINCIPLE DIRECTION
FORCEx NET FORCE ACTING ON THE VEHICLE C.G. IN THE X DIRECTION IN THE (F) COORDINATE SYSTEM
FORCEy NET FORCE ACTING ON THE VEHICLE C.G. IN THE Y DIRECTION IN THE (F) COORDINATE SYSTEM
FORCEz NET FORCE ACTING ON THE VEHICLE C.G. IN THE Z DIRECTION IN THE (F) COORDINATE SYSTEM
IFORCEx SAME AS FORCEx
IFORCey SAME AS FORCEy
IFORCez SAME AS FORCEz
TURQZ COMPONENT OF TURQUE ACTING ON THE VEHICLE C.G. IN THE

Figure 6-16. Subroutine - INIT (Continued)
INITIALIZING

COMMON/LS1/BETAP,BETTDT,BTAMIN,CRUSH(12),DT,E,ETA,FORCEZ,FOTAC(4),
FXP(4),FORCEX,FORCEY,FSUMZ,FSUMY,FXGRAV,FYGRAV,G1,G2,
GND,GND2,GND3,GNDF,ITORE,JXPSI,MAX,SCOS(3,3),SPRRI(3,3),SSTRUT(4,3),
TIME,TYPE(3),TOK,1,T2,11,U2,12,U3,YAW,VELOC,VFO(4,3),VFOT(4,3)

SUBROUTINE INIT

PREPARE LINE COUNT, ETC. FOR PRINTOUT

ITORE=1
PRINT INTERVAL IS SNTRVL
SNTRVL=.04
LINE=1
IF(SNTRVL-.05)1100,1110,1101
1100 LCOUNT=40
GO TO 1103
1101 LCOUNT=50
1103 DO 441 I=.1,12

 INITIALIZE CRUSH AND FOTAC(I) AND VFO

Figure 6-16. Subroutine - INIT (Continued)
441 CRUSH (I) = 0
DO 200 I = 1, 4
FOTAC(I) = FOTAC(I)

INITIALIZE INPUT PARAMETERS FOR THE RUN

DO 200 J = 1, 3
200 VFOT(I, J) = VFOT(I, J)
XCG = XCG1
YCG = YCG1
ZCG = ZCG1
XCG = ZCG1
W2E = W2E1
W3E = W3E1
PITCH = PITCH1
ROLL = ROLL1
YAW = YAW1
ETA = ETA1

CONVERT THE VELOCITY VECTOR OF THE C.G. TO THE (F) COORDINATE SYSTEM

YDCG = HVEL*COS(ETA) + VVEL*SIN(ETA)
XDCG = VVEL*(-COS(ETA)) + HVEL*(SINIETA)

INITIALIZE XSFOTP

DO 38 II = 1, 4
38 XSFOTP(II) = 0.0

INITIALIZE SPRR, THE SHORTEST PREVIOUS STRUT LENGTH AND SSTRUT, THE PRESENT STRUT LENGTH

DO 82 II = 1, 4
DO 83 I = 1, 3
CALL LEGTYP

SPRR(II, I) = SQRT((VFOT(II, I) - XB(I, II))**2 + (VFOT(II, 2) - YB(I, II))**2 + (VFOT(II, 3) - ZB(I, II))**2)
SSTRUT(II, I) = SQRT((VFOT(II, 1) - XB(I, II))**2 + (VFOT(II, 2) - YB(I, II))**2 + (VFOT(II, 3) - ZB(I, II))**2)

RETAP = 0
GND = 0
GNDF = 0
FORCEX = 0
FORCEY = 0
TORGZ = 0
DMIN = 1E2
PSIMAX = -10
TIME = 0
GSUM = -1E20
PRNI = 0
DT = .0005
RTMIN = 1E2

THE GRAVITY FORCES ARE CONSTANT

Figure 6-16. Subroutine - INIT (Continued)
FXGRAV = -(32.2/6.0) * VASLUG * COS(ETA)
FYGRAV = (32.2/6.0) * VASLUG * SIN(ETA)

COMPUTE SCOS FROM ROLL, PITCH AND YAW

SCOS(1,1) = COS(PITCH) * COS(ROLL) + SIN(PITCH) * SIN(YAW) * SIN(ROLL)
SCOS(2,1) = -SIN(PITCH) * COS(ROLL) + COS(PITCH) * SIN(YAW) * SIN(ROLL)
SCOS(3,1) = -COS(YAW) * SIN(ROLL)
SCOS(1,2) = SIN(PITCH) * COS(YAW)
SCOS(2,2) = COS(PITCH) * COS(YAW)
SCOS(3,2) = -SIN(YAW)
SCOS(1,3) = COS(PITCH) * SIN(ROLL) - SIN(PITCH) * SIN(YAW) * COS(ROLL)
SCOS(2,3) = -SIN(PITCH) * SIN(ROLL) - COS(PITCH) * SIN(YAW) * COS(ROLL)
SCOS(3,3) = COS(YAW) * COS(ROLL)
SC1 = $STRUT(1,1)
SC2 = $STRUT(1,2)
SC3 = $STRUT(1,3)
RETURN
END

Figure 6-16. Subroutine - INIT (Concluded)
TITLE: STAR

THREE DIMENSIONAL STABILITY DETERMINATION

AUTHOR: J.C. GIBSON

BENDIX PRODUCTS AEROSPACE DIVISION

DATE: NOVEMBER, 1964

PURPOSE: THIS SUBROUTINE DETERMINES THE MINIMUM VEHICLE STABILITY ANGLE BETA FOR A THREE DIMENSIONAL VEHICLE LANDING

METHOD: SUBROUTINE DETERMINES THE MINIMUM ANGLE BETWEEN A GRAVITY VECTOR THROUGH THE VEHICLE C.G., AND A LINE THROUGH THE SURFACE CONTACT REGION IN THE DIRECTION OF MOTION. SEE BENDIX REPORT MM 64 - 9 FOR DETAILS

CALL: CALL STAR

NOTES: THIS PROGRAM WAS WRITTEN IN FORTRAN IV

THIS PROGRAM EXECUTES ON UNIVAC 1107 COMPUTER

NOTE: ALL UNITS ARE FEET, SECONDS, RADIANS AND SLUGS

(FT) DENOTES COORDINATE SYSTEM MOVING WITH THE VEHICLE
(F) DENOTES COORDINATE SYSTEM FIXED IN SPACE

APPEARANCE OF 1 AFTER ANY VARIABLE INDICATES THE INITIAL VALUE OF THAT VARIABLE. ABSENCE OF THE 1 INDICATES THE INSTANTANEOUS VALUE.

INPUT: BY EQUIVALENCE TO COMMON STORAGE

INPUT DEFINITIONS:

ETA (OR ETA) ANGLE OF SLOPE IN PRINCIPLE DIRECTION

PSI MAX MAXIMUM PSI ANGLE REACHED DURING THE ENTIRE COMPUTER RUN

XCG, YCG, ZCG DIRECTION COSINES RELATING FIXED (F) AND MOVING (M) COORDINATE SYSTEMS

VFOT1(I,J) (OR VFOT1(I,J)) I TH COMPONENT OF THE VECTOR POSITION OF THE J TH FOOTPAD IN (F) COORDINATE SYSTEM

I = 1 IS X COMPONENT

I = 2 IS THE Y COMPONENT

I = 3 IS THE Z COMPONENT

XCG, YCG, ZCG X POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

XCG, YCG, ZCG Y POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

XCG, YCG, ZCG Z POSITION OF THE VEHICLE C.G. IN (F) COORDINATE SYSTEM

XDCG VELOCITY OF VEHICLE C.G. IN X DIRECTION IN (F) COORDINATE SYSTEM

YDCG VELOCITY OF VEHICLE C.G. IN Y DIRECTION IN (F) COORDINATE SYSTEM

ZDCG VELOCITY OF VEHICLE C.G. IN Z DIRECTION IN (F) COORDINATE SYSTEM

OUTPUT: BY EQUIVALENCE TO COMMON STORAGE

Figure 6-17. Subroutine - STAB

BENDIX PRODUCTS AEROSPACE DIVISION
**Figure 6-17. Subroutine - STAB (Concluded)**
TITLE       INTEQM  SUBROUTINE INTEGRATION

AUTHOR      J.C. GIBSON  BENDIX PRODUCTS AEROSPACE DIVISION

DATE        NOVEMBER 1, 1964

PURPOSE     THIS SUBROUTINE DETERMINES VEHICLE MOTION AS THE RESULT
OF THE FORCES IMPOSED ON THE VEHICLE AND SETS ITERATION
INTERVAL ON THE BASIS OF ERROR TESTS

METHOD      SIMPLE RECTANGULAR INTEGRATION IS USED TO DETERMINE
VEHICLE POSITION AT THE END OF THE TIME INTERVAL DT
AN ERROR TEST ON THE THIRD TERM OF THE TAYLOR SERIES
IS USED TO DETERMINE THE ITERATION INTERVAL SIZE

CALL        CALL INTEQM

NOTES       THIS PROGRAM WAS WRITTEN IN FORTRAN IV
THIS PROGRAM EXECUTES ON UNIVAC 1107 COMPUTER

NOTE        ALL UNITS ARE FEET, SECONDS, RADIANS AND SLUGS
(V) DENOTES COORDINATE SYSTEM MOVING WITH THE VEHICLE
(F) DENOTES COORDINATE SYSTEM FIXED IN SPACE

APPEARANCE OF 1 AFTER ANY VARIABLE INDICATES THE
INITIAL VALUE OF THAT VARIABLE.
ABSENCE OF THE 1 INDICATES THE INSTANTANEOUS VALUE.

INPUT       BY EQUIVALENCE TO COMMON STORAGE

INPUT DEFINITIONS

DT          ITERATION TIME INTERVAL
FORCEX      NET FORCE ACTING ON THE VEHICLE C.G. IN THE X DIRECTION
            IN THE (F) COORDINATE SYSTEM
FORCEY      NET FORCE ACTING ON THE VEHICLE C.G. IN THE Y DIRECTION
            IN THE (F) COORDINATE SYSTEM
FORCEZ      NET FORCE ACTING ON THE VEHICLE C.G. IN THE Z DIRECTION
            IN THE (F) COORDINATE SYSTEM
IFORCX      SAME AS FORCEX
IFORCY      SAME AS FORCEY
IFORCZ      SAME AS FORCZ
TORQX       COMPONENT OF TORQUE ACTING ON THE VEHICLE C.G. IN THE
            X-Y PLAN
ITORQ       SAME AS TORQZ
FXP          FORCE IN THE X DIRECTION ON THE FOOTPAD
FYP          FORCE IN THE Y DIRECTION ON THE FOOTPAD
FZP          FORCE IN THE Z DIRECTION ON THE FOOTPAD
G           TORQUES IN THE (M) COORDINATE SYSTEM
SCOS        DIRECTION COSINES RELATING FIXED (F) AND MOVING (M)
            COORDINATE SYSTEMS
U1          MASS MOMENT OF INERTIA OF VEHICLE ABOUT X AXIS IN (M)
            COORDINATE SYSTEM
U2          MASS MOMENT OF INERTIA OF VEHICLE ABOUT Y AXIS IN (M)
            COORDINATE SYSTEM
U3          MASS MOMENT OF INERTIA OF VEHICLE ABOUT Z AXIS IN (M)

Figure 6-22. Subroutine - INTEQM
SUBROUTINE INTEQM

DIMENSION BRKT(3,3),VID(3,3),VIO(3,3),VIN(3,3),SN(3,3),BKP(3,3)
COMMCN/LS1/ETAP*BETDAT/ETAMIN,CRUSHER(D),U,T,E,GAMEZ,TOTAL(4),
1FXP(4),FORCEX,FORCEY,FCSUMZ,FCSUMX,FCSUMY,FXGRAV,FXHAV,G1,G2,
2G3*G4D5*G6D7*G8,TIME*TYPE(3),VELOCITY(3),VACSLUG,VEQUT(4,3),W2E,
4W2E,W3E*XCG,XDCG,X(1),X(2),X(3),Y(1),Y(2),Y(3),ZCG,ZDCG
EQUIVALENCE (VACSLUG,VEQUT),(XH(1),Y(1)),(YH(1),Y(1)),
1ZH(1),Z(1),(FSUMX,FCSUMX),(FSUMY,FCSUMY),(FCSUMZ,FCSUMZ)

EULER EQUATIONS

W1=G1/U1
W2=(1/U2)*(G2+(G3-U1)*W1E*W2E)
W3=(1/U3)*(G3+(G3-U2)*W2E*W1E)

FIND THE AVERAGE VALUES OF ANGULAR VELOCITY IN THE TIME INTERVAL
OF LENGTH DT

W1=W1F+W1I*DT/2
W2=W2E+W2D*DT/2
W3=W3E+W3D*DT/2

CALCULATE NEW V VECTOR (1,J,K AXES)

BRKT(1,1)=-W2W2+W3W3
BRKT(1,2)=W1W2+W3D
BRKT(1,3)=W1W3-W2D
BRKT(2,1)=W2W1-W3D
BRKT(2,2)=-W1W1+W3W3
BRKT(2,3)=W2W3+W1D
BRKT(3,1)=W3W1+W2D
BRKT(3,2)=W3W2-W1D
BRKT(3,3)=-(W2W2+W1W1)

Figure 6-22. Subroutine - INTEQM (Continued)
ERROR CHECK ON 3RD TERM ON THE TAYLOR SERIES

1. ERROR = D.T

2. DO 3 I = 1, 3

3. ERROR = ERROR + ABS(ARCT(1, J) - EXP(I, J1) * DT * DT / 6.0)

4. IF (ERROR > 10D0 * DT) STOP

PRINT 100, DT

10 FORMAT (4H OUTPUT ON ERROR CHECKING, DT = , F14.5)

5. GOTO 2

6. END

STORE ANGULAR ACCELERATIONS FOR USE IN NEXT ERROR CHECK ON DT INTERVAL SIZE

7. DO 2 1 = 1, 3

8. DO 2 J = 1, 3

9. ERROR(I, J) = ARCT(I, J)

10. VINC(I, J) = 0

11. VINC(I, J) = 0

12. VINC(I, J) = 0

13. VINC(I, J) = 0

14. VINC(I, J) = 0

15. VINC(I, J) = 0

16. VINC(I, J) = 0

17. VINC(I, J) = 0

18. VINC(I, J) = 0

19. VINC(I, J) = 0

20. DO 41 1 = 1, 3

21. DO 41 J = 1, 3


23. VINC(I, J) = ARCT(I, J) * DT * DT / 2 + VINC(I, J) * DT + VINC(I, J)

24. DO 42 1 = 1, 3

25. DO 42 J = 1, 3

26. ERROR(I, J) = SCOS(I, 1) * VINC(J, 1) + SCOS(I, 2) * VINC(J, 2) + SCOS(I, 3) * VINC(J, 3)

27. GOTO 2

28. DO 43 1 = 1, 3

29. DO 43 J = 1, 3

30. AVOID SCOS GREATER THAN UNITY

31. IF (ABS(SN(I, J) / SN(I, J)) = 1) STOP

32. SN(I, J) = SIN(I, J) / ABS(SN(I, J))

33. COS(I, J) = SN(I, J)

34. INTEGRATE TO FIND ANGULAR VELOCITY AT THE END OF THE TIME INTERVAL DT

35. W1E = W1E + WDI * DT

Figure 6-22. Subroutine - INTEQM (Continued)
Figure 6-22. Subroutine - INTEQM (Concluded)
<table>
<thead>
<tr>
<th>NAME</th>
<th>XE (1+1)</th>
<th>XE (1+2)</th>
<th>XE (3+1)</th>
<th>XE (3+2)</th>
<th>AB (2+2)</th>
<th>XE (1+4)</th>
<th>XE (2+4)</th>
<th>XE (3+4)</th>
<th>XE (4+1)</th>
<th>XE (4+2)</th>
<th>XE (4+4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.16</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
</tr>
<tr>
<td>2.17</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
</tr>
<tr>
<td>2.18</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
</tr>
<tr>
<td>2.19</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
</tr>
<tr>
<td>2.20</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-4.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
<td>-6.75E00</td>
</tr>
</tbody>
</table>

**Figure 6-23. Input Data Printout**
<table>
<thead>
<tr>
<th>TIME</th>
<th>F1x</th>
<th>F2x</th>
<th>F3x</th>
<th>F4x</th>
<th>XFFT1</th>
<th>ABFT2</th>
<th>ABFT3</th>
<th>ABFT4</th>
<th>ON/OFF</th>
<th>THEOD</th>
<th>PH1D1</th>
<th>BLAST</th>
<th>PSI1D1</th>
<th>CRUSH</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.23</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.34</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.45</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.56</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.67</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.78</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Figure 6-25. Final Printout of Stored Data
SECTION VII

GENERAL NON-PLANAR LUNAR LANDING COMPUTER PROGRAM

INTRODUCTION

This computer program is basically an extension of the Planar Lunar Landing Computer Program. An attempt was made to use as much of the original planar program as possible and to keep the flow of the non-planar program about the same.

METHOD

The motion of the vehicle and that of the footpads is computed in the ground coordinate system while the geometry is computed in the vehicle coordinate system. These two coordinate systems are related by a set of direction cosines.

The following procedure is used to set up the initial orientation of the vehicle.

First, consider the vehicle oriented in the ground coordinate system with leg No. 1 in the "+Y" direction, leg No. 3 in the "-Y" direction and leg No. 4 in the "+Z" direction. A right-handed coordinated system is used. From this basic orientation, the vehicle is rotated through a pitch angle, yaw angle, and roll angle in that order. This sets the initial direction cosine matrix relating the vehicle coordinate system to the ground coordinate system. This matrix of original direction cosines is:

\[
[X] = \begin{bmatrix}
\cos \alpha \cos \gamma - \\
\sin \alpha \sin \beta \sin \gamma
\end{bmatrix}
\begin{bmatrix}
(-\sin \alpha \cos \beta) \\
\sin \alpha \sin \beta \cos \gamma
\end{bmatrix}
\begin{bmatrix}
\cos \alpha \sin \gamma + \\
\cos \alpha \cos \gamma
\end{bmatrix}
\begin{bmatrix}
\sin \gamma \\
\cos \gamma
\end{bmatrix}
\begin{bmatrix}
(-\cos \beta \sin \gamma - \\
\sin \beta)
\end{bmatrix}
\begin{bmatrix}
\cos \beta \cos \gamma
\end{bmatrix}
\]

where for this matrix

\[\alpha = \text{pitch angle}\]
\[\beta = \text{yaw angle}\]
\[\gamma = \text{roll angle}\]
NOTE: \[ [XL] [XL]^T = [I] \text{ or } [XL]^T = [XL]^{-1} \]

That is, the transpose of this matrix is equal to the inverse of the matrix.

The vehicle coordinate system and the ground coordinate system are related by the following matrix equations:

\[
\begin{bmatrix}
{x_p} \\
{y_p} \\
{z_p}
\end{bmatrix} = [XL] \begin{bmatrix}
{x_{pv}} \\
{y_{pv}} \\
{z_{pv}}
\end{bmatrix} + \begin{bmatrix}
{X} \\
{Y} \\
{Z}
\end{bmatrix}
\]

where

- \( x_p, y_p, \) and \( z_p \) are pad coordinates in the ground coordinate system.
- \( x_{pv}, y_{pv}, \) and \( z_{pv} \) are pad coordinates in the vehicle coordinate system.
- \( X, Y, \) and \( Z \) are vehicle coordinates in ground system.

The vehicle is initially set so that the lowest leg is into the ground .001 feet.

The initial velocities are given in the ground coordinate system. The right hand rule is used to determine the signs of the angular velocities (See Figure 7-1).

The strut geometry is computed in the vehicle coordinate system. This part of the program is therefore basically the same as the planar program.

The footpad motion is computed in the ground coordinate system as was the case in the planar program. This portion of the non-planar program is again very similar to the planar program.

Since some computations are done in the ground coordinate system and others are done in the vehicle coordinate system, a set of directions cosines relating the two sets of axes must be kept updated. This is done for every time increment after the equations of motion are integrated. The updating of direction cosines and the integration of the equations of motion are handled in subroutine INTEQM. A description of this subroutine can be found on page 5-36 (Reference 1).
NOZZLE CLEARANCE

\[ XN = X - HN \left\{ XL(1, 1) \right\} - RN \left\{ SIN11 \right\} \]
\[ SIN11 = \sqrt{1.0 - XL(1, 1)^2} \]

STABILITY CHECK

\[
\begin{bmatrix}
\dot{X}_{grav.} \\
\dot{Y}_{grav.} \\
\dot{Z}_{grav.}
\end{bmatrix} =
\begin{bmatrix}
\cos \zeta & -\sin \zeta & 0 \\
\sin \zeta & \cos \zeta & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
\dot{X} \\
\dot{Y} \\
\dot{Z}
\end{bmatrix}
\]
\[
\begin{bmatrix}
X_{Grav.} \\
Y_{Grav.} \\
Z_{Grav.}
\end{bmatrix} =
\begin{bmatrix}
\cos \gamma & -\sin \gamma & 0 \\
\sin \gamma & \cos \gamma & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

\[
\begin{bmatrix}
X_{PM} \\
Y_{PM} \\
Z_{PM}
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \phi' & \sin \phi' \\
0 & -\sin \phi' & \cos \phi'
\end{bmatrix}
\begin{bmatrix}
X_{Grav.} \\
Y_{Grav.} \\
Z_{Grav.}
\end{bmatrix}
\]

where

- \(X, Y, \) and \(Z\) are coordinates of vehicle in ground coordinate system.
- \(\dot{X}, \dot{Y}, \) and \(\dot{Z}\) are vehicle velocities in ground coordinate system.
- \(X_{Grav.}, Y_{Grav.}, \) and \(Z_{Grav.}\) are vehicle coordinates in the gravitational coordinate system.
- \(XP, YP,\) and \(ZP\) are pad coordinates in the ground coordinate system.
XM, YM, and ZM are vehicle coordinates in the coordinate system of motion.

XPM, YPM, and ZPM are pad coordinates in the coordinate system of motion.

Using these basic relationships between coordinate systems, the following is obtained.

\[
YM = X \sin \phi' + Y \cos \phi' + Z \sin \phi'
\]

\[
YPM(i) = X_P(i) \sin \phi' + Y_P(i) \cos \phi' + Z_P(i) \sin \phi'
\]

If \( YPM(i) > YM \) for all legs, the vehicle is considered unstable.

The stability angle tabulated in the output (BETAF) is computed for the leg with the largest value of YPM.

\[
\beta_F = \tan^{-1} \left[ \frac{YPM(IFRONT) - YM}{XM - XPM(IFRONT)} \right]
\]

where

IFRONT = index of front leg in direction of motion.

CONCLUSIONS

This non-planar program has not been completely checked out.

The planar run included in this report is the same run that is included in the Planar Lunar Landing Program. Some of the differences in these two runs would be due to the different integration routines used.

Perfect symmetry is not maintained in this planar run. Some of this can be attributed to roundoff error. For instance, the initial \( x_P \) values are slightly different and therefore the
forces developed in the two legs on the ground will not be completely symmetrical.

In non-planar runs made, the motion looked reasonable although no detailed studies were made. No comparisons have been made with non-planar test data.

Figure 7-2 illustrates the input data format. Figures 7-3 through 7-6 are flow diagrams for the program and its subroutines. Figures 7-7 through 7-10 are complete program listings. Figures 7-11 through 7-18 are sample output data from the program. Figure 7-11 is a printout of the input data used. Figure 7-12 prints out some auxiliary inputs required by the main program. These are computed by the subroutine INCON. Figure 7-13 output data indicates the instantaneous vehicle orientation. Figure 7-14 is a summary printout of pertinent information for the run. Figure 7-15 presents the footpad positions and forces for footpads 2 and 3. Figure 7-16 presents the forces acting on the footpads from the struts.

Figures 7-17 and 7-18 are similar to Figures 7-15 and 7-16 but present this information for footpads 1 and 4.
### Figure 7-2. Input Data Form

<table>
<thead>
<tr>
<th>Columns</th>
<th>Alpha(1)</th>
<th>Alpha(2)</th>
<th>Alpha(3)</th>
<th>Alpha(4)</th>
<th>Alpha(5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>D(1)</td>
<td>D(2)</td>
<td>D(3)</td>
<td>ETC.</td>
<td></td>
</tr>
<tr>
<td>Used</td>
<td>D11(1)</td>
<td>D11(2)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>For</td>
<td>F11(1)</td>
<td>F11(2)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I.D.</td>
<td>F22(1)</td>
<td>F22(2)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G.R.D.M.U.(1)</td>
<td>G.R.D.M.U.(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1(1)</td>
<td>P1(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P2(1)</td>
<td>P2(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P3(1)</td>
<td>P3(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R1(1)</td>
<td>R1(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R2(1)</td>
<td>R2(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Theta(1)</td>
<td>Theta(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R.P(1)</td>
<td>R.P(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sks(1)</td>
<td>Sks(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeltaP</td>
<td>DeltaT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eps3</td>
<td>Eps4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fint</td>
<td>Grav</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P mass</td>
<td>RunnΦΦ</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sk1</td>
<td>Sk2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ske1</td>
<td>Ske2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SerΦΦ</td>
<td>V mass</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PitchΦΦ</td>
<td>YawΦΦ</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RollΦΦ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This program determines detailed non-planar motions of Lunar Landing Vehicle from touchdown to rest or instability.

Figure 7-3. Main Computer Program
STOP → Yes: Is run being duplicated?

No:

- Zero out initial forces for parabolic integration of footpad motion
- Determine initial strut angles and trigonometric relationships
- Determine initial footpad positions and velocities
- Determine initial coordinates and velocities of vehicle C.G.
- Determine initial strut lengths

140: Determine strut angles and assign to proper quadrants

280: Test (IFLAG)

250: 0

Figure 7-3. Main Computer Program (Continued)
Figure 7-3. Main Computer Program (Continued)
Is footpad moving with vehicle?

Set footpad forces equal to zero

NOGR(I) = -1

Find footpad positions and velocities by rigid body motion with vehicle

Footpad is moving independently of the vehicle, calculate strut forces and footpad energy

Figure 7-3. Main Computer Program (Continued)
Determine forces acting on the vehicle

Call Subroutine INTEQM
Determine vehicle C.G. motion and update direction cosines

Index previous forces for the next integration of footpad motion

Increment time

CONS = CONS + 1.0

Test nozzle clearance and store minimum value

Test stability angle and store minimum value

Figure 7-3. Main Computer Program (Continued)
Call Subroutine STABAN
Compute stability angle

Is vehicle unstable?

Has final time been exceeded?

\[ K_M = K_M + 1 \]

\[ \text{Test (CONS-KPRINT)} \]

\[ \geq 0 \]

\[ \text{CONS} = 0.0 \]

\[ \text{Test (K}_M\text{-MULT * KPRINT)} \]

\[ > 0 \]

\[ M = M + 1 \]

Store variables for final print

Print ON LINE output

Figure 7-3. Main Computer Program (Continued)
Figure 7-3. Main Computer Program (Continued)
IFLAGX = -1

Vehicle is unstable
Print output data

Assign 785 to MM

IFLAGX = 0

Time limit has been exceeded
Print output data

Figure 7-3. Main Computer Program (Continued)
Figure 7-3. Main Computer Program (Continued)
Figure 7-3. Main Computer Program (Continued)
Figure 7-3. Main Computer Program (Concluded)
This subroutine converts the input data to a form that is used in the program.

Figure 7-4. Subroutine - INCON
This subroutine determines the critical leg for stability and calculates the stability angle for this leg.

Figure 7-5. Subroutine - STABAN
This subroutine integrates the equations of motion for the vehicle and updates the direction cosines relating the ground coordinate system to the vehicle coordinate system.

**Figure 7-6. Subroutine - INTEQM**
TITLE MAIN PROGRAM GENERAL THREE DIMENSIONAL LUNAR LANDING DYNAMICS COMPUTER PROGRAM

AUTHOR J. CADORET BENDIX PRODUCTS AEROSPACE DIVISION

DATE NOV. 1964

PURPOSE THIS PROGRAM COMPUTES THE DETAILED VEHICLE MOTIONS FOR A THREE DIMENSIONAL LUNAR LANDING

NOTE THIS PROGRAM WAS WRITTEN IN FORTRAN IV

NOTE THIS PROGRAM WAS COMPILLED ON THE UNIVAC 1107 DIGITAL COMPUTER

THIS PROGRAM HAS BEEN CHECKED OUT TO A LIMITED EXTENT ONLY. FURTHER CHECK OUT AND CORRELATION WITH A PHYSICAL DROP TEST MODEL ARE CURRENTLY IN PROGRESS. AS THE RESULT OF THIS LIMITED CHECKOUT, THE RESULTS OF THIS PROGRAM SHOULD BE USED WITH CAUTION

6 DEGREES OF FREEDOM

LUNAR LANDING DYNAMICS COMPUTER PROGRAM FOR ASSYMMETRIC VEHICLE LANDING GEAR CONFIGURATION USING PARABOLIC INTEGRATION ON FOOTPAD MOTIONS

INPUT BY EQUIVALENCE TO COMMON STORAGE

NOTE AN INPUT PARAMETER FOLLOWED BY A 0 IMPLIES THE INITIAL VALUE AS B10(I). THE INSTANTANEOUS VALUE OF THE PARAMETER IS DEFINED WITHOUT THE 0 AS B1(I).

ALPHA ANGLE (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE) SUBTENDED BY THE LOWER HARDPOINTS AND VEHICLE C.G.

BETAF STABILITY ANGLE FOR FRONT LEG IN THE DIRECTION OF MOTION

BETMIN MINIMUM STABILITY ANGLE

B10 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE

B20 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 2 AND VEHICLE CENTERLINE

B2PREV VALUE OF B2 AT THE END OF THE PREVIOUS TIME INCREMENT

B30 ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION) BETWEEN STRUT NO. 3 AND VEHICLE CENTERLINE

B3PREV VALUE OF B3 AT THE END OF THE PREVIOUS TIME INCREMENT

CONS COUNTER FOR DETERMINING ON LINE PRINT FREQUENCY

COSC COSTHETA)

COSD COS(B1)

COSE COS(THEETA+ALPHA/2.0)

COSG COS(THEETA+ALPHA/2.0)

C10,C20,C30 ANGLE , IN PLANE FORMED BY STRUT AND A NORMAL TO THE DIRECTION OF MOTION, BETWEEN STRUT AND A PLANE NORMAL TO THE VEHICLE CENTERLINE – FOR STRUTS 1,2,3 RESPECTIVELY

C1PREV VALUE OF C1 AT THE END OF THE PREVIOUS TIME INCREMENT

C2PREV VALUE OF C2 AT THE END OF THE PREVIOUS TIME INCREMENT

C3PREV VALUE OF C3 AT THE END OF THE PREVIOUS TIME INCREMENT

D VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS

Figure 7-7. Main Program
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELTAP</td>
<td>Distance from bottom of footpad to intersection of the leg struts</td>
</tr>
<tr>
<td>DELTAT</td>
<td>Time interval between program calculations</td>
</tr>
<tr>
<td>DELTTT</td>
<td>Time increment used in the integration of footpad motion. DELTTT = DELTAT/KCONMX</td>
</tr>
<tr>
<td>D10</td>
<td>Initial length (projected in plane normal to direction of vehicle motion) of the component (in plane perpendicular to the vehicle centerline) of strut No 1 length</td>
</tr>
<tr>
<td>D11</td>
<td>Vertical distance from vehicle C.G. to upper hardpoint</td>
</tr>
<tr>
<td>EGBAL1</td>
<td>Energy dissipated based on vehicle velocities and C.G. drop</td>
</tr>
<tr>
<td>EGBAL2</td>
<td>Energy dissipated based on plastic stroke of all struts</td>
</tr>
<tr>
<td>EGBAL3</td>
<td>Energy dissipated based on plastic and full elastic stroke of all struts</td>
</tr>
<tr>
<td>ENPRO(I)</td>
<td>Percent of total energy absorbed by stroking of the struts of leg set 1</td>
</tr>
<tr>
<td>EPSEN</td>
<td>Program constant equal to 10 percent of the possible potential energy which could be stored in a footpad as the result of elastic stroking of the upper strut</td>
</tr>
<tr>
<td>EPS2</td>
<td>Minimum allowable footpad sliding velocity</td>
</tr>
<tr>
<td>EPS3</td>
<td>Limiting minimum velocity of vehicle C.G. in X direction</td>
</tr>
<tr>
<td>EPS4</td>
<td>Limiting minimum velocity of vehicle C.G. in Y and Z directions</td>
</tr>
<tr>
<td>EPS5</td>
<td>Limiting minimum angular velocities</td>
</tr>
<tr>
<td>NOTE</td>
<td>If XVEL is less than EPS3, YVEL and ZVEL are less than EPS4 and the angular velocities are less than EPS5 simultaneously and the footpads are all less than 1 ft. from the ground, the program terminates</td>
</tr>
<tr>
<td>E1(I)</td>
<td>Potential energy stored in strut No.1 of leg 1 due to compression or extension of the leg</td>
</tr>
<tr>
<td>E2(I)</td>
<td>Potential energy stored in strut No.2 of leg 1 due to compression or extension of the leg</td>
</tr>
<tr>
<td>E3(I)</td>
<td>Potential energy stored in strut No.3 of leg 1 due to compression or extension of the leg</td>
</tr>
<tr>
<td>FA(I)</td>
<td>Force in the Z direction acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set</td>
</tr>
<tr>
<td>FAG(I)</td>
<td>Force in the Z direction acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set (Ground coordinate system)</td>
</tr>
<tr>
<td>FB(I)</td>
<td>Force, parallel to the vehicle centerline acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set</td>
</tr>
<tr>
<td>FBG(I)</td>
<td>Force in the X direction acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set (Ground coordinate system)</td>
</tr>
<tr>
<td>FC(I)</td>
<td>Force, normal to vehicle centerline in the X-Y plane acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set</td>
</tr>
<tr>
<td>FCG(I)</td>
<td>Force in the Y direction acting on the vehicle C.G. as the resultant of the strut forces in the three struts of the 1st leg set (Ground coordinate system)</td>
</tr>
<tr>
<td>FIN</td>
<td>Maximum allowable time for computer run</td>
</tr>
<tr>
<td>FX</td>
<td>Same as FXP</td>
</tr>
<tr>
<td>FXP(I)</td>
<td>Force in the X direction on the footpad (1)</td>
</tr>
<tr>
<td>FXPLG1</td>
<td>Average force in the X direction acting on footpad No.1 during the time interval DT. This force is defined by the parabolic integration procedure</td>
</tr>
<tr>
<td>FXPLG2</td>
<td>Average force in the X direction acting on footpad No.2 during the time interval DT. This force is defined by the parabolic integration procedure</td>
</tr>
</tbody>
</table>

Figure 7-7. Main Program (Continued)
C FXPLG3 AVERAGE FORCE IN THE X DIRECTION ACTING ON FOOTPAD NO. 3
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FXPL31 FXPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT
C FXPL33 FXPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT
C FXPL35 FXPLG3 VALUE FOR LEG IIX STORED FOR FINAL PRINT
C FXPL37 FXPLG3 VALUE FOR LEG JJX STORED FOR FINAL PRINT
C FX1 FX VALUE FOR LEG II STORED FOR FINAL PRINT
C FX3 FX VALUE FOR LEG JJ STORED FOR FINAL PRINT
C FX5 FX VALUE FOR LEG IIX STORED FOR FINAL PRINT
C FX7 FX VALUE FOR LEG JJX STORED FOR FINAL PRINT
C FYPLG1 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 1
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FYPLG2 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 2
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FYPLG3 AVERAGE FORCE IN THE Y DIRECTION ACTING ON FOOTPAD NO. 3
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FYPL31 FYPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT
C FYPL33 FYPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT
C FYPL35 FYPLG3 VALUE FOR LEG IIX STORED FOR FINAL PRINT
C FYPL37 FYPLG3 VALUE FOR LEG JJX STORED FOR FINAL PRINT
C FZPLG1 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 1
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FZPLG2 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 2
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FZPLG3 AVERAGE FORCE IN THE Z DIRECTION ACTING ON FOOTPAD NO. 3
C DURING THE TIME INTERVAL DT. THIS FORCE IS DEFINED BY
C THE PARABOLIC INTEGRATION PROCEDURE
C FZPL31 FZPLG3 VALUE FOR LEG II STORED FOR FINAL PRINT
C FZPL33 FZPLG3 VALUE FOR LEG JJ STORED FOR FINAL PRINT
C FZPL35 FZPLG3 VALUE FOR LEG IIX STORED FOR FINAL PRINT
C FZPL37 FZPLG3 VALUE FOR LEG JJX STORED FOR FINAL PRINT
C F11(I) PLASTIC STROKE FORCE IN STRUT NO. 1 OF LEG I
C F21(I) PLASTIC STROKE FORCE IN STRUT NO. 2 OF LEG I
C F31(I) PLASTIC STROKE FORCE IN STRUT NO. 3 OF LEG I
C F11 PLASTIC STROKE FORCE FOR UPPER STRUT (NO. 1)
C F22 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 2
C F33 PLASTIC STROKE FORCE FOR LOWER STRUT NO. 3
C GRAV LOCAL GRAVITY
C GRDMU COEFFICIENT OF FRICTION BETWEEN VEHICLE FOOTPADS AND GROUND
C GRFX FORCE ACTING IN THE X DIRECTION ON THE VEHICLE C.G.
C (GROUND COORDINATE SYSTEM ) THIS FORCE IS USED
C IN THE INTEGRATION OF EQUATIONS OF MOTION
C GRFY FORCE ACTING IN THE Y DIRECTION ON THE VEHICLE C.G.
C (GROUND COORDINATE SYSTEM ) THIS FORCE IS USED
C IN THE INTEGRATION OF EQUATIONS OF MOTION
C GRFZ FORCE ACTING IN THE Z DIRECTION ON THE VEHICLE C.G.
C (GROUND COORDINATE SYSTEM ) THIS FORCE IS USED
C IN THE INTEGRATION OF EQUATIONS OF MOTION
C H DISTANCE FROM THE BOTTOM OF THE FOOTPAD TO THE VEHICLE
C CENTER OF GRAVITY
C HN VERTICAL DISTANCE BETWEEN VEHICLE C.G. AND THE LOWEST
C POINT ON THE NOZZLE CONE
C IBEMIN INDEX OF CRITICAL LEG FOR MINIMUM STABILITY ANGLE
C IFLAG FLAG FOR INITIALIZING THE PROGRAM
C IFLAGE FLAG USED TO PRINT ADDITIONAL OUTPUT

Figure 7-7. Main Program (Continued)
FLAG USED IN FINAL PRINT  
IFLAGX=+1  VEHICLE WAS STABLE,
IFLAGX=0  MAXIMUM ALLOWABLE TIME WAS EXCEEDED,
IFLAGX=-1  VEHICLE WAS UNSTABLE

INDEX OF FRONT LEG (THE FRONT LEG IN THE DIRECTION OF
MOTION IS THE CRITICAL LEG FOR STABILITY)

PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE
PRINTED AS OUTPUT (INITIAL SUMMARY OUTPUT)

PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE
PRINTED AS OUTPUT (ADDITIONAL SUMMARY OUTPUT)

PRINT INDICATOR

PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE
PRINTED AS OUTPUT (INITIAL SUMMARY OUTPUT)

PARAMETER SPECIFYING WHICH LEG SET DATA WILL BE
PRINTED AS OUTPUT (ADDITIONAL SUMMARY OUTPUT)

TIME INTERVAL

COUNTER FOR DETERMINING THE FREQUENCY OF THE STORED
PRINTING

COMPUTATION INCREMENTS BETWEEN PRINTOUT INTERVALS

PRINTOUT LINE COUNTER

INDEX USED TO STORE VARIABLES FOR PRINT AT END OF RUN

INDICATOR FLAG USED TO PRINT STORED OUTPUT DATA AT LESS
FREQUENT INTERVALS IF FINT IS SUCH THAT THE DIMENSIONED
STORAGE CAPACITY FOR THE STORED VARIABLES WILL BE EXCEEDED.

NUMBER OF LEGS ON THE VEHICLE

INDICATES IF FOOTPAD (I) IS MOVING WITH THE VEHICLE.  IF
NOGR(I)=-1, FOOTPAD IS MOVING WITH THE VEHICLE.  IF
NOGR(I)=+1, FOOTPAD IS MOVING INDEPENDENTLY

STARTING COLUMN IN VELOCITY INPUT ARRAY

ENDING COLUMN IN VELOCITY ARRAY

STARTING ROW IN VELOCITY INPUT ARRAY

ENDING ROW IN VELOCITY ARRAY

SEE WRITEUP FOR DISCUSSION OF VELOCITY ARRAY

FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE

FLAG FOR DETERMINING STABILITY PROFILE INPUT SEQUENCE

ANGLE BETWEEN VEHICLE CENTERLINE AND GRAVITY VECTOR

INSTANTANEOUS PITCH ANGLE OF THE VEHICLE

INITIAL PITCH ANGLE OF THE VEHICLE

INSTANTANEOUS PITCH VELOCITY OF THE VEHICLE

MASS OF EACH FOOTPAD

IN MULTIPLE RUNS, THIS IS THE INITIAL X VELOCITY OF THE
PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.

IN MULTIPLE RUNS, THIS IS THE INITIAL Y VELOCITY OF THE
PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.

IN MULTIPLE RUNS, THIS IS THE INITIAL Z VELOCITY OF THE
PREVIOUS RUN. IT IS USED TO PREVENT DUPLICATE RUNS.

INITIAL PITCH VELOCITY OF

J

FRICTION FORCE IN STRUT NO. 1

FRICTION FORCE IN STRUT NO. 2

FRICTION FORCE IN STRUT NO. 3

MOMENT OF INERTIA OF FOOTPADS ABOUT X AXIS (YAW)

THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE
GROUND AT THE INSTANT UNDER INVESTIGATION

MOMENT OF INERTIA OF FOOTPADS ABOUT Y AXIS (ROLL)

THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE
GROUND AT THE INSTANT UNDER INVESTIGATION

MOMENT OF INERTIA OF FOOTPADS ABOUT Z AXIS (PITCH)

THIS TERM INCLUDES ONLY THOSE FOOTPADS WHICH ARE OFF THE
GROUND AT THE INSTANT UNDER INVESTIGATION

Figure 7-7. Main Program (Continued)
Figure 7-7. Main Program (Continued)
IN THE INTEGRATION OF EQUATIONS OF MOTION
VEMY
TORQUE ABOUT THE Y AXIS OF THE VEHICLE
(GROUND COORDINATE SYSTEM) THIS TORQUE IS USED
IN THE INTEGRATION OF EQUATIONS OF MOTION
VEMZ
TORQUE ABOUT THE Z AXIS OF THE VEHICLE
(GROUND COORDINATE SYSTEM) THIS TORQUE IS USED
IN THE INTEGRATION OF EQUATIONS OF MOTION
VMASS
VEHICLE MASS
X
INSTANTANEOUS X POSITION OF THE VEHICLE C.G.
XL(I,J)
SET OF DIRECTION COSINES RELATING VEHICLE COORDINATE
SYSTEM TO FIXED COORDINATE SYSTEM
XMIN
MINIMUM VALUE OF XP (USED ONLY FOR INITIAL POSITIONING
OF VEHICLE)
XMO MX
MOMENT OF INERTIA OF VEHICLE ABOUT X AXIS (YAW)
XMO MY
MOMENT OF INERTIA OF VEHICLE ABOUT Y AXIS (ROLL)
XMO MZ
MOMENT OF INERTIA OF VEHICLE ABOUT Z AXIS (PITCH)
XN
INSTANTANEOUS NOZZLE CLEARANCE NORMAL TO THE SURFACE
XM NMIN
MINIMUM VALUE OF NOZZLE CLEARANCE
XP(I)
X POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
XPV(I)
X POSITION OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
XPVEL(I)
VELOCITY OF THE ITH FOOTPAD IN THE X DIRECTION IN THE
FIXED COORDINATE SYSTEM
XPVVEL(I)
X VELOCITY OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
WITH RIGID BODY MOTION
XP1
XP VALUE FOR LEG II STORED FOR FINAL PRINT
XP3
XP VALUE FOR LEG JJ STORED FOR FINAL PRINT
XP5
XP VALUE FOR LEG IIX STORED FOR FINAL PRINT
XP7
XP VALUE FOR LEG JXI STORED FOR FINAL PRINT
XVEL
INSTANTANEOUS X VELOCITY OF THE VEHICLE C.G.
XVELO
INITIAL VERTICAL VELOCITY OF VEHICLE C.G.
XVELOO
SAME AS XVELO
Y
INSTANTANEOUS Y POSITION OF THE VEHICLE C.G.
YAW
INSTANTANEOUS YAW ANGLE OF THE VEHICLE
YAWO
INITIAL YAW ANGLE OF THE VEHICLE
YAWV
INSTANTANEOUS YAW VELOCITY OF THE VEHICLE
YAWVO
INITIAL YAW VELOCITY OF THE VEHICLE
YP(I)
Y POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
YPV(I)
Y POSITION OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
YPVEL(I)
VELOCITY OF THE ITH FOOTPAD IN THE Y DIRECTION IN THE
FIXED COORDINATE SYSTEM
YPVVEL(I)
Y VELOCITY OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
WITH RIGID BODY MOTION
YP1
YP VALUE FOR LEG II STORED FOR FINAL PRINT
YP3
YP VALUE FOR LEG JJ STORED FOR FINAL PRINT
YP5
YP VALUE FOR LEG IIX STORED FOR FINAL PRINT
YP7
YP VALUE FOR LEG JXI STORED FOR FINAL PRINT
YVEL
INSTANTANEOUS Y VELOCITY OF THE VEHICLE C.G.
YVELO
INITIAL HORIZONTAL VELOCITY OF VEHICLE C.G.
YVELOO
SAME AS YVELO
Z
INSTANTANEOUS Z POSITION OF THE VEHICLE C.G.
ZETA
GROUND SLOPE
ZP(I)
Z POSITION OF FOOTPAD I IN THE FIXED COORDINATE SYSTEM
ZPV(I)
Z POSITION OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
ZPVEL(I)
VELOCITY OF THE ITH FOOTPAD IN THE Z DIRECTION IN THE
FIXED COORDINATE SYSTEM
ZPVVEL(I)
Z VELOCITY OF FOOTPAD I IN THE VEHICLE COORDINATE SYSTEM
WITH RIGID BODY MOTION
ZP1
ZP VALUE FOR LEG II STORED FOR FINAL PRINT

Figure 7-7. Main Program (Continued)
ZP3  ZP  VALUE FOR LEG JJ  STORED FOR FINAL PRINT
ZP5  ZP  VALUE FOR LEG IIX  STORED FOR FINAL PRINT
ZP7  ZP  VALUE FOR LEG JJX  STORED FOR FINAL PRINT
ZVEL  INSTANTANEOUS Z VELOCITY OF THE VEHICLE C.G.
ZVELO  INITIAL TRANSVERSE VELOCITY OF VEHICLE C.G.
ZVEL00  SAME AS ZVELO

1B10(5),B20(5),B30(5),B10(5),B20(5),B30(5),B10(5),B20(5),B30(5),
1B2M1(5),B22M(5),B3M(5),B32M(5),B33M(5),C1M(5),
1XVELO(14,8),YVELO(14,8),ZVELO(14,8),PADENG(5),E1(5),
1E2(5),E3(5),TIM(200),XP3(200),YP3(200),ZP3(200),NOGR(5)

DIMENSION FXPLG3(5),FYPLG3(5),FZPLG3(5),
1COSC(5),SINC(5),SIND(5),COSD(5),SING(5),SINDJ(5),
1SIN(5),COS(5),COSG(5),SIN(5),SINK(5),SINL(5),
1FXPLG1(5),FXPLG2(5),FYPLG(5),FYPLG2(5),FZPLG1(5),FZPLG2(5),
1XP1(200),YP1(200),ZP(200),FX1(200),FX3(200),FXPL33(200),
1FXPL33(200),FXPL33(200),FXPL33(200),FYPL31(200),FYPL31(200),
1FXP5(200),YP5(200),ZP5(200),XP7(200),YP7(200),ZP7(200),
1FX5(200),FX7(200),FXPL37(200),FYPL37(200),FZPL37(200),
1DIMENSION FXPL35(200),FYPL35(200),FZPL35(200),
1ENGY(5),ENPRO(5),EPSEN(5),ALPHA(5),
1XL(3),XPV(5),YPV(5),ZPV(5),XPVELV(5),YPVEL(5),ZPVVEL(5),
1FAG(5),FBG(5),FCG(5),XPVELV(5),YPVEL(5),ZPVVEL(5),ZPVVEL(5),
1FXPL35(200),FYPL35(200),FZPL35(200),
1READ INITIAL DATA
1READ 1000,(ALPHA(I),I=1,5),(D(1),I=1,5),(D11(I),I=1,5),
1(F11(I),I=1,5),(F21(I),I=1,5),(F33(I),I=1,5),
1(R1(I),I=1,5),R2(I),P1(1),P2(1),P3(1),R1(I),I=1,5),
3(R2(I),I=1,5),(THETA(I),I=1,5),(R1(I),I=1,5),
3READ 1005,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,
1RUNNOO,RN,NS0,GNO,NSMAX,GNO,NSMAX,GNO,NSMAX,GNO,NSMAX,GNO,NSMAX,
1READ 1010,(XVELO(ML,NL),YVELO(ML,NL),ZVELO(ML,NL),ML=1,NQMAX,NL=1,
1NSMAX)

READ INITIAL DATA

READ 1000,(ALPHA(I),I=1,5),(D(1),I=1,5),(D11(I),I=1,5),
1(F11(I),I=1,5),(F21(I),I=1,5),(F33(I),I=1,5),
1(R1(I),I=1,5),R2(I),P1(1),P2(1),P3(1),R1(I),I=1,5),
3(R2(I),I=1,5),(THETA(I),I=1,5),(R1(I),I=1,5),
3READ 1005,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,DELTAP,
1RUNNOO,RN,NS0,GNO,NSMAX,GNO,NSMAX,GNO,NSMAX,GNO,NSMAX,
1READ 1010,(XVELO(ML,NL),YVELO(ML,NL),ZVELO(ML,NL),ML=1,NQMAX,NL=1,
1NSMAX)

CONVERT INPUT DATA TO PROPER FORM TO BE USED BY THE PROGRAM
CALL INCON(R1,R2,THETA,ALPHA,SL1,T,D10,
1B10,B20,B30,C10,C20,C30,N+1)

INITIALIZE PROGRAM CONSTANTS

DO 5 I=1,N
5 EPSEN(I)=F11(I)*F11(I)/(SK1*20.0)

Determine printout frequency of stored output

FACTOR=FINT/200.0
XPRINT=KPRINT
IF (FACTOR-(DELTAP*XPRINT))7,8,8
7 MULT=1

Figure 7-7. Main Program (Continued)
GO TO 15
8 IF(FACTOR-2.0*(DELTAT*XPRINT)) 9,12,12
9 MULT=2
GO TO 15
12 MULT=5

SET STARTING POINT IN VELOCITY ARRAY

15 NS=NSO
NQ=NQO

INITIALIZE PROGRAM VARIABLES

IFLAGE=-1
RUNNO=RUNNOO-1.0
PRXVEL=100.0
PRYVEL=100.0
PRZVEL=100.0
XMIN=100.0
20 NST=0
NUN=0
40 RUNNO=RUNNO+1.0
LINE=0

ZERO OUT LINE COUNT

KM=0

CHOOSE INITIAL TOUCHDOWN CONDITIONS

PITCH=PITCHO
YAW=YAWO
ROLL=ROLLO
PITCHV=PITCHVO
YAWV=YAWVO
ROLLV=ROLLVO
XVELOO=XVELOO(NQ,NS)
YVELOO=YVELOO(NQ,NS)
ZVELOO=ZVELOO(NQ,NS)
XVEL=XVELO(NQ,NS)
YVEL=YVELO(NQ,NS)
ZVEL=ZVELO(NQ,NS)

FORM INIT. DIRECTION COSINE MATRIX GROUND AXES TO VEHICLE AXES

XL(1,1)=COS(PITCH)*COS(ROLL)-SIN(PITCH)*SIN(YAW)*SIN(ROLL)
XL(2,1)=COS(PITCH)*SIN(YAW)*SIN(ROLL)+SIN(PITCH)*COS(ROLL)
XL(3,1)=-COS(YAW)*SIN(ROLL)
XL(1,2)=-SIN(PITCH)*COS(YAW)
XL(2,2)=COS(PITCH)*COS(YAW)
XL(3,2)=SIN(YAW)
XL(1,3)=COS(PITCH)*SIN(ROLL)+SIN(PITCH)*SIN(YAW)*COS(ROLL)
XL(2,3)=SIN(PITCH)*SIN(ROLL)-COS(PITCH)*SIN(YAW)*COS(ROLL)
XL(3,3)=COS(YAW)*COS(ROLL)

PRINT INPUT DATA

PRINT 1066
PRINT 1025,(ALPHA(I),I=1,5)
PRINT 1026,(D(I),I=1,5)
PRINT 1027,(D1(I),I=1,5)

Figure 7-7. Main Program (Continued)
Figure 7-7. Main Program (Continued)
Determine initial strut angles and trigonometric relationships

\[
B1(I) = B10(I)
\]
\[
\cos(I) = \cos(\theta(I))
\]
\[
\sin(I) = \sin(\theta(I))
\]
\[
S1(I) = S10(I)
\]
\[
B2PREF(I) = B20(I)
\]
\[
B3PREF(I) = B30(I)
\]
\[
C1PREF(I) = C10(I)
\]
\[
C2PREF(I) = C20(I)
\]
\[
C3PREF(I) = C30(I)
\]

Determine footpad positions and velocities

\[
\text{DO } 75 \ I = 1, N
\]
\[
XPV(I) = -D1(I) - S10(I) \times \cos(B10(I))
\]
\[
YPV(I) = R1(I) \times \cos(I) + S10(I) \times \sin(B10(I))
\]
\[
ZPV(I) = -R1(I) \times \sin(I) - D10(I)
\]
\[
XP(I) = XL(1,1) \times XPV(I) + XL(1,2) \times YPV(I) + XL(1,3) \times ZPV(I)
\]
\[
YP(I) = XL(2,1) \times XPV(I) + XL(2,2) \times YPV(I) + XL(2,3) \times ZPV(I)
\]
\[
ZP(I) = XL(3,1) \times XPV(I) + XL(3,2) \times YPV(I) + XL(3,3) \times ZPV(I)
\]

Do 78 I = 1, N

\[
\text{IF } (XP(I) - XMIN) \text{ THEN } 77, 78, 78
\]

77 XMIN = XP(I)

78 CONTINUE

\[
\text{DO } 79 \ I = 1, N
\]
\[
XP(I) = XP(I) - XMIN + \text{DELTAP } + 0.01
\]

Fix coordinates of vehicle C.G.

\[
x = -XMIN + \text{DELTAP}
\]
\[
y = 0.0
\]
\[
z = 0.0
\]
\[
XO = X
\]

\[
\text{DO } 82 \ I = 1, N
\]
\[
XPVEL(I) = -\text{PITCH} \times YPV(I) + \text{ROLL} \times ZPV(I)
\]
\[
YPVEL(I) = \text{PITCH} \times XPV(I) - \text{YAW} \times ZPV(I)
\]
\[
ZPVEL(I) = \text{YAW} \times YPV(I) - \text{ROLL} \times XPV(I)
\]
\[
XPVEL(1) = XL(1,1) \times XPVEL(I) + XL(1,2) \times YPVEL(I) + XL(1,3) \times ZPVEL(I)
\]
\[
YPVEL(1) = XL(2,1) \times XPVEL(I) + XL(2,2) \times YPVEL(I) + XL(2,3) \times ZPVEL(I)
\]
\[
ZPVEL(1) = XL(3,1) \times XPVEL(I) + XL(3,2) \times YPVEL(I) + XL(3,3) \times ZPVEL(I)
\]

Initialize program for strut length calculations

\[
\text{DO } 100 \ I = 1, N
\]
\[
NOGR(I) = 1
\]
\[
SL1PRE(I) = 0.0
\]
\[
SL2PRE(I) = 0.0
\]
\[
SL3PRE(I) = 0.0
\]
\[
SL1M(I) = F11(I) / SK1
\]
\[
SL2M(I) = F22(I) / SK2
\]
\[
SL3M(I) = F33(I) / SK3
\]
\[
TIME = 0.0
\]
\[
CON5 = 0.0
\]
\[
XMIN = 100.0
\]

Figure 7-7. Main Program (Continued)
BETMIN=10.0
I8EMIN=O
IFLAG=-1

DETERMINE TRIGONOMETRIC RELATIONSHIPS FOR STRUT LENGTH
CALCULATIONS.

DO 130 I=1,N
SIND(I)=SIN(B1(I))
COSD(I)=COS(B1(I))
COSG(I)=COS(THETA(I)-ALPHA(I)/2.0)
SINK(I)=SIN(THETA(I)+ALPHA(I)/2.0)

DETERMINE STRUT LENGTHS

SL1TO(I)=SQRT(SL1(I)*SL1(I)+D1(I)*D1(I))
HALF3=(SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))*(SL1(I)*
1SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))
HALF4=(SL1(I)*COSD(I)-D(I))*(SL1(I)*COSD(I)-D(I))
SL2(I)=SQRT(HALF3+HALF4)
SL2TO(I)=SQRT(SL2(I)*SL2(I)+(D1(I)+R1(I)*SINC(I)-R2(I)*SINJ(I))*
1(D1(I)+R1(I)*SINC(I)-R2(I)*SINJ(I)))
HALF1=(SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))*
1(SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))
HALF2=(SL1(I)*COSD(I)-D(I))*(SL1(I)*COSD(I)-D(I))
SL3(I)=SQRT(HALF1+HALF2)
SL3TO(I)=SQRT(SL3(I)*SL3(I)+(D1(I)-R2(I)*SINK(I)+R1(I)*SINC(I))*
1(D1(I)-R2(I)*SINK(I)+R1(I)*SINC(I)))
SL1T(I)=SL1TO(I)
SL2T(I)=SL2TO(I)
SL3T(I)=SL3TO(I)

DETERMINE STRUT ANGLES AND ASSIGN TO PROPER QUADRANTS

140 DO 240 I=1,N
IF(NGR(I))240,240,145
145 B2(I)=ATAN((SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))/(SL1(I)*
1COSD(I)-D(I)))
B2MI(I)=B2(I)-3.14159265359
B2PL(I)=B2(I)+3.14159265359
DIFF1=ABS(B2MI(I)-B2PREV(I))
DIFF2=ABS(B2PL(I)-B2PREV(I))
DIFF3=ABS(B2PL(I)-B2PREV(I))
IF(DIFF1-DIFF2)150,152,152
150 IF(DIFF1-DIFF3)154,158,158
152 IF(DIFF2-DIFF3)160,158,158
154 B2(I)=B2MI(I)
GO TO 160
158 B2(I)=B2PL(I)
B3(I)=ATAN((SL1(I)*SIND(I)+R1(I)*COSC(I)-R2(I)*COSG(I))/(SL1(I)*
1COSD(I)-D(I)))
B3MI(I)=B3(I)-3.14159265359
B3PL(I)=B3(I)+3.14159265359
DIFF1=ABS(B3MI(I)-B3PREV(I))
DIFF2=ABS(B3PL(I)-B3PREV(I))
DIFF3=ABS(B3PL(I)-B3PREV(I))
IF(DIFF1-DIFF2)170,172,172
170 IF(DIFF1-DIFF3)174,178,178

Figure 7-7. Main Program (Continued)

7-34
172 IF (DIFF2 - DIFF3) > 180.178 > 178
174 B3(I) = B3MI(I)
GO TO 180
178 B3(I) = B3PL(I)
180 B3PREV(I) = B3(I)
IF (D1(I)) > 182 > 181 > 182
181 C1(I) = 1.5708
GO TO 183
182 C1(I) = ATAN (SL1(I) / D1(I))
183 C1MI(I) = C1(I) - 3.14159265359
C1PL(I) = C1(I) + 3.14159265359
DIFF1 = ABS (C1MI(I) - C1PREV(I))
DIFF2 = ABS (C1MI(I) - C1PL(I))
DIFF3 = ABS (C1PL(I) - C1PREV(I))
IF (DIFF1 - DIFF2) > 190 > 192 > 192
190 IF (DIFF1 - DIFF3) > 194 > 198
192 IF (DIFF2 - DIFF3) > 200 > 198
194 C1(I) = C1MI(I)
GO TO 200
198 C1(I) = C1PL(I)
200 C1PREV(I) = C1(I)
C2(I) = ATAN (SL2(I) / (D1(I) + R1(I) * SINC(I) - R2(I) * SINJ(I)))
C2MI(I) = C2(I) - 3.14159265359
C2PL(I) = C2(I) + 3.14159265359
DIFF1 = ABS (C2MI(I) - C2PREV(I))
DIFF2 = ABS (C2MI(I) - C2PL(I))
DIFF3 = ABS (C2PL(I) - C2PREV(I))
IF (DIFF1 - DIFF2) > 210 > 212 > 212
210 IF (DIFF1 - DIFF3) > 214 > 218
212 IF (DIFF2 - DIFF3) > 220 > 218
214 C2(I) = C2MI(I)
GO TO 220
218 C2(I) = C2PL(I)
220 C2PREV(I) = C2(I)
C3(I) = ATAN (SL3(I) / (D1(I) + R1(I) * SINC(I) - R2(I) * SINJ(I)))
C3MI(I) = C3(I) - 3.14159265359
C3PL(I) = C3(I) + 3.14159265359
DIFF1 = ABS (C3MI(I) - C3PREV(I))
DIFF2 = ABS (C3MI(I) - C3PL(I))
DIFF3 = ABS (C3PL(I) - C3PREV(I))
IF (DIFF1 - DIFF2) > 230 > 232 > 232
230 IF (DIFF1 - DIFF3) > 234 > 236
232 IF (DIFF2 - DIFF3) > 238 > 236
234 C3(I) = C3MI(I)
GO TO 238
236 C3(I) = C3PL(I)
238 C3PREV(I) = C3(I)
240 CONTINUE
IF (IFLAG) > 250 > 250 > 280
STORE INFORMATION FOR FINAL PRINTOUT

M = 1
TIM(M) = TIME
XP1(M) = XP(I)
YP1(M) = YP(I)
ZP1(M) = ZP(I)
XP3(M) = XP(J)
YP3(M) = YP(J)
ZP3(M) = ZP(J)
FX1(M) = 0.0

Figure 7-7. Main Program (Continued)
FX3(M)=0.0
FXPL31(M)=FXPLG3(M)
FYPL31(M)=FYPLG3(M)
FZPL31(M)=FZPLG3(M)
FXPL33(M)=FXPLG3(JJ)
FYPL33(M)=FYPLG3(JJ)
FZPL33(M)=FZPLG3(JJ)
XP5(M)=XP(M)
YP5(M)=YP(M)
ZP5(M)=ZP(M)
XP7(M)=XP(M)
YP7(M)=YP(M)
ZP7(M)=ZP(M)
FX5(M)=0.0
FX7(M)=0.0
FXPL35(M)=FXPLG3(M)
FYPL35(M)=FYPLG3(M)
FZPL35(M)=FZPLG3(M)
FXPL37(M)=FXPLG3(JJ)
FYPL37(M)=FYPLG3(JJ)
FZPL37(M)=FZPLG3(JJ)

COMPUTE INITIAL STABILITY ANGLE

CALL STABAN(XVEL,YVEL,ZVEL,X,Y,Z,XP,YP,ZP,ZETA,N,BETAF,IFRON)
IF(TIME)275,275,277

PRINT INPUTTED ANGLES AS CALCULATED BY THE COMPUTER AT TIME = 0
AS CHECK ON DATA PROGRAMING

275 PRINT 940
PRINT 1084,B10(I),I=1,N
PRINT 1085,B20(I),I=1,N
PRINT 1086,B30(I),I=1,N
PRINT 1087,C10(I),I=1,N
PRINT 1088,C20(I),I=1,N
PRINT 1089,C30(I),I=1,N
PRINT 1090,SL10(I),I=1,N
PRINT 1091,D10(I),I=1,N
PRINT 1056
PRINT 1070,XP(I),I=1,N
PRINT 1071,YP(I),I=1,N
PRINT 1072,ZP(I),I=1,N
PRINT 1073,XPV(I),I=1,N
PRINT 1074,YPV(I),I=1,N
PRINT 1075,ZPV(I),I=1,N
PRINT 1056
PRINT 1076,XPVEL(I),I=1,N
PRINT 1077,YPVEL(I),I=1,N
PRINT 1078,ZPVEL(I),I=1,N
PRINT 1079,XPVEL(I),I=1,N
PRINT 1080,YPVEL(I),I=1,N
PRINT 1081,ZPVEL(I),I=1,N

PRINT ON LINE DATA

277 PRINT 1066
PRINT 1095,SERNO,RUNNO
PRINT 1096
PRINT 1097,TIME,X,Y,Z,XVEL,YVEL,ZVEL,PITCHV,ROLLV,YAWV,BETAF,IFRON

Figure 7-7. Main Program (Continued)
TEST IF FOOTPAD IS ON THE GROUND

IF(XP(I)-DELTAP < 330,330,290

COMPUTE FOOTPAD VELOCITY RELATIVE TO C,G AND DETERMINE RELATIVE ENERGY OF FOOTPAD

XPVVLR(I) = -PITCHV*YPV(I)+ROLLV*ZPV(I)
YPVVLR(I) = PITCHV*XPV(I)-YAWV*ZPV(I)
ZPVVR(I) = YAWV*YPV(I)-ROLLV*XPV(I)
PADENG(I) = PMASS/2.*((XPVVLR(I)-XPVEL(I)@2+(YPVVLR(I)-
YPVEL(I))@2+ZPVVR(I)-ZPVVEL(I)@2)+E1(I)+E2(I)+E3(I)

TEST IF FOOTPAD IS MOVING WITH THE VEHICLE

IF(PADENG(I)<EPSLEN(I)) 300,300,330

FOOTPAD IS MOVING WITH THE VEHICLE. SET FORCES EQUAL TO ZERO

FA(I) = 0.0
FB(I) = 0.0
FC(I) = 0.0
FAG(I) = 0.0
FBG(I) = 0.0
FCG(I) = 0.0
FXPLG1(I) = 0.0
FXPLG2(I) = 0.0
FXPLG3(I) = 0.0
FYPLG1(I) = 0.0
FYPLG2(I) = 0.0
FYPLG3(I) = 0.0
FZPLG1(I) = 0.0
FZPLG2(I) = 0.0
FZPLG3(I) = 0.0
RMOMGX = RMOMGX+PMASS*(YPV(I)@2+ZPV(I)@2)
RMOMGY = RMOMGY+PMASS*(XPV(I)@2+ZPV(I)@2)
RMOMGZ = RMOMGZ+PMASS*(XPV(I)@2+YPV(I)@2)
NOGR(I) = -1

FIND FOOTPAD POSITIONS AND VELOCITY BY RIGID BODY MOTION WITH THE VEHICLE

XPV(I) = -D11(I)-SL1(I)*COS(B1(I))
YPV(I) = R1(I)*COS(I)+SL1(I)*SIN(B1(I))
ZPV(I) = -R1(I)*SIN(I)-D11(I)
XPVEL(I) = -PITCHV*YPV(I)+ROLLV*ZPV(I)
YPVEL(I) = PITCHV*XPV(I)-YAWV*ZPV(I)
ZPVEL(I) = YAWV*YPV(I)-ROLLV*XPV(I)
X(I) = XL(1,1)*XPV(I)+XL(1,2)*Y(I)+XL(1,3)*ZPV(I)+X
Y(I) = XL(2,1)*XPV(I)+XL(2,2)*Y(I)+XL(2,3)*ZPV(I)+Y
Z(I) = XL(3,1)*XPV(I)+XL(3,2)*Y(I)+XL(3,3)*ZPV(I)+Z
XPVEL(I) = XL(1,1)*XPVEL(I)+XL(1,2)*YPVEL(I)+XL(1,3)*ZPVVEL(I)
+XVEL
YPVEL(I) = XL(2,1)*XPVEL(I)+XL(2,2)*YPVEL(I)+XL(2,3)*ZPVVEL(I)
+YVEL
ZPVVEL(I) = XL(3,1)*XPVEL(I)+XL(3,2)*YPVEL(I)+XL(3,3)*ZPVVEL(I)

Figure 7-7. Main Program (Continued)
FOOTPAD IS MOVING INDEPENDENTLY OF THE VEHICLE, CALCULATE STRUT FORCES AND FOOTPAD ENERGY

330 SL=SL1TO(I)-SL1T(I)
    NOGR(I)=1
    IF(SL-SL1M(I)+F1(I)/SK1)334,334,375
334 IF(SL)340,367,367
340 IF(SL-SL1PRE(I))343,360,360
343 F1(I)=SKE1*SL-P1(I)
    E1(I)=SKE1*SL/2.0
    GO TO 393
350 F1(I)=-P1(I)
    E1(I)=0.0
    GO TO 393
360 F1(I)=SKE1*SL+P1(I)
    E1(I)=SKE1*SL/2.0
    GO TO 393
367 IF(SL-SL1PRE(I))350,370,370
370 F1(I)=P1(I)
    E1(I)=0.0
    GO TO 393
375 IF(SL-SL1M(I))378,390,390
378 IF(SL-SL1PRE(I))385,380,380
380 F1(I)=SK1*(SL-SL1M(I)+F1(I)/SK1)+P1(I)
    E1(I)=(SK1/2.0)*(SL-SL1M(I)+F1(I)/SK1)+P1(I)
    GO TO 393
385 F1(I)=SK1*(SL-SL1M(I)+F1(I)/SK1)-P1(I)
    E1(I)=(SK1/2.0)*(SL-SL1M(I)+F1(I)/SK1)-P1(I)
    GO TO 393
390 F1(I)=F1(I)+P1(I)
    E1(I)=(SK1/2.0)*(SL-SL1M(I)+F1(I)/SK1)+P1(I)
393 IF(SL-SL1M(I))400,406,425
396 SL1M(I)=SL
400 SL1PRE(I)=SL
    SL=SL2TO(I)-SL2T(I)
    IF(SL-SL2M(I)+F22(I)/SK2)403,403,433
403 IF(SL)406,425,425
406 IF(SL-SL2PRE(I))410,420,420
410 F2(I)=SKE2*SL-P2(I)
    E2(I)=SKE2*SL/2.0
    GO TO 450
412 F2(I)=-P2(I)
    E2(I)=0.0
    GO TO 450
420 F2(I)=SKE2*SL+P2(I)
    E2(I)=SKE2*SL/2.0
    GO TO 450
425 IF(SL-SL2PRE(I))412,430,430
430 F2(I)=P2(I)
    E2(I)=0.0
    GO TO 450
433 IF(SL-SL2M(I))436,446,446
436 IF(SL-SL2PRE(I))443,440,440
440 F2(I)=SK2*(SL-SL2M(I)+F22(I)/SK2)+P2(I)
    E2(I)=(SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)+P2(I)
    GO TO 450
443 F2(I)=SK2*(SL-SL2M(I)+F22(I)/SK2)-P2(I)

Figure 7-7. Main Program (Continued)
E2(I) = (SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)*(SL-SL2M(I)+F22(I)/SK2)
GO TO 450
446 F2(I) = F22(I) + P2(I)
   E2(I) = (SK2/2.0)*(SL-SL2M(I)+F22(I)/SK2)*(SL-SL2M(I)+F22(I)/SK2)
450 IF (SL-SL2M(I)) 454
452 SL2M(I) = SL
454 SL2PRE(I) = SL
   SL = SL3TO(I) - SL3T(I)
   IF (SL-SL3M(I)+F33(I)/SK3) 458
458 IF (SL) 460
460 IF (SL-SL3PRE(I)) 465
465 F3(I) = SK3*SL-P3(I)
   E3(I) = SK3*SL/2.0
   GO TO 505
470 F3(I) = -P3(I)
   E3(I) = 0.0
   GO TO 505
475 F3(I) = SK3*SL+P3(I)
   E3(I) = SK3*SL/2.0
   GO TO 505
480 IF (SL-SL3PRE(I)) 483
483 F3(I) = P3(I)
   E3(I) = 0.0
   GO TO 505
486 IF (SL-SL3M(I)) 490
490 IF (SL-SL3PRE(I)) 496
493 F3(I) = SK3*(SL-SL3M(I)+F33(I)/SK3)+P3(I)
   E3(I) = SK3/2.0
   IF (SL-SL3M(I)+F33(I)/SK3) 498
   GO TO 505
496 F3(I) = SK3*(SL-SL3M(I)+F33(I)/SK3)-P3(I)
   E3(I) = SK3/2.0
   IF (SL-SL3M(I)+F33(I)/SK3) 498
   GO TO 505
500 F3(I) = F33(I)+P3(I)
   E3(I) = SK3/2.0
   IF (SL-SL3M(I)+F33(I)/SK3) 498
   GO TO 505
505 IF (SL-SL3M(I)) 510
507 SL3M(I) = SL
510 SL3PRE(I) = SL

CALCULATE COMPONENTS OF STRUT FORCE

SING(I) = SIN (C1(I))
SINL(I) = SIN (C2(I))
SINI(I) = SIN (C3(I))
FA(I) = F1(I)*COS (C1(I)) + F2(I)*COS (C2(I)) + F3(I)*COS (C3(I))
FB(I) = F1(I)*SING(I)*COSD(I) + F2(I)*SINL(I)*COS (B2(I)) + F3(I)*SINI(I)*COS (B3(I))
FC(I) = F1(I)*SING(I)*SIND(I) + F2(I)*SINL(I)*SIN (B2(I)) + F3(I)*SINI(I)*SIN (B3(I))
FBG(I) = -XL(1,1)*FB(I) + XL(1,2)*FC(I) - XL(1,3)*FA(I)
FCG(I) = -XL(2,1)*FB(I) + XL(2,2)*FC(I) - XL(2,3)*FA(I)
FAG(I) = XL(3,1)*FB(I) - XL(3,2)*FC(I) + XL(3,3)*FA(I)

DETERMINE FORCES ACTING ON FOOTPAD AND FOOTPAD MOTIONS

KCON = 0
515 KCON = KCON + 1
516 SLVEL = SQRT ((YPVEL(I))^2 + (ZPVEL(I))^2)
517 IF (SLVEL-EPS2) 530
530 SLVEL = EPS2
535 FXP = SKS(I) * (DELTAP-XP(I))
540 IF (FXP) 545

Figure 7-7. Main Program (Continued)
540 FX(I)=0.0
GO TO 550
545 FX(I)=FXP
550 TEMP2=GRDMU(I)*FX(I)/(PMASS*SLVEL)
  FYPLG3(I)=(FCG(I)/PMASS+GRAV*SIN (ZETA))*PMASS
  YP(I)=YP(I)+YPVEL(I)*DELTST+(19.0*FYPLG3(I)-10.0*FYPLG2(I)+5.0*FYPLG1(I))*
  1FYPLG1(I)*DELTST*DELTST/(24.0*PMASS)-TEMP2*YPVEL(I)*DELTST*
  2DELTST/2.0
  YPVEL(I)=YPVEL(I)+(23.0*FYPLG3(I)-16.0*FYPLG2(I)+5.0*FYPLG1(I))*
  1DELTST/(12.0*PMASS)-TEMP2*YPVEL(I)*DELTST
  FZPLG3(I)=FAG(I)
  ZP(I)=ZP(I)+ZPVEL(I)*DELTST+(19.0*FZPLG3(I)-10.0*FZPLG2(I)+3.0*FZPLG1(I))*
  1FZPLG1(I)*DELTST*DELTST/(24.0*PMASS)-TEMP2*ZPVEL(I)*DELTST*
  2DELTST/2.0
  ZPVEL(I)=ZPVEL(I)+(23.0*FZPLG3(I)-16.0*FZPLG2(I)+5.0*FZPLG1(I))*
  1DELTST/(12.0*PMASS)-TEMP2*ZPVEL(I)*DELTST
  XPVEL(I)=XPVEL(I)+(23.0*FXPLG3(I)-16.0*FXPLG2(I)+5.0*FXPLG1(I))*
  1DELTST/(12.0*PMASS)-TEMP2*XPVEL(I)*DELTST
  XPVEL(I)=XPVEL(I)+(23.0*FXPLG3(I)-16.0*FXPLG2(I)+5.0*FXPLG1(I))*
  1DELTST/(12.0*PMASS)+FX(I)*DELTST/PMASS
  XPVEL(I)=XL(I)+XPVEL(I)-X+XL(I)*SL(I)*SIN (B(I))+XI(I)*COSC(I)+FC(I)*
  1(R(I)*SINC(I)+D1(I))
  VEMX=VEMX-FA(I)*(SL(I)*SIN (B(I))+R(I)*COSC(I)+FC(I)*
  1(R(I)*SINC(I)+D1(I))
  VEMY=VEMY-FA(I)*(SL(I)*COS (B(I))+D1(I))+FB(I)*
  1(R(I)*SINC(I)+D1(I))
610 VEMZ=VEMZ+FB(I)*(SL(I)*COSC(I)+SL(I)*SIN (B(I)))-FC(I)*
  1(SL(I)*COS (B(I))+D1(I))
  RMOMX=XMOMX+RMOMGX
  RMOMY=XMOMY+RMOMGY
  RMOMZ=XMOMZ+RMOMGZ
  RMOMGX=0.0
  RMOMGY=0.0
  RMOMGZ=0.0
C C
Determine forces acting on the vehicle

GRFX=-FBG(1)-FBG(2)-FBG(3)-FBG(4)-VMASS*GRAV*COS (ZETA)
GRFY=-FCG(1)-FCG(2)-FCG(3)-FCG(4)+VMASS*GRAV*SIN (ZETA)
GRFZ= FAG(1)+FAG(2)+FAG(3)+FAG(4)
VEMX=0.0
VEMY=0.0
VEMZ=0.0
DO 610 I=1,N
  VEMX=VEMX-FA(I)*(SL(I)*SIN (B(I))+R(I)*COSC(I)+FC(I)*
  1(R(I)*SINC(I)+D1(I))
  VEMY=VEMY-FA(I)*(SL(I)*COS (B(I))+D1(I))+FB(I)*
  1(R(I)*SINC(I)+D1(I))
610 VEMZ=VEMZ+FB(I)*(SL(I)*COSC(I)+SL(I)*SIN (B(I)))-FC(I)*
  1(SL(I)*COS (B(I))+D1(I))
  RMOMX=XMOMX+RMOMGX
  RMOMY=XMOMY+RMOMGY
  RMOMZ=XMOMZ+RMOMGZ
  RMOMGX=0.0
  RMOMGY=0.0
  RMOMGZ=0.0
C C
Determine vehicle C.G. motions

PITCHV=-PITCHV
YAWV=-YAWV
ROLLV=-ROLLV

Figure 7-7. Main Program (Continued)
CALL INTEQMV(VEMX, VEMY, VEMZ, YAWV, ROLLV, PITCHV, DELTAT, XVEL, GRFX, 
YVEL, GRFY, ZVEL, GRFZ, XL, VMAS, RMOMX, RMOMY, RMOMZ)
PITCHV = -PITCHV
YAWV = -YAWV
ROLLV = -ROLLV

INDEX PREVIOUS FORCES FOR THE NEXT INTEGRATION OF FOOTPAD
MOTION

DO 639 I = 1, N
   IF(NOGR(I)) 639, 639, 638
       FXPLG1(I) = FXPLG2(I)
       FXPLG2(I) = FXPLG3(I)
       FYPLG1(I) = FYPLG2(I)
       FYPLG2(I) = FYPLG3(I)
       FZPLG1(I) = FZPLG2(I)
       FZPLG2(I) = FZPLG3(I)
   CONTINUE
638

INCREMENT TIME FOR NEXT CALCULATION
TIME = TIME + DELTAT
CONS = CONS + 1.0

TEST NOZZLE = GROUND CLEARANCE AND STORE MINIMUM VALUE

XN = X - HN*XL(1,1) - RN*SQRT((1.0 - XL(1,1)*XL(1,1))
   IF(XN - XNMIN) 650, 651, 651
       XNMIN = XN
       TMINXN = TIME - DELTAT
   IF(TIME - 2.0*DELTAT) 653, 653, 651
650

TEST STABILITY ANGLE AND STORE MINIMUM VALUE

IF(BETAF - BETMIN) 652, 653, 653
   BETMIN = BETAF
   TIMBMI = TIME - DELTAT
   IBEMIN = IFRONT
651

TEST IF VEHICLE IS STOPPED

IF(ABS(XVEL) - EPS3) 654, 654, 668
   IF(ABS(YVEL) - EPS4) 655, 655, 668
   IF(ABS(ZVEL) - EPS4) 656, 656, 668
   IF(ABS(YAWV) - EPS5) 657, 657, 668
   IF(ABS(ROLLV) - EPS5) 658, 658, 668
653

DO 665 I = 1, N
   IF(XP(I) - 1.0) 665, 665, 668
666

CONTINUE
GO TO 810

COMPUTE STABILITY ANGLE

668

CONTINUE
CALL STABAN(XVEL, YVEL, ZVEL, X, Y, Z, XP, YP, ZP, ZETA, N, BETAF, IFRONT)

TEST IF VEHICLE IS UNSTABLE

IF(BETAF) 770, 674, 674
674

IF(TIME - FINT) 675, 800, 800

Figure 7-7. Main Program (Continued)
SET LINE COUNT AND STORAGE FOR PLOT ROUTINE

675 KM=KM+1
IF(CONS-KPRINT) 710, 680, 680
680 CONS=0.0
687 IF(KM-MULT*KPRINT) 700, 688, 688

STORE VARIABLES FOR THE PRINT TO BE MADE AT THE END OF THE RUN

688 M=M+1
TIM(M)=TIME
XP3(M)=XP(JJ)
YP3(M)=YP(JJ)
ZP3(M)=ZP(JJ)
XP1(M)=XP(II)
YP1(M)=YP(II)
ZP1(M)=ZP(II)
FX1(M)=FX(II)
FX3(M)=FX(JJ)
FXPL33(M)=FXPLG3(JJ)
FYPL33(M)=FYPLG3(JJ)
FZPL33(M)=FZPLG3(JJ)
FXPL31(M)=FXPLG3(II)
FYPL31(M)=FYPLG3(II)
FZPL31(M)=FZPLG3(II)
XP5(M)=XP(IIX)
YP5(M)=YP(IIX)
ZP5(M)=ZP(IIX)
XP7(M)=XP(JJX)
YP7(M)=YP(JJX)
ZP7(M)=ZP(JJX)
FX5(M)=0.0
FX7(M)=0.0
FXPL35(M)=FXPLG3(IIX)
FYPL35(M)=FYPLG3(IIX)
FZPL35(M)=FZPLG3(IIX)
FXPL37(M)=FXPLG3(JJX)
FYPL37(M)=FYPLG3(JJX)
FZPL37(M)=FZPLG3(JJX)
KM=0

RECORD LINE COUNT FOR PRINT HEADINGS

700 LINE=LINE+1
IF(LINE-49) 705, 705, 707
705 PRINT 1097, TIME, X, Y, Z, VEL, VEL, VEL, PITCH, ROLL, YAW, BETA,
1 IFront
GO TO 710
707 PRINT 1066
PRINT 1096
PRINT 1097, TIME, X, Y, Z, VEL, VEL, VEL, PITCH, ROLL, YAW, BETA,
1 IFront
LINE=0
710 DO 730 I=1, N
IF(NOGR(I)) 730, 730, 715

IF FOOTPAD IS ON THE GROUND, DETERMINE STRUT GEOMETRY

715 SL1(I)=SQRT((YPV(I)-R1(I)*COSC(I))*2+(XPV(I)+D11(I))*2)
B1(I)=ATAN((YPV(I)-R1(I)*COSC(I))/(-D11(I)-XPV(I)))

Figure 7-7. Main Program (Continued)
RESET PROGRAM CONSTANTS

\[
\begin{align*}
\text{COSD}(I) &= \text{COS} \left( \text{B1}(I) \right) \\
\text{SIND}(I) &= \text{SIN} \left( \text{B1}(I) \right)
\end{align*}
\]

CALCULATE NEW STRUT LENGTHS

\[
\begin{align*}
D1(I) &= -ZP(V)(I) - R1(I) * \text{SINC}(I) \\
SL1T(I) &= \sqrt{SL1[I] * SL1[I] + D1[I] * D1[I]} \\
\text{HALF3} &= (SL1[I] * \text{SIND}(I) + R1[I] * \text{COSC}(I) - R2[I] * \text{COSG}(I)) * (SL1[I] \\
\text{SIND}(I) + R1[I] * \text{COSC}(I) - R2[I] * \text{COSG}(I)) \\
\text{HALF4} &= (SL1[I] * \text{COSD}(I) - D(I)) * (SL1[I] * \text{COSD}(I) - D(I)) \\
SL2(I) &= \sqrt{\text{HALF3} + \text{HALF4}} \\
SL2T(I) &= \sqrt{SL2[I] * SL2[I] + (D1[I] + R1[I] * \text{SINC}(I) - R2[I] * \text{SINJ}(I)) * (D1[I] + R1[I] * \text{SINC}(I) - R2[I] * \text{SINJ}(I))} \\
\text{HALF1} &= (SL1[I] * \text{SIND}(I) + R1[I] * \text{COSC}(I) - R2[I] * \text{COSG}(I)) * (SL1[I] * \text{SIND}(I) + R1[I] * \text{COSC}(I) - R2[I] * \text{COSG}(I)) \\
\text{HALF2} &= (SL1[I] * \text{COSD}(I) - D(I)) * (SL1[I] * \text{COSD}(I) - D(I)) \\
SL3(I) &= \sqrt{\text{HALF1} + \text{HALF2}} \\
\text{SL3T}(I) &= \sqrt{SL3[I] * SL3[I] + (D1[I] - R2[I] * \text{SINK}(I) + R1[I] * \text{SINC}(I)) * (D1[I] - R2[I] * \text{SINK}(I) + R1[I] * \text{SINC}(I))}
\end{align*}
\]

CONTINUE

\[
\text{IFLAG} = 1 \\
\text{GO TO 140}
\]

VEHICLE IS UNSTABLE PRINT OUTPUT DATA

\[
\text{IFLAGX} = -1 \\
\text{PRINT 1066} \\
\text{PRINT 1095, SERNO, RUNNO} \\
\text{PRINT 905, TIME} \\
\text{ASSIGN 785 TO MM} \\
\text{GO TO 820}
\]

CHOOSE NEW VELOCITY CONDITIONS FROM INPUTED ARRAY FOR THE NEXT RUN

\[
\begin{align*}
\text{IF (NST) 790, 790, 796} \\
\text{NUN} &= 1 \\
\text{NO} &= \text{NO} - 1 \\
\text{IF (NO) 794, 794, 40} \\
\text{NO} &= \text{NO} + 1 \\
\text{NS} &= \text{NS} + 1 \\
\text{IF (NS - NSMAX) 20, 20, 840} \\
\text{IFLAGX} &= 0 \\
\text{PRINT 1066} \\
\text{PRINT 1095, SERNO, RUNNO} \\
\text{PRINT 906, TIME} \\
\text{GO TO 819}
\end{align*}
\]

VEHICLE IS STABLE AND HAS STOPPED PRINT OUTPUT DATA

\[
\text{IFLAGX} = 1 \\
\text{PRINT 1066} \\
\text{PRINT 1095, SERNO, RUNNO} \\
\text{PRINT 904, TIME} \\
\text{ASSIGN 824 TO MM}
\]

PRINT SUMMARY OUTPUT

Figure 7-7. Main Program (Continued)
820  PRINT 907,(SL1M(I),I=1,N)
     PRINT 908,(SL2M(I),I=1,N)
     PRINT 909,(SL3M(I),I=1,N)
     PRINT 910,XMIN,TMIN*XN
     PRINT 936,BE_MIN,BEMP,TIMBI
     EGBAL1=VMASS*2.0*(XVELOO**2-YVEL**2-YVEL**2+ZVELOO**2
     1-ZVEL**2)+XMOMX/2.0*(YAWVO**2-YAWV**2)+XMOMY/2.0*(ROLLVO**2
     2-ROLLV**2)+XMOMZ/2.0*(PITCHV**2-PITCHV**2)
     EGBAL2=EGBAL1+((XO-X)*COS(ZETA)+Y*SSIN(ZETA))*VMASS*GRAV
     EGBAL2=EGBAL2+ENGY(1)
     EGBAL3=EGBAL3+((F11(I)+P1(I))*(SL1M(I)-F11(I)/SK1)+
     (F22(I)+P2(I))*(SL2M(I)-F22(I)/SK2)+(F33(I)+P3(I))*(SL3M(I)-F33(I)/SK3))
     822  EGBAL3=EGBAL3+(F22(I)+P2(I))*(SL2M(I)-F22(I)/(2.0*SK2))+
     (F33(I)+P3(I))*(SL3M(I)-F33(I)/(2.0*SK3))
     DO 2822 I=1,N
     ENGY(1)=ENGY(1)+100.0
     2822  PRINT 917,EGBAL1
     PRINT 918,EGBAL2
     PRINT 934,EGBAL3
     IF(FRONT,BETA)=1
     IP=1
     IQ=50
     IR=-1
     IF(M-IQ)1823,1825,1825
     IQ=M
     IR=I
     IF(IR)1827,1830,1830
     IP=IP+50
     IQ=IQ+50
     GO TO 1822
1825  IP=IP+50
     IQ=IQ+50
     GO TO 1822
1826  IP=1
     IQ=50
     IR=-1
     IF(M-IQ)1828,1829,1829
1827  IP=IP+50
     IQ=IQ+50
     GO TO 1822
1828  IP=1
     IQ=50
     IR=-1
1829  PRINT 1066
     PRINT 931,TIM(I),XP1(I),YP1(I),ZP1(I),XP3(I),YP3(I),ZP3(I),
     IF(IR)1830,1831,1831
     IP=IP+50
     IQ=IQ+50
     GO TO 1827
1830  IP=IP+50
     IQ=IQ+50
     GO TO 1827

C
C PLOTS MAY BE INSERTED HERE

Figure 7-7. Main Program (Continued)
IF (IFLAGE) 1905, 1935, 1935
1905 IFLAG = 1
II = II X
JJ = JJ X
DO 1910 I = J, M
XPI (1) = XP5 (I)
YP1 (1) = YP5 (I)
ZP1 (1) = ZP5 (I)
XP3 (1) = XP7 (I)
YP3 (1) = YP7 (I)
ZP3 (1) = ZP7 (I)
FX (1) = FX5 (I)
FX3 (1) = FX7 (I)
FXPL31 (1) = FXPL35 (I)
FYPL31 (1) = FYPL35 (I)
FZPL31 (1) = FZPL35 (I)
FXPL33 (1) = FXPL37 (I)
FYPL33 (1) = FYPL37 (I)
1910 FZPL33 (1) = FZPL37 (I)
IF (IFLAGX) 770, 800, 810
1935 GO TO MM

CHOOSE NEW VELOCITY CONDITIONS FROM INPUTED ARRAY FOR THE NEXT RUN.

824 IF (NUX) 825, 825, 796
825 NST = 1
NO = NO + 1
IF (NO - NQMAX) 140, 40, 830
830 NO = NO - 1
NS = NS + 1
IF (NS - NSMAX) 833, 833, 840
833 IF (NO - NQMAX) 20, 20, 836
836 NO = NO - 1
GO TO 833
840 PRINT 1066
GO TO 1

REMEMBER INITIAL STARTING VELOCITIES TO PREVENT DUPLICATE Runs BECAUSE OF INPUT ERRORS

845 PRINT 1066
PRINT 935
STOP

FORMAT STATEMENTS FOR OUTPUT INFORMATION

904 FORMAT (11H) "STABLE, F7.3, 5HSECS, ///")
905 FORMAT (13H) "UNSTABLE, F7.3, 5HSECS, ///")
906 FORMAT (9H) "TIME, F7.3, 5HSECS, ///")
907 FORMAT (32H) "MAXIMUM STROKE NO. 1 STRUT, F12.3, ///")
908 FORMAT (32H) "NO. 2 STRUT, F12.3, ///")
909 FORMAT (32H) "NO. 3 STRUT, F12.3, ///")
910 FORMAT (///32H) "MINIMUM CLEARANCE OF NOZZLE = F7.3, 5H"
1TIME WHEN THE MINIMUM CLEARANCE OCCURS = F7.3, ///")
911 FORMAT (54H) "MINIMUM CLEARANCE BETWEEN SHOCK STRUT AND FRAME =")

Figure 7-7. Main Program (Continued)
912 FORMAT(32H*,5F12.3///)
914 FORMAT(32H MAXIMUM STROKE NO. 1 STRUT,*4F12.3///)
915 FORMAT(32H NO. 2 STRUT,*4F12.3///)
916 FORMAT(32H NO. 3 STRUT,*4F12.3///)
917 FORMAT(66H ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND IC. DROP=*F11.3///)
918 FORMAT(48H ENERGY DISSIPATED BASED ON PLASTIC STROKE =*F11.3//1/)  
920 FORMAT(21H TIME XP(*I1,11H) YP(*I1,11H) ZP(*I1,11H) 
FX(*I1,11H) F(*I1,11H)///)
921 FORMAT(7F12.3,2F14.1)
922 FORMAT(24H TIME FXPLG3(*I1,13H) FYPLG3(*I1,13H) 
FXPLG3(*I1,13H) FYPLG3(*I1,13H) FZPLG 
13(*I1,1H)/)
923 FORMAT(F12.3,6F14.1)
924 FORMAT(65H ENERGY DISSIPATED BASED ON PLASTIC AND FULL ELASTIC 
1 STROKE =*F11.3///)
925 FORMAT(98H X AND Y VELOCITIES ARE IDENTICAL TO THOSE OF THE PREVI 
OUS RUN. CHECK THE INPUT DATA FOR ERRORS )
926 FORMAT(29H MINIMUM STABILITY ANGLE B*I1,1H=*F7.3,55H 
1 TIME WHEN THIS STABILITY ANGLE OCCURS =*F7.3///)
927 FORMAT(27H FINAL STABILITY ANGLE B*I1,1H=*F7.3///)
928 FORMAT(48H ENERGY DISTRIBUTION BETWEEN LEGS - PERCENT =*F5F8.3// 
1/)
929 FORMAT(///26H1 CALCULATED INPUT VALUES///)
930 FORMAT(10X,S12.5)
931 FORMAT(IOX,3F12.5)
932 FORMAT(IOX,6110)
933 FORMAT(6F10.3)
934 FORMAT(65H0 INIT. DIRECTION COSINE MATRIX GROUND AXES TO VEHIC 
1LE AXES/)
935 FORMAT(3F20.8)
936 FORMAT(10X,10HALPHA =,5F15.5)
937 FORMAT(10X,10HD =,5F15.5)
938 FORMAT(10X,10HD11 =,5F15.5)
939 FORMAT(10X,10HF11 =,5F15.5)
940 FORMAT(10X,10HF22 =,5F15.5)
941 FORMAT(10X,10HF33 =,5F15.5)
942 FORMAT(10X,10GRDMU =,5F15.5)
943 FORMAT(10X,10HP1 =,5F15.5)
944 FORMAT(10X,10HP2 =,5F15.5)
945 FORMAT(10X,10HP3 =,5F15.5)
946 FORMAT(10X,10HR1 =,5F15.5)
947 FORMAT(10X,10HR2 =,5F15.5)
948 FORMAT(10X,10HRP =,5F15.5)
949 FORMAT(10X,10HTHETA =,5F15.5)
950 FORMAT(10X,10HSKS =,5F15.5)
951 FORMAT(10X,10HDELTAP =,F15.5,10H DELTAT =,F15.5,10H EPS2 =,F15.5)
952 FORMAT(10X,10HEPS3 =,F15.5,10H EPS4 =,F15.5,10H EPS5 =,F15.5)
953 FORMAT(10X,10HFINT =,F15.5,10H GRAV =,F15.5,10H HN =,F15.5)
954 FORMAT(10X,10HPMASS =,F15.5,10H RUNNOG =,F15.5,10H RN =,F15.5)
955 FORMAT(10X,10HSK1 =,F15.5,10H SK2 =,F15.5,10H SK3 =,F15.5)
956 FORMAT(10X,10HSKE1 =,F15.5,10H SKE2 =,F15.5,10H SKE3 =,F15.5)
957 FORMAT(10X,10HSERNO =,F15.5,10H VMASS =,F15.5,10H ZETA =,F15.5)

Figure 7-7. Main Program (Continued)
Figure 7-7. Main Program (Concluded)
**TITLE**
INCON  SUBROUTINE INCON

**AUTHOR**
J. GIBSON  BENDIX PRODUCTS AEROSPACE DIVISION

**DATE**
NOV. 1964

**PURPOSE**
THIS SUBROUTINE CONVERTS THE INPUT DATA TO THE PROPER
FORM FOR USE IN THE MAIN PROGRAM

**CALL**
CALL INCON

**NOTE**
THIS PROGRAM WAS WRITTEN IN FORTRAN IV

**NOTE**
THIS PROGRAM WAS COMPILED ON THE UNIVAC 1107 DIGITAL
COMPUTER

**INPUT**
BY EQUIVALENCE TO COMMON

**SYMBOL**
DEFINITION

**ALPHA**
ANGLE (IN PLANE PERPENDICULAR TO THE VEHICLE CENTERLINE)
SUBTENDED BY THE LOWER HARDPOINTS AND VEHICLE C.G.

**D**
VERTICAL DISTANCE BETWEEN UPPER AND LOWER HARDPOINTS

**DELTAP**
DISTANCE FROM BOTTOM OF FOOTPAD TO INTERSECTION OF THE
LEG STRUTS

**D11**
VERTICAL DISTANCE FROM VEHICLE C.G. TO UPPER HARDPOINT

**H**
DISTANCE FROM THE BOTTOM OF THE FOOTPAD TO THE VEHICLE
CENTER OF GRAVITY

**RP(I)**
RADIUS OF FOOTPAD (I)

**R1**
RADIUS OF UPPER HARDPOINT MOUNTING CIRCLE

**R2**
RADIUS OF LOWER HARDPOINT MOUNTING CIRCLE

**THETA**
ANGLE BETWEEN PLANE PARALLEL TO VEHICLE CENTERLINE IN
DIRECTION OF VEHICLE MOTION AND PLANE THROUGH VEHICLE
CENTERLINE AND UPPER HARDPOINT

**OUTPUT**
THROUGH Labeled COMMON

**SYMBOL**
DEFINITION

**B10**
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 1 AND VEHICLE CENTERLINE

**B20**
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 2 AND VEHICLE CENTERLINE

**B30**
ANGLE (PROJECTED IN PLANE PARALLEL TO VEHICLE MOTION)
BETWEEN STRUT NO. 3 AND VEHICLE CENTERLINE

**C10,C20,C30**
ANGLE IN PLANE FORMED BY STRUT AND A NORMAL TO
THE DIRECTION OF MOTION, BETWEEN STRUT AND A PLANE NORMAL
TO THE VEHICLE CENTERLINE - FOR STRUTS 1,2,3 RESPECTIVELY

**D10**
INITIAL LENGTH (PROJECTED IN PLANE NORMAL TO DIRECTION
OF VEHICLE MOTION) OF THE COMPONENT (IN PLANE PERPENDICULAR
TO THE VEHICLE CENTERLINE) OF STRUT NO 1 LENGTH

**SL10**
INITIAL VALUE OF SL1

**SUBROUTINE INCON(R1,R2,RP,H,D11,D,DELTAP,THETA,ALPHA,SL10,D10,**
**B10,B20,B30,C10,C20,C30,N,K)
DIMENSION R1(5),R2(5),RP(5),THETA(5),ALPHA(5),SL10(5),D10(5),B10(5,**
**1,B20(5),B30(5),C10(5),C20(5),C30(5),D11(5),D(5),C1ANG(5),**
**1,C2ANG(5),C3ANG(5))**

Figure 7-8. Subroutine INCON
I

1 DO 100 I=1,N+K
   SL10(I)=SQRT((H-DELTAP-D11(I))*((RP(I)-R1(I)))*COS(THETA(I)))*COS(THETA(I))
1COS(THETA(I)))*((RP(I)-R1(I))*COS(THETA(I)))
   D10(I)=(RP(I)-R1(I))*SIN(THETA(I))
   B10(I)=ATAN((RP(I)-R1(I))*COS(THETA(I))/(H-DELTAP-D11(I)))
   B20(I)=ATAN((RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)-ALPHA(I)/2.0)
1))/(H-DELTAP-D11(I)-D(I))
   B30(I)=ATAN((RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)+ALPHA(I)/2.0)
1))/(H-DELTAP-D11(I)-D(I))
   IF (DIO(I)) 10,11,10
10 C1ANG(I)=ATAN(ABS(SL10(I)/D10(I)))
11 S2X=RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)-ALPHA(I)/2.0)
   SL200=SQRT(S2X*S2X+(H-DELTAP-D11(I)-D(I))*((RP(I)-R1(I)))*COS(THETA(I))))
   C2ANG(I)=ATAN(ABS(SL200/(RP(I)*SIN(THETA(I)))-R2(I)*SIN(THETA(I)-1))
   1ALPHA(I)/2.0)))
   S3X=RP(I)*COS(THETA(I))-R2(I)*COS(THETA(I)+ALPHA(I)/2.0)
   SL300=SQRT(S3X*S3X+(H-DELTAP-D11(I)-D(I))*((RP(I)-R1(I)))*COS(THETA(I))))
   C3ANG(I)=ATAN(ABS(SL300/(RP(I)*SIN(THETA(I))-R2(I)*SIN(THETA(I)+1))
   1ALPHA(I)/2.0)))
64 IF (THETA(I)) 68,70,72
68 C10(I)=3.14159265359-C1ANG(I)
   GO TO 76
70 C10(I)=1.5707963
   GO TO 76
72 C10(I)=C1ANG(I)
76 IF (RP(I)*SIN(THETA(I))-R2(I)*SIN(THETA(I)-ALPHA(I)/2.0)) 80,82,84
80 C20(I)=3.14159265359-C2ANG(I)
   GO TO 88
82 C20(I)=1.5707963
   GO TO 88
84 C20(I)=C2ANG(I)
88 IF (RP(I)*SIN(THETA(I))-R2(I)*SIN(THETA(I)+ALPHA(I)/2.0)) 92,94,96
92 C30(I)=3.14159265359-C3ANG(I)
   GO TO 100
94 C30(I)=1.5707963
   GO TO 100
96 C30(I)=C3ANG(I)
100 CONTINUE
   RETURN
END

Figure 7-8. Subroutine INCON (Concluded)
SUBROUTINE STABAN(XVEL,YVEL,ZVEL,X,Y,Z,XP,YP,ZP,ZETA,N,BETAF,IFront)
DIMENSION XP(5),YP(5),ZP(5),XPM(5),YPM(5)

YVELGV=XVEL*SIN(ZETA)+YVEL*COS(ZETA)
ZVELGV=ZVEL

PHIPR=ATAN(ZVELGV/YVELGV)
YM=X*SIN(ZETA)*COS(PHIPR)+Y*COS(ZETA)*COS(PHIPR)+Z*SIN(PHIPR)

DO 10 IX=1,N
   10 YPM(IX)=XP(IX)*SIN(ZETA)*COS(PHIPR)+YP(IX)*COS(ZETA)*COS(PHIPR)+ZP(IX)*SIN(PHIPR)

FIND LARGEST VALUE OF YPM(I)

IFRON=-1,N
   DO 20 IX=2,N
      IF(YPM(IX)-YPM(IFRON)) 20*20_18
   20 CONTINUE

Figure 7-9. Subroutine STABAN
XM = X * COS(ZETA) - Y * SIN(ZETA)
XPM(IFRONT) = XP(IFRONT) * COS(ZETA) - YP(IFRONT) * SIN(ZETA)
BETAF = ATAN((YPM(IFRONT) - YM) / (XM - XPM(IFRONT)))
RETURN
END
Title: INTEQM

Subroutine INTEQM

Author: R. Dix

Bendix Products Aerospace Division

Date: Nov., 1964

Purpose: This subroutine integrates the equations of motion

Call: Call INTEQM

Note: This program was written in Fortran IV

Note: This program was compiled on the Univac 1107 Digital Computer

Input: By equivalence to common

Symbol Definition

G1 Torque about the X axis of the vehicle
    (Ground coordinate system)

G2 Torque about the Y axis of the vehicle
    (Ground coordinate system)

G3 Torque about the Z axis of the vehicle
    (Ground coordinate system)

Q1 Total moment of inertia (vehicle and footpads) about X axis

Q2 Total moment of inertia (vehicle and footpads) about Y axis

Q3 Total moment of inertia (vehicle and footpads) about Z axis

Xmass Vehicle mass

Output: By equivalence to common

Symbol Definition

DT Time interval between program calculations

ForceX Force acting in the X direction on the vehicle C.G.
    (Ground coordinate system)

ForceY Force acting in the Y direction on the vehicle C.G.
    (Ground coordinate system)

ForceZ Force acting in the Z direction on the vehicle C.G.
    (Ground coordinate system)

W1E Instantaneous Yaw velocity of the vehicle

W2E Instantaneous Roll velocity of the vehicle

W3E Instantaneous Pitch velocity of the vehicle

XCOS(I,J) Set of direction cosines relating vehicle coordinate
    system to fixed coordinate system

XCG Instantaneous X position of the vehicle C.G.

XDCG Instantaneous X velocity of the vehicle C.G.

YCG Instantaneous Y position of the vehicle C.G.

YDCG Instantaneous Y velocity of the vehicle C.G.

ZCG Instantaneous Z position of the vehicle C.G.

ZDCG Instantaneous Z velocity of the vehicle C.G.

Subroutine INTEQM(G1,G2,G3,W1E,W2E,W3E,DT,XCG,XDCG,FORCEX,
    FORCEY,FORCEZ,XCOS,XM ASS,Q1,Q2,Q3)

Integrate equations of motion

Dimension BRKT(3,3),VID(3,3),VIO(3,3),VIN(3,3),XN(3,3),XCOS(3,3)

Integrate equations

Figure 7-10. Subroutine INTEQM
Figure 7-10. Subroutine INTEQM (Concluded)
<table>
<thead>
<tr>
<th>ALPHA</th>
<th>1.49200</th>
<th>1.49200</th>
<th>1.49200</th>
<th>1.49200</th>
<th>1.49200</th>
<th>0.00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>5.25000</td>
<td>5.25000</td>
<td>5.25000</td>
<td>5.25000</td>
<td>5.25000</td>
<td>0.00000</td>
</tr>
<tr>
<td>D11</td>
<td>1.60000</td>
<td>1.60000</td>
<td>1.60000</td>
<td>1.60000</td>
<td>1.60000</td>
<td>0.00000</td>
</tr>
<tr>
<td>F11</td>
<td>5411.00000</td>
<td>5411.00000</td>
<td>5411.00000</td>
<td>5411.00000</td>
<td>5411.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>F22</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>F33</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>1737.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>GMU</td>
<td>2.00000</td>
<td>2.00000</td>
<td>2.00000</td>
<td>2.00000</td>
<td>2.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P1</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P2</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P3</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P1</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P2</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>6.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>RP</td>
<td>12.00000</td>
<td>12.00000</td>
<td>12.00000</td>
<td>12.00000</td>
<td>12.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>THETA</td>
<td>0.00000</td>
<td>1.57000</td>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
</tr>
<tr>
<td>SKS</td>
<td>125000.00000</td>
<td>125000.00000</td>
<td>125000.00000</td>
<td>125000.00000</td>
<td>125000.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>DELTAP</td>
<td>0.25000</td>
<td>DELTAT =</td>
<td>0.02000</td>
<td>FPS2 =</td>
<td>0.50000</td>
<td>0.00000</td>
</tr>
<tr>
<td>FPS3</td>
<td>0.40000</td>
<td>FPS4 =</td>
<td>0.40000</td>
<td>FPS5 =</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>FINIT</td>
<td>2.00000</td>
<td>GRAV =</td>
<td>5.32000</td>
<td>UN =</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>PMASS</td>
<td>4.00000</td>
<td>FURNOC =</td>
<td>0.00000</td>
<td>PN =</td>
<td>1.87500</td>
<td>1.00000</td>
</tr>
<tr>
<td>SK1</td>
<td>125000.00000</td>
<td>SK2 =</td>
<td>125000.00000</td>
<td>SK3 =</td>
<td>125000.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>SKE1</td>
<td>125000.00000</td>
<td>SKE2 =</td>
<td>125000.00000</td>
<td>SKE3 =</td>
<td>125000.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>G1RNO</td>
<td>1608.00000</td>
<td>VMAS =</td>
<td>400.00000</td>
<td>ETA =</td>
<td>0.2685</td>
<td>0.00000</td>
</tr>
<tr>
<td>PITCHO</td>
<td>-226.5</td>
<td>YAKO =</td>
<td>-7.9540</td>
<td>ROLLU =</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>PTCWNO</td>
<td>0.00000</td>
<td>YAWNO =</td>
<td>0.00000</td>
<td>PULLV0 =</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>XMOMX</td>
<td>999.99000</td>
<td>XMCHY =</td>
<td>11673.00000</td>
<td>XMCH7 =</td>
<td>11673.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>H</td>
<td>12.83000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 7-11. Input Data Printout**
CALCULATED INPUT VALUES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>B10</td>
<td>50187261</td>
</tr>
<tr>
<td>B20</td>
<td>92109223</td>
</tr>
<tr>
<td>B30</td>
<td>92109223</td>
</tr>
<tr>
<td>C10</td>
<td>1.570769360</td>
</tr>
<tr>
<td>C20</td>
<td>1.7099242</td>
</tr>
<tr>
<td>C30</td>
<td>1.97000025</td>
</tr>
<tr>
<td>SL1C</td>
<td>12.63854410</td>
</tr>
<tr>
<td>D10</td>
<td>0.00000000</td>
</tr>
<tr>
<td>XP</td>
<td>4.09128360</td>
</tr>
<tr>
<td>YP</td>
<td>11.15236430</td>
</tr>
<tr>
<td>ZP</td>
<td>-6.59184990</td>
</tr>
<tr>
<td>XPV</td>
<td>-12.57999990</td>
</tr>
<tr>
<td>YPV</td>
<td>12.07999990</td>
</tr>
<tr>
<td>ZPV</td>
<td>0.00000000</td>
</tr>
<tr>
<td>XPVEL</td>
<td>-9.74370000</td>
</tr>
<tr>
<td>YPVEL</td>
<td>2.24950000</td>
</tr>
<tr>
<td>ZPVEL</td>
<td>0.00000000</td>
</tr>
<tr>
<td>XPVVEL</td>
<td>-0.00000000</td>
</tr>
<tr>
<td>YPVVEL</td>
<td>0.00000000</td>
</tr>
<tr>
<td>ZPVVEL</td>
<td>0.00000000</td>
</tr>
</tbody>
</table>

Figure 7-12. Calculated Input Values
<table>
<thead>
<tr>
<th>TIME</th>
<th>A</th>
<th>Y</th>
<th>Z</th>
<th>XREF</th>
<th>YREF</th>
<th>ZREF</th>
<th>PILLOW</th>
<th>ROLLY</th>
<th>VRH</th>
<th>DELAY</th>
<th>LOG FLG</th>
</tr>
</thead>
<tbody>
<tr>
<td>.000</td>
<td>14.629</td>
<td>.000</td>
<td>.000</td>
<td>-6.726</td>
<td>2.256</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.020</td>
<td>14.622</td>
<td>.000</td>
<td>.000</td>
<td>-6.699</td>
<td>2.253</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.040</td>
<td>14.611</td>
<td>.000</td>
<td>.000</td>
<td>-6.672</td>
<td>2.249</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.060</td>
<td>14.589</td>
<td>.143</td>
<td>.000</td>
<td>-6.627</td>
<td>2.468</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.080</td>
<td>14.565</td>
<td>.194</td>
<td>.000</td>
<td>-6.572</td>
<td>2.422</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.100</td>
<td>14.532</td>
<td>1.496</td>
<td>.000</td>
<td>-6.509</td>
<td>2.346</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.120</td>
<td>14.492</td>
<td>2.318</td>
<td>.000</td>
<td>-6.434</td>
<td>1.505</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.140</td>
<td>14.453</td>
<td>3.387</td>
<td>.000</td>
<td>-6.364</td>
<td>1.567</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.160</td>
<td>14.415</td>
<td>4.683</td>
<td>.000</td>
<td>-6.287</td>
<td>1.726</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.180</td>
<td>14.389</td>
<td>6.211</td>
<td>.000</td>
<td>-6.207</td>
<td>1.948</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.200</td>
<td>14.389</td>
<td>7.069</td>
<td>.000</td>
<td>-6.137</td>
<td>2.126</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.220</td>
<td>14.389</td>
<td>7.174</td>
<td>.000</td>
<td>-6.067</td>
<td>2.276</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.240</td>
<td>14.389</td>
<td>7.307</td>
<td>.000</td>
<td>-5.997</td>
<td>2.376</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.260</td>
<td>14.389</td>
<td>7.458</td>
<td>.000</td>
<td>-5.927</td>
<td>2.372</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.280</td>
<td>14.389</td>
<td>7.629</td>
<td>.000</td>
<td>-5.857</td>
<td>2.366</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.300</td>
<td>14.389</td>
<td>7.829</td>
<td>.000</td>
<td>-5.787</td>
<td>2.356</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.320</td>
<td>14.389</td>
<td>8.054</td>
<td>.000</td>
<td>-5.717</td>
<td>2.346</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.340</td>
<td>14.389</td>
<td>8.304</td>
<td>.000</td>
<td>-5.647</td>
<td>2.332</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.360</td>
<td>14.389</td>
<td>8.579</td>
<td>.000</td>
<td>-5.577</td>
<td>2.317</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.380</td>
<td>14.389</td>
<td>8.879</td>
<td>.000</td>
<td>-5.507</td>
<td>2.294</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.400</td>
<td>14.389</td>
<td>9.198</td>
<td>.000</td>
<td>-5.437</td>
<td>2.260</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.420</td>
<td>14.389</td>
<td>9.530</td>
<td>.000</td>
<td>-5.367</td>
<td>2.212</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.440</td>
<td>14.389</td>
<td>9.874</td>
<td>.000</td>
<td>-5.297</td>
<td>2.144</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.460</td>
<td>14.389</td>
<td>10.234</td>
<td>.000</td>
<td>-5.227</td>
<td>2.060</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.480</td>
<td>14.389</td>
<td>10.599</td>
<td>.000</td>
<td>-5.157</td>
<td>1.964</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.500</td>
<td>14.389</td>
<td>10.979</td>
<td>.000</td>
<td>-5.087</td>
<td>1.858</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.520</td>
<td>14.389</td>
<td>11.369</td>
<td>.000</td>
<td>-5.017</td>
<td>1.743</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.540</td>
<td>14.389</td>
<td>11.769</td>
<td>.000</td>
<td>-4.947</td>
<td>1.627</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.560</td>
<td>14.389</td>
<td>12.189</td>
<td>.000</td>
<td>-4.877</td>
<td>1.509</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.580</td>
<td>14.389</td>
<td>12.629</td>
<td>.000</td>
<td>-4.807</td>
<td>1.389</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
<tr>
<td>.600</td>
<td>14.389</td>
<td>13.089</td>
<td>.000</td>
<td>-4.738</td>
<td>1.267</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
</tbody>
</table>

Figure 7-13. Output Data (Part 1)
<table>
<thead>
<tr>
<th>TIME</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>XVEL</th>
<th>YVEL</th>
<th>F17TV</th>
<th>NVELV</th>
<th>YAVV</th>
<th>JLF1E</th>
<th>LEGTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>-0.05</td>
<td>-0.01</td>
<td>2.09</td>
<td>2.88</td>
<td>0.06</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>-0.17</td>
<td>2.16</td>
<td>2.76</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.04</td>
<td>-0.19</td>
<td>2.08</td>
<td>2.72</td>
<td>-0.06</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.06</td>
<td>-0.18</td>
<td>1.99</td>
<td>2.71</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.08</td>
<td>-0.18</td>
<td>1.93</td>
<td>2.66</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>-0.18</td>
<td>1.87</td>
<td>2.62</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.12</td>
<td>-0.18</td>
<td>1.82</td>
<td>2.57</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.14</td>
<td>-0.18</td>
<td>1.77</td>
<td>2.53</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.16</td>
<td>-0.18</td>
<td>1.73</td>
<td>2.49</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.18</td>
<td>-0.18</td>
<td>1.70</td>
<td>2.46</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.20</td>
<td>-0.18</td>
<td>1.67</td>
<td>2.44</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.22</td>
<td>-0.18</td>
<td>1.65</td>
<td>2.41</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.24</td>
<td>-0.18</td>
<td>1.63</td>
<td>2.39</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.26</td>
<td>-0.18</td>
<td>1.61</td>
<td>2.37</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.28</td>
<td>-0.18</td>
<td>1.60</td>
<td>2.35</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.30</td>
<td>-0.18</td>
<td>1.58</td>
<td>2.33</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.32</td>
<td>-0.18</td>
<td>1.57</td>
<td>2.31</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.34</td>
<td>-0.18</td>
<td>1.55</td>
<td>2.29</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.36</td>
<td>-0.18</td>
<td>1.54</td>
<td>2.27</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.38</td>
<td>-0.18</td>
<td>1.52</td>
<td>2.25</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.40</td>
<td>-0.18</td>
<td>1.51</td>
<td>2.23</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.42</td>
<td>-0.18</td>
<td>1.49</td>
<td>2.21</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.44</td>
<td>-0.18</td>
<td>1.48</td>
<td>2.19</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.46</td>
<td>-0.18</td>
<td>1.46</td>
<td>2.17</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.48</td>
<td>-0.18</td>
<td>1.45</td>
<td>2.15</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.50</td>
<td>-0.18</td>
<td>1.43</td>
<td>2.13</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.52</td>
<td>-0.18</td>
<td>1.41</td>
<td>2.11</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.54</td>
<td>-0.18</td>
<td>1.39</td>
<td>2.09</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.56</td>
<td>-0.18</td>
<td>1.37</td>
<td>2.07</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.58</td>
<td>-0.18</td>
<td>1.35</td>
<td>2.05</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.60</td>
<td>-0.18</td>
<td>1.33</td>
<td>2.03</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.62</td>
<td>-0.18</td>
<td>1.31</td>
<td>2.01</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.64</td>
<td>-0.18</td>
<td>1.29</td>
<td>1.99</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.66</td>
<td>-0.18</td>
<td>1.27</td>
<td>1.97</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.68</td>
<td>-0.18</td>
<td>1.25</td>
<td>1.95</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.70</td>
<td>-0.18</td>
<td>1.23</td>
<td>1.93</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.72</td>
<td>-0.18</td>
<td>1.21</td>
<td>1.91</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.74</td>
<td>-0.18</td>
<td>1.19</td>
<td>1.89</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.76</td>
<td>-0.18</td>
<td>1.17</td>
<td>1.87</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.78</td>
<td>-0.18</td>
<td>1.15</td>
<td>1.85</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.80</td>
<td>-0.18</td>
<td>1.13</td>
<td>1.83</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.82</td>
<td>-0.18</td>
<td>1.11</td>
<td>1.81</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.84</td>
<td>-0.18</td>
<td>1.09</td>
<td>1.79</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.86</td>
<td>-0.18</td>
<td>1.07</td>
<td>1.77</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.88</td>
<td>-0.18</td>
<td>1.05</td>
<td>1.75</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.90</td>
<td>-0.18</td>
<td>1.03</td>
<td>1.73</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.92</td>
<td>-0.18</td>
<td>1.01</td>
<td>1.71</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.94</td>
<td>-0.18</td>
<td>0.99</td>
<td>1.69</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.96</td>
<td>-0.18</td>
<td>0.97</td>
<td>1.67</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.98</td>
<td>-0.18</td>
<td>0.95</td>
<td>1.65</td>
<td>0.04</td>
<td>0.17</td>
<td>-0.14</td>
<td>0.34</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 7-13. Output Data (Part 1 - Concluded)
SERIES NO. 1608.00  RUN NO. 1.00

TIME  2.002 SECS.

<table>
<thead>
<tr>
<th>MAXIMUM STROKE</th>
<th>NO. 1 STRUT</th>
<th>NO. 2 STRUT</th>
<th>NO. 3 STRUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>.719</td>
<td>.676</td>
<td>.514</td>
</tr>
<tr>
<td></td>
<td>1.975</td>
<td>1.691</td>
<td>.996</td>
</tr>
<tr>
<td></td>
<td>.709</td>
<td>.452</td>
<td>.86</td>
</tr>
</tbody>
</table>

MINIMUM CLEARANCE OF NOZZLE = 3.400
TIME WHEN THE MINIMUM CLEARANCE OCCURS = .570

MINIMUM STABILITY ANGLE B1 = .234
TIME WHEN THIS STABILITY ANGLE OCCURS = 2.000

ENERGY DISSIPATED BASED ON VEHICLE VELOCITIES AND C.G. DROP = 23894.445

ENERGY DISSIPATED BASED ON PLASTIC STROKE = 32150.126

ENERGY DISSIPATED BASED ON PLASTIC AND FULL ELASTIC STROKE = 32726.353

FINAL STABILITY ANGLE B1 = .234

ENERGY DISTRIBUTION BETWEEN LEGS - PERCENT = 17.902  32.302  32.378  17.469

Figure 7-14. Summary Output Data
<table>
<thead>
<tr>
<th>TIPF</th>
<th>FXPLG3(1)</th>
<th>FYPLG3(1)</th>
<th>FXPLG3(2)</th>
<th>FYPLG3(2)</th>
<th>FXPLG5(1)</th>
<th>FYPLG5(1)</th>
<th>FXPLG5(2)</th>
<th>FYPLG5(2)</th>
<th>FXPLG5(3)</th>
<th>FYPLG5(3)</th>
<th>FXPLG5(4)</th>
<th>FYPLG5(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 7-16. Output Data (Part 3 - Concluded)**
<table>
<thead>
<tr>
<th>TIPF</th>
<th>KF (1)</th>
<th>VF (1)</th>
<th>ZP (1)</th>
<th>YP (1)</th>
<th>ZF (1)</th>
<th>FX (1)</th>
<th>FY (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.091</td>
<td>11.152</td>
<td>0.042</td>
<td>4.917</td>
<td>11.117</td>
<td>0.144</td>
<td>10.971</td>
<td></td>
</tr>
<tr>
<td>3.014</td>
<td>11.197</td>
<td>0.477</td>
<td>3.115</td>
<td>11.234</td>
<td>0.562</td>
<td>10.922</td>
<td></td>
</tr>
<tr>
<td>3.718</td>
<td>11.236</td>
<td>0.424</td>
<td>3.716</td>
<td>11.234</td>
<td>0.562</td>
<td>10.922</td>
<td></td>
</tr>
<tr>
<td>3.537</td>
<td>11.277</td>
<td>0.432</td>
<td>3.528</td>
<td>11.312</td>
<td>0.662</td>
<td>10.977</td>
<td></td>
</tr>
<tr>
<td>3.379</td>
<td>11.310</td>
<td>0.429</td>
<td>3.279</td>
<td>11.377</td>
<td>0.764</td>
<td>10.984</td>
<td></td>
</tr>
<tr>
<td>3.190</td>
<td>11.344</td>
<td>0.426</td>
<td>3.183</td>
<td>11.434</td>
<td>0.864</td>
<td>11.026</td>
<td></td>
</tr>
<tr>
<td>3.057</td>
<td>11.376</td>
<td>0.428</td>
<td>3.087</td>
<td>11.477</td>
<td>0.964</td>
<td>11.080</td>
<td></td>
</tr>
<tr>
<td>2.836</td>
<td>11.461</td>
<td>0.427</td>
<td>2.776</td>
<td>11.401</td>
<td>0.642</td>
<td>11.010</td>
<td></td>
</tr>
<tr>
<td>2.585</td>
<td>11.543</td>
<td>0.404</td>
<td>2.487</td>
<td>11.401</td>
<td>0.642</td>
<td>11.010</td>
<td></td>
</tr>
<tr>
<td>2.179</td>
<td>11.707</td>
<td>0.392</td>
<td>2.032</td>
<td>11.671</td>
<td>0.542</td>
<td>11.041</td>
<td></td>
</tr>
<tr>
<td>1.739</td>
<td>11.705</td>
<td>0.364</td>
<td>1.536</td>
<td>11.624</td>
<td>0.442</td>
<td>11.141</td>
<td></td>
</tr>
<tr>
<td>1.493</td>
<td>11.690</td>
<td>0.361</td>
<td>1.429</td>
<td>11.617</td>
<td>0.442</td>
<td>11.141</td>
<td></td>
</tr>
<tr>
<td>1.284</td>
<td>11.678</td>
<td>0.353</td>
<td>1.306</td>
<td>11.577</td>
<td>0.342</td>
<td>11.141</td>
<td></td>
</tr>
<tr>
<td>1.082</td>
<td>11.665</td>
<td>0.341</td>
<td>1.375</td>
<td>11.526</td>
<td>0.242</td>
<td>11.181</td>
<td></td>
</tr>
<tr>
<td>0.910</td>
<td>11.612</td>
<td>0.332</td>
<td>1.302</td>
<td>11.470</td>
<td>0.142</td>
<td>11.203</td>
<td></td>
</tr>
<tr>
<td>0.760</td>
<td>11.558</td>
<td>0.321</td>
<td>1.213</td>
<td>11.405</td>
<td>0.042</td>
<td>11.273</td>
<td></td>
</tr>
<tr>
<td>0.620</td>
<td>11.485</td>
<td>0.301</td>
<td>1.120</td>
<td>11.327</td>
<td>0.000</td>
<td>11.313</td>
<td></td>
</tr>
<tr>
<td>0.505</td>
<td>11.411</td>
<td>0.285</td>
<td>1.036</td>
<td>11.238</td>
<td>0.000</td>
<td>11.313</td>
<td></td>
</tr>
<tr>
<td>0.408</td>
<td>11.337</td>
<td>0.268</td>
<td>0.952</td>
<td>11.142</td>
<td>0.000</td>
<td>11.313</td>
<td></td>
</tr>
<tr>
<td>0.334</td>
<td>11.261</td>
<td>0.254</td>
<td>0.869</td>
<td>11.038</td>
<td>0.000</td>
<td>11.313</td>
<td></td>
</tr>
<tr>
<td>0.274</td>
<td>11.186</td>
<td>0.242</td>
<td>0.784</td>
<td>10.925</td>
<td>0.000</td>
<td>11.313</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7-16. Output Data (Part 4)
<table>
<thead>
<tr>
<th>TIME</th>
<th>XP(1)</th>
<th>YF(1)</th>
<th>ZP(1)</th>
<th>XP(4)</th>
<th>YF(4)</th>
<th>F(4)</th>
<th>FX(1)</th>
<th>FY(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>0.780</td>
<td>12.322</td>
<td>-6.169</td>
<td>0.777</td>
<td>12.198</td>
<td>0.758</td>
<td>1.000</td>
<td>0.500</td>
</tr>
<tr>
<td>1.020</td>
<td>0.784</td>
<td>12.333</td>
<td>-6.167</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.020</td>
<td>0.500</td>
</tr>
<tr>
<td>1.040</td>
<td>0.786</td>
<td>12.345</td>
<td>-6.166</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.040</td>
<td>0.500</td>
</tr>
<tr>
<td>1.060</td>
<td>0.787</td>
<td>12.356</td>
<td>-6.165</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.060</td>
<td>0.500</td>
</tr>
<tr>
<td>1.080</td>
<td>0.788</td>
<td>12.367</td>
<td>-6.164</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.080</td>
<td>0.500</td>
</tr>
<tr>
<td>1.100</td>
<td>0.788</td>
<td>12.378</td>
<td>-6.163</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.100</td>
<td>0.500</td>
</tr>
<tr>
<td>1.120</td>
<td>0.788</td>
<td>12.389</td>
<td>-6.162</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.120</td>
<td>0.500</td>
</tr>
<tr>
<td>1.140</td>
<td>0.788</td>
<td>12.400</td>
<td>-6.161</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.140</td>
<td>0.500</td>
</tr>
<tr>
<td>1.160</td>
<td>0.788</td>
<td>12.411</td>
<td>-6.160</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.160</td>
<td>0.500</td>
</tr>
<tr>
<td>1.180</td>
<td>0.788</td>
<td>12.422</td>
<td>-6.159</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.180</td>
<td>0.500</td>
</tr>
<tr>
<td>1.200</td>
<td>0.788</td>
<td>12.433</td>
<td>-6.158</td>
<td>0.792</td>
<td>12.191</td>
<td>0.772</td>
<td>1.200</td>
<td>0.500</td>
</tr>
</tbody>
</table>

**Figure 7-17. Output Data (Part 4 - Concluded)**
<table>
<thead>
<tr>
<th>TEMP</th>
<th>FXPLG3(1)</th>
<th>FYPLG3(1)</th>
<th>FFLPG3(1)</th>
<th>FFLPG3(1)</th>
<th>FYPLG3(1)</th>
<th>FYPLG3(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.200</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 7-18. Output Data (Part 5 - Concluded)