Progress Report on

PHASE II:
DEVELOPMENT OF A SIMPLE, SELF-CONTAINED
FLIGHT TEST DATA ACQUISITION SYSTEM

KU-FRL-407-6

NASA Grant NSG-4019

Prepared by: Ronald R. L. Renz

Principal Investigators: Jan Roskam
                     Dale I. Rummer

Work Performed by: Mark A. Mosser
                   Alex Kotsabasis
                   Robert Clarke
                   Douglas Shane

University of Kansas Flight Research Laboratory
Lawrence, Kansas 66045
February 1981
ABSTRACT

DEVELOPMENT OF A SIMPLE, SELF-CONTAINED FLIGHT TEST DATA ACQUISITION SYSTEM

This report describes work done under a continuing program to develop a simple, self-contained flight test data acquisition system. In the past, instrumenting an airplane for flight testing has taken a great deal of time and money. With recent advances in sensor and microprocessor technology, a simple, low-cost system could be developed which would be applicable to general aviation airplanes.

This system was conceived to obtain performance and stability characteristics of airplanes. The design criteria for the system were that it be easy to install, self-contained, and simple; that it require no special/difficult flight techniques; and that it be applicable to general aviation airplanes and low in cost.

The system developed meets these criteria for doing longitudinal and lateral stability analysis. The package consists of three modules. These are 1) microprocessor controller and data acquisition module, 2) transducer module, and 3) power supply module. The system is easy to install and occupies space in the cabin or baggage compartment of the airplane. All transducers are contained in these modules except the total pressure tube, static pressure air temperature transducer, and control position transducers.

The data reduction technique used was the NASA-developed MILE program. This has been placed on a microcomputer, and all data reduction is done on the microcomputer. This greatly reduces the cost of the data reduction. Also, when compared with the analogue recording techniques, still being used, there has been a large improvement in the accuracy of results.

The flight testing program undertaken has proven both the flight testing hardware and the data reduction method to be applicable to the current field of general aviation airplanes.

This report describes the instrumentation system developed, the data reduction method used, and important results of the flight test program.

PRECEDING PAGE BLANK NOT FILMED
ACKNOWLEDGEMENTS

We would like to thank NASA Dryden Flight Research Center via Mr. Ken Szalai, Technical Monitor, and Dr. Z. Kordes for the financial support of this program. Special thanks also go to Cessna Aircraft Company for the assistance they have given in this phase of the program. Sincerest thanks go to our secretary and grammarians, Mrs. Nancy Hanson, for the many hours spent typing and debugging this manuscript.
TABLE OF CONTENTS

Page

ABSTRACT. ................................................................. iii
ACKNOWLEDGEMENTS. ....................................................... v
TABLE OF CONTENTS ........................................................ v
LIST OF SYMBOLS ............................................................ xi
LIST OF FIGURES .............................................................. xxv
LIST OF TABLES. ............................................................. xxix

1. INTRODUCTION. ............................................................ 1

2 PURPOSE OF PROJECT. .................................................. 7

3. INSTRUMENTATION SYSTEM. ................................................. 9

3.1 Data Management. ...................................................... 15

3.2 Transducers. ............................................................. 20

3.2.1 Accelerometers. .................................................... 27

3.2.2 Filtering .............................................................. 29

3.2.3 Attitude Gyro ....................................................... 31

3.2.4 Rate Gyros. .......................................................... 31

3.2.5 Control Position Transducers. ................................... 33

3.2.6 Static and Dynamic Pressure Transducer ....................... 38

3.2.7 Temperature Transducer. ......................................... 41

3.3 Power Supply ........................................................... 44

3.4 Pilot Control. ........................................................... 48

3.4.1 System Control. .................................................... 48

3.4.2 Transducer Readout Control. ..................................... 48

4. GROUND COMPUTER SYSTEM. ........................................... 51

4.1 Integer Speed Routine. ............................................... 51
# TABLE OF CONTENTS (continued)

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2</td>
<td>Floating Point Speed Routine</td>
<td>53</td>
</tr>
<tr>
<td>4.3</td>
<td>Description of System</td>
<td>55</td>
</tr>
<tr>
<td>5.</td>
<td>DATA REDUCTION METHOD</td>
<td>59</td>
</tr>
<tr>
<td>5.1</td>
<td>Data Acquisition</td>
<td>59</td>
</tr>
<tr>
<td>5.2</td>
<td>Transfer Data to Ground-Based Computer</td>
<td>62</td>
</tr>
<tr>
<td>5.3</td>
<td>Engineering Conversion</td>
<td>63</td>
</tr>
<tr>
<td>5.4</td>
<td>Quick-Look Plots</td>
<td>64</td>
</tr>
<tr>
<td>5.5</td>
<td>Detailed Engineering Conversion</td>
<td>64</td>
</tr>
<tr>
<td>5.6</td>
<td>Modified Maximum Likelihood Estimator</td>
<td>65</td>
</tr>
<tr>
<td>5.6.1</td>
<td>Parameter Estimation</td>
<td>66</td>
</tr>
<tr>
<td>5.6.2</td>
<td>Mathematical Model</td>
<td>67</td>
</tr>
<tr>
<td>5.6.3</td>
<td>Assumptions Used in Data Reduction</td>
<td>76</td>
</tr>
<tr>
<td>5.7</td>
<td>Time History Plotting</td>
<td>77</td>
</tr>
<tr>
<td>6.</td>
<td>KU-FRL FLIGHT TEST PROGRAM</td>
<td>79</td>
</tr>
<tr>
<td>6.1</td>
<td>Flight Test Maneuver</td>
<td>79</td>
</tr>
<tr>
<td>6.2</td>
<td>Results of Flight Test Program</td>
<td>80</td>
</tr>
<tr>
<td>7.</td>
<td>CESSNA FLIGHT TEST PROGRAM</td>
<td>91</td>
</tr>
<tr>
<td>7.1</td>
<td>Instrumentation</td>
<td>91</td>
</tr>
<tr>
<td>7.2</td>
<td>Data Reduction</td>
<td>96</td>
</tr>
<tr>
<td>7.3</td>
<td>Results of Spin Program</td>
<td>100</td>
</tr>
<tr>
<td>8.</td>
<td>CONCLUSIONS</td>
<td>101</td>
</tr>
<tr>
<td>9.</td>
<td>RECOMMENDATIONS FOR PHASE III</td>
<td>103</td>
</tr>
<tr>
<td>9.1</td>
<td>Equipment</td>
<td>103</td>
</tr>
<tr>
<td>9.2</td>
<td>Calibration</td>
<td>103</td>
</tr>
<tr>
<td>9.3</td>
<td>Data Reduction</td>
<td>103</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>9.4 Effect of Control Surface Float</td>
<td>104</td>
<td></td>
</tr>
<tr>
<td>9.5 Proof-of-Test Capability</td>
<td>104</td>
<td></td>
</tr>
<tr>
<td>10. REFERENCES</td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>10.1 Instrumentation System Reports</td>
<td>108</td>
<td></td>
</tr>
<tr>
<td>APPENDICES:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A. PROGRAMS</td>
<td>109</td>
<td></td>
</tr>
<tr>
<td>.1) Data Acquisition</td>
<td>111</td>
<td></td>
</tr>
<tr>
<td>.2) Data Transfer</td>
<td>115</td>
<td></td>
</tr>
<tr>
<td>.3) Data Receive</td>
<td>119</td>
<td></td>
</tr>
<tr>
<td>.4) Engineering Conversion</td>
<td>121</td>
<td></td>
</tr>
<tr>
<td>.5) Quick Look Plots</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>.6) Detailed Engineering Conversion</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>.7) MMLE Programs</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>.8) Time History Plotting</td>
<td>165</td>
<td></td>
</tr>
<tr>
<td>.9) Cessna Programs</td>
<td>171</td>
<td></td>
</tr>
<tr>
<td>B. TRANSFORMATION OF AXES SYSTEMS</td>
<td>179</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS

All parameters in this report are referenced to a system of body axes as shown in Figure 1.1.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A = -X )</td>
<td>Force in A direction</td>
<td>lb</td>
</tr>
<tr>
<td>( A_X, A_X )</td>
<td>Longitudinal acceleration</td>
<td>g</td>
</tr>
<tr>
<td>( A_Y, A_Y )</td>
<td>Lateral acceleration</td>
<td>g</td>
</tr>
<tr>
<td>( A_Z, A_Z )</td>
<td>Vertical acceleration</td>
<td>g</td>
</tr>
<tr>
<td>( A_N = -A_Z )</td>
<td>Normal acceleration</td>
<td>g</td>
</tr>
<tr>
<td>([A])</td>
<td>Stability matrix</td>
<td></td>
</tr>
<tr>
<td>([B])</td>
<td>Control matrix</td>
<td></td>
</tr>
<tr>
<td>( b )</td>
<td>Wing span</td>
<td>ft</td>
</tr>
<tr>
<td>( {c} )</td>
<td>Vector of unknowns for MMLE</td>
<td></td>
</tr>
<tr>
<td>( \bar{c} )</td>
<td>Mean aerodynamic chord</td>
<td>ft</td>
</tr>
<tr>
<td>( C_A = \frac{A}{qS} = -C_X )</td>
<td>Coefficient of force in A direction ((A = -X))</td>
<td></td>
</tr>
<tr>
<td>( C_A = \frac{3C_A}{3\alpha} )</td>
<td>Variation of body A coefficient with angle of attack</td>
<td>rad(^{-1})</td>
</tr>
<tr>
<td>( C_A = \frac{3C_A}{3(U_U)^-} )</td>
<td>Variation of body A coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>( C_A = \frac{3C_A}{3\delta_{E,c}} )</td>
<td>Variation of body A coefficient with elevator or canard angle</td>
<td>rad(^{-1})</td>
</tr>
<tr>
<td>( C_A )</td>
<td>Nondimensional longitudinal force equation bias</td>
<td></td>
</tr>
<tr>
<td>( C_D = \frac{D}{qS} )</td>
<td>Drag force coefficient</td>
<td></td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>$C_{D} = \frac{3C_{D}}{\partial \alpha}$</td>
<td>Variation of drag coefficient with angle of attack</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{D_{u}} = \frac{3C_{D}}{\partial \left( \frac{u}{U_1} \right)}$</td>
<td>Variation of drag coefficient with speed</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{D_{E}} = \frac{3C_{D}}{\partial \delta_{E}}$</td>
<td>Variation of drag coefficient with elevator angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{L} = \frac{L}{q_{s}}$</td>
<td>Lift force coefficient</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{\alpha}} = \frac{3C_{L}}{\partial \alpha}$</td>
<td>Variation of lift coefficient with angle of attack</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{L_{C}} = \frac{3C_{L}}{\partial \left( \frac{C}{2U_1} \right)}$</td>
<td>Variation of lift coefficient with rate of change of angle of attack</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{q}} = \frac{3C_{L}}{\partial \left( \frac{q}{2U_1} \right)}$</td>
<td>Variation of lift coefficient with pitch rate</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{u}} = \frac{3C_{L}}{\partial \left( \frac{u}{U_1} \right)}$</td>
<td>Variation of lift coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{E}} = \frac{3C_{L}}{\partial \delta_{E}}$</td>
<td>Variation of lift coefficient with elevator angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{L} = \frac{L}{q_{s} s_{b}}$</td>
<td>Rolling moment coefficient</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{\beta}} = \frac{3C_{L}}{\partial \beta}$</td>
<td>Variation of rolling moment coefficient with sideslip angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>$C_\ell_p$</td>
<td>Variation of rolling moment coefficient with roll rate</td>
<td>$\text{rad}^{-1}$</td>
</tr>
<tr>
<td>$C_\ell_r$</td>
<td>Variation of rolling moment coefficient with yaw rate</td>
<td>$\text{rad}^{-1}$</td>
</tr>
<tr>
<td>$C_{A,R}$</td>
<td>Variation of rolling moment coefficient with aileron or rudder angle</td>
<td>$\text{rad}^{-1}$</td>
</tr>
<tr>
<td>$C_m = \frac{M}{qS_C}$</td>
<td>Pitching moment coefficient</td>
<td></td>
</tr>
<tr>
<td>$C_m_\alpha$</td>
<td>Variation of pitching moment coefficient with angle of attack</td>
<td>$\text{rad}^{-1}$</td>
</tr>
<tr>
<td>$C_m_\dot{\alpha}$</td>
<td>Variation of pitching moment coefficient with rate of change of angle of attack</td>
<td></td>
</tr>
<tr>
<td>$C_m_q$</td>
<td>Variation of pitching moment coefficient with pitch rate</td>
<td></td>
</tr>
<tr>
<td>$C_m_u$</td>
<td>Variation of pitching moment coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$C_m_T$</td>
<td>Pitching moment coefficient due to thrust</td>
<td></td>
</tr>
<tr>
<td>$C_m_{\alpha}$</td>
<td>Variation of thrust pitching moment coefficient with angle of attack</td>
<td>$\text{rad}^{-1}$</td>
</tr>
<tr>
<td>$C_m_{\dot{\alpha}}$</td>
<td>Variation of thrust pitching moment coefficient with angle of attack</td>
<td></td>
</tr>
<tr>
<td>$C_m_{U}$</td>
<td>Variation of thrust pitching moment coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$C_m_{E,c}$</td>
<td>Variation of pitching moment coefficient with elevator or canard angle</td>
<td>$\text{rad}^{-1}$</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS (continued)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n_0}$</td>
<td>Nondimensional pitching moment equation bias</td>
<td></td>
</tr>
<tr>
<td>$C_N = \frac{N}{qS} = -C_z$</td>
<td>Normal force coefficient. $(N = -z)$</td>
<td></td>
</tr>
<tr>
<td>$C_{N_a} = \frac{3C_N}{3a}$</td>
<td>Variation of normal force coefficient with angle of attack</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{N_u} = \frac{3C_N}{3(\frac{U}{U_1})}$</td>
<td>Variation of normal force coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$C_{N_{E,c}} = \frac{3C_N}{3\delta_{E,c}}$</td>
<td>Variation of normal force coefficient with elevator or canard angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{N_0}$</td>
<td>Nondimensional normal force equation bias</td>
<td></td>
</tr>
<tr>
<td>$C_n = \frac{N}{qSb}$</td>
<td>Yawing moment coefficient</td>
<td></td>
</tr>
<tr>
<td>$C_{n_B} = \frac{3C_n}{3\delta}$</td>
<td>Variation of yawing moment coefficient with sideslip angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{n_r}$</td>
<td>Yawing moment coefficient due to thrust</td>
<td></td>
</tr>
<tr>
<td>$C_{n_T} = \frac{3C_n}{3\delta}$</td>
<td>Variation of thrust yawing moment coefficient with sideslip angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{n_p} = \frac{3C_n}{3p}$</td>
<td>Variation of yawing moment coefficient with roll rate</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{n_r} = \frac{3C_n}{3\delta}$</td>
<td>Variation of yawing moment coefficient with yaw rate</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_{n_{A,R}} = \frac{3C_n}{3\delta_{A,R}}$</td>
<td>Variation of yawing moment coefficient with aileron or rudder angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>$C_T = \frac{T_X}{qS}$</td>
<td>Thrust force coefficient in X direction</td>
<td></td>
</tr>
<tr>
<td>$\frac{\partial C_T}{\partial u}$</td>
<td>Variation of thrust force coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$C_X = \frac{X}{qS}$</td>
<td>Force coefficient in X direction</td>
<td></td>
</tr>
<tr>
<td>$\frac{\partial C_X}{\partial \alpha}$</td>
<td>Variation of longitudinal force coefficient with angle of attack</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$\frac{\partial C_X}{\partial \alpha}$</td>
<td>Variation of longitudinal force coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>$\frac{\partial C_X}{\partial \alpha}$</td>
<td>Variation of longitudinal force coefficient with elevator or canard angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$C_X^o$</td>
<td>Nondimensional longitudinal force equation bias</td>
<td></td>
</tr>
<tr>
<td>$C_Y = \frac{Y}{qS}$</td>
<td>Force coefficient in Y direction</td>
<td></td>
</tr>
<tr>
<td>$\frac{\partial C_Y}{\partial \beta}$</td>
<td>Variation of side force coefficient with sideslip angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$\frac{\partial C_Y}{\partial \phi}$</td>
<td>Variation of side force coefficient with roll rate</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$\frac{\partial C_Y}{\partial \gamma}$</td>
<td>Variation of side force coefficient with yaw rate</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>$\frac{\partial C_Y}{\partial \delta_{A,R}}$</td>
<td>Variation of side force coefficient with aileron or rudder angle</td>
<td>rad$^{-1}$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>( C_Z )</td>
<td>Force coefficient in Z direction</td>
<td></td>
</tr>
<tr>
<td>( \frac{\delta\alpha}{\delta\alpha} C_Z )</td>
<td>Variation of vertical force coefficient with angle of attack</td>
<td>rad(^{-1})</td>
</tr>
<tr>
<td>( \frac{\delta C_Z}{\delta u} )</td>
<td>Variation of vertical force coefficient with speed</td>
<td></td>
</tr>
<tr>
<td>( \frac{\delta C_Z}{\delta\delta_{E,C}} )</td>
<td>Variation of vertical force coefficient with elevator or canard angle</td>
<td>rad(^{-1})</td>
</tr>
<tr>
<td>( C_{Z_o} )</td>
<td>Nondimensional vertical force equation bias</td>
<td></td>
</tr>
<tr>
<td>( C_c )</td>
<td>Y axis force coefficient in wind tunnel axes</td>
<td></td>
</tr>
<tr>
<td>( D )</td>
<td>Drag force</td>
<td>lb</td>
</tr>
<tr>
<td>[( D )]</td>
<td>MMLE weighting matrix</td>
<td></td>
</tr>
<tr>
<td>( g, G )</td>
<td>Force of gravity</td>
<td>ft sec(^{-2})</td>
</tr>
<tr>
<td>[( G )]</td>
<td>MMLE observation matrix</td>
<td></td>
</tr>
<tr>
<td>[( H )]</td>
<td>MMLE observation matrix</td>
<td></td>
</tr>
<tr>
<td>( H_p )</td>
<td>Pressure altitude</td>
<td>ft</td>
</tr>
<tr>
<td>[( I )]</td>
<td>Identity matrix</td>
<td></td>
</tr>
<tr>
<td>( I_{xx}, I_{yy}, I_{zz} )</td>
<td>Moment of inertia about the X, Y, and Z axes respectively</td>
<td>slug ft(^2)</td>
</tr>
<tr>
<td>( I_{xz} )</td>
<td>Product of inertia</td>
<td>slug ft(^2)</td>
</tr>
<tr>
<td>( J )</td>
<td>MMLE cost function</td>
<td></td>
</tr>
<tr>
<td>KTAS</td>
<td>True airspeed</td>
<td>knots</td>
</tr>
<tr>
<td>( \ell, L )</td>
<td>Rolling moment (perturbed, total)</td>
<td>ft lb</td>
</tr>
<tr>
<td>( L )</td>
<td>Lift force</td>
<td>lb</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>$L$</td>
<td>Iteration number</td>
<td></td>
</tr>
<tr>
<td>$L_\beta$</td>
<td>Dimensional variation of rolling moment with sideslip angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$L_p$</td>
<td>Dimensional variation of rolling moment with roll rate</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$L_r$</td>
<td>Dimensional variation of rolling moment with yaw rate</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$L_{\delta A,R}$</td>
<td>Dimensional variation of rolling moment with aileron or rudder angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$L_o$</td>
<td>Rolling moment equation bias</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$m, M$</td>
<td>Pitching moment (perturbed, total), ft lb</td>
<td></td>
</tr>
<tr>
<td>$m$</td>
<td>Mass</td>
<td>slug</td>
</tr>
<tr>
<td>$M_P$</td>
<td>Engine manifold pressure</td>
<td></td>
</tr>
<tr>
<td>$M_\alpha$</td>
<td>Dimensional variation of pitching moment with angle of attack</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$M_{\alpha}$</td>
<td>Dimensional variation of pitching moment with rate of change of angle of attack</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$M_q$</td>
<td>Dimensional variation of pitching moment with pitch rate</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$M_u$</td>
<td>Dimensional variation of pitching moment with speed</td>
<td>ft$^{-1}$ sec$^{-1}$</td>
</tr>
<tr>
<td>$M_{\alpha}$</td>
<td>Dimensional variation of pitching moment due to thrust with angle of attack</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$M_{\alpha}$</td>
<td>Dimensional variation of pitching moment due to thrust with speed</td>
<td>ft$^{-1}$ sec$^{-1}$</td>
</tr>
</tbody>
</table>
## LIST OF SYMBOLS (continued)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_\delta$</td>
<td>Dimensional variation of pitching moment due to elevator or canard angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$E_c$</td>
<td>Moment due to elevator or canard angle</td>
<td></td>
</tr>
<tr>
<td>$M_o$</td>
<td>Pitching moment equation bias</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$M_\theta$</td>
<td>Dimensional variation of pitching moment with pitch angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$n,N$</td>
<td>Yawing moment (perturbed,total)</td>
<td>ft lb</td>
</tr>
<tr>
<td>$N = -Z$</td>
<td>Normal force</td>
<td>lb</td>
</tr>
<tr>
<td>$N_\beta$</td>
<td>Dimensional variation yawing moment with sideslip angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$N_{T\beta}$</td>
<td>Dimensional variation of yawing moment due to thrust with sideslip angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$N_p$</td>
<td>Dimensional variation of yawing moment with roll rate</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$N_r$</td>
<td>Dimensional variation of yawing moment with yaw rate</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$N_{\delta A,R}$</td>
<td>Dimensional variation of yawing moment with aileron or rudder angle</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$N_o$</td>
<td>Yawing moment equation bias</td>
<td>sec$^{-2}$</td>
</tr>
<tr>
<td>$p,P$</td>
<td>Roll rate</td>
<td>rad sec$^{-1}$, deg sec$^{-1}$</td>
</tr>
<tr>
<td>$P_D$</td>
<td>Dynamic pressure</td>
<td>knots (speed), lb ft$^{-2}$</td>
</tr>
<tr>
<td>$P_S$</td>
<td>Static pressure</td>
<td>ft (altitude), lb ft$^{-2}$</td>
</tr>
<tr>
<td>$P_T$</td>
<td>Total pressure</td>
<td>lb ft$^{-2}$</td>
</tr>
<tr>
<td>$q,Q$</td>
<td>Pitch rate</td>
<td>rad sec$^{-1}$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td>( \bar{q} )</td>
<td>Dynamic pressure</td>
<td>lb ft(^{-2})</td>
</tr>
<tr>
<td>( r, R )</td>
<td>Yaw rate</td>
<td>rad sec(^{-1})</td>
</tr>
<tr>
<td>RPM</td>
<td>Engine rotational speed</td>
<td>deg sec(^{-1})</td>
</tr>
<tr>
<td>([R])</td>
<td>Acceleration transformation matrix</td>
<td>ft(^2)</td>
</tr>
<tr>
<td>( S )</td>
<td>Wing area</td>
<td>ft(^2)</td>
</tr>
<tr>
<td>( t, T )</td>
<td>Time point</td>
<td>sec</td>
</tr>
<tr>
<td>( T )</td>
<td>Temperature</td>
<td>°F</td>
</tr>
<tr>
<td>( T_X )</td>
<td>Thrust force in X direction</td>
<td>lb</td>
</tr>
<tr>
<td>( u, U )</td>
<td>Speed (perturbed, total)</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( {u(t)} )</td>
<td>Control vector</td>
<td>mph</td>
</tr>
<tr>
<td>( v )</td>
<td>Perturbed sideward velocity</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( {v} )</td>
<td>MMLE variable bias vector</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( V_X, V_x )</td>
<td>Longitudinal velocity</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( V_Y, V_y )</td>
<td>Lateral velocity</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( V_Z, V_z )</td>
<td>Normal velocity</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( w )</td>
<td>Perturbed downward velocity</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( {x(t)} )</td>
<td>State vector</td>
<td>ft sec(^{-2})</td>
</tr>
<tr>
<td>( X )</td>
<td>Force in X direction</td>
<td>lb</td>
</tr>
<tr>
<td>( \bar{X} )</td>
<td>Distance in the X direction from the center of gravity</td>
<td>ft</td>
</tr>
<tr>
<td>( X_a )</td>
<td>Dimensional variation of X-force with angle of attack</td>
<td>ft sec(^{-2})</td>
</tr>
<tr>
<td>( X_u )</td>
<td>Dimensional variation of X-force with speed</td>
<td>sec(^{-1})</td>
</tr>
<tr>
<td>( X_{T_u} )</td>
<td>Dimensional variation of X-force due to thrust with speed</td>
<td>sec(^{-1})</td>
</tr>
</tbody>
</table>
**LIST OF SYMBOLS (continued)**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_{E,c}$</td>
<td>Dimensional variation of X-force with elevator or canard angle</td>
<td>ft sec$^{-2}$</td>
</tr>
<tr>
<td>$X_0$</td>
<td>Longitudinal force equation bias</td>
<td>ft sec$^{-2}$</td>
</tr>
<tr>
<td>$(y(t))'$</td>
<td>Computed observation vector</td>
<td></td>
</tr>
<tr>
<td>$y_i = (y(i))$</td>
<td>Computed observation vector at time i</td>
<td></td>
</tr>
<tr>
<td>$Y$</td>
<td>Force in Y direction</td>
<td>lb</td>
</tr>
<tr>
<td>$\bar{Y}$</td>
<td>Distance in Y direction from the center of gravity</td>
<td>ft</td>
</tr>
<tr>
<td>$Y_\beta$</td>
<td>Dimensional variation of Y-force with sideslip angle</td>
<td>sec$^{-1}$, ft sec$^{-2}$</td>
</tr>
<tr>
<td>$'\rho$</td>
<td>Dimensional variation of Y-force with roll rate</td>
<td>ft sec$^{-1}$</td>
</tr>
<tr>
<td>$Y_\gamma$</td>
<td>Dimensional variation of Y-force with yaw rate</td>
<td>ft sec$^{-1}$</td>
</tr>
<tr>
<td>$Y_{\delta_{A,R}}$</td>
<td>Dimensional variation of Y-force with aileron or rudder angle</td>
<td>ft sec$^{-2}$, sec$^{-1}$</td>
</tr>
<tr>
<td>$YO$</td>
<td>Lateral acceleration equation bias</td>
<td>sec$^{-1}$</td>
</tr>
<tr>
<td>$(z(t))$</td>
<td>Measured observation vector</td>
<td></td>
</tr>
<tr>
<td>$z_i = (z(i))$</td>
<td>Measured observation vector at time i</td>
<td></td>
</tr>
<tr>
<td>$Z = -N$</td>
<td>Force in the Z direction</td>
<td>lb</td>
</tr>
<tr>
<td>$\bar{Z}$</td>
<td>Distance in Z direction from the center of gravity</td>
<td>ft</td>
</tr>
<tr>
<td>$Z_a$</td>
<td>Dimensional variation of Z-force with angle of attack</td>
<td>ft sec$^{-2}$</td>
</tr>
</tbody>
</table>
### LIST OF SYMBOLS (continued)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Z'_{\alpha} )</td>
<td>Dimensional variation of Z-force with rate of change of angle of attack</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( Z_q )</td>
<td>Dimensional variation of Z-force with pitch rate</td>
<td>ft sec(^{-1})</td>
</tr>
<tr>
<td>( Z_u )</td>
<td>Dimensional variation of Z-force with speed</td>
<td>sec(^{-1})</td>
</tr>
<tr>
<td>( Z_{\delta E,c} )</td>
<td>Dimensional variation of Z-force with elevator or canard angle</td>
<td>ft sec(^{-2})</td>
</tr>
<tr>
<td>( Z_o )</td>
<td>Vertical force equation bias</td>
<td>ft sec(^{-1})</td>
</tr>
</tbody>
</table>

#### Greek Symbol

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha )</td>
<td>Angle of attack</td>
<td>rad</td>
</tr>
<tr>
<td>( \beta )</td>
<td>Angle of sideslip</td>
<td>rad</td>
</tr>
<tr>
<td>( \psi )</td>
<td>Euler heading angle</td>
<td>rad</td>
</tr>
<tr>
<td>( \theta )</td>
<td>Euler pitch angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \phi )</td>
<td>Euler roll angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \delta_o )</td>
<td>Bias in Euler pitch rate equation</td>
<td>rad sec(^{-1})</td>
</tr>
<tr>
<td>( \delta_{E,e} )</td>
<td>Elevator angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \delta_{A,a} )</td>
<td>Aileron angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \delta_{R,r} )</td>
<td>Rudder angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \delta_c )</td>
<td>Canard angle</td>
<td>deg, rad</td>
</tr>
<tr>
<td>( \rho )</td>
<td>Air density</td>
<td>slugs ft(^{-3})</td>
</tr>
<tr>
<td>( \phi_o )</td>
<td>Bias in Euler roll rate equation</td>
<td>rad sec(^{-1})</td>
</tr>
<tr>
<td>( \omega_{n SP} )</td>
<td>Undamped natural frequency of the short period mode</td>
<td>Hz</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
<td>Dimension</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td>( n_p )</td>
<td>Undamped natural frequency of the phugoid mode</td>
<td>Hz</td>
</tr>
<tr>
<td>( n_D )</td>
<td>Undamped natural frequency of the dutch roll mode</td>
<td>Hz</td>
</tr>
<tr>
<td>{n (t)}</td>
<td>Noise vector</td>
<td></td>
</tr>
<tr>
<td>( V_c )</td>
<td>First gradient with respect to c</td>
<td></td>
</tr>
<tr>
<td>( V_c^2 )</td>
<td>Second gradient with respect to c</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subscript</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>l</td>
<td>Initial</td>
</tr>
<tr>
<td>B</td>
<td>At body axis at center of gravity</td>
</tr>
<tr>
<td>M</td>
<td>As measured by transducer</td>
</tr>
<tr>
<td>I</td>
<td>As installed wrt body axis at center of gravity</td>
</tr>
<tr>
<td>L</td>
<td>Left hand</td>
</tr>
<tr>
<td>R</td>
<td>Right hand</td>
</tr>
<tr>
<td>s</td>
<td>Flight stability axes</td>
</tr>
<tr>
<td>w</td>
<td>Wind axes</td>
</tr>
<tr>
<td>wt</td>
<td>Wind tunnel stability axes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Superscript</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>^t</td>
<td>Transpose</td>
</tr>
<tr>
<td>^'</td>
<td>State vector derivatives</td>
</tr>
</tbody>
</table>

A dot over a quantity denotes the time derivative of that quantity.
Note: All directions are shown positive

Figure 1.1 Body axes system used in this report
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Body axes system used in this report</td>
<td>xxiii</td>
</tr>
<tr>
<td>1.1</td>
<td>Experimental configuration of the test airplane</td>
<td>4</td>
</tr>
<tr>
<td>3.1</td>
<td>Overall system block diagram</td>
<td>10</td>
</tr>
<tr>
<td>3.2</td>
<td>Major components of the airborne system</td>
<td>11</td>
</tr>
<tr>
<td>3.3</td>
<td>Block diagram of airborne system</td>
<td>12</td>
</tr>
<tr>
<td>3.4</td>
<td>Data transfer system</td>
<td>13</td>
</tr>
<tr>
<td>3.5</td>
<td>Block diagram of data transfer system</td>
<td>13</td>
</tr>
<tr>
<td>3.6</td>
<td>Battery and computer module installation</td>
<td>14</td>
</tr>
<tr>
<td>3.7</td>
<td>Transducer module installation</td>
<td>14</td>
</tr>
<tr>
<td>3.8</td>
<td>Computer module block diagram</td>
<td>17</td>
</tr>
<tr>
<td>3.9</td>
<td>MDAS-16 data acquisition card</td>
<td>18</td>
</tr>
<tr>
<td>3.10</td>
<td>Block diagram of MDAS-16 data acquisition module</td>
<td>19</td>
</tr>
<tr>
<td>3.11</td>
<td>TEAC MT2-02 digital cassette tape transport</td>
<td>19</td>
</tr>
<tr>
<td>3.12</td>
<td>Transducer pallet</td>
<td>26</td>
</tr>
<tr>
<td>3.13</td>
<td>Schaevitz Engineering LSB series accelerometers</td>
<td>28</td>
</tr>
<tr>
<td>3.14</td>
<td>Measured airframe vibration</td>
<td>30</td>
</tr>
<tr>
<td>3.15</td>
<td>Measured filter frequency response</td>
<td>30</td>
</tr>
<tr>
<td>3.16</td>
<td>Filtered and unfiltered in flight measurements</td>
<td>30</td>
</tr>
<tr>
<td>3.17</td>
<td>Humphrey VG-24 vertical gyroscope</td>
<td>32</td>
</tr>
<tr>
<td>3.18</td>
<td>Northrop 3-axis rate sensor</td>
<td>34</td>
</tr>
<tr>
<td>3.19</td>
<td>Space Age Controls linear displacement transducer</td>
<td>35</td>
</tr>
<tr>
<td>3.20(a)</td>
<td>Control position transducer mounting detail</td>
<td>36</td>
</tr>
<tr>
<td>3.20(b)</td>
<td>Aileron control position transducer</td>
<td>36</td>
</tr>
<tr>
<td>3.20(c)</td>
<td>Rudder and elevator control position transducers</td>
<td>37</td>
</tr>
</tbody>
</table>
**LIST OF FIGURES (continued)**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.21</td>
<td>Mounting technique for external devices</td>
<td>37</td>
</tr>
<tr>
<td>3.22</td>
<td>B&amp;D Instruments 2054 pressure transducer</td>
<td>39</td>
</tr>
<tr>
<td>3.23</td>
<td>Pitot tube</td>
<td>39</td>
</tr>
<tr>
<td>3.24(a)</td>
<td>Pitot tube mounted on airplane</td>
<td>40</td>
</tr>
<tr>
<td>3.24(b)</td>
<td>Pitot tube mounted on airplane</td>
<td>40</td>
</tr>
<tr>
<td>3.25</td>
<td>Pitot tube mounting location</td>
<td>41</td>
</tr>
<tr>
<td>3.26</td>
<td>Temperature transducer specifications</td>
<td>42</td>
</tr>
<tr>
<td>3.27</td>
<td>Temperature probe</td>
<td>42</td>
</tr>
<tr>
<td>3.28(a)</td>
<td>Temperature probe mounted on airplane</td>
<td>43</td>
</tr>
<tr>
<td>3.28(b)</td>
<td>Temperature probe mounted on airplane</td>
<td>43</td>
</tr>
<tr>
<td>3.29</td>
<td>Battery discharge curve</td>
<td>45</td>
</tr>
<tr>
<td>3.30</td>
<td>Battery module schematic and specifications</td>
<td>47</td>
</tr>
<tr>
<td>3.31</td>
<td>Pilot control console</td>
<td>49</td>
</tr>
<tr>
<td>3.32</td>
<td>Transducer monitor</td>
<td>49</td>
</tr>
<tr>
<td>4.1</td>
<td>Digital Equipment Corp. MINC 11/03 computer</td>
<td>57</td>
</tr>
<tr>
<td>4.2</td>
<td>Block diagram of MINC 11/03 computer</td>
<td>57</td>
</tr>
<tr>
<td>5.1</td>
<td>Data processing flow chart</td>
<td>60</td>
</tr>
<tr>
<td>5.2</td>
<td>Maximum likelihood estimation concept</td>
<td>66</td>
</tr>
<tr>
<td>6.1</td>
<td>Flight time history; Flight 19/10/80 Run 23A; Longitudinal</td>
<td>82</td>
</tr>
<tr>
<td>6.2</td>
<td>Flight time history; Flight 19/10/80 Run 23B; Longitudinal</td>
<td>83</td>
</tr>
<tr>
<td>6.3</td>
<td>Flight time history; Flight 19/10/80 Run 51A; Longitudinal</td>
<td>84</td>
</tr>
<tr>
<td>6.4</td>
<td>Flight time history; Flight 23/10/80 Run 11; Lateral</td>
<td>86</td>
</tr>
<tr>
<td>6.5</td>
<td>Flight time history; Flight 19/10/80 Run 26C; Lateral</td>
<td>87</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>6.6</td>
<td>Flight time history; Flight 19/10/80 Run 33B; Lateral.</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>Block diagram of Cessna spin test installation.</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>Cessna spin test airplane.</td>
<td></td>
</tr>
<tr>
<td>7.3(a)</td>
<td>Cessna wing tip booms (supplied by NASA Langley).</td>
<td></td>
</tr>
<tr>
<td>7.3(b)</td>
<td>Cessna wing tip booms (supplied by NASA Langley).</td>
<td></td>
</tr>
<tr>
<td>7.3(c)</td>
<td>Cessna wing tip booms (supplied by NASA Langley).</td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td>Cessna spin test instrumentation installation.</td>
<td></td>
</tr>
<tr>
<td>7.5</td>
<td>Cessna spin test computer installation.</td>
<td></td>
</tr>
<tr>
<td>7.6(a)</td>
<td>Cessna spin test, spin traces, spin No. 9.</td>
<td></td>
</tr>
<tr>
<td>7.6(b)</td>
<td>Cessna spin test, spin traces, spin No. 10.</td>
<td></td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Transducers Used in Various Flight Test Programs</td>
<td>22</td>
</tr>
<tr>
<td>3.2</td>
<td>Transducer Accuracies Used in Various Flight Test Programs</td>
<td>23</td>
</tr>
<tr>
<td>3.3</td>
<td>Transducer Accuracy and Range Used</td>
<td>24</td>
</tr>
<tr>
<td>3.4</td>
<td>Power Requirements for Data Acquisition Package</td>
<td>47</td>
</tr>
<tr>
<td>4.1</td>
<td>Integer Speed Routine (FORTRAN LISTING)</td>
<td>52</td>
</tr>
<tr>
<td>4.2</td>
<td>Integer Speed Comparison</td>
<td>53</td>
</tr>
<tr>
<td>4.3</td>
<td>Floating Point Speed Routine (FORTRAN LISTING)</td>
<td>54</td>
</tr>
<tr>
<td>4.4</td>
<td>Floating Point Speed Comparison</td>
<td>55</td>
</tr>
<tr>
<td>5.1</td>
<td>Qualified Data Cassettes</td>
<td>62</td>
</tr>
<tr>
<td>5.2a</td>
<td>Longitudinal Dimensional Stability Derivatives</td>
<td>68</td>
</tr>
<tr>
<td>5.2b</td>
<td>Lateral-Directional Dimensional Stability Derivatives</td>
<td>69</td>
</tr>
<tr>
<td>5.3a</td>
<td>Longitudinal Dimensional State Vector Stability Derivatives</td>
<td>72</td>
</tr>
<tr>
<td>5.3b</td>
<td>Lateral Dimensional State Vector Stability Derivatives</td>
<td>73</td>
</tr>
<tr>
<td>5.4</td>
<td>Matrices Used in Observation Equation</td>
<td>74</td>
</tr>
<tr>
<td>6.1</td>
<td>Cessna 172 Flight Test Conditions</td>
<td>80</td>
</tr>
<tr>
<td>6.2</td>
<td>Comparison of Results, Longitudinal</td>
<td>85</td>
</tr>
<tr>
<td>6.3</td>
<td>Comparison of Results, Lateral</td>
<td>89</td>
</tr>
<tr>
<td>7.1</td>
<td>Cessna Spin Test Instrumentation Requirements</td>
<td>93</td>
</tr>
<tr>
<td>B.1</td>
<td>Designation of Force and Moment Coefficients for Different Axes Systems</td>
<td>182</td>
</tr>
<tr>
<td>B.2</td>
<td>Comparison of Non-Dimensional Derivatives</td>
<td>184</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

This report describes work completed during the second phase of a continuing program sponsored by the NASA Dryden Flight Research Center.* This program was accomplished during the period January 21, 1979, through February 15, 1981. The program encompasses the development of a simple, self-contained flight test data acquisition system. To date the program has consisted of two phases:

PHASE I

* A literature survey of flight testing methods (presented in Reference 1).
* The development and testing of a proof-of-concept system capable of longitudinal stability analysis (presented in Reference 2).

PHASE II

* Development and testing of a system capable of longitudinal and lateral stability analysis.

This report describes in detail the system concepts selected, as well as results of the flight test program used to show the validity of these concepts, as of the completion of Phase II.

The purpose of this project, and the design criteria developed are contained in Chapter 2. The literature survey (Reference 1) has been used as a primary data base for establishment of these

*Funding provided under NASA Grant NSG 4019 (FRL/CRINC 4070).
criteria. Other inputs have come from talks with personnel in the general aviation industry and of NASA Dryden Flight Research Center.

Chapter 3 describes the hardware selected and manufactured to meet the design requirements.

The instrumentation package employs transducers to allow both longitudinal and lateral stability analysis of general aviation type airplanes, although it can easily encompass most other types of airplanes. Due to the nature of the data reduction method utilized, a minimum number of high-accuracy transducers are required. Data from the transducers are recorded using an on-board microprocessor and digital cassette recorder. This has proven a simple, reliable method to obtain accurate flight data.

The system has been designed to allow it to be placed in the aircraft with a minimum amount of aircraft modification. A rechargeable battery pack was selected for airborne power to reduce the number of airplane modifications required. This has allowed total isolation from the aircraft electrical system, which simplifies installation, enhances safety, and eliminates many electrical noise problems in the transducer signals. The transducers are all contained in one module, except for the following:
- total pressure probe,
- temperature probe,
- static pressure probe, and
- control position transducers.

A minimum of installation is also required for these devices, as they are literally "sticky-taped" to the airframe.
Presented in Chapter 4 is an evaluation procedure used to select a ground-based data reduction computer. Due to the extensive mathematical procedure used for data analysis, a powerful, high-level language microcomputer is required. The evaluation method used is described here, as well as the computer selected and used for this program.

The total flight testing process is included as Chapter 5. Discussed are the various computer programs and operating techniques developed. The heart of this system is the Modified Maximum Likelihood Estimation (MMLE) method which has been used for data reduction. The mathematics of this technique are included as Section 5.6.

The flight test program used for system development is included in Chapter 6. Tests have been performed using the KU-FRL* Cessna 172 airplane (shown in Figure 1.1). The type of flight test maneuver required is discussed, and results of the actual flight testing are presented.

A flight test program was conducted at Cessna Aircraft to evaluate the spin properties of their model 172 airplane. The data management portion of the KU-FRL system was used in conjunction with Cessna-supplied transducers for data acquisition and analysis. This program is described in Chapter 7.

Conclusions to be drawn as a result of the work carried out under this program, and recommendations for further work are included in Chapters 8 and 9.

*KU-FRL = University of Kansas Flight Research Laboratory.
References, and reports describing this project are presented in Chapter 10.

Appendix A includes descriptions of all programs required for system operation.

There appears to be some confusion over the many reference axes systems used in airplane analysis. Included in the list of symbols is Figure 1.1, which explicitly defines the axes system used in this report. Appendix B is included to allow conversion of results in this report to several other standard axes definitions.
The system constructed under this program has proven under flight test the validity of the concept selected for longitudinal and lateral stability analysis. Throughout this flight test program, using the Cessna 172 airplane, the acquisition package performed reliably.
2. PURPOSE OF PROJECT

Flight testing has always required a high degree of complex instrumentation to get accurate results. This, in the past, and still evident today, has taken a great deal of time and money to equip each individual flight test article. Traditional systems are placed on aircraft on an individual basis, utilizing what is available at that time, coupled with the specific requirements of a particular test program. This has never really led to ideal or totally thought-out systems, and normally results in high costs or in too much time being required for instrumenting the airplane.

With the accurate instrumentation available today, and with the recent advances in microcomputer technology, it was seen that an accurate, multipurpose data acquisition system could be developed. The system described here has been developed to do just that.

The basis for design of this system is as laid out here.

EASE OF INSTALLATION - This has been a major design consideration. If possible, NO permanent modification should be done to the airplane. The system must be universally easy to install and should require a minimum of installation time and no special procedures. This factor includes calibration of the system installed on the airplane.

SELF-CONTAINED - The system should be totally self contained. This should include all data sources, data recording methods, power requirements and data reduction techniques.
SIMPLE - The system must be simple in concept and easy to use. The need for complex instrumentation, difficult calibration, and specialized operator knowledge must be kept to a minimum.

FLIGHT TESTING - The system should not require any specialized piloting techniques to obtain accurate results.

CLASS OF AIRCRAFT - The system to be developed is primarily applicable to the general aviation type airplane. This criterion does not restrict the methods and theories, but it does define the requirements for the transducer ranges and accuracies.

RESULTS - The system is aimed at stability and performance parameter identification, but it must permit adaptation to other test requirements.

COSTS - The system should meet all of the above requirements, yet reduce the expenditure required for the instrumentation system as compared with current methods.

The system described in this report has been developed to prove that the concepts selected meet the above design requirements.
3. **INSTRUMENTATION SYSTEM**

The system described and constructed under this phase meets the objectives stated in Chapter 2. The instrumentation system can be broken up into four parts:

1) Data Management;
2) Transducers;
3) Power Supply;
4) Pilot Control.

The package is shown in the block diagram of Figure 3.1. The system is used in two forms: airborne for recording of flight data (Figures 3.2 and 3.3), and the ground-based portion for data transfer to the data reduction computer (Figures 3.4 and 3.5).

Installation of this system is straightforward and requires no permanent modifications to the airplane. The major modules are shown installed in the KU-FRL's Cessna 172 in Figures 3.6 and 3.7. The other components are shown installed on the airplane as they are described in Section 3.2. It is seen that the major modules are essentially strapped into the cabin compartment. The transducer module does require a more rigid attachment and is, therefore, held firmly in place by clamping it to the seat tracks.

Following is a detailed description of the instrumentation system, as well as the trade-offs considered in its design.
Figure 3.1 Overall system block diagram
WEIGHTS

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight (lb.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATTERY MODULE</td>
<td>60.5</td>
</tr>
<tr>
<td>COMPUTER MODULE</td>
<td>34.5</td>
</tr>
<tr>
<td>TRANSUDCER MODULE</td>
<td>32.5</td>
</tr>
<tr>
<td>CONTROL POSITION TRANSDUCERS *</td>
<td>1.5</td>
</tr>
<tr>
<td>PITOT PROBE *</td>
<td>0.2</td>
</tr>
<tr>
<td>TEMPERATURE PROBE *</td>
<td>0.2</td>
</tr>
<tr>
<td>PILOT CONTROL CONSOLE</td>
<td>1.0</td>
</tr>
<tr>
<td>MISCELLANEOUS (cables,clamps,etc.)*</td>
<td>2.0</td>
</tr>
<tr>
<td><strong>TOTAL:</strong></td>
<td><strong>132.4</strong></td>
</tr>
</tbody>
</table>

*not shown

Figure 3.2 Major components of the airborne system
Figure 3.3 Block diagram of airborne system
Figure 3.4 Data transfer system

Figure 3.5 Block diagram of data transfer system
Figure 3.6 Battery and computer module installation

Figure 3.7 Transducer module installation
3.1 Data Management

It was decided to use a microprocessor controlled data management system. Using a commercially available computer simplified the design task, as well as reducing overall system complexity and cost. Also with this type of controller, versatility is easily achieved, especially if programs are stored on cassette tape rather than in computer hardware. The trade-offs considered prior to selecting this data management system were those of analog vs digital data storage, and airborne recording vs telemetry. Following is discussion of these trade-offs and a detailed description of the system constructed.

In the past, most on-board systems made use of analog recording, due primarily to the high cost and complexity of digital systems. In recent years, however, progress has been made in the digital field, resulting in small, inexpensive, and reliable digital devices, most available as solid-state integrated circuits. The recent advances in digital electronics technology have reduced both the complexity and cost. Coupling this with the lower likelihood of error in digital systems, it was decided to use a totally digital system for this package.

In the past, telemetry has been a much-used means of transmitting data to be recorded on the ground. Telemetry has an important place in aircraft flight testing, specifically in high-risk operations (such as flutter testing, spin testing, etc.). Its major disadvantages are the requirement of a ground station, and the associated high cost and complexity. Telemetry, however, has been primarily used in the
past, due to the large size, complexity, and inaccuracies of the older recording media. Many improvements have been made in this regard with the introduction of small, reliable cartridge and cassette recording systems. This improvement is largely attributed to the recent advances in solid state electronics technology. For this system, on-board recording, making use of a digital cassette recorder, has been chosen.

The heart of the unit constructed is a Rockwell AIM 65 microcomputer. This is coupled through a Rockwell expansion interface to the other components. The other two major components of the airborne package’s recording system are the Datel MDAS-16 multiplexer and analog-to-digital converter, the TEAC MT2-02 digital cassette tape transport, and RS232* interfacing port. These are shown in the block diagram of Figure 3.8.

The AIM-65 is an interactive single board computer using an 8-bit 6502 microprocessor. Contained on the computer board is 4K bytes of memory, as well as a monitor and symbolic assembler. (An 8K BASIC programming ROM** is also available for this computer.) A 20 character display, 20 column thermal printer and alphanumeric keyboard allow the user to interact with the computer. Two application connectors increase the computer's versatility. One allows interfacing to audio cassette recorder and other computer terminals. The second allows adding an expansion interface which facilitates

*RS232 = serial interfacing standard.

**ROM = read only memory.
additional features to be adapted to the standard computer. These features provide an easy-to-use Data Management Controller.

The user is able to easily program the computer using the symbolic assembler and monitor functions provided. Programs presently are stored on the audio cassette recorder. Using the additional ROM slots on the computer, or the addition of a ROM board on the expansion interface would allow regularly used programs to be permanently placed in the system.

The AIM-65 is coupled, through the expansion board, by use of the MDAS-16, to the transducer package. The MDAS-16 is a 16-channel multiplexer coupled with a 12-bit analog-to-digital converter. This unit has the capability of addressing channels as desired.
(either randomly or sequentially), using a microprocessor controller. Voltage input ranges can be selected (-5 volt to +5 volt was chosen for this system). The unit has a 50 KHz through-put rate with 20 µ sec access time per channel. The MDAS-16 is shown in Figures 3.9 and 3.10. The MDAS-16 does require calibration. This procedure is described in detail in Reference 3.

The other major component of the data acquisition system is the TEAC tape transport (see Figure 3.11). This unit is a low-cost magnetic tape unit designed specifically for digital applications. It makes use of standard audio type cassette tapes for data storage. All interfacing required is included in the
Figure 3.10 Block diagram of MDAS-16 data acquisition module

Figure 3.11 TEAC MT2-02 digital cassette tape transport
package. Input requirements are TTL*-compatible; and the tape unit
requires only control signals, provided by the AIM-65 microcomputer,
and parallel data input. All detailed control functions required
by the tape unit are handled on board by the unit for both recording
and playback. Only simple control signals are required to initiate
the various functions.

The data management system is also used for data playback and
transfer to the ground-based data reduction computer. It was decided
to use the same recorder and computer system for playback of data
and in-flight recording. This avoids possible problems due to mis-
match of tape drives and also reduces overall system costs. An
interface system compatible with standard computer RS232 ports was
designed and constructed. A hard wire connection, or use of a modem
through the telephone can thus be utilized. This type of interface
allows data transfer to virtually any computer. A program on the
AIM-65 controls the TEAC tape transport and sends the data over
the line to the other computer. Once all the data are on the
other computer, the Rockwell system is no longer required in the
data reduction process. (See Chapter 5 for a complete description
of the data reduction process.)

3.2 Transducers

At the outset of this program, it was decided to keep the
number of containers in the total system to a minimum. Thus, most
transducers, as well as their required signal conditioning, are

*TTL = Transistor-Transistor-Logic: Electrical standard.
contained in the transducer module. This module contains all filtering, all voltage regulation, and the transducer pallet.

It was not possible to place all transducers in this module, as measurements such as control positions and outside air conditions were required to be measured. Following is a description of the methods used for selecting transducers required, as well as descriptions of the actual equipment selected.

The primary input to aid in the selection of the parameters to be measured was the literature describing the data reduction methods to be used (References 4-20). The transducers discussed in the references above are summarized in Table 3.1. Discussion with personnel at NASA, Dryden Flight Research Center, was the secondary input for transducer selection. The transducers selected allow optimal use of the data reduction technique considered (basically a maximum likelihood parameter estimation method; see Chapter 5 for a detailed description of this method).

The literature (References 4-17 and 20) was also used as the primary reference for selection of transducer accuracies required. The results are summarized in Table 3.2. The transducer ranges were selected after discussion with the general aviation manufacturers (the secondary reference), and consideration of the performance characteristics of this class of airplane.

The ranges and accuracies required for the various transducers selected are summarized in Table 3.3.
Table 3.1 Transducers used in various flight test programs

<table>
<thead>
<tr>
<th></th>
<th>SHEMARA ref. 16</th>
<th>DELFT ref. 4-11</th>
<th>SORENSON ref. 19</th>
<th>BORSES PROGRAM ref. 15</th>
<th>KLEIN ref. 17</th>
<th>SELECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_x )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( A_y )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( A_z )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( V_s )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( V_y )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( V_s )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>Alt.</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>Temp.</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \theta )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \phi )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \psi )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( p )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( q )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( r )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \dot{\beta} )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \dot{\delta}_k )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \dot{\delta}_A )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \delta_R )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>RPM</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>M.P.</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( P_S )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( P_T )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( P_D )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( a )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \beta )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
<tr>
<td>( \rho )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
<td>( * )</td>
</tr>
</tbody>
</table>

- \( * \) indicates that the transducer is used.
- \( \text{N.R.} \) indicates that the transducer is not reported.

\( \text{can be derived} \)

- \( \text{not normally needed} \)

- \( \text{may be req'd or desirable for performance data} \)

- \( \text{can be derived} \)
### TABLE 3.2 Transducer accuracies used in various flight test programs

<table>
<thead>
<tr>
<th></th>
<th>DELFT ref: 4-11</th>
<th>ECKHOLD ref: 6-20</th>
<th>KLEIN ref: 17</th>
<th>ILIFFE &amp; MAINE ref: 12-16</th>
<th>SELECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_x$</td>
<td>.001 g</td>
<td>.002 g</td>
<td>.005 g</td>
<td></td>
<td>.002 g</td>
</tr>
<tr>
<td>$A_y$</td>
<td>.02 g</td>
<td>or</td>
<td></td>
<td></td>
<td>.002 g</td>
</tr>
<tr>
<td>$A_z$</td>
<td>.02 g</td>
<td>2 %</td>
<td></td>
<td></td>
<td>.002 g</td>
</tr>
<tr>
<td>$\theta$</td>
<td>1/2 °</td>
<td>.2° or</td>
<td></td>
<td></td>
<td>.5°</td>
</tr>
<tr>
<td>$\phi$</td>
<td>1/2 °</td>
<td>2 %</td>
<td></td>
<td></td>
<td>.5°</td>
</tr>
<tr>
<td>$p$</td>
<td>.15°/sec</td>
<td>.2°/sec</td>
<td></td>
<td></td>
<td>.5°/sec</td>
</tr>
<tr>
<td>$q$</td>
<td>.02°/sec</td>
<td>.15°/sec</td>
<td>or</td>
<td></td>
<td>.5°/sec</td>
</tr>
<tr>
<td>$r$</td>
<td>.15°/sec</td>
<td>2 %</td>
<td></td>
<td></td>
<td>.5°/sec</td>
</tr>
<tr>
<td>$\delta_E$</td>
<td>.4°</td>
<td>.2°</td>
<td></td>
<td></td>
<td>.5°</td>
</tr>
<tr>
<td>$\delta_A$</td>
<td>.4°</td>
<td>or</td>
<td></td>
<td></td>
<td>.5°</td>
</tr>
<tr>
<td>$\delta_R$</td>
<td>.4°</td>
<td>2 %</td>
<td></td>
<td></td>
<td>.5°</td>
</tr>
<tr>
<td>$T$</td>
<td>2° F</td>
<td></td>
<td></td>
<td></td>
<td>2° F</td>
</tr>
<tr>
<td>$P_S$</td>
<td>.1 m, 160 ft</td>
<td>10 ft</td>
<td></td>
<td></td>
<td>10 ft</td>
</tr>
<tr>
<td>$P_D$</td>
<td>5 knots</td>
<td>2 knots</td>
<td></td>
<td></td>
<td>2 knots</td>
</tr>
</tbody>
</table>
Table 3.3 Transducer Accuracy and Range Used

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Sensor</th>
<th>Accuracy</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_X$</td>
<td>longitudinal acceleration</td>
<td>.002 g</td>
<td>±1 g</td>
</tr>
<tr>
<td>$A_Y$</td>
<td>lateral acceleration</td>
<td>.002 g</td>
<td>±0.5 g</td>
</tr>
<tr>
<td>$A_N$</td>
<td>normal acceleration</td>
<td>.002 g</td>
<td>-1.5 g to 4 g</td>
</tr>
<tr>
<td>$\theta$</td>
<td>pitch angle</td>
<td>0.5°</td>
<td>±30°</td>
</tr>
<tr>
<td>$\phi$</td>
<td>roll angle</td>
<td>0.5°</td>
<td>±30°</td>
</tr>
<tr>
<td>$p$</td>
<td>pitch rate</td>
<td>0.5°/sec</td>
<td>±50°/sec</td>
</tr>
<tr>
<td>$q$</td>
<td>roll rate</td>
<td>0.5°/sec</td>
<td>±50°/sec</td>
</tr>
<tr>
<td>$r$</td>
<td>yaw rate</td>
<td>0.5°/sec</td>
<td>±50°/sec</td>
</tr>
<tr>
<td>$\delta_E$</td>
<td>elevator position</td>
<td>0.5°</td>
<td></td>
</tr>
<tr>
<td>$\delta_A$</td>
<td>aileron position</td>
<td>0.5°</td>
<td></td>
</tr>
<tr>
<td>$\delta_R$</td>
<td>rudder position</td>
<td>0.5°</td>
<td></td>
</tr>
<tr>
<td>$T$</td>
<td>temperature</td>
<td>*</td>
<td>2°F -65 to +120°F</td>
</tr>
<tr>
<td>$P_S$</td>
<td>static pressure</td>
<td>*</td>
<td>10 feet 0 to 25K feet</td>
</tr>
<tr>
<td>$P_D$</td>
<td>dynamic pressure</td>
<td>*</td>
<td>2 knots 40 to 150 knots</td>
</tr>
</tbody>
</table>

*Indicates transducers used to define initial and final conditions.

During a specific maneuver, $T$, $P_S$ and $P_D$ need only be measured at the start and finish to define the initial and final conditions. The other 11 channels require measurement throughout the maneuver to determine the dynamic characteristics and analyze stability and performance properties of the airplane.

To select the data acquisition rate required, the following factors must be considered:

- Minimum rate must be higher than the undamped natural frequency of the airplane to be tested.
- Minimum rate must be high enough to avoid time skewing of the data points.
- Minimum rate must be as low as possible to allow economy in the recording media and data reduction process.
In data analysis, to obtain reasonable representations of the frequency response, an acquisition rate of at least five times the undamped natural frequency should be used (Reference 21, Volume 1, Chapter 6). In the class of aircraft considered for this instrumentation system, the natural frequencies are of the following order (from Reference 22):

\[
\begin{align*}
\omega_{n_{SP}} & : 0.5 - 1.0 \text{ Hz} \\
\omega_{n_{P}} & : 0.01 - 0.03 \text{ Hz} \\
\omega_{n_{D}} & : 0.25 - 0.60 \text{ Hz}.
\end{align*}
\]

Therefore, the maximum frequency (\(\omega_{n_{SP}}\)) requires an acquisition rate of

\[
1.0 \times 5 = 5 \text{ samples/sec}. *
\]

This is the minimum data requirement.

From References 12 and 14 and discussion with the authors it was determined that an acquisition rate of 100/sec is required to avoid time skewing problems. From the practical applications of the maximum likelihood estimation method, this rate (100/sec) also results in an excess of data that unnecessarily increases the computation time and costs.

Using a computer-controlled acquisition system allows scanning of the transducers as rapidly as possible (20 \(\mu\) sec/channel, 220 \(\mu\) sec total **), and then waiting until the next data point is required.

---

*10 samples/sec was chosen for the KU-FRL system, as this then definitely meets the minimum data requirement. This rate also seems to be somewhat of an acceptable industry standard.

**Values for the KU-FRL system.
(0.1 sec later*). These data are temporarily stored in memory and then output to the TEAC tape. This technique allows a high scanning rate to avoid time skewing between channels (equivalent to 4545/sec*) and a low overall acquisition rate (10/sec*) to provide economy and still satisfy the minimum data requirement.

The transducers were primarily mounted on one pallet. This is shown in Figure 3.12. It was possible to include most transducers on one pallet with the exception of the
- pitot tube,
- temperature probe,
- static cone, and
- control position transducers.

Figure 3.12 Transducer pallet

*Values for the KU-FRL system.
The pallet, contained within the transducer module, was mounted as close to the center of gravity of the airplane as possible. In this flight test program the transducer module has been clamped to the seat tracks of the Cessna 172, in the copilot's position.

---

Following are descriptions of the individual transducers used in this program.

3.2.1 Accelerometers

The accelerometers used in this package are of the force feedback (or closed loop) type. This type of accelerometer derives its measurement from determining the force required to maintain a mass at a zero location. This technique reduces the errors caused by mass displacement and also does not rely on springs (and their associated inaccuracies) as do the displacement (or open loop) type accelerometers. The disadvantage to the force feedback accelerometer is its relatively high cost.

It is essential to note that linear (as opposed to vibration) accelerometers be used for this type of package.

The accelerometers chosen are manufactured by Schaevitz Engineering. Their specifications are shown in Figure 3.13. These accelerometers are intended for the measurement of linear accelerations such as required for guidance control systems, or vehicle ride analysis. Both a precision sensor and electronics are integrated into the ac-
### Specifications at 20°C

<table>
<thead>
<tr>
<th>Range (g)</th>
<th>Nominal Natural Frequency (Hz)</th>
<th>Nominal Output Impedance (kilohms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>±0.25</td>
<td>50</td>
<td>20</td>
</tr>
<tr>
<td>±1.0</td>
<td>100</td>
<td>5</td>
</tr>
<tr>
<td>±2.0</td>
<td>110</td>
<td>2.5</td>
</tr>
<tr>
<td>±5.0</td>
<td>125</td>
<td>5</td>
</tr>
<tr>
<td>±10.0</td>
<td>140</td>
<td>2.5</td>
</tr>
<tr>
<td>±20.0</td>
<td>160</td>
<td>5</td>
</tr>
<tr>
<td>±50.0</td>
<td>200</td>
<td>5</td>
</tr>
</tbody>
</table>

- **Input Voltage**: ±15V DC nominal
- **Input Current**: 10 mA DC maximum (6mA DC average)
- **Full-Range Open-Circuit Output Voltage**: ±5.0V DC
- **Damping Ratio**: 0.6 typical (0.3 to 1.0 on request)
- **Linearity (Notes 1 & 2)**: ±0.05% of full scale output
- **Hysteresis (Note 2)**: 0.02% of full scale
- **Resolution (Note 2)**: 0.0005% of full scale
- **Cross-Axis Sensitivity (Note 3)**: ±0.002 g per g up to ±10 g range, inclusive
- **Bias**: ±0.005 g per g over ±10 g range
- **Bias Less than 0.1% of full scale**
- **Sensitive Axis to Case Alignment**: ±1°
- **Noise Output**: 5mV rms maximum
- **Operating Temperature**: -40°C to +95°C
- **Storage Temperature**: -55°C to +105°C
- **Thermal Coefficient of Sensitivity**: 0.02% per °C
- **Thermal Coefficient of Bias**: 0.002% per °C
- **Shock Survival**: 100 g – 11 ms
- **Weight**: 3 oz.

---

**Figure 3.13** Schaevitz Engineering LSB series accelerometers
celerometer case. Interfacing is relatively simple, requiring only a DC input voltage, and then a measurement of the DC voltage output, corresponding to the acceleration sensed.

3.2.2 Filtering

The response characteristics of the accelerometers were such that they picked up the aircraft vibration caused by the engine.

The graph of Figure 3.14 shows the airframe vibration characteristics (measured using the accelerometers as transducers, and observing the output on an oscilloscope) as a function of engine speed. It is obvious from these curves that the vibration is caused by the engine and is a function of the engine speed. Also of note is the fact that all the vibration is at a frequency above 40 Hz.

A low pass filter with a cutoff frequency at 10 Hz would eliminate this vibration from the measurement signal. Using a two-pole, active filter with a response as shown in Figure 3.15 virtually eliminated this unwanted vibrational noise, yet leaves the desired measurement (occurring in the order of 1 Hz) essentially unchanged. (The measurements of the $A_N$ accelerometer are presented as filtered and unfiltered measurements in Figure 3.16 to show this.)

In general, as was the case with this instrumentation package and the Cessna installation (see Chapter 7), only the accelerometers required any filtering.
Figure 3.14 Measured airframe vibration

Figure 3.15 Measured filter frequency response

Figure 3.16 Filtered and unfiltered in flight measurements
One drawback of filtering signals is the introduction of a phase shift due to the filter. To counter this problem, all signals should be filtered the same amount, thus eliminating the problems of the phase shift.

3.2.3 Attitude Gyro

Both roll attitude and pitch attitude are obtained from a Humphrey VG-24 vertical gyroscope. Full specifications for this gyro are shown in Figure 3.17. This is a DC gyro, with potentiometers for determining the measurement (28 volt DC used for the motor, ±5 volt DC used for potentiometer excitation). This gyro has operated reliably during both phases of this program.

3.2.4 Rate Gyros

A three-axis DC/DC rate gyro package was used for roll-, pitch-, and yaw-rate measurement. The advantage of using a three-axis package rather than three separate gyros is that alignment for orthogonality upon installation is eliminated. Of course, failure of a single gyro will require the entire package to be removed for repair.

The gyros selected are of the displacement type (or open circuit). Closed circuit (or integrating gyros) will provide better accuracy; however, cost of these is approximately 10 times higher. The accuracy of a good quality displacement type gyro will meet the requirement (see Tables 3.2 and 3.3), especially considering the type of power input used (free of oscillations or any high frequencies).
SPECIFICATIONS

RANGE - MECHANICAL

- ELECTRICAL

OUTPUT

STATIC ERROR BAND

RESISTANCE

CONTACT RESISTANCE

RESOLUTION

POWER DISSIPATION

WIPER CURRENT

ELECTRICAL REQUIREMENTS

SPIN MOTOR

VOLTAGE

CURRENT - STARTING

- RUNNING

ERUPTION

VOLTAGE

CURRENT

PERFORMANCE

SPIN MOTOR TIME TO SPEED

TIME TO EFFECT FROM MOTOR OFF

NORMAL OPERATING EJECTION RATE

VERTICAL ACCURACY

FREE DRIFT RATE

ENVIRONMENTAL CONDITIONS

VIBRATION

SHOCK

ACCELERATION - NON OPERATING

- OPERATING

TEMPERATURE - OPERATING

- STORAGE

ALTITUDE

SEA WATER IMMERSION

MILDITTY

SALT SPRAY

SAND AND DUST

FUNGUS

EXPLOSION PROOF

RADIO NOISE INTERFERENCE

SERVICE LIFE

SHELF LIFE

INSULATED RESISTANCE

WEIGHT

SEALING

Figure 3.17 Humphrey VG-24 vertical gyroscope

vertical accuracy of $2.0^\circ$ shall be maintained during vibration of 0.01 inch D.A. at 5 to 65 Hz; 5g, 65 to 500 Hz.
15g; 11 axes; all axes
30g; 1 min; vertical axis
10g; 1 min; applied in pitch or roll axis shall not produce a drift of greater than 10 $^\circ$/min.

-60 to +165°F
-40 to +125°F
sea level to 40000 ft
3 ft for 3 hrs.
to 95% including condensation for 240 hrs.
as encountered on shipboard or at coastal regions
as encountered in desert regions external surfaces non-nutritive shall not produce an explosion when operated in a fuel vapor rich area
MIL-1-6161; paragraph 4.3.1 & 4.3.2
100 hrs minimum
3 yrs minimum
20 megohms minimum at 100 volts DC motor circuit exempt
3.0 lb. maximum shall not leak under vacuum equivalent to 40000 ft.
The gyros selected are manufactured by Northrop. Voltage input required is 28 volts DC, and output voltage is from -5 to +5 volts DC. The gyros are Northrop G5 subminiature rate sensors. The gyro package specifications are included in Figure 3.18.

3.2.5 Control Position Transducers

Linear displacement transducers manufactured by Space-Age Control, Inc., were used to measure elevator position. This transducer is depicted in Figure 3.19. Due to the small size of this unit, it was decided to place it externally on the airframe. These transducers are installed as shown in Figures 3.20.

A novel technique for attaching the control position transducer (as well as the total pressure tube and temperature probe) has been used. Double-sided foam tape attaches the external devices onto the airframe. The mounting technique is depicted in Figure 3.21. The mounting method was first tested in the KU-FRL subsonic wind tunnel for wind speeds up to 119 mph. The tests in the tunnel were run for periods of up to 4 hours, with no degradation in rigidity of the mount (see Reference 23). The method has proven to give excellent results in the flight test program. The tape used is 3M number 4265 neoprene foam, the properties of which are included in the table on Figure 3.21.

It was anticipated that the mounting locations for the control position transducers would result in non-linear calibration curves. However, the calibration curves appeared to have a linear character.
Weight
2.0 lb. (max)

Outline dimensions
3.75 x 3.75 x 2.13 in

Power input
13 w. (max) (31 vdc)

Input voltage limits
±3 vdc

Full-scale output
±5 vdc

Output impedance
5000 ohms (max)

Output load resistance
500K ohms (nominal)

Ripple
25 mv. peak-peak (max)

Zero rate setting
±1/2 ± FS

Input range
50°/sec

Maximum input rate
600°/sec

Output voltage
±7 vdc

Output stability
1/2 ± FS

Input voltage variations
1 ± FS

Repeatability
0.01 °/sec

Threshold
0.01 °/sec

Resolution
0.1 °/sec

Hysteresis
0.1 °/sec

Operating temperature
0 - 160 °F

Temperature sensitivity
Zero output: 12 FS/100°F
Scale Factor: 3 FS/100°F

Warm up time
10 min

Motor acceleration time
30 sec (max)

±2° typical

Acceleration sensitivity
Linear
Angular

Linearity

Service life
100 hr (typical 14000 hr)

Insulation resistance
10 mohms (min), 30 vdc

Damping ratio
0.5 to 0.9

Natural frequency
35 Hz (min)

Ripple
25 mv. peak-peak (max)

Zero rate setting
1/2 ± FS

Threshold
0.01 °/sec

Resolution
0.01 °/sec

Hysteresis
0.1 °/sec

Operating temperature
0 - 160 °F

Temperature sensitivity
Zero output: 12 FS/100°F
Scale Factor: 3 FS/100°F

Environments

Shock

Vibration

Storage temperature

Radio interference

Figure 3.18 Northrop 3 - axis rate sensor
SAC Linear Displacement Transducers (LDT) consist of an extension cable, spirally wound on a spring-loaded rewind drum, which is coupled to a precision, wire-wound, rotary potentiometer. The cable end is attached to the object whose movements are to be monitored. As the cable is extended or retracted, the cable drum rotates the potentiometer wiper, varying the voltage at the wiper tap (No. 2) of the potentiometer. The voltage may be measured to reflect the position, direction, or rate of motion of the object attached to the cable.

Figure 3.19 Space Age Controls linear displacement transducer
Figure 3.20(a) Control position transducer mounting detail

Figure 3.20(b) Aileron control position transducer
Figure 3.20(c) Rudder and elevator control position transducers

Figure 3.21 Mounting technique for external devices

NOTE:
- lightly sand surface of airplane
- clean with isopropyl alcohol
- surface must be room temperature during attachment
- fair with duct tape

3M #4265 - DOUBLE COATED NEOPRENE FOAM TAPE

- Adhesive: A-20 Firm Acrylic
- Thickness: 3/64 in.
- Tensile: 60 psi
- Static Shear: 66 psi
- Temp max.: 225 °F
- Temp min.: -20 °F
(linear regression correlation coefficient of between 0.9976 and 0.9998) for the mounting locations used.

3.2.6 Static and Dynamic Pressure Transducer

A B&D Instruments Company 2504 series transducer (see Figure 3.22) was used for the static and dynamic pressure measurement. This device includes its own signal conditioning and converts the pressures to electrical signals utilizing semiconductor pressure transducers. Semiconductor transducers are largely affected by the ambient temperature; the B&D unit allows for this by heating the case and maintaining a constant temperature.

The pitot tube was designed and constructed according to Reference 24. (See Figure 3.23.) The pitot tube is attached to the underside of the wing (see Figures 3.24) using the foam tape method shown in Figure 3.21. The pitot tube allows a high angularity of the flow and still provides true readings. The distance from the wing is such that the tube is out of the boundary layer and thus provides a true total pressure reading as long as the pitot tube axis is close to the direction of airflow (±15°). The tube is mounted along the wing, halfway between the propeller arc and the wing tip (see Figure 3.25). This location minimizes flow effects due to the propeller slip stream and the wing tip vortices.

For the accurate measurement of static pressure, a trailing static cone is recommended (see Reference 25). Initial flights showed difficulty in deployment of the static cone after takeoff.
Figure 3.22 B&D Instruments 2054 pressure transducer

Figure 3.23 Pitot tube
Figure 3.24(a) Pitot tube mounted on airplane

Figure 3.24(b) Pitot tube mounted on airplane
The cone has not been used and is not essential if only stability analysis is performed. The airplane static system is sufficient for stability analysis; however, a more accurate method would be required for any performance testing.

3.2.7 Temperature Transducer

An Analog Devices Company Semiconductor temperature transducer was used for measurement of air temperature. Specifications are shown in Figure 3.26. The transducer is mounted in a probe, as shown in Figure 3.27. The temperature probe is mounted the same way as the pitot tube, using the double-sided tape method. The temperature probe is shown mounted on the airplane in Figures 3.28. The location of the probe is identical to that of the pitot tube, but on the opposite wing.
Figure 3.26 Temperature transducer specifications

Figure 3.27 Temperature probe
Figure 3.28(a) Temperature probe mounted on airplane

Figure 3.28(b) Temperature probe mounted on airplane
The transducers selected have shown that the basic decisions regarding specific transducers, ranges and accuracies were correct. They have all proved reliable, with no failures encountered; and none required any specialized signal conditioning or difficult calibration procedures.

3.3 Power Supply

There were two options considered for supplying power to this instrumentation system:

1) Tap off the aircraft electrical system, or
2) Carry a separate battery package on the flight.

Considering option one, using the aircraft power system, offered several advantages. These were reduction in size of the instrumentation system, and no limited usage time due to battery rundown. It was realized, however, that there are several voltage standards on the current general aviation fleet. This would therefore require either a complex voltage control system or several systems to account for the various voltages available in the airplanes to be considered. Coupled with this is the high cost of voltage conversion systems. Also, modification would then be required to the airplane's electrical system to install the instrumentation package.

It was decided to explore the second option. A suitable rechargeable battery was found, manufactured by Eagle-Picher. These lead acid batteries are sealed, rechargeable, and maintenance free. A typical discharge curve is shown in Figure 3.29. These batteries when used in a deep cyclic regime (i.e., removing 50–100% of the
battery rated capacity prior to recharge) have a recharge time of
12 to 20 hours. They have an expected lifetime of 100 to 150 complete
charge/discharge cycles, with longer life expectancy when less than
100% depth of discharge is used. These batteries can also be used
in any position. The cost of these batteries is such that several
battery packs could be purchased for less than the price of one regu-
lated voltage divider required if the airplane electrical system were
used.

Figure 3.29 Battery discharge curve
Another advantage of a battery system is stability of the voltage supplied to the system. This advantage stems from two conditions. One is the fact that no external loads are on the power supply; thus, the power being used is steady and unchanging. Second is the fact that no ripple or noise will be in the power supplied. With a ship-supplied system, ripple will be present in the voltage system due to the means of supplying power (from the generator or alternator systems). This steady voltage supply, and the lack of ripple when the batteries are used, results in transducers being able to normally exceed their advertised specifications.

The voltages required for the complete on-board data acquisition system are shown in Table 3.4. Batteries were selected to match the power requirements at the various voltages. The wiring schematic, as well as the specifications of the batteries selected, are shown in Figure 3.30. The batteries allow a minimum of 3 hours running time between recharge. (The 12-volt battery supplying the TEAC tape drive is discharged first.)

The biggest disadvantage when batteries are used is that of weight. The battery module, complete, weighs 60.5 lbs. This is the heaviest component in the entire system (see the table of Figure 3.2). Total weight of the entire instrumentation system including all cables is 132.4 lbs. This system weight is not a problem for the majority of general aviation airplanes.
Table 3.4 Power Requirements for Data Acquisition Package

<table>
<thead>
<tr>
<th>BATTERY VOLTAGE</th>
<th>REGULATED VOLTAGE</th>
<th>REQUIREMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>+36</td>
<td>+28</td>
<td>Heater ($P_S, P_D$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Gyro motors ($\theta, \phi, p, q, r$)</td>
</tr>
<tr>
<td>+24</td>
<td>+15</td>
<td>Accelerometers ($A_x, A_y, A_n$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Filters, MDAS-16</td>
</tr>
<tr>
<td></td>
<td>+12</td>
<td>TEAC tape drive</td>
</tr>
<tr>
<td>+12</td>
<td>+5,5</td>
<td>$P_S$ and $P_D$ reference voltage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Potentiometers ($\theta, \phi, \delta_E, \delta_A, \delta_R$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AIM 65 computer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Temperature transducer</td>
</tr>
<tr>
<td>-12</td>
<td>-5</td>
<td>Potentiometers ($\theta, \phi, \delta_E, \delta_A, \delta_R$)</td>
</tr>
<tr>
<td>-24</td>
<td>-15</td>
<td>Accelerometers ($A_x, A_y, A_n$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Filters, MDAS-16</td>
</tr>
</tbody>
</table>

![Battery module schematic](image)

* maximum current requirement

<table>
<thead>
<tr>
<th>BATTERY NUMBER</th>
<th>BATTERY VOLTAGE</th>
<th>NOMINAL CAPACITY</th>
<th>DIMENSIONS (INCHES)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>20 HR</td>
<td>10 HR</td>
</tr>
<tr>
<td>12V8</td>
<td>12</td>
<td>20.0</td>
<td>19.0</td>
</tr>
<tr>
<td>12V15</td>
<td>12</td>
<td>15.0</td>
<td>14.5</td>
</tr>
<tr>
<td>12V20</td>
<td>12</td>
<td>8.0</td>
<td>7.7</td>
</tr>
<tr>
<td>12V1.5</td>
<td>12</td>
<td>1.5</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Figure 3.30 Battery module schematic and specifications
3.4 Pilot Control

The pilot controls the instrumentation system using a box which can be placed on the seat beside him (see Figure 3.31). The control box performs essentially the same function as the ground keyboard, the switches on the box replacing the keys (which are really just momentary contact switches). The controls are described below.

3.4.1 System Control

This consists of three switches.

First is the "INITIALIZE" tape switch. This is a momentary contact switch which is used only after insertion of a fresh data tape. This function prepares the data cassette to accept data.

Second, the "RUN/STBY" toggle switch is used to control when data is being recorded. In the STBY position the system is non-active. In the RUN position, data is recorded. There are two of these switches, one of which is located on the pilot control wheel and the other, on the pilot control box.

Third is the "REWIND" switch. This is used at the end of a cassette or flight. Activation of this switch places an "end" mark on the data tape and rewinds the tape.

3.4.2 Transducer Readout Control

A high-impedance analog voltmeter is provided to the pilot so that he can observe a particular transducer as he requires. The meter's installation is shown in Figure 3.32. A rotary switch
Figure 3.31 Pilot control console

Figure 3.32 Transducer monitor
(on the pilot control console) controls the signal which is observed.

This feature is also used to verify that all transducers are operating correctly prior to a test flight.
4. GROUND COMPUTER SYSTEM

The MMLE* data reduction process described in this report requires a powerful computer capable of being programmed in a high level language. Phase I (Reference 2) pointed out the requirement for a computer system operating under a compiled language. This requirement is due to the lengthy execution times associated with interpretive languages. (A Hewlett Packard 9825 was used in Phase I, programmed in interpretive Basic.) This chapter presents a benchmark process which has been used to evaluate the capability of the computer systems to perform the data reduction tasks. Also, a description of the selected computer is presented.

A two-step evaluation process was used. The first program in this process, the INTEGER SPEED ROUTINE, is short and easy to implement and gives a ball-park speed estimate. Secondly, the FLOATING POINT SPEED ROUTINE is a lengthier program, more closely resembling the operations performed in the MMLE process. These programs are described below.

4.1 Integer Speed Routine

This is a short, easy-to-implement program giving a rough benchmark of the operating speed of computers. The idea for this routine was originally conceived in Reference 26. A listing is presented in Table 4.1. The program does not realistically reflect the MMLE data reduction process, but it can be easily implemented in virtually

*MMLE = Modified Maximum Likelihood Estimation (see Chapter 5).
Table 4.1 Integer Speed Routine
(Fortran Listing)

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>DO 100 M=5,10000,2</td>
</tr>
<tr>
<td>20</td>
<td>I=M/2</td>
</tr>
<tr>
<td>30</td>
<td>DO 200 K=3,I,2</td>
</tr>
<tr>
<td>40</td>
<td>J=(M/K)*K</td>
</tr>
<tr>
<td>50</td>
<td>IF(J.EQ.M) GO TO 100</td>
</tr>
<tr>
<td>60</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>70</td>
<td>PRINT,M</td>
</tr>
<tr>
<td>80</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>90</td>
<td>STOP</td>
</tr>
<tr>
<td>100</td>
<td>END</td>
</tr>
</tbody>
</table>

any language on most computers in little time. This increases the ease with which a benchmark can be run and gives a ball park estimate of a computer's speed performance. The results of this speed comparison are presented in Table 4.2.

For evaluation of this data, it was assumed that once through this program was equal to two iterations of the MMLE routine. Therefore, to obtain the desired data reduction time through MMLE of 5-20 minutes, the Integer Speed Routine needs to run at 2-8 minutes on an acceptable computer. From Table 4.2 it is seen that all acceptable computers had both a compiler, which compiled down to machine code, and a floating point hardware package. Also, all acceptable computers were either using 16-bit microprocessors or could be considered main frame machines. It was obvious that current 8-bit microcomputers would not be capable of performing the data reduction task in any reasonable time frame. This is evident by the fact that the AIM-65 (using a 6502, 8-bit microprocessor) could not meet the speed requirements even in assembly language.

This study narrowed the number of acceptable machines considerably.
Table 4.2 Integer Speed Comparison

<table>
<thead>
<tr>
<th>PROCESSOR</th>
<th>MACHINE</th>
<th>LANGUAGE</th>
<th>INTERPRETER</th>
<th>COMPILER</th>
<th>FLOATING POINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-BIT MICRO</td>
<td>AIM 65</td>
<td>BASIC (PRINTER OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>4: 16: 44</td>
</tr>
<tr>
<td></td>
<td>TRS 80</td>
<td>ASSEMBLY (LED OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 23: 36</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASSEMBLY (PRINTER OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 33: 40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LEVEL I BASIC</td>
<td>*</td>
<td>*</td>
<td>7: 12: 27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LEVEL II BASIC</td>
<td>*</td>
<td>*</td>
<td>6: 31: 10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASSEMBLY</td>
<td>*</td>
<td>*</td>
<td>0: 21: 55</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FORTRAN</td>
<td>*</td>
<td>*</td>
<td>0: 56: 18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODEL 7X BASIC</td>
<td>*</td>
<td>*</td>
<td>3: 15: 00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INTEGER BASIC</td>
<td>*</td>
<td>*</td>
<td>2: 24: 31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FLOATING POINT BASIC</td>
<td>*</td>
<td>*</td>
<td>3: 56: 23</td>
</tr>
<tr>
<td>16-BIT MICRO</td>
<td>TERAQ 8510</td>
<td>PASCAL (COMPILE TO P CODE)</td>
<td>*</td>
<td>*</td>
<td>0: 30: 35</td>
</tr>
<tr>
<td></td>
<td>TEKTRONIX (4052)</td>
<td>BASIC</td>
<td>*</td>
<td>*</td>
<td>1: 23: 00</td>
</tr>
<tr>
<td></td>
<td>HP 9825</td>
<td>BASIC</td>
<td>*</td>
<td>*</td>
<td>1: 41: 17</td>
</tr>
<tr>
<td></td>
<td>HP 1000</td>
<td>FORTRAN RTE IV B (CRT OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 01: 23</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (NO OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 00: 48</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FORTRAN RTE M (CRT OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 00: 57</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (NO OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 00: 44</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FORTRAN (NO OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 01: 30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (PRINTER OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 04: 30</td>
</tr>
<tr>
<td></td>
<td>IBM SERIES I</td>
<td>FORTRAN (RSX 11 M) (CRT OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 07: 10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (PRINTER OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 11: 20</td>
</tr>
<tr>
<td></td>
<td>PDP 11/34*</td>
<td>FORTRAN RT11-IV (CRT OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 03: 36</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (DISC OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 03: 29</td>
</tr>
<tr>
<td></td>
<td>MINC 11/23</td>
<td>FORTRAN RT11-IV PLUS (CRT OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 03: 10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; (DISC OUTPUT)</td>
<td>*</td>
<td>*</td>
<td>0: 03: 00</td>
</tr>
<tr>
<td>MAIN FRAME</td>
<td>HONEYWELL 60/66</td>
<td>FORTRAN</td>
<td>*</td>
<td>*</td>
<td>0: 00: 44</td>
</tr>
<tr>
<td></td>
<td>CDC CYBER 70</td>
<td>FORTRAN (NON OPTIMIZED)</td>
<td>*</td>
<td>*</td>
<td>0: 02: 13</td>
</tr>
<tr>
<td></td>
<td>CDC CYBER 148</td>
<td>FORTRAN (OPTIMIZED)</td>
<td>*</td>
<td>*</td>
<td>0: 00: 39</td>
</tr>
<tr>
<td></td>
<td>IBM 370-148</td>
<td>PL/1 (OPTIMIZED)</td>
<td>*</td>
<td>*</td>
<td>0: 00: 37</td>
</tr>
</tbody>
</table>

*The PDP 11/34 was operating in a multi-user mode. Its performance is estimated to be approximately 2-3 times faster than the 11/23 series computer in single-user mode.

4.2 Floating Point Speed Routine

To more closely resemble the MMLE data reduction process, yet still use a simple-to-implement program, the routine shown in Table 4.3 was developed. The program is made up of floating point matrix mathematics, which is what MMLE primarily contains.
Table 4.3 Floating Point Speed Routine
(Fortran Listing)

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>REAL A(20,20),B(20,20),C(20,20),E(20,20),T(20,20),D(20,20),F</td>
</tr>
<tr>
<td>20</td>
<td>INTEGER I,J,K,M</td>
</tr>
<tr>
<td>30</td>
<td>PRINT,&quot;START&quot;</td>
</tr>
<tr>
<td>40</td>
<td>F=.098625</td>
</tr>
<tr>
<td>50</td>
<td>DO 400 M=1,40</td>
</tr>
<tr>
<td>60</td>
<td>DO 200 I=1,20</td>
</tr>
<tr>
<td>70</td>
<td>DO 200 J=1,20</td>
</tr>
<tr>
<td>80</td>
<td>E(I,J)=0</td>
</tr>
<tr>
<td>90</td>
<td>A(I,J)=F<em>I</em>J</td>
</tr>
<tr>
<td>100</td>
<td>B(I,J)=F*I</td>
</tr>
<tr>
<td>110</td>
<td>C(I,J)=F*J</td>
</tr>
<tr>
<td>120</td>
<td>D(I,J)=F</td>
</tr>
<tr>
<td>130</td>
<td>T(I,J)=0</td>
</tr>
<tr>
<td>140</td>
<td>200 CONTINUE</td>
</tr>
<tr>
<td>150</td>
<td>DO 300 I=1,20</td>
</tr>
<tr>
<td>160</td>
<td>DO 300 J=1,20</td>
</tr>
<tr>
<td>170</td>
<td>DO 300 K=1,20</td>
</tr>
<tr>
<td>180</td>
<td>T(I,J)=T(I,J)+(A(I,K)*B(K,J))</td>
</tr>
<tr>
<td>190</td>
<td>E(I,J)=E(I,J)+(E(I,K)*D(K,J))</td>
</tr>
<tr>
<td>200</td>
<td>300 CONTINUE</td>
</tr>
<tr>
<td>210</td>
<td>DO 400 I=1,20</td>
</tr>
<tr>
<td>220</td>
<td>DO 400 J=1,20</td>
</tr>
<tr>
<td>230</td>
<td>400 E(I,J)=E(I,J)+T(I,J)</td>
</tr>
<tr>
<td>240</td>
<td>PRINT,&quot;E=&quot;</td>
</tr>
<tr>
<td>250</td>
<td>DO 100 I=1,20</td>
</tr>
<tr>
<td>260</td>
<td>DO 100 J=1,20</td>
</tr>
<tr>
<td>270</td>
<td>PRINT,E(I,J)</td>
</tr>
<tr>
<td>280</td>
<td>100 CONTINUE</td>
</tr>
<tr>
<td>290</td>
<td>PRINT,M</td>
</tr>
<tr>
<td>300</td>
<td>STOP</td>
</tr>
<tr>
<td>310</td>
<td>END.</td>
</tr>
</tbody>
</table>

The program approximates one iteration of the MMLE method.

This is indicated by the 48 minute run time on the Hewlett Packard 9825, which requires approximately 50 minutes to perform one iteration of the MMLE program. In order to deem a computer acceptable for the MMLE process, it must be able to complete the floating point speed routine in the order of 1-5 minutes. It was decided that an MMLE execution time of 5-20 minutes would be acceptable (assuming 5 iterations).
The results of this test are presented in Table 4.4. It is seen that the 16-bit machines tested, operating in compiled Fortran, meet the speed requirement.

Table 4.4 Floating Point Speed Comparison

<table>
<thead>
<tr>
<th>MACHINE</th>
<th>MIN:SECS</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP9825 (BASIC)</td>
<td>48:15</td>
</tr>
<tr>
<td>HONEYWELL 60/66</td>
<td>0:20.6</td>
</tr>
<tr>
<td>HP1000 (NO OUTPUT)</td>
<td>1:08.7</td>
</tr>
<tr>
<td>(DISC OUTPUT)</td>
<td>2:07</td>
</tr>
<tr>
<td>IBM SERIES 1 (DISC OUTPUT)</td>
<td>0:58</td>
</tr>
<tr>
<td>MINC 11/03 (DISC OUTPUT)</td>
<td>5:35</td>
</tr>
<tr>
<td>MINC 11/23 (DISC OUTPUT)</td>
<td>4:00</td>
</tr>
</tbody>
</table>

4.3 Description of System

The results of the benchmark evaluation left several computers that were deemed acceptable. To select the best machine for the KU-FRL requirements, the following factors were also considered:

- Memory expansion capability
- Floating Point Hardware available
- RS232 ports/IEEE 488 ports* installed
- CRT Graphics capability
- Hard/Flexible disc storage
- Programming languages available
- Users group existing
- Delivery
- Cost

*Industry interfacing standards
Evaluating the acceptable computers, the DEC* MINC 11/03 computer was selected as best meeting the requirements. A description follows.

The MINC 11/03 is shown in Figure 4.1. The block diagram of Figure 4.2 shows the basic features and some of the options available.

The computer uses a 16-bit DEC LSI 11/03 processor, capable of addressing 64K bytes of memory, and contains a floating point hardware package, 4 RS232 ports, and an IEEE 488 port.

Data and program storage is handled using the dual RX02 flexible disc drives. These use 8" flexible discs, capable of holding 500 K bytes of information each.

Computer and program interaction is handled using the DEC-VT 105 graphics terminal. This permits inputting and outputting of data, as well as allowing graphical representation of the flight test results.

The RS232 ports are used for input and output of the data. Four are provided. One is used for the VT 105 terminal, two are configured to allow data transfer from the Rockwell AIM-65, and one is used to control a hard copy printer.

The IEEE 488 port allows ease of interfacing to many industry standard components. Planned future use of this port is for a hard copy plotter for analysis and report quality plots of flight test data.

*DEC = Digital Equipment Corporation.
Figure 4.1 Digital Equipment Corp. MINC 11/03 computer

Figure 4.2 Block diagram of MINC 11/03 computer

*Options available (not on KU-FRL system)
*Installed on KU-FRL system
The standard MINC comes with BASIC language software. The KU-FRL package has the RT11-FORTRAN IV software option. This version of FORTRAN allows compiling programs to machine level, which was determined necessary to perform the data reduction task as indicated in Sections 4.1 and 4.2.

The MINC computer has been found capable of performing the function intended. The MMLE process takes approximately 20 minutes for 5 iterations, which is close to the prediction of Section 4.2.

It is recommended that a hard copy printer and plotter be added to the standard MINC to make it a complete data analysis system.
5. DATA REDUCTION METHOD

This chapter describes the data analysis procedures used for longitudinal and lateral stability analysis. The overall method is best depicted via the flow chart shown in Figure 5.1.

For this phase the system described in Chapter 3 was used for airborne data acquisition. The KU-FRL's DEC-MINC 11/03 microcomputer was used for all further data processing. This computer is described in detail in Chapter 4. Segmenting the various data reduction programs into the blocks as shown in Figure 5.1 allowed effective data analysis.

This section describes theoretical aspects of the computer programs used. Flow charts and program listings are included as Appendix A.

5.1 Data Acquisition

This program is used as part of the airborne data recording system. It is written on the AIM-65 in machine level language to allow rapid execution. The program controls the MDAS-16 module, as well as the TEAC cassette recorder. (See Appendix A.1 for flow-chart and listing.)

The program has three control inputs, which are located on the pilot control console. The first is the "INITIALIZE" tape button. This is used for getting the data cassette ready to record the signals. It is used only once per data cassette. This command
Figure 5.1 Data processing flow chart
rewinds the tape (if required), advances the tape past the beginning of the tape hole, and then writes a beginning-of-tape file mark.

Second, the "RUN/STBY" toggle switch is used to control the recording of data. Placing this switch in the RUN position begins the data recording process. The computer then sends control to the MDAS to sample the \( P_D \), \( P_S \) and \( T \) channels. These are sampled 10 times each and then output to the TEAC cassette drive in one block. The program then runs through the other channels (\( A_X \), \( A_Y \), \( A_N \), \( \theta \), \( \phi \), \( p \), \( q \), \( r \), \( \delta_E \), \( \delta_A \), \( \delta_R \)). These data are temporarily stored in computer memory. After a total of 0.1 seconds has elapsed, the computer then samples these channels again and also temporarily stores them in memory. After 10 such time points are in the computer memory, the AIM-65 outputs this to the TEAC in one block; and the process continues until the "RUN/STBY" switch is placed in STBY. Then the computer samples the \( P_D \), \( P_S \) and \( T \) channels again and outputs these to the tape. After this the system idles, waiting for the next command.

To reduce the possibility of error, the highest order bits on the measurement channels are recorded twice. This is easily done, as the analog-to-digital conversion comes out as a 12-bit word, available as a tri-state output. The AIM-65 operates on the basis of 8-bit words; therefore, the 4 highest order bits of the data are recorded twice, resulting in two 8-bit words. These higher order bits are compared on readback as a means of error checking.

Third, the "REWIND" switch causes an end-of-tape mark to be written on the cassette and then rewinds the tape back to the start.
This program also keeps track of the run number, which is output at the beginning of each run to the cassette.

5.2 Transfer Data to Ground Based Computer

This operation requires both the AIM-65 and MINC 11/03 and a program for each to allow the two to be coupled. A standard RS232 serial interface is available on both computers. The data can be transferred across telephone lines if desired.

The data, before being transferred from the AIM-65, is checked for errors. This is done by comparing the high order bits, which have been recorded twice. A running total of any errors is kept and printed out by the AIM-65 on its display printer. Errors have not been significant in number, and therefore no correction is made. All errors to date have been caused by poor quality data cassettes. Using the qualified cassettes (see Table 5.1 and Reference 27) no data errors have been found in the flight data.

Table 5.1 Qualified Data Cassettes

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Type</th>
<th>Part No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3M</td>
<td>Scotch</td>
<td>834A/1-300</td>
</tr>
<tr>
<td>TDK</td>
<td>Data Cassette</td>
<td>HR-850 90C</td>
</tr>
<tr>
<td>MAXELL</td>
<td>Data Cassette</td>
<td>M-90</td>
</tr>
<tr>
<td>BASF</td>
<td>Digital Power Typing Cassette</td>
<td>52346</td>
</tr>
</tbody>
</table>

(Qualified as per Reference 27)
The AIM-65 program is shown in Appendix A.2; and the MINC 11/03 program, in Appendix A.3. These programs are used to transfer the flight data from the TEAC cassette tape to the MINC 11/03 disc. In this mode the AIM-65 keyboard is used for controlling the data transfer off of itself. The MINC 11/03 program loads the transferred data into its memory and then transfers this data to the data disc.

5.3 Engineering Conversion

The first step in the actual data analysis procedure is converting the raw data bits into their corresponding engineering units. The process involved first converts the bit pattern of each measurement to the voltage representation. (See Reference 3 for detailed explanation of this process.) Then, utilizing the particular transducer calibration curve, this voltage representation is converted to the units of the actual motion measured. Resulting from this, then, is the transducer measurement in the correct engineering unit. (See Appendix A.4 for program listing.)

This two-step process is presently required due to the calibration process utilized in this phase. Currently, transducers are excited using known inputs; and the transducer response is measured using a voltmeter. A suggested improvement in this process is to bypass the voltmeter, using the digital recording system in the calibration process. This improvement is planned to be implemented upon construction of the calibration rig suggested in Chapter 9.
5.4 **Quick-Look Plots**

The next step in the data analysis procedure is making the quick-look plots. The program of Appendix A.5 is used to do this. Basically this program plots the transducer outputs (uncorrected for C.G. location, etc.) on the graphics CRT. This is a rapid means of determining the portion of the recorded data that has the proper aircraft modes excited and is thus suitable for further analysis. Operator interaction has been minimized to reduce the overall time required for this step.

5.5 **Detailed Engineering Conversion**

This program is used to do a rigorous conversion of the data into the form required for the MMLE technique. Accounted for in this procedure are accurate transducer calibrations and instrument position corrections.

The first step in the instrument position correction process is to account for the misalignment between the transducers and the aircraft body axis. Secondly a correction must be applied to correct for the distance from the transducer center of gravity to the airplane center of gravity. The following equations are used. (See Reference 28 for a more rigorous presentation.)

\[
\begin{align*}
\theta_B &= \theta_M - \theta_I \\
 p_B &= p_M \cos(\theta_I) + r_M \sin(\theta_I) \\
 r_B &= -p_M \sin(\theta_I) + r_M \cos(\theta_I)
\end{align*}
\]
where

\[ A_X = A_M \cos(\theta_I) - A_N \sin(\theta_I) + \frac{r B}{g} \frac{X}{g} - (p q - r) \frac{V}{g} - (p r + q) \frac{Z}{g} \]

\[ A_Y = A_M - (p q + r) \frac{X}{g} + (p^2 + r^2) \frac{V}{g} - (q r - p) \frac{Z}{g} \]  \hspace{1cm} [5.1(b)]*

\[ A_N = A_M \sin(\theta_I) + A_N \cos(\theta_I) + (p r - q) \frac{X}{g} + (q r + p) \frac{V}{g} - (p^2 + q^2) \frac{Z}{g} \]

B indicates Body axis at airplane center of gravity

M indicates as Measured by transducer

I indicates as Installed wrt Body axis at airplane center of gravity

This step also involved checking for and correcting any obvious data errors. If any filtering of unwanted noise is required, it would also be done at this stage; however, none has been needed to date. The quick-look plots are used as the major aid in this process.

A program listing is contained in Appendix A.6.

5.6 Modified Maximum Likelihood Estimator

The flight data were processed through the Modified Maximum Likelihood Estimator (MMLE) developed by NASA (see References 12-16). This technique has been used by NASA for over 12 years. A simplified program (NASA Dryden "BONES" version of MMLE) has been placed on the MINC 11/03 computer. The actual program listings are included in Appendix A.7. Described here is the theory used in this technique, and some of the assumptions made for the KU-FRL version.

*Where \( \dot{p} \), \( \dot{q} \), and \( \dot{r} \) are required, these are determined by digitally differentiating the \( p \), \( q \), and \( r \) measurements.
5.6.1 Parameter Estimation

The MMLE estimator is an iterative process that determines the coefficients of a given set of linear equations describing the motion of the aircraft. It does this by comparing the difference between actual in-flight measured responses of various states, and the predicted responses of these states using an estimate of the coefficients. The actual measured control input is used as the input for the estimating procedure. The estimated coefficients are updated each iteration, using the differences as determined above. The flow chart below shows the MMLE concept.

Figure 5.2 Maximum likelihood estimation concept (from Reference 13)
5.6.2 Mathematical Model

The mathematical model used to describe the airplane is derived from the small perturbation equations of motion (see Reference 22).* These are shown here explicitly, in the non-dimensional form.

- for longitudinal (from Reference 22, Equation 6.1):

\[
\dot{\mu} = -mg\cos \theta_1 + \delta_1 S\left(-C_D + 2C_D L_1 \right) \frac{U}{U_1} + \left( C_T + 2C_T L_1 \right) \frac{U}{U_1} - \left( C_D - C_L L_1 \right) \alpha - C_D \delta_E
\]

\[
m(\dot{\theta} - U_1 q) = -mg\sin \theta_1 + \delta_1 S\left(-C_D + 2C_D L_1 \right) \frac{U}{U_1} - \left( C_D + C_D L_1 \right) \alpha - C_D \frac{\alpha}{U_1} - C_L \frac{\alpha}{U_1} - C_L \delta_E \]  

[5.2(a)]

- for lateral (from Reference 22, Equation 6.2):

\[
m(\dot{\phi} + U_1 r) = mg\cos \theta_1 + \delta_1 S\left(C_y b + C_y p \frac{p}{U_1} + C_y r \frac{r}{U_1} + C_y \delta_A + C_y \delta_R\right)
\]

\[
l_{xz} \dot{p} - l_{xz} r = \delta_1 S\left(C_y b + C_y p \frac{p}{U_1} + C_y r \frac{r}{U_1} + C_y \delta_A + C_y \delta_R\right) \]  

[5.2(b)]

Using the definitions shown in Table 5.2, Equations [5.2] can be converted to the dimensional form shown below.

- for longitudinal (from Reference 22, Equation 6.72):

\[
\dot{\mu} = -g\theta \cos \theta_1 + X_\mu + X_T u + X_\alpha + X_\delta E
\]

\[
\dot{\theta} - U_1 q = -g\theta \sin \theta_1 + Z_u + Z_\alpha + Z_\delta + Z_q + Z_\delta E \]  

[5.3(a)]

\[
q = M_\mu + M_T u + M_\alpha + M_\delta \]  

*The derivatives in Reference 22 are for the stability axes system. See Appendix B for conversion to the Body axes used in this report.
Table 5.2(a) Longitudinal Dimensional Stability Derivatives *

\[
\begin{align*}
X_u &= \frac{-q_1S(C_D + 2C_{D_1})}{mU_1} \quad \text{(sec}^{-1}) \\
X_{T_1} &= \frac{-q_1S(C_{T_1} + 2C_{T_1})}{mU_1} \quad \text{(sec}^{-1}) \\
X_a &= \frac{-q_1S(C_D - C_L)}{m} \quad \text{(ft sec}^{-2}) \\
X_{\delta E} &= \frac{-q_1S C_D \delta E}{m} \quad \text{(ft sec}^{-2}) \\
Z_u &= -\frac{q_1S(C_a + 2C_{L_1})}{mU_1} \quad \text{(sec}^{-1}) \\
Z_a &= -\frac{q_1S(C_a + C_{D_1})}{m} \quad \text{(ft sec}^{-2}) \\
Z_\alpha &= -\frac{q_1S C_L \alpha}{2mU_1} \quad \text{(ft sec}^{-1}) \\
Z_q &= -\frac{q_1S C_L q}{2mU_1} \quad \text{(ft sec}^{-1}) \\
Z_\delta E &= -\frac{q_1S C_L \delta E}{m} \quad \text{(ft sec}^{-2}) \\
M_u &= \frac{-q_1S C_m + 2C_{m_1}}{I_{yy} U_1} \quad \text{(ft}^{-1} \text{ sec}^{-1}) \\
M_{T_1} &= \frac{-q_1S C_{m_T} + 2C_{m_{T_1}}}{I_{yy} U_1} \quad \text{(ft}^{-1} \text{ sec}^{-1})
\end{align*}
\]

* from Reference 22, Table 6.3, page 413
Table 5.2(b) Lateral-Directional Dimensional Stability Derivatives *

\[
\begin{align*}
Y_\beta &= \frac{-q_1 SC_y}{m} \text{ (ft sec}^{-2}\text{)} \\
Y_p &= \frac{-q_1 SbCy_p}{2mU_1} \text{ (ft sec}^{-1}\text{)} \\
Y_r &= \frac{-q_1 SbCy_r}{2mU_1} \text{ (ft sec}^{-1}\text{)} \\
Y_\delta_A &= \frac{-q_1 SC_y \delta_A}{m} \text{ (ft sec}^{-2}\text{)} \\
Y_\delta_R &= \frac{-q_1 SC_y \delta_R}{m} \text{ (ft sec}^{-2}\text{)} \\
L_\delta &= \frac{-q_1 SbC_\delta}{I_{xx}} \text{ (sec}^{-2}\text{)} \\
L_\delta_A &= \frac{-q_1 SbC_\delta A}{I_{xx}} \text{ (sec}^{-2}\text{)} \\
L_\delta_R &= \frac{-q_1 SbC_\delta R}{I_{xx}} \text{ (sec}^{-2}\text{)} \\
N_\beta &= \frac{-q_1 SbC_n \beta}{I_{zz}} \text{ (sec}^{-2}\text{)} \\
N_T\beta &= \frac{-q_1 SbC_n \beta}{I_{zz}} \text{ (sec}^{-2}\text{)} \\
N_p &= \frac{-q_1 Sb^2C_n p}{2I_{zz} U_1} \text{ (sec}^{-1}\text{)} \\
N_r &= \frac{-q_1 Sb^2C_n r}{2I_{zz} U_1} \text{ (sec}^{-1}\text{)} \\
N_\delta_A &= \frac{-q_1 SbC_n \delta_A}{I_{zz}} \text{ (sec}^{-2}\text{)} \\
N_\delta_R &= \frac{-q_1 SbC_n \delta_R}{I_{zz}} \text{ (sec}^{-2}\text{)}
\end{align*}
\]

* from Reference 22, Table 6.8, page 445
- for lateral (from Reference 22, Equation 6.141):

\[ v + U_l r = g \phi \cos \theta_1 + Y_B \beta + Y_P p + Y_r r + Y_A \delta_A + Y_R \delta_R \]

\[ \dot{r} = \frac{I_{XZ}}{I_{XX}} \dot{p} = L_B \beta + L_P p + L_r r + L_A \delta_A + L_R \delta_R \]  

\[ \ddot{p} = \frac{I_{XZ}}{I_{Zz}} \dot{r} = N_B \beta + N_T \beta + N_P p + N_r r + N_A \delta_A + N_R \delta_R \]

Using the concept of state variable theory (see Reference 22), Equation [5.3] can be written in the following form:

\[ [R] \{ \dot{x}(t) \} = [A] \{ x(t) \} + [B] \{ u(t) \} \]

where

\{ x(t) \} = state vector

[R] = acceleration transformation matrix

[A] = stability matrix

[B] = control matrix

\{ u(t) \} = control vector.

Equation 5.4 can be written more explicitly in the form which follows:

- for longitudinal (where [R] = identity matrix):

\[
\begin{bmatrix}
q' \\
U' \\
\alpha' \\
\theta'
\end{bmatrix}
= \begin{bmatrix}
M_q' & M_u' & M_a' & M_0' \\
0 & X_u' & X_a' & -g \cos(\theta_1) \\
\frac{z_u + U_1}{U_1 - z_a} & z_u' & z_a' & \frac{-R}{U_1 - z_a} \sin(\theta_1) \cos(\phi_1) \\
\cos(\phi_1) & 0 & 0 & 0
\end{bmatrix}
\cdot
\begin{bmatrix}
q \\
U \\
\alpha \\
\theta
\end{bmatrix}
+ \begin{bmatrix}
M_{x'} & M_{x'} & M_{z'} & M_{\gamma} \\
X_{x'} & X_{x'} & X_{z'} & X_{\gamma} \\
Z_{x'} & Z_{x'} & Z_{z'} & Z_{\gamma} \\
0 & 0 & 0 & \delta_{x'}
\end{bmatrix}
\begin{bmatrix}
\delta_x \\
\delta_y \\
\delta_z \\
1
\end{bmatrix}
\]

[5.5(a)]

(See Table 5.3 for explicit definition of these terms.)
To allow determination of states other than the ones contained in \{x(t)\}, the following expression can be derived:

\[
\{y(t)\} = \begin{bmatrix} \frac{1}{G} \\ \frac{0}{H} \end{bmatrix} \{x(t)\} + \begin{bmatrix} 0 \\ \frac{0}{V} \end{bmatrix} \{u(t)\} + \{v\} \quad [5.6]
\]

where

\[
\{y(t)\} = \text{computed observation vector}
\]
\[
[G] = \text{observation matrix}
\]
\[
[H] = \text{observation matrix}
\]
\[
\{v\} = \text{variable bias vector}.
\]

(See Table 5.4 for explicit definition of these terms.)

The computed observation vector, \{y(t)\}, corresponds to the measured observation vector, shown here:

\[
\{z(t)\} = \{y(t)\} + \{\eta(t)\} \quad [5.7]
\]

where

\[
\{z(t)\} = \text{measured observation vector} = \{\theta, \phi, p, q, r, A_x, A_y, A_n, \delta_E, \delta_A, \delta_R, P_S, P_D, T\}
\]
\[
\{\eta(t)\} = \text{measured noise vector}.
\]

From the terms of Equations [5.4], [5.6], [5.7], the vector

\^From Reference 16
| $M'_{q} = M_{q} + M'_{a}$ | $Z_{q} + \frac{U_{1}}{1 - Z_{a}} = M_{q} + M'_{a}$ (sec$^{-1}$) |
| $M'_{u} = M_{u} + M_{T_{u}} + \frac{M_{a} Z_{a}}{U_{1}-Z_{a}}$ (ft$^{-1}$ sec$^{-1}$) |
| $M'_{a} = M_{a} + M_{T_{a}} + \frac{M_{a} Z_{a}}{U_{1}-Z_{a}}$ (sec$^{-2}$) |
| $M'_{6} = \frac{-M_{a} g \sin(\theta_{1}) \cos(\phi_{1})}{U_{1}-Z_{a}} = 0$ (sec$^{-2}$) |
| $M'_{6, E, c} = M_{6, E, c} + \frac{M_{a} Z_{6, E, c}}{U_{1}-Z_{a}}$ (sec$^{-2}$) |
| $X'_{o} = \text{longitudinal acceleration equation bias (ft sec}^{-2})^*$ |
| $M'_{o} = \text{pitching moment equation bias (sec}^{-2})^*$ |
| $X'_{u} = X_{u} + X_{T_{u}}$ (sec$^{-1}$) |
| $X'_{a}$ (ft sec$^{-2}$) |
| $X'_{6, E, c} = X_{6, E, c}$ (ft sec$^{-2}$) |

*Note: The equation bias terms are used to allow prediction of the complete state which is made up of the steady state and the perturbed state.*

†Note: With the approximations above, Equation [5.5(a)] is rewritten as:

$$\begin{bmatrix} q \\ U \\ a \\ \theta \end{bmatrix} = \begin{bmatrix} M'_{q} & M'_{u} & M'_{a} & 0 \\ 0 & X'_{u} & X'_{a} & -\cos(\theta_{1})g \\ 1 & Z'_{u} & Z'_{a} & -\sin(\theta_{1})\cos(\phi_{1}) \frac{g}{U_{1}} + \frac{X'_{6, E} & X'_{6, c} & X'_{o}}{U_{1}} + \frac{Z'_{6, E} & Z'_{6, c} & Z'_{o}}{U_{1}} \end{bmatrix} \begin{bmatrix} q \\ U \\ a \\ \theta \end{bmatrix} = \begin{bmatrix} M'_{6, E} & M'_{6, c} & M'_{o} \\ \delta'_{E} \\ \delta'_{c} \end{bmatrix} \begin{bmatrix} \delta'_{E} \\ \delta'_{c} \end{bmatrix}$$
Table 5.3(b) Lateral, Dimensional State Vector Stability Derivatives

<table>
<thead>
<tr>
<th>Term</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_p'$ = $L_p$ (sec$^{-1}$)</td>
<td>$N_{\delta_A}' = N_{\delta_A}$ (sec$^{-2}$)</td>
</tr>
<tr>
<td>$L_r'$ = $L_r$ (sec$^{-1}$)</td>
<td>$N_{\delta_r}' = N_{\delta_r}$ (sec$^{-2}$)</td>
</tr>
<tr>
<td>$L_\beta' = L_\beta$ (sec$^{-1}$)</td>
<td>$N_\beta' = N_\beta + N_{T_\beta}$ (sec$^{-1}$)</td>
</tr>
<tr>
<td>$L_{\delta_A}' = L_{\delta_A}$ (sec$^{-2}$)</td>
<td>$Y_\beta' = \frac{Y_\beta}{U_1}$ (sec$^{-1}$)</td>
</tr>
<tr>
<td>$L_{\delta_r}' = L_{\delta_r}$ (sec$^{-2}$)</td>
<td>$Y_{\delta_A}' = \frac{Y_{\delta_A}}{U_1}$ (sec$^{-1}$)</td>
</tr>
<tr>
<td>$N_p' = N_p$ (sec$^{-1}$)</td>
<td>$Y_{\delta_r}' = \frac{Y_{\delta_r}}{U_1}$ (sec$^{-1}$)</td>
</tr>
<tr>
<td>$N_r' = N_r$ (sec$^{-1}$)</td>
<td>$Y_\gamma' = \frac{Y_\gamma}{U_1}$ (sec$^{-1}$)</td>
</tr>
<tr>
<td>$Y_o' = \text{lateral acceleration equation bias (sec$^{-1}$)}$</td>
<td>$\phi_o' = \text{roll rate equation bias (sec$^{-1}$)}$</td>
</tr>
<tr>
<td>$L_o' = \text{rolling moment equation bias (sec}^{-2})$</td>
<td>$N_o' = \text{yawing moment equation bias (sec}^{-2})$</td>
</tr>
</tbody>
</table>

*NOTE: The equation bias terms are used to allow prediction of the complete state which is made up of the steady state and the perturbed state.

$$[R] = \begin{bmatrix}
1.0 & -\frac{I_{xz}}{I_{xx}} & 0 & 0 \\
\frac{I_{xz}}{I_{xx}} & 1.0 & 0 & 0 \\
0 & 0 & 1.0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}$$

for $I_{xz} = 0$; $[R] = \text{identity matrix}$
Table 5.4 Matrices Used in Observation equation

### Longitudinal

\[(y(t))^\dagger = \{\dot{q}, \dot{U}, \alpha, \beta, \dot{\dot{q}}, A_X, A_N\}\]

\[
\begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 \\
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
M_{\delta}' \\
M_{\delta c}' \\
M_0'
\end{bmatrix}
\]

\[
\begin{bmatrix}
M_q' \\
H_{\dot{u}}' \\
M_{\alpha}'
\end{bmatrix}
\]

\[
\begin{bmatrix}
X_{\delta}' \\
X_{\delta c}' \\
X_0'
\end{bmatrix}
\]

\[
\begin{bmatrix}
x_{\dot{u}}' \\
\frac{X_{\dot{u}}'}{g} \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
-U_{1z}' \\
\frac{-U_{1z}'}{g} \\
\frac{-U_{1z}'}{g_0}
\end{bmatrix}
\]

### Lateral

\[(y(t))^\dagger = \{p, r, \beta, \dot{\phi}, \dot{\theta}, \dot{r}, A_Y\}\]

\[
\begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 \\
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
L_{\delta}' \\
L_{\delta R}' \\
L_0'
\end{bmatrix}
\]

\[
\begin{bmatrix}
L_p' \\
L_r' \\
L_{\beta}'
\end{bmatrix}
\]

\[
\begin{bmatrix}
N_{\delta}' \\
N_{\delta R}' \\
N_0'
\end{bmatrix}
\]

\[
\begin{bmatrix}
N_p' \\
N_r' \\
N_{\beta}'
\end{bmatrix}
\]

\[
\begin{bmatrix}
U_{1Y}' \\
\frac{U_{1Y}'}{g} \\
\frac{U_{1Y}'}{g_0}
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
Y_{\beta}' \frac{U_1}{g_0}
\end{bmatrix}
\]
(c) = f([A], [B], [G], [H], (v)) \quad [5.8]

(where f indicates "a function of") is defined as the vector of unknowns. It is this vector that the MMLE method estimates. MMLE determines the unknowns ((c)) by minimizing the cost function given by:

\[ J = \frac{1}{T} \int_0^T (z(t) - y(t))^T [D] (z(t) - y(t)) \, dt \quad [5.9] \]

(T, t: indicates time)
or approximately in the discrete case:

\[ J = \frac{1}{(N-1)} \sum_{i=1}^N (z_i - y_i)^T [D] (z_i - y_i) \Delta t \quad [5.10] \]

(where i is the time index, and N the number of time points).

The weighting matrix, [D], is used to provide emphasis on the various measured states; in other words, to allow greater emphasis on the more accurate transducers, or the transducers that are more important to describe the maneuver performed.

The value of the cost functional, J, is minimized using the Newton-Raphson* method. This technique is an iterative procedure, utilizing an estimated value of the vector of unknowns, (c), and the first and second gradients of the cost functional, J, with respect to the vector of unknowns, (c). The equation

\[ (c)_L = (c)_{L-1} - (\nabla^2 J)_L^{-1} (\nabla J)_L^+ \quad [5.11] \]

(where L is the iteration number) is used to revise estimates for the vector of unknowns, (c). The first and second gradients are given by:

*From Reference 16

\[ \nabla J \]

\[ \nabla^2 J \]

\[ \nabla J \]

\[ \nabla^2 J \]
\[
(v_c^J) = \frac{2}{N-1} \sum_{i=1}^{N} (z_i - y_i) \dagger [D] v_c (z_i - y_i) \tag{5.12}
\]

\[
(v_c^2) = \frac{2}{N-1} \sum_{i=1}^{N} v_c (z_i - y_i) \dagger [D] v_c (z_i - y_1) + \frac{2}{N-1} \sum_{i=1}^{N} (z_i - y_i) \dagger [D] v_c^2 (z_i - y_i) \tag{5.13}
\]

The Baiakrishnan* modification makes use of the fact that the term \(v_c^2 (z_i - y_1)\) approaches zero with convergence and is thus neglected. The expression for the second gradient becomes:

\[
(v_c^2) = \frac{2}{N-1} \sum_{i=1}^{N} v_c (z_i - y_i) \dagger [D] v_c (z_i - y_1) \tag{5.14}
\]

After several iterations the cost function converges near some small value. At this point the parameters of Equation [5.5] have been modified to obtain their most likely value which results in the best fit of the measured states.

5.6.3 Assumptions Used in Data Reduction

The following inputs and modifications were made to the MMLE method, allowing effective use of the technique on the MINC 11/03 computer.

Initial estimates of the derivatives in Equation [5.5] were obtained using the analytical methods of Reference 22. Although the MMLE technique does not require accurate knowledge of these derivatives, this procedure does speed convergence.

The MMLE program usually uses a modified least squares method for the first iteration to estimate the derivatives, as an aid to

*From Reference 16
speed convergence. This, however, requires measurement of most of the states indicated in (5.5). The instrumentation package uses only a minimum of transducers, and all the states required for this least squares estimate are not measured. Using the least squares procedure would result in divergence of the first iteration. Therefore, the least squares estimate was not used, which did slow convergence of the derivatives.

A diagonal multiplying factor allows control over how large a change is made to the derivatives after each iteration. Too large a value of this factor causes sluggishness in the convergence, and too small a value will cause divergence. Further analysis into this factor will indicate its optimum value for best convergence.

The weighting matrix, \([D]\), of Equation \([5.9]\), was chosen after analysis of the instrumentation error magnitudes. The first run through the MMLE program, with measurements from this instrumentation package, provided a weighted error for each measurement state. As suggested in Reference 16, the values for the weighting matrix were chosen to attempt to equalize the weighted errors. After the values for the weighting matrix were chosen for the instrument package, they were then left at this for further maneuver analysis.

5.7 Time History Plotting

The MMLE reduction method not only produces the estimates for the derivatives, but also calculates the estimated time history for the various states. This is stored on the data disc by the MMLE program. The programs presented in Appendix A.8 retrieve both the
predicted time histories and the measured time histories and plot them together on the graphics CRT. These graphs of the flight test maneuver are the visual indication of the goodness of the predicted airplane derivatives.

----- ********** -----

As is evident from the many programs provided, the final results of a flight test maneuver are obtained only after a multi-step procedure. This is primarily due to the nature of the methods being used in aircraft flight testing, as well as the limitations of computer technology being used.
6. KU-FRL FLIGHT TEST PROGRAM

Two series of flight tests have been conducted using the KU-FRL Cessna 172. The first series, conducted under Phase I, is presented in Reference 2. Presented here are the basic concepts of the type of flight maneuvers required, and results of the Phase II test program.

6.1 Flight Test Maneuver

Traditional flight testing methods have utilized primarily steady-state flight paths for data collection. This was due mostly to the data acquisition systems available. Unfortunately, this required a highly trained and competent test pilot to obtain realistic and valuable results.

With the current transducer and acquisition system technology available, flight testing need no longer rely on steady-state maneuvers to allow accurate state measurement. This development has resulted in the newer flight testing methods utilizing dynamic maneuvers.

When techniques such as the MMLE are used, the literature (Reference 29) indicates that the nature of the maneuver is not critical to determine the aircraft characteristics. What is important when using these techniques is to ensure that the proper aircraft modes have been excited. For example, a longitudinal maneuver should excite both the short-period and phugoid modes of the airplane. This realization (i.e., non-critical flight path) leads to the possibility of using lesser qualified pilots and still obtaining accurate results. All testing done on this program has been done by a pilot who had no previous flight test experience.
The control inputs presented in the traces of Chapter 6.2 are typical of the type of maneuver required. Several frequencies are excited, which tends to increase the validity of the results obtained. Also the total energy input is approximately symmetrical. In other words, the motion produced in one direction is offset by the motion produced in the opposite direction a short time later.

6.2 Results of Flight Test Program

Presented here are results of the Phase II flight test program. All flights were done at the conditions of Table 6.1.*

<table>
<thead>
<tr>
<th>Table 6.1 Cessna 172 Flight Test Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wing area (S) 174 ft²</td>
</tr>
<tr>
<td>Wing span (b) 35.8 ft</td>
</tr>
<tr>
<td>Inertias</td>
</tr>
<tr>
<td>( I_{xx} ) 1029 slug ft²</td>
</tr>
<tr>
<td>( I_{zz} ) 1891 slug ft²</td>
</tr>
<tr>
<td>( I_{yy} ) 1092 slug ft²</td>
</tr>
<tr>
<td>Mass (m) 59.46 slug</td>
</tr>
<tr>
<td>Weight 1913 lb</td>
</tr>
<tr>
<td>Center of Gravity (Body Station) 41.3 inch</td>
</tr>
<tr>
<td>Mean chord (c) 4.9 ft</td>
</tr>
<tr>
<td>Speed ((U_1)) 176 ft/sec</td>
</tr>
<tr>
<td>Dynamic Pressure ((q_1)) 33.69 lb/ft²</td>
</tr>
<tr>
<td>Altitude 3000 ft</td>
</tr>
</tbody>
</table>

The plots following show the absence of noise in the measurement, as well as the typical maneuver required.

*Estimated by Reference 30
The fit of the estimated states compared to the actual states in the longitudinal maneuvers is good (Figures 6.1-6.3). The only state that is off consistently is the $A_X$ term, which appears to be affected by a phase shift. The cause of this phase shift has not been determined.

The estimated parameters have been compared with the analytical methods of Reference 22 and with flight test results obtained by NASA Langley on a Cessna 172 (Reference 30). This correlation is shown in Table 6.2 for the longitudinal maneuvers. It is seen that there is good correlation between some derivatives, but not between others. The best correlation appears to be with the one of run 23B, in which the speed derivatives have been held constant for the MMLE analysis. This would tend to be the predicted result due to the mismatch in the $A_X$ term, which is a major contributor to the speed prediction.

The lateral maneuvers are presented in Figures 6.4-6.6; and the correlation of derivatives, in Table 6.3. The fit of the measured and predicted states is again reasonable. Run 11 has the best fit as well as the overall best fit to the parameters. Again, however, the predicted coefficients are not within acceptable limits. The cause of this is not known.

Observing the rudder trace on Figure 6.6, what appears to be rudder float is evident (especially between 10 sec and 12 sec).

*This is the same procedure performed by NASA Langley, which makes no attempt to predict any speed derivatives.*
TRIM POWER; 120 mph; 3000 ft; 75 °F;

Figure 6.1 Flight time history; Flight 19/10/80 Run 23A; Longitudinal
TRIM POWER; 120 mph; 3000 ft; 75 °F

Figure 6.2 Flight time history; Flight 19/10/80 Run 23B; Longitudinal
Figure 6.3 Flight time history; Flight 19/10/80 Run 51A; Longitudinal
Table 6.2 Comparison of Results, Longitudinal

<table>
<thead>
<tr>
<th>Estimation Method</th>
<th>KU-PR. WHEL</th>
<th>NASA LANGLEY¹,²</th>
<th>ANALYTICAL²,³</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gross Weight (lb)</td>
<td>1913</td>
<td>1948</td>
<td>2160</td>
</tr>
<tr>
<td>Center of Gravity (body station, in.)</td>
<td>41.3</td>
<td>42.5</td>
<td>40.3</td>
</tr>
<tr>
<td>Flight No.</td>
<td>19/10/80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Run No.</td>
<td>23A (Fig. 6.1)</td>
<td>23B (Fig. 6.2)</td>
<td>21A (Fig. 6.3)</td>
</tr>
</tbody>
</table>

| | C_{0q} ' | C_{0q} | C_{0d} | C_{0r} | C_{0g} | C_{0e} | C_{0f} | C_{0g} |
| | (-132) | 0.093 | (208) | -0.095 | 0.093 | (208) | -0.095 | 0.093 |
| | -21.64 | 0.097 | -17.60 | -0.046 | NOT PREDICTED | -0.046 | NOT PREDICTED | -0.046 |


* As compared with NASA Langley results.
* Wrong sign.
1 Reference 30, Table IV, page 28, Maximum Likelihood Method, average of the two runs at Full Trim.
2 See Appendix B for conversion to the body axes system used in this report.
3 Reference 12, Airplane A, page 590.
4 C_{0q} ' has a large C_{0d} component which cannot be predicted.
TRIM POWER; 120 mph; 3000 ft; 75 °F

Figure 6.4 Flight time history; Flight 23/10/80 Run 11; Lateral
Figure 6.5 Flight time history; Flight 19/10/80 Run 26C; Lateral
Figure 6.6 Flight time history; Flight 19/10/80 Run 33B; Lateral
### Table 6.3 Comparison of Results, Lateral

<table>
<thead>
<tr>
<th>Estimation Method</th>
<th>KE-FPL MOE</th>
<th>NASA LANGLEY (^1,^2)</th>
<th>ANALYTICAL (^1,^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gross Weight (lb)</td>
<td>1913</td>
<td>1848</td>
<td>2160</td>
</tr>
<tr>
<td>Center of Gravity (Body station, inches)</td>
<td>41.3</td>
<td>42.5</td>
<td>40.3</td>
</tr>
<tr>
<td>Flight No.</td>
<td>23/10/80</td>
<td>19/10/80</td>
<td></td>
</tr>
<tr>
<td>Run No.</td>
<td>11 (Fig. 6.6)</td>
<td>26C (Fig. 6.3)</td>
<td>33B (Fig. 6.6)</td>
</tr>
<tr>
<td>(C_{x, y}^{'\prime}) (C_{y, y}^{'\prime})</td>
<td>(-0.402) ((-0.698) ((-0.331) (-0.461) (-0.470)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, z}^{'\prime}) (C_{y, z}^{'\prime})</td>
<td>(0.062) (0.068) (0.139) (0.076) (0.096)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, x}^{'\prime}) (C_{y, x}^{'\prime})</td>
<td>(-0.049) (-0.057) (-0.034) (-0.074) (-0.089)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, A}^{'\prime}) (C_{y, A}^{'\prime})</td>
<td>(0.208) (0.207) (0.208) (0.206) (0.178)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, A}^{'\prime}) (C_{y, A}^{'\prime})</td>
<td>(0.009) (0.128) (0.023) (0.010) (-0.053)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, A}^{'\prime}) (C_{y, A}^{'\prime})</td>
<td>(-0.046) (-0.050) (-0.053) (\text{NOT PREDICTED}) (0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, R}^{'\prime}) (C_{y, R}^{'\prime})</td>
<td>(0.010) (0.039) (-0.164) (0.044) (0.013)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{x, R}^{'\prime}) (C_{y, R}^{'\prime})</td>
<td>(-0.040) (-0.058) (-0.035) (-0.052) (-0.066)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C_{y, R}^{'\prime}) (C_{y, R}^{'\prime})</td>
<td>(0.003) (0.066) (-0.481) (0.091) (0.187)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

( ) As compared with NASA Langley results.
\(^1\) Wrong sign.
\(^2\) Reference 30, Table VII, p. 32, case 34.
\(^3\) See Appendix B for conversion to the body axes system used in this report.
\(^4\) Reference 22, Airplane A, page 590.
This maneuver was performed by holding the rudder pedals fixed, yet a float of 2°-3° is seen in the rudder. This magnitude of input could affect the parameters determined. This effect is due to a second order control surface term introduced by this float but not predicted by the MMLE mathematical model.

It is suggested that further work be done to evaluate whether this is the case, and perhaps to include control surface float into the mathematical representation. The effect of control surface float can be determined by varying the tension of the cable which moves the surface.

Possible problems that could be responsible for the differences in parameter prediction are listed here:

- Calibration of transducers. It is suggested that part of the error in parameters is due to inaccuracies in transducer calibration.

- Uniqueness. It has not yet been determined if the methods such as MMLE have a unique solution. The possibility does exist of more than one solution to any given maneuver.

- Control surface float. No attempt was made in this flight test program to ensure a minimum of float of the control surfaces. It is suggested that cable tensions be tightened to allowable maximums prior to flight testing.
7. CESSNA FLIGHT TEST PROGRAM

The versatility of this flight test package was demonstrated in a spin test program conducted by Cessna Aircraft. In this program the KU-FRL provided the data management portion of the instrumentation system described in this report. Cessna supplied the instrumentation and the airplane. A block diagram of this installation is shown in Figure 7.1

7.1 Instrumentation

The purpose of this program was to investigate the spin characteristics of Cessna's latest model 172 airplane. To do this, Cessna approached the KU-FRL as to the applicability of the instrumentation system for this type of test. After initial evaluation it was decided that the measurements described in Table 7.1 would be required. It was apparent that the KU-FRL transducer package was unable to meet these needs; however, the data management portion of the package would be able to.

The airplane used in this program is shown in Figure 7.2. The external modifications to the airplane include a spin chute as well as right-hand and left-hand wing tip booms.

The spin chute was added for safety reasons. A device for deploying the chute is provided to the pilot, allowing him to retrieve the airplane from an unrecoverable spin. Also a release mechanism is provided to release the chute after deployment and spin recovery. The pilot also wears a parachute in the event the
Figure 7.1 Block diagram of Cessna spin test installation
## Table 7.1 Cessna Spin Test Measurement Requirements

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>TRANSDUCER</th>
<th>RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta_e$</td>
<td>ELEVATOR POSITION</td>
<td>FULL TRAVEL</td>
</tr>
<tr>
<td>$\delta_a$</td>
<td>AILERON POSITION</td>
<td>FULL TRAVEL</td>
</tr>
<tr>
<td>$\delta_r$</td>
<td>RUDDER POSITION</td>
<td>FULL TRAVEL</td>
</tr>
<tr>
<td>$p$</td>
<td>ROLL RATE</td>
<td>$\pm 360 \degree/\text{sec}$</td>
</tr>
<tr>
<td>$q$</td>
<td>PITCH RATE</td>
<td>$\pm 360 \degree/\text{sec}$</td>
</tr>
<tr>
<td>$r$</td>
<td>YAW RATE</td>
<td>$\pm 360 \degree/\text{sec}$</td>
</tr>
<tr>
<td>$A_z$</td>
<td>NORMAL ACCELERATION</td>
<td>$-3 \text{ to } +5 \text{ g}$</td>
</tr>
<tr>
<td>$A_y$</td>
<td>LATERAL ACCELERATION</td>
<td>$\pm 3 \text{ g}$</td>
</tr>
<tr>
<td>$\alpha_L$</td>
<td>ANGLE OF ATTACK LEFT HAND</td>
<td>$-20 \text{ to } +80 \degree$</td>
</tr>
<tr>
<td>$\alpha_R$</td>
<td>ANGLE OF ATTACK RIGHT HAND</td>
<td>$-20 \text{ to } +80 \degree$</td>
</tr>
<tr>
<td>$\beta_L$</td>
<td>SIDESLIP ANGLE LEFT HAND</td>
<td>$\pm 45 \degree$</td>
</tr>
<tr>
<td>$\beta_R$</td>
<td>SIDESLIP ANGLE RIGHT HAND</td>
<td>$\pm 45 \degree$</td>
</tr>
<tr>
<td>$\text{KTAS}_L$</td>
<td>TRUE AIRSPEED LEFT HAND</td>
<td>$20 \text{ to } 180 \text{ knots}$</td>
</tr>
<tr>
<td>$\text{KTAS}_R$</td>
<td>TRUE AIRSPEED RIGHT HAND</td>
<td>$20 \text{ to } 180 \text{ knots}$</td>
</tr>
<tr>
<td>$H_p$</td>
<td>PRESSURE ALTITUDE</td>
<td>$0 \text{ to } 15000 \text{ ft}$</td>
</tr>
</tbody>
</table>

Figure 7.2 Cessna spin test airplane
spin chute does not deploy or will not release, permitting him to leave the airplane in safety.

The right-hand and left-hand wing tip booms are shown in more detail in Figures 7.3. The booms utilize a flow direction and airspeed sensor (described in detail in Reference 31). This sensor allows determination of the airspeed, angle of attack, and angle of sideslip (as shown in Figures 7.3). A probe is included on each wing tip to allow determining the true properties of the spin. The axis of a spin is generally not at the center of gravity of the airplane. (In the C172 it appears to be ahead of the center of gravity.) Providing both left-hand and right-hand measurements allows determining where this spin axis is by using the differences between the measurements from each side.

Figure 7.3(a) Cessna wing tip booms (supplied by NASA Langley)
Figure 7.3(b) Cessna wing tip booms (supplied by NASA Langley)

Figure 7.3(c) Cessna wing tip booms (supplied by NASA Langley)
Inside the airplane cockpit the Cessna inertial reference transducers ($p, q, r, A_x, A_y$) were mounted on the sensor pallet as shown in Figure 7.4. As can be seen, the KU-FRL power supply system was used in this installation. This was necessary to provide power for the computer and was also utilized to provide power for some of the transducers. Figure 7.1 shows the power sources used for the specific devices.

The KU-FRL data management computer is shown installed in the Cessna airplane in Figure 7.5.

A chase airplane was used in this flight test program. This was for safety purposes to provide an outside observer who could warn the pilot of the spin test airplane (over the communications radio) of any unexpected problems. Also, a video camera was carried onboard the chase airplane to record the spin visually.

7.2 Data Reduction

Data analysis for this spin program was done by Cessna on their Hewlett Packard 9825 microcomputer. Data was transferred from the KU-FRL system to the Cessna computer, using the standard RS232 ports on each machine (see Appendix A.9 for Hewlett Packard 9825 programs). After transfer, the data was plotted on Cessna's computer using the program in Appendix A.9. Figures 7.6 present the traces of several of the spins. It can be seen that the data recorded produce results capable of analysis.
Figure 7.4  Cessna spin test instrumentation installation

Figure 7.5  Cessna spin test computer installation
Figure 7.6(a) Cessna spin test, spin traces, spin No.9
Figure 7.6(b) Cessna spin test, spin traces, spin No.10
7.3 Results of Spin Program

The results of the spin program, from the aspect of this report, show the adaptability of the KU-FRL-designed data management system. The versatility specifically designed into this portion of the system allows virtually any 16-channel instrumentation combination to provide the measurements. Also shown is the feasibility of using different data reduction computers by the use of the standard RS232 port for data transfer.

No real problems were encountered by Cessna personnel in using the data management system, even though none of them had any extensive microcomputer experience.
8. CONCLUSIONS

The flight test system designed and evaluated under this program has met the objectives outlined in Chapter 2. The system

- is easy to install,
- is virtually self contained,
- is simple in operation,
- requires no complex flight maneuvers,
- is applicable to general aviation airplanes,
- is capable of longitudinal and lateral stability analysis, and
- is low in cost.

This system has shown that the technology used is capable of the tasks to be performed.

In the data reduction method all the derivatives contained in Equation [5.5] can be determined. The method also allows determining any combination of these derivatives. It must be noted that these are the state vector dimensional derivatives which can be converted to the normally accepted stability derivatives (as per Reference 22) using Tables 5.2 and 5.3.

Areas have been discovered where further work is required. A comprehensive list is included in Chapter 9.
9. RECOMMENDATIONS FOR PHASE III

Four areas have been suggested throughout this report for improvement of the KU-FRL instrumentation system. These are summarized here.

9.1 Equipment

- Equipment is required for accurate transducer calibration. A pendulum arrangement as per Reference 32 is suggested as an excellent means of calibrating the transducers.

- Size reduction of equipment is suggested. To allow easier placement in aircraft, the size of the system could be reduced significantly, especially if the number of packages is increased to form more efficient space utilization.

9.2 Calibration

- All transducers should be calibrated as a system. Using the actual data acquisition package for transducer calibration is suggested as a means to reduce calibration errors. This should be done in conjunction with the calibration pendulum of 9.1 above.

9.3 Data Reduction

- Refinements are required to the current MMLE "BONES" program to simplify its use and add to the versatility.

- Further study is suggested to allow Performance analysis (i.e., Drag Polar) of the test airplane. Methods similar to those of References 4-11, 13, and 18 seem to provide promising solutions.
- Some of the features of the latest version of MMLE (Reference 33) should be added. Specifically, the Cramer-Rao bounds addition, and the correction for center of gravity offsets can be added directly into the MMLE program.

- The addition of the acceleration transformation matrix ([R] of Equation [5.4]) to the MMLE program should be explored.

- Determine the validity of the prediction of $\alpha$ and $\beta$ by comparing with measured values.

9.4 **Effect of Control Surface Float**

- The effect of control surface cable tension should be evaluated to determine the influence this has on the parameters predicted.

9.5 **Proof-of-Test Capability**

- Tests are recommended in other general aviation airplanes to demonstrate the system's adaptability. Recommended are tests on a high performance, single-engine retractable, and on a light-to-medium, twin-engine airplane.

- The tests suggested above would also aid in providing further insight into the possible "Uniqueness" problem. This should be a definite area of research, to validate the MMLE (or similar) concepts.
10. REFERENCES


22. Roskam, J.; "Airplane Flight Dynamics and Automatic Flight Controls"; Roskam Aviation and Engineering Corporation; Ottawa, Kansas;(Rte 4, Box 274);1979


10.1 **Instrumentation System Reports**

<table>
<thead>
<tr>
<th>KU-FRL Number</th>
<th>Title</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>407-1</td>
<td>A Literature Survey of Performance and Stability Flight Testing</td>
<td>1979</td>
</tr>
<tr>
<td>407-4</td>
<td>Calibration of MDAS-16 Analog-to-Digital Converter</td>
<td>1981</td>
</tr>
</tbody>
</table>
APPENDIX A

PROGRAMS

This appendix includes descriptions, flow charts, and listings of the computer programs required by this flight test system.

A.1 Data Acquisition (AIM-65)
A.2 Data Transfer (AIM-65)
A.3 Data Receive (MINC 11/03)
A.4 Engineering Conversion (MINC 11/03)
A.5 Quick Look Plots (MINC 11/03)
A.6 Detailed Engineering Conversion (MINC 11/03)
A.7 MMLE BONES Routines (MINC 11/03)
   .1) MMLE Set-Up
   .2) Main MMLE Programs
   .3) MMLE Output Format
A.8 Time History Plotting (MINC 11/03)
A.9 Cessna Programs
   .1) Data Acquisition
   .2) Data Readback
   .3) Data Receive
   .4) Data Plotting
Description: This program, which runs on the AIM 65, collects and saves the measured state time histories. The information is collected and stored on the cassette tape in one-second real-time blocks. The data for each channel is coded as two binary eight-bit words totalling sixteen bits. The first word holds the eight most significant bits. The second word holds the four most significant bits and the four least significant bits. This gives a redundancy check of the highest order bits.

Flowchart:
PROGRAM LISTING

; DATA ACQUISITION

RNCNT=0
BLKCNT=2
BUFCTN=4
ISBUF=5
OBUF=7
CNT=9
BUF1=$200
BUF2=$300
KDDRA2=$A481
KDDRB2=$A483
KDDRA2=$AA480
KDDRB2=$AA482
DBR=$900B
WDC=$9009
CDBR=$900A
NDRO=$900B
CSR=$900C
ESR=$900D
ISR=$900E
KDDRA2=$A481
KDDRB2=$A483
KDR82=$A482
KIM=$A483
KDDRS2=$A483
KIM2=$A483
KDDRA2=$A481
KDDRB2=$A483
MAIN
LDA #$00
STA KDDRB2
STA KDDRA2
LDA #$C0
STA UACR
START
LDA #$12
STA KDRO
LDA FREW
JSR COND
LDA FREW
JSR COND
MAIN
LDA #REW
JSR SWAP
JSR WRITE
RECI
JSR WRITE
RECI
JSR WRITE
RECI
INC BLKCNT
BNE RECI
INC BLKCNT+1
JMP RECI
RECI
LDA BUFCTN
LDA #$20
BNE RECI
LDA #$A0
STA UIER
JSR SWAP
INC BLKCNT
BNE RECI
INC BLKCNT+1
RECI
JSR WRITE
JSR ENREC
JSR SWAP
LDA #$FF
STA BLKCNT
STA BLKCNT+1
JSR WRITE
INC RNCNT
BNE RECI
INC RNCNT+1
RECI
JMP MAIN2

--- N ---

LDA 0BUF2
*$0400 STA OBUF+1
LDA 0$92 JSR ENDREC
STA OBUF
STA OBUFF
LDA #>BUF1
LDA IBUFF
STA IBUFF+1
LDA #>BUF2
STA OBUFF+1
JSR ENDREC
LDA #<INT
STA S6405
LDA #<INT
STA S6404
LDA #$C0
STA UIER
LDA #<$C14E
STA UTIL
LDA #$C14E
STA UTICH
CLI
LDA #0
STA BLKCNT
STA BLKCNT+1
RECI
JSR SWAP
JSR WRITE
RECI
JSR WRITE
RECI
INC BLKCNT
BNE RECI
INC BLKCNT+1
JMP RECI
RECI
LDA BUFCTN
LDA #$20
BNE RECI
LDA #$A0
STA UIER
JSR SWAP
INC BLKCNT
BNE RECI
INC BLKCNT+1
RECI
JSR WRITE
JSR ENREC
JSR SWAP
LDA #$FF
STA BLKCNT
STA BLKCNT+1
JSR WRITE
INC RNCNT
BNE RECI
INC RNCNT+1
RECI
JMP MAIN2

112
PROGRAM LISTING (continued)

______________________________
GKEY
LDA KDRB2
PHA
LDA #TIME1L
STA UT2L
LDA #TIME1H
STA UT2H
GKEY1
LDA UIFR
AND #8153
BEQ GKEY1
PLA
CMP KDRB2
BNE GKEY
RTS
______________________________
WAIT
JSR WAITX
WAITX
RTS

______________________________
WRITE
LDA ESR
LDA #224
STA WDC
LDA AVRT
STA CDR
LDA RNCN1
JSR WWORD
LDA RNCN1+1
JSR WWORD
LDA BLCN1
JSR WWORD
LDA BLCN1+1
JSR WWORD
LDC #0
WRT1
LDA (OBUF),Y
JSR WWORD
LDA RNCN1+2
JSR WWORD
LDA SEKCT
JSR WWORD
LDA SLKCNT+1
JSR WWORD
LDY 10
WRITEI
LDA (OBUF),Y
JSR WWORD
INY
CPY 1220
SHE WRITEI
WRITE2
JMP COMD2

______________________________
SWAP
LDA OBUF+1
PHA
LDA IBUF+1
STA OBUF+1
PLA
STA IBUF+1
LDA #0
STA BUFCNT
RTS

______________________________
WWORD
PHA
WWORD1
LDA ISR
AND #DBRE
BEQ WWORD1
PLA
STA DBR
RTS

______________________________
INT
PHA
LDA UDRB
BPL INTEX
TYA
PHA
LDA #BUFCNT
LDA #11
STA CNT
LDA $8000
JSR WAIT
ILOOP
LDA $8002
STA (IBUF),Y
LDA $8011
INY
LDA $803
STA (IBUF),Y
INY
DEC CNT
BNE ILOOP
STY BUFCNT
PLA
TAY
INTEX
LDA UTIL
PLA
RTI
END

______________________________
A.2) **DATA TRANSFER**

Description: This program allows the AIM 65 to read the information stored in the DATA ACQUISITION program. The information is passed to the MINC 11/03 computer using the RS 232 port.

Flowchart:
PROGRAM LISTING

; DATA RECOVERY

RNCHT=0
BLKCNT=2
VRUN=4
BLK=5
VRUN=4
BUFI=8200
DBR=9008
WOC=9009
CDR=890A
NDR=900B
CSR=900C
ISR=900D
MDR=900F
SLP=8CB
RLD=3CA
REM=3CA
MDY=910
TDRE=502
CR=50D
SCR=9006
SDR=9007
LOAD=4AC
READ=552
CLOSE=443
INALL=8E993
NUMA=8E406
READR=8E93C
OUTALL=8E98C
OUTPUT=8E97A

UTILITY=5A004
UTICH=5A005
UACR=5A00B

;#3000
CCZ
.BYTE $80
DA
.BYTE $2B
NO
.BYTE CR,'TAPE ERROR',5A0
HARUN
.BYTE CR,'WHICH RUN NUMBER',5BF
MBLK
.BYTE CR,'HOW MANY BLOCKS',5BF
MEND
.BYTE CR,'LAST BLOCK HIS RU',5CE
MINV
.BYTE CR,'INVALID COMMAND',5CA
MERR
.BYTE CR,'FILE MARK FOUN',5CA
NRENT
.BYTE CR,'RUN NUMBER',5A0
MERROR
.BYTE CR,'DATA ERROR',5BF
TEMPO
.BYTE $00

;#4000
RESETB
LDA #592
.STA MDR0
LDA #5C0
STA UACR
LDA #688
;568=300BAUD
;34=400
;51A=1200
;50=2400

STA UTIL
LDA #0
STA UTICH
LDA #811
SKR #FF
STA SCR

MAIN
JSR GCCNT
CMR #LOADC
BEQ MAIN2
JSR INVALID
JSR COMDA
LDA #SLP
JSR COMDA
MAIN3
JSR GCCNT
CMP #LOADC
BEQ READ
CMP #LOADC
BEQ READ
CMP #LOADC
BEQ READ
JSR INVALID
JSR MAIN3

READ
LDY #VRUN-MO
JSR MESS
JSR GCCNT
CMR #0
BEQ CLOSE
STA VRUN
READ1
JSR RSLK
BCLCLOSE
LDA BLKCNT+1
ORA BLKCNT+1
BNE READ3
LRY #NRCNT-MO
JSR MESS

LDA RNCHT
JSR NVR
READ3
LDA RNCHT
CMR VRUN
RE4 READ1
BEQ READ2
LDA #REV
JSR COMDA
LDA #SLP
JSR COMDA
JMP READ1
READ2
LRY #MBLK-MO
JSR MESS

JSR GCCNT
CMR #0
BEQ CLOSE
STA VBLK

;CHANGE TO NOPS
;TO TRANSMIT COUNTS
SEND1
JMP SENDB1
LDA RNCHT
JSR SEND
LDA RNCHT+1
JSR SEND
LDA BLKNT
JSR SEND
LDA RBLKNT+1
JSR SEND
SEND2
LDA BUFX
;COMPARE HIGH BITS
AND #$FF
STA TEMPO
INX
LDA BUFX
AND #$FF
CMP TEMPO
BEQ CWNT1
JSR FIX
; IF BAD FIX HERE
CWNT1

DUX
LDA BUFX
; CONVERT TO PRINTABLE
SEC
ROR BUFX
;
LDA BUFX
AND #$33
CLC
ROL A
ROL A
ROL A
ROL A
ROL A
ROL A
ROL A
ROL A

BNE TEMP0
INX
LDA BLKNT
AND #$FF
ORA TEMPO
ORA #$40
STA BUFX

CPE #220
BNE CVNT
LRY #0
SEND2
LDA BUFX
JSR SEND
INY
CPE #20
BNE SEND3
LDA BLKNT
CMP #$FF
BNE SEND3

CPE BLKNT+1
BEQ END
SEND3
DEC VBLK
BNE SEND5
SEND4
LRY #MBLK-MO

116
JSR MESS
JSR GCNT
CMP #0
BEQ CLOS/E
STA VBLK
SEND5
JSR RBLK
BCS CLOS/E
JMP SENDB
CLOSE5
JMP CLOSE
END
LDY #MEND-MO
JSR MESS
JMP MAIN
----------
MESS
LDA NO,Y
PHA
AND #$7F
JSR OUTPUT
INY
PLA
BPL MESS
RTS
----------
GCNT
LDA #0
STA CNT
GCNT1
JSR INALL
JSR DPACK
BCC GCNT1
LDA CNT
RTS
----------
DPACK
CMP #'0'
BCC RSPAC
CMP #$3A
BCE RSPAC
AND #$FF
PHA
LDA CNT
ASL A
STA CNT
CLC
ADC CNT
ASL A
STA CNT
CLC
RST
RST
----------
SEND
PHA
SEND1
LDA SCR
AND #FDBE
BNE SEND1
PLA
EDR #$FF ;CHG FOR KNOWN CHAR
STA SDR
RTS
---------
RBLK
LDA CSR
AND #HDBE
BNE RBLK1
LDA #$24
STA WDC
LDA #$2D
STA CDR
JSR RWORD
BCE RBLK1
STA RMCT
JSR RWORD
BCE RBLK2
STA RMCT+1
JSR RWORD
BCE RBLK2
STA BLKCT
JSR RWORD
BCE RBLK2
STA BLKCT+1
LDY #0
RBLK1
JSR RWORD
BCE RBLK2
STA BUF1,Y
INY
CPY #$220
BNE RBLK1
JMP CONDA2
RBLK2
JMP CONDA4
----------
CONDA
PHA
LDA ESR
CONDA1
LDA CSR
AND #HDBE
BNE CONDA1
PLA
STA CDR
CONDA4
LDA ISR
AND CCE
BEQ CONDA2
COMDA4
LDA CSR
PHA
AND #$2
BEQ CONDA5
LDY #ERRL-MO
JSR MESS
PLA
STA CNT
RTS
COMDA5
PLA
----------
AND #$81
BNE CONDA3
CLC
RTS
CONDA3
LDY #MO-MO
PHA
JSR MESS
PLA
JSR MESS
LDA ESR
JSR MESS
CLC
--------
RWORD
LDA ISR
BIT CCE
BNE RWORD2
BIT DA
BEQ RWORD
LDA DSR
CLC
RTS
--------
RWORD2
SEC
RTS
----------
FIX
LDY #ERROR-MO
JSR MESS
RTS
----------
;INITIAL SET UP
<<$700
LDA #$C0
STA UACK
LDA #$60
STA UTIL
LDA #$50
STA UTICN
JMP UTIC
END
A.3) DATA RECEIVE

Description: This program accepts raw data from the AIM 65. This raw, formatted data is collected in files to be used in the ENGINEERING CONVERSION routine.

Program listing

```
0001 PROGRAM AIMIN
0002 C AIM TO MINC PROGRAM
0003 C WRITTEN BY MARK A MOSSER
0004 C
0005 C This program inputs data from the AIM - 65 through SLU-1 as characters (22 at a time) to fill a 600 X 22 character array. When full, this array is outputed to the user specified file.
0006 C
0007 DIMENSION IADDR(4), IDATA(600,22), ICHAR(22)
0008 TYPE # 'THIS PROGRAM READS 22 CHARACTER WORDS FROM SLU1 AT A TIME AND PLACES THESE WORDS ON A FILE'
0009 TYPE # 'I WILL ATTACH SLU 1'
0010 IERR=MATCH(2)
0011 TYPE # 'WHAT IS THE NAME OF YOUR OUTPUT FILE ?'/
0012 CALL ASSIGN (lrr-1, NEW)
0013 TYPE # '1ST I WILL ATTACH SLU 1'
0014 IADDR(1) = '50010
0015 IADDR(2) = 0
0016 IADDR(3) = 0
0017 IADDR(4) = 0
0018 IERR = MGET(2, IADDR(1))
0019 TYPE # 999, IERR
0020 TYPE # 'NOW YOU HAVE 2 CHOICES, STOP OR READ IN DATA'
0021 TYPE # '1 = READ IN DATA'
0022 TYPE # '2 = STOP'
0023 TYPE # 'WHICH ?'
0024 ACCEPT $, IFCT
0025 FORMAT (I1)
0026 IF (IFCT .EQ. 2) GOTO 10
0027 GOTO (100, 200), IFC
0028 C
0029 100 TYPE $,
0030 110 DO I=1,600
0031 120 J=1,22
0032 IERR = MGET(2, ICHAR(J), I)
0033 IDATA(I,J)=ICHAR(J)
0034 120 CONTINUE
0035 10 TYPE $, 'COUNT = ', I
0036 110 CONTINUE
0037 10 TYPE $, 'WAIT FOR DISK OUTPUT'
0038 DO 135 K=1,600
0039 WRITE (1,169) (IDATA(K,L),L=1,22)
0040 169 FORMAT (22I41)
0041 135 CONTINUE
0042 10 GOTO 10
0043 C STOP
0044 TYPE $, 'DETACH INPUT PORT'
0045 IERR = MDETACH(2)
0046 TYPE # 999, IERR
0047 STOP
0048 END
```
A.4) ENGINEERING CONVERSION

Description: This process presently uses two programs which run separately. The first routine, AIMCNV, converts the raw coded AIM 65 data to voltages. The AIMCNV program makes use of a macro assembly language routine (CONVRT) which performs the bit manipulations necessary to turn the AIM 65 coded data into a form useable by the MINC Fortran programs. The second routine, EGRCNV, converts the voltages into engineering units.

Program listing:

```
C
C	 AIM TO VOLT CONVERSION PROGRAM
C	 IT MUST BE LINKED TO CONVRT TO WORK
C
0001 PROGRAM AIMCNV
0002 DIMENSION IN(22),VOLT(11),IVOLT(11)
0003 TYPE $ ' AIM FORMAT TO VOLT CONVERSION '
0004 TYPE $ ' INSERT A DATA DISKETTE '
0005 5 TYPE $ ' what is the full name of the INPUT file?'
0006 TYPE $
0007 CALL ASSIGN('1',-1,RDO)
0008 10 TYPE $ ' What is the full name of the OUTPUT file?'
0009 TYPE $
0010 CALL ASSIGN('2',-1,NEW)
C
0011 20 TYPE $ ' How many SECONDS (10 time pts) do You want?'
0012 ACCEPT $NUM
C
C convert this mess

0013 NUM=NUM+10
0014 DO 150 I=1+NUM
0015 READ (1,109,END=300) (IN(J),J=1,22)
0016 109 FORMAT(22A1)
0017 DO 150 K=1+I
0018 KH=(K#2)-1
0019 KL=K#2
0020 CALL CONVRT(IN(KH),IN(KL),IVOLT(K))
0021 VOLT(K)=IVOLT(K)/16
0022 150 VOLT(K)=$1.02435632861+.06330029324
0023 WRITE (2,119) (VOLT(J),J=1+I)
0024 119 FORMAT (11F12.9)
0025 TYPE $ ' COUNT=',I
0026 100 CONTINUE
C
C Here is the series of questions
C
0027 TYPE $ ' More in this file?'
0028 ACCEPT 209+NYR
0029 209 FORMAT(A1)
0030 IF (NYR .EQ. 'Y') GOTO 20
0031 TYPE $ ' Another OUTPUT file from this INPUT file?'
0032 ACCEPT 209+NYR
0033 209 IF (NYR .EQ. 'Y') GOTO 210
0034 TYPE $ ' Another INPUT file in this OUTPUT file?'
0035 ACCEPT 209+NYR
0036 210 IF (NYR .EQ. 'Y') GOTO 310
0037 TYPE $ ' Are you done?'
0038 ACCEPT 209+NYR
0039 209 IF (NYR .EQ. 'N') GOTO 5
0040 GOTO 500
0041 210 CALL CLOSE(2)
0042 GOTO 10
0043 300 TYPE $ ' END OF INPUT FILE ERROR'
```

121
PROGRAM EORCNV

C Volts to Engineering Units conversion program
C By Mark A. Mosser Dec. 1980

COMMON VOLTS(1:11), ENG(10:11)

10 TYPE 8, 'Volts to Engineering Units conversion'
0004 TYPE 5, 'INSERT DATA DISK'
0005 TYPE 8, 'What is the full name of your INPUT file?'
0006 TYPE 8
0007 CALL ASSIGN(1,-1,RDO)
0010 30 TYPE 8, 'What is the full name of your OUTPUT file?'
0011 TYPE 8
0010 CALL ASSIGN(2,-1,NEW)

C INPUT first block (TPS+PS) convert 3 output
C
READ(1,109) ((VOLTS(J,K),K=1,11),J=1,10)
   109 FORMAT(11F12.9)
0013 CALL TOTPSD(T,PS,PD)
0014 119 FORMAT(3E12.4)
0015 READ (1,105) ((VOLTB(J,K),K=1,11),J=1,10)
0016 CALL VTDEGR
0017 100 READ(1,109,END=300,ERR=1000) ((VOLTS(J,K),K=1,11),J=1,10)
0018 WRITE (2,129) (ENG(J,K),K=1,11),J=1,10
0019 129 FORMAT(11E12.4)
0020 L. VTDEGR
0021 GOTO 100
C
C LOOP EXITED BY EOF IN READ
C
0022 300 WRITE (2,119) T,PS,PD
0023 CALL TOTPSD(T,PS,PD)
0024 WRITE (2,119) T,PS,PD
0025 CALL CLOSE(1)
0026 CALL CLOSE(2)
0027 TYPE ' ANOTHER FILE?'
0028 ACCEPT 777: HORY
0029 FORMAT(A1)
0030 IF ( HORY .EQ. 'Y') GOTO 10
0032 GOTO 500

C
C ERROR
C
0033 1000 TYPE ' ERROR IN READ'
0034 500 STOP
0035 END

C
C Subroutine TOTPSD
C This subroutine converts the data to T (temperature) and PD (dynamic pressure)
C
0001 SUBROUTINE TOTPSD(T,PS,PD)
0002 COMMON VOLTS(10:11), ENOS(10:11)
0003 II=2
0004 JJ=1
0005 T=0
0006 PS=0
0007 PD=0

C
C Loop to fill T,PS,PD
C
0008 11 T=T+VOLTS(II,JJ)
0009 JJ=JJ+1
0010 IF (JJ .LE. 11) GOTO 21
0012 JJ=JJ-11
0013 II=II+1
0014 21 PD=PD+VOLTS(II,JJ)
0015 JJ=JJ+1
0016 IF (JJ .LE. 11) GOTO 31
0018 JJ=JJ-11
0019 II=II+1
0020 31 PS=PS+VOLTS(II,JJ)
0021 JJ=JJ+3
0022 IF (JJ .LE. 11) GOTO 11
0024 IF (II .EQ. 10) GOTO 101
0026 JJ=JJ-11
0027 II=II+1
0028 GOTO 11

C
C Average T,PD,PS
C
0029 101 T=T/20
0030 PD=PD/20
0031 PS=PS/20

C
C Convert to engineering units
C
0032 T=((((280*T)-487)-273.16)*(9/5))+32
0033 IF (PD .LE. 4.526) GOTO 206
0035 PD=6658.67-(1436.61*PD)
0036 GOTO 216
0037 204 PD=2646.58-(550.6*PD)
0038 214 IF (PS .LE. 2.303) GOTO 226
0040 PS=(12917.896*PS)-19754.521
0041 GOTO 236
0042 224 PS=(9451.796*PS)-11767.49
0043 234 CONTINUE
0044 RETURN
0045 END
Subroutine VTOEOR

This subroutine converts the data from volts to engineering units. It then reorders them as follows:

\[ \theta, \phi, \delta, \phi, \Delta, \delta, \phi, \Delta \]

```
0001 SUBROUTINE VTOEOR
0002 COMMON VOLTS(10,11), ENO(10,11)
0003 DO 102 IA = 1, 10
0004    ENO(IA+2) = ((VOLTS(IA, 9)) * 0.5111) + 0.00733
0005    ENO(IA+3) = ((VOLTS(IA, 5) + 0.002) / 0.001) - 1.0
0006    ENO(IA+4) = ((VOLTS(IA, 6)) / 2.499
0007    ENO(IA+5) = ((VOLTS(IA, 6)) / 2.499
0008    ENO(IA+6) = ((VOLTS(IA, 6)) / 2.499
0009    ENO(IA+7) = ((VOLTS(IA, 6)) / 2.499
0010    ENO(IA+8) = VOLTS(IA, 6) / 10.027
0011    ENO(IA+9) = ((VOLTS(IA, 7)) - 0.011) / 10.134
0012    CONTINUE
0013    RETURN
0014    END
```
A.5) **QUICK LOOK PLOTS**

Description: The QUICK LOOK PLOT program is used as an aid in choosing appropriate flight data for further analysis. The routine collects the engineering units data and plots it on a graphics CRT terminal.

Program listing:
0037 TYPE 51
0038 FORMAT(' DO YOU WANT TO TAKE ANOTHER LOOK AT THE DATA? (Y/N)')
0039 READ(5,24) ANS
0040 IF(ANS.EQ.YES) GO TO 223
0042 TYPE 52
0043 FORMAT(' DO YOU WANT TO LOOK AT ANOTHER DATA FILE? (Y/N)')
0044 READ(3,24) ANS
0045 CALL CLOSE(2)
0046 IF(ANS.EQ.YES) GO TO 1
0047 CALL PLOT55(2,512,1+2+4+32+64,ISTAT)
0048 CALL PLOT55(0,-1,0,ISTAT)
0050 RETURN
0051 END

0001 SUBROUTINE INIT
0002 COMMON/STATUS/ISTAT(16)
0003 DATA ISTAT/16#0/
0004 CALL PLOT55(13,72,,ISTAT)
0005 CALL PLOT55(13,74,,ISTAT)
0006 CALL PLOT55(2,1+512,,ISTAT)
0007 RETURN
0008 END

0001 SUBROUTINE FOUR(NZ)
0002 COMMON/STATUS/ISTAT(16)
0003 DATA ISTAT/16#0/
0004 CALL PLOT55(13,72,,ISTAT)
0005 CALL PLOT55(13,74,,ISTAT)
0006 CALL PLOT55(2,1+512,,ISTAT)
0007 RETURN
0008 END

0013 K1 = 3
0014 K2 = 2
0015 K3 = 1
0016 K4 = 5
0017 GO TO 54
0018 52 CONTINUE
0019 K1 = 9
0020 K2 = 11
0021 K3 = 4
0022 K4 = 5
0023 GO TO 54
0024 53 CONTINUE
0025 K1 = 8
0026 K2 = 7
0027 K3 = 6
0028 K4 = 10
0029 54 CONTINUE

0030 DO 41 K=1,N
0031 IARRAY(2*K) = DATA(K*K1)*.57295*GAIN(K1)+45
0032 IARRAY(2*K-1) = DATA(K*K2)*.57295*GAIN(K2)+90
0033 41 CONTINUE
0034 CALL GRAPH(2*N,IARRAY)
0035 DO 42 K=1,N
0036 IARRAY(28K) = DATA(K,K3)*.57295#GAIN(K3)+135
0037 IARRAY(28K-1) = DATA(K,K4)*.57295#GAIN(K4)+180
0038 42 CONTINUE
0039 CALL GRAPH(2N+IARRAY)

C

CALL PLOT55(9,25,1,ISTAT)
CALL PLOT55(12,'G# QUICK LOOK DATA PLOTS ***',ISTAT)

C

IF(NZ.EQ.1.) GO TO 31
IF(NZ.EQ.2.) GO TO 32
IF(NZ.EQ.3.) GO TO 33

0048 31 CONTINUE
0049 CALL PLOT55(9,50,4,ISTAT)
0050 CALL PLOT55(12,'ELEVATOR POSN. 20 DEG.',ISTAT)
0051 CALL PLOT55(9,50,8,ISTAT)
0052 CALL PLOT55(12,'PITCH ATTITUDE, 30 DEG.',ISTAT)
0053 CALL PLOT55(9,50,13,ISTAT)
0054 CALL PLOT55(12,'PITCH RATE, 50 DEG/SEC.',ISTAT)
0055 CALL PLOT55(9,50,17,ISTAT)
0056 CALL PLOT55(12,'NORMAL ACCEL., 2 G.',ISTAT)

0057 32 CONTINUE
0058 CALL PLOT55(9,50,4,ISTAT)
0059 CALL PLOT55(12,'ELEVATOR POSN. 20 DEG.',ISTAT)
0060 CALL PLOT55(9,50,8,ISTAT)
0061 CALL PLOT55(12,'LONGITUDINAL ACCEL., 5 G',ISTAT)
0062 CALL PLOT55(9,50,13,ISTAT)
0063 CALL PLOT55(12,'Rudder Posh ., 20 DEG.',ISTAT)
0064 CALL PLOT55(9,50,17,ISTAT)
0065 CALL PLOT55(12,'YAW RATE, 50 DEG/SEC.',ISTAT)

0066 33 CONTINUE
0067 CALL PLOT55(9,50,4,ISTAT)
0068 CALL PLOT55(12,'AILERON POSH., 20 DEG.',ISTAT)
0069 CALL PLOT55(9,50,8,ISTAT)
0070 CALL PLOT55(12,'BANK ANGLE, 60 DEG.',ISTAT)
0071 CALL PLOT55(9,50,13,ISTAT)
0072 CALL PLOT55(12,'ROLL RATE, 50 DEG/SEC.',ISTAT)
0073 CALL PLOT55(9,50,17,ISTAT)
0074 CALL PLOT55(12,'LATERAL ACCEL., .5 G.',ISTAT)

0075 34 CONTINUE
0076 CALL PLOT55(9,0,20,ISTAT)
0077 DO 1 J=1,5000
0078 DAMMY= COS(45)
0079 1 CONTINUE
0080 RETURN
0081 END

0001 DO 1 J=1,5000
0002 DAMMY= COS(45)
0003 1 CONTINUE
0004 RETURN
0005 END
0006 SUBROUTINE GRID(IDX, IDY)
0007 COMMON/STATUS/ISTAT(16)
0008 CALL PLOT55(2,1+512,ISTAT)
0009 CALL PLOT55(9,0,0,ISTAT)
0010 CALL PLOT55(10,ISTAT)
0011 DO 3 I=1,512,IDX
0012 CALL PLOT55(5,I-I+1,ISTAT)
0013 DD 4 I=1,512,IDX
0014 CALL PLOT55(4,1,1,1,ISTAT)
0015 RETURN
0016 END

0007 DO 3 I=1,512,IDX
0008 CALL PLOT55(S,I-I+1,1,ISTAT)
0009 DD 4 I=1,512,IDX
0010 CALL PLOT55(4,1,1,1,ISTAT)
0011 RETURN
0012 END

0001 SUBROUTINE GRID(1N+IARRAY)
0002 COMMON/STATUS/ISTAT(16)
0003 DIMENSION IARRAY(512)
0004 COMMON STATUS/ISTAT(16)
0005 CALL PLOT55(7,0,0,ISTAT)
0006 CALL PLOT55(8,512,0,ISTAT)
0007 CALL PLOT55(2,1+(NUMBER+1)*2,(NUMBER+1)*10,ISTAT)
0008 CALL PLOT55(3,-N+IARRAY,ISTAT)
0009 CALL PLOT55(1,1-NUMBER,ISTAT)
0010 CALL PLOT55(9,10,1,ISTAT)
0011 END
A.6) DETAILED ENGINEERING CONVERSION

Description: The CRINST program performs the detailed corrections for instrument offsets from the body axes. Biases on the accelerometers are also removed in the corrections.

Program listing:

0001 PROGRAM CRINST
0002 C.... PROGRAM TO MODIFY THE RAW ENGINEERING
0003 C.... DATA FOR INSTRUMENT CORRECTIONS
0004 C.... THAI MEASURED FROM BODY TO INST AXES
0005 C.... XBAR, YBAR, AND ZBAR FROM BODY TO INST AXES
0006 BYTE NAME(15)
0007 DIMENSION FI(11),FIM1(11),FIP1(11),DATA(600,11)
0008 DATA DORP /57.2957SP329174/
0009 DATA THAI /-5.696/
0010 DATA XBARPYBARPZBAR /+0.052r+1.1799+1.630/
0011 C.... TRANSDUCER POSITIONS RECALCULATED ON 6-FEB-81
0012 4 FORMAT(14A)
0013 5 FORMAT(I10,BF12.4)
0014 C.... SET LAST BYTE OF CHARACTER STRING TO NULL
0015 NAME(15)=0
0016 C.... ENTER THE FILE NAME FOR THE DATA TO BE CORRECTED
0017 10 FORMATobicenter THE FILE NAME FOR THE RAW ENGINEERING DATAbic
0018 TYPE 10
0019 ACCEPT 4P(NAME(I)PINIP14)
0020 OPEN(UNIT
0021 NAME=NAMEPTYPEn'OLD'PACCESSn'SEQUENTIAL'.
0022 READONLY.Form='FORMATTED',RECORDSIZE=132)
0023 C.... ENTER THE FILE NAME TO HOLD THE CONVERTED DATA
0024 20 FORMATobicenter THE FILE NAME TO HOLD THE CONVERTED DATAbic
0025 TYPE 20
0026 ACCEPT 4P(NAME(I)PINIP14)
0027 OPEN(UNIT
0028 NAME=NAMEPTYPEn'NEW'PACCESS='SEQUENTIAL'.
0029 UNFORMATTED'.
0030 BUFFERCOUNT=2)
0031 C.... READ DATA FROM HP DATA FILE
0032 28 FORMATobicenter THE NUMBER OF TIME POINTS TO BE CORRECTEDbic
0033 TYPE 28
0034 29 FORMAT(I10)
0035 ACCEPT 29:NIEND
0036 30 FORMAT(11E12.4)
0037 DO 31 I=1:NIEND
0038 READ(1,30)(DATA(I,J),J=1,11)
0039 31 CONTINUE
0040 CLOSE(UNIT=1)
0041 C.... TRANSFER FIRST TWO DATA POINTS
0042 DO 35 I=1:11
0043 FIM1(I)=DATA(I,1)
0044 35 CONTINUE
0045 C.... CORRECT FOR SIGN ERRORS IN CALIBRATIONS
0046 FIM1(4)=-FIM1(4)
0047 FIM1(6)=-FIM1(6)
0048 FIM1(8)=-FIM1(8)
0049 FIM1(9)=-FIM1(9)
0050 FIM1(11)=-FIM1(11)
0051 C.... CORRECT FOR GYRO MISALIGNEMENT
0052 COSTHI=COS(THAI/DGR)
0053 SINTHI=SIN(THAI/DGR)
0054 FIM1(1)=FIM1(1)-THAI/DGR
0055 FIM1(7)=FIM1(7)*COSTHI+FIM1(9)*SINTHI
0056 FIM1(7)=FIM1(7)*SINTHI+FIM1(9)*COSTHI
0057 FIM1(7)=PI
0058 FIM1(9)=RI
0059 C.... WRITE THESE VALUES TO THE OUTPUT FILE
0060 WRITE(2)(FIM1(I),I=1,11)
C.... TRANSFER NEXT DATA POINT
0044  DO 36 I=1,11
0047   FI(I)=DATA(I)
0048  36 CONTINUE
C.... CORRECT FOR SIGN ERRORS IN CALIBRATIONS
0049   FI( 4) =-FI( 4)
0050   FI( 6) =-FI( 6)
0051   FI( 8) =-FI( 8)
0052   FI( 9) =-FI( 9)
0053   FI(11) =-FI(11)
C.... ADD CORRECTION FOR GYRO MISALIGNMENT
0054   FI(1)=FI(1)-THAI/DOR
0055   PI=FI(7)*COSTH+FI(9)*SINTH
0056   RI=FI(7)*SINTH+FI(9)*COSTH
0057   FI(7)=PI
0058   FI(9)=RI
C.... PRINT OUT AVERAGE VALUES FOR THI1 AND PHI1
0059   THAI=((FI1(1)+FI(1))/2.)*DOR
0060   PHI1=((FI1(4)+FI(4))/2.)*DOR
0061   40 FORMAT(3F12.6)
0062   TYPE 40=THAI
0063   KOUNT=2
0064   COSTH=COS(THAI/DOR)
0065   SINTH=SIN(THAI/DOR)
0066   COSPHI=COS(PHI1/DOR)
0067   SINFPhi=SIN(PHI1/DOR)
C.... START CORRECTION LOOP
0068   50 CONTINUE
0069   KOUNT=KOUNT+1
C.... IF KOUNT IS GREATER THAN IEND GO TO 1000
0070   IF(KOUNT.GT.IEND)GO TO 1000
C.... TRANSFER NEXT TIME POINT
0072   DO 37 1
0073     FIP1(I)=DATA(KOUNT+I)
0074  37 CONTINUE
C.... CORRECT FOR SIGN ERRORS IN CALIBRATION
0075   FIP1( 4)=-FIP1( 4)
0076   FIP1( 6)=-FIP1( 6)
0077   FIP1( 8)=-FIP1( 8)
0078   FIP1( 9)=-FIP1( 9)
0079   FIP1(11)=-FIP1(11)
C.... CORRECT FOR MISALIGNMENT ANGLE
0080   FIP1(1)=FIP1(1)-THAI/DOR
0081   PI= FIP1(7)*COSTH+FIP1(9)*SINTH
0082   RI=FIP1(7)*SINTH+FIP1(9)*COSTH
0083   FIP1(7)=PI
0084   FIP1(9)=RI
C.... GET BACK TO ORIGINAL SIGNAL
0085   FI(3)=FI(3)+1.00
C.... COMPUTE PDOT, QDOT, AND RDOT
0086   PDET=(FIP1(7)-FI1(7))/0.2
0087   QDET=(FIP1(2)-FI1(2))/0.2
0088   RDET=(FIP1(9)-FI1(9))/0.2
C.... CORRECT ACCELERATIONS FOR OFFSET
0089   P =FI(7)
0090   Q =FI(2)
0091   R =FI(9)
0092   AXP1 =FI(4)*COSTH
0093   AXP2 =FI(3)*SINTH
0094   AXP3 = (R*R+Q*Q)*XBAR/G
0095   AXP4 = (P*Q-RDET)*YBAR/G
0096   AXP5 = (P*Q+RDET)*ZBAR/G
0097   AXP6 = AXP1*AXP2*AXP3*AXP4*AXP5
0098   AX = AXP6
0099   AY =FI(8) -(P*Q+RDET)*XBAR/G+(P*R+Q*R)*YBAR/G
0100   AZ =FI(4)*SINTH-FI(3)*COSTH-(P*R-QDET)*XBAR/G
0101   - (Q*R+PDET)*YBAR/G+(P*P+Q*Q)*ZBAR/G
0102   FI(3) =AZ
0103   FI(4) = AX
0104   FI(5) = AY
130
C.... CORRECT FOR ACCELEROMETER BIAS
0104   FI(3) = FI(3) - COS(THETA) * COS(THETA)
0105   FI(4) = FI(4) - SIN(THETA)
0106   FI(5) = FI(5) + COS(THETA) * SIN(THETA)
C.... WRITE VALUES ON OUTPUT
C.... TYPE OUT KOUNTER
0107   TYPE S KOUNT
0108   WRITE(2) (FI(I), I = 1, 11)
C.... BUCKET BRIGADE VALUES THRU TIME
0109   DO 100 I = 1, 11
0110     FIM(I) = FI(I)
0111     FI(I) = FIP(I)
0112    100 CONTINUE
0113   GO TO 50
0114   1000 CONTINUE
C.... TRANSFER LAST DATA POINT
0115   WRITE(2) (FI(I), I = 1, 11)
C.... CLOSE DATA FILE
0116   STOP
0117   END
A.7) MMLE BONES ROUTINES

This appendix describes the MMLE programs. The first program required is the one that sets up the input matrices, as well as defining for the MMLE program which parameters it is to estimate. The MMLE programs, as well as their output format is also presented.

A.7.1) MMLE SETUP

Description: The setup program is an interactive program which sets up the input data for the MMLE BONES routine. Non-dimensional derivatives, geometric, and inertia data are input and used to form the initial estimate to the MMLE program.

Program listing:

```
0001 PROGRAM SETUP
C.... THIS PROGRAM SETS UP THE DATA USED IN BONES MMLE.
C.... DIMENSIONAL DERIVATIVES ARE BUILT UP FROM NON-DIMENSIONAL
C.... INPUT DATA AND AIRPLANE GEOMETRIC DATA.
C.... DEFAULT VALUES (IF THEY EXIST) ARE SHOWN AFTER EACH QUESTION.
0002 DIMENSION A(5,4);B(5,4);AB(5,4);AP(8,4);BP(8,3)
0003 DIMENSION ZERO(4);BIAS(4);DI(7,7)
0004 DOUBLE PRECISION CASE,TTEMP
0005 BYTE BANNER(4,80)
0006 DATA VALUE,VALUE;A,AP;BIAS,BP;BP,0.0,0.0,20.0,20.0,32.0,24.0,/
0007 DATA D1,BIAS,ZERO/49.0,4.0,3.0,/
0008 DATA CASE,TTEMP/2,
0009 DATA BANNER/320,
C.... SET DEFAULT VALUES
0010 NN =200
0011 ITR =10
0012 MZ =7
0013 MPR =0
0014 HH =0.10
0015 EPS =0.0
0016 TIME =0.0
0017 ALPHA =0.0
0018 XLA =1.0
C.... UNIT 1 WILL BE THE FILE NUMBER OF THE FILE FOR
C.... THE DATA DISK WHICH IS ASSUMED ON DY1:
C.... OPEN UNIT 1
0019 2 FORMAT(9901)
0020 9 FORMAT(' ENTER A BANNER OF UP TO FOUR LINES.')
0021 TYPE 9
0022 1 FORMAT(' ENTER LINE: PI1')
0023 DO 3 I =1,4
0024 ACCEPT 29(BANNER(IPJ)PJN1,80)
0025 3 CONTINUE
0026 10 CONTINUE
0027 30 FORMAT(' ENTER 'LONG' OR 'LATR' FOR THE TYPE OF CASE/,
0028 ' TO BE SET UP,')
0029 31 CONTINUE
0030 32 CONTINUE
0031 33 CONTINUE
0032 IF(CASE.EQ.'LONG')OPEN (UNIT=I,NAM= 'DY1:MML10.DAT',TYPE='NEW',
0033 RECORDSIZE=96,INITIALSIZE=50,DISPOSE='SAVE')
0034 IF(CASE.EQ.'LATR')OPEN (UNIT=I,NAM= 'DY1:MML1D.DAT',TYPE='NEW',
0035 RECORDSIZE=96,INITIALSIZE=50,DISPOSE='SAVE')
0036 IF(CASE.NE.'LONG')AND.(CASE.NE.'LATR')GO TO 10
```
C.... ERROR TRAP IF RESPONSE IS NOT 'LONG' OR 'LATR'
C.... BASIC DATA FOR EITHER LONITUDINAL OR LATERAL-DIRECTIONAL CASE

0030 FORMAT(' ENTER THE NUMBER OF DATA POINTS TO BE PROCESSED.',/,'(DEFAULT IS 200)')
0039 TYPE 50
0040 60 FORMAT(2F10.0)
0041 61 FORMAT(115)
0042 ACCEPT 61*IVALUE
0043 IF(IVALUE.GT.0)NH=IVALUE
0045 IVALUE=0
0046 70 FORMAT(' ENTER THE NUMBER OF ITERATIONS TO BE PERFORMED.',/,'(DEFAULT IS 10)')
0047 TYPE 70
0048 ACCEPT 61*IVALUE
0049 IF(IVALUE.GT.0)ITR=IVALUE
0051 IVALUE=0
0052 80 FORMAT(' ENTER THE NUMBER OF OBSERVATIONS.',/,'(DEFAULT IS 7)')
0053 TYPE 80
0054 ACCEPT 61*IVALU9
0055 IF(IVALUE.GT.0)MZ=IVALUE
0057 IVALUE=0
0058 90 FORMAT(' ENTER THE CONTROL NUMBER FOR THE APRIORI OPTION.',/,'(DEFAULT IS 01 WHICH IS NO APRIORI VALUES)')
0059 TYPE 90
0060 ACCEPT 61*IVALUE
0061 MAPR=0
0062 IF(IVALUE.NE.0)MAPR*IVALUE
0064 IVALUE=0
0065 100 FORMAT(' ENTER THE DELTA TIME INCREMENT.',/,'(DEFAULT IS 0.10)')
0066 TYPE 100
0067 ACCEPT 60*VALUE
0068 IF(VALUE.GT.0)NH=VALUE
0070 VALUE=0.
0071 110 FORMAT(' ENTER THE VALUE FOR EPS.',/,'(DEFAULT IS 0.0)')
0072 TYPE 110
0073 ACCEPT 60*VALUE
0074 IF(VALUE.GT.0)EPS=VALUE
0076 VALUE=0.
0077 120 FORMAT(' ENTER THE VALUE FOR TIME.',/,'(DEFAULT IS 0.0)')
0078 TYPE 120
0079 ACCEPT 60*VALUE
0080 IF(VALUE.GT.0)TIME=VALUE
0082 VALUE=0.
0083 130 FORMAT(' ENTER THE VALUE FOR ALPHA.',/,'(DEFAULT IS 0.0)')
0084 TYPE 130
0085 ACCEPT 60*VALUE
0086 IF(VALUE.GT.0)ALPHA=VALUE
0088 VALUE=0.
0089 140 FORMAT(' ENTER THE VALUE FOR XLA.',/,'(DEFAULT IS 1.0)')
0090 TYPE 140
0091 ACCEPT 60*VALUE
0092 IF(VALUE.GT.0)XLA=VALUE
0094 DO 141 I=1,4
0095 WRITE(1,2)(BANNER(I,J),J=1,80)
0096 141 CONTINUE
0097 WRITE(1,150)NH,ITR,HI,MAPR
0098 WRITE(1,160)NH, EPS, TIME, ALPHA, XLA
0099 150 FORMAT(7I10)
0100 160 FORMAT(8F10.4)

C.... ENTER THE MASS AND GEOMETRIC DATA

0101 170 FORMAT(' ENTER THE AIRPLANE WEIGHT. (IN LBS)')
0102 TYPE 170
0103 ACCEPT 60*WEIGHT
0104 AMSS =WEIGHT/32.174
0105 180 FORMAT(' ENTER THE AIRPLANE WING AREA. (IN FT**2)')
0106 TYPE 180
0107 ACCEPT 60*W
0108 190 FORMAT(' ENTER THE AIRPLANE CBAR. (IN FT)')
0109 TYPE 190
0110 ACCEPT 60*CBAR
GILL 195 FORMAT(' ENTER THE WING SPAN. (IN FT)')
0112 TYPE 195
0113 ACCEPT $0PASS
0114 200 FORMAT(' ENTER THE ALTITUDE OF THE FLIGHT/Run. (IN FT)')
0115 TYPE 200
0116 ACCEPT $0H

C.... COMPUTE ATMOSPHERIC CONDITIONS FROM APPROXIMATE RELATIONS
0117 TA = 518.7 - 960.0030
0119 IF(TA.LT.390.)TA = 390.
0120 PA = 2116.229*(1.0 - 0.00000687843H)**5.2532
0121 RHO = PA/(1716.56*TA)
0122 AVEL = 49.02*SQRT(TA)

C.... ENTER THE STEADY-STATE FLIGHT CONDITIONS
0123 210 FORMAT(' ENTER THE STEADY STATE VELOCITY. (IN FT/SEC)')
0124 TYPE 210
0125 ACCEPT $0UL
0126 CL1 = 2.8*WEIGHT/(RHO*U1*U1)

C.... ASSUME L/D OF 10.
0127 CD1 = CL1/10.
C.... ASSUME CXT1 = CD1
0128 CXT1 = CD1

C.... ENTER THE STEADY STATEtheta. (IN DEG)'/,
0129 220 FORMAT(' ENTER THE STEADY STATE THETA. (IN DEG)'/,
0130   ' (DEFAULT IS 0.0)')
0131 DDR = 57.29578
0132 TYPE 220
0133 ACCEPT $0THA
0134 THA = THA/DDR

C.... ENTER THE STEADY州E ANGLE OF ATTACK. (IN DEG)'/,
0135 230 FORMAT(' ENTER THE STEADY STATE ANGLE OF ATTACK. (IN DEG)'/,
0136   ' (DEFAULT IS theta)')
0137 TYPE 230
0138 VALUE = 0.
0139 ACCEPT $0VALUE
0140 IF(VALUE.NE.0.)ALP = VALUE
0141 ALP = THA/DDR
0142 SINALP = SIN(ALP)
0143 COSALP = COS(ALP)
0144 SINTHA = SIN(THA)
0145 COSTHA = COS(THA)
0146 SINPHI = SIN(PHI)
0147 COSPHI = COS(PHI)
0148 TANTHA = SINTHA/COSTHA

C.... ENTER THE INERTIAL DATA
0149 260 FORMAT(' ENTER IYYB. (IN SLUG*FT*SEC)')
0150 TYPE 260
0151 ACCEPT $0AIY
0152 270 FORMAT(' ENTER IXXB. (IN SLUG*FT*SEC)')
0153 TYPE 270
0154 ACCEPT $0AIX
0155 280 FORMAT(' ENTER IZZB. (IN SLUG*FT*SEC)')
0156 TYPE 280
0157 ACCEPT $0AIZ

C.... SPLIT FOR CASES
0158 IF(CASE.EQ.'LONG')GO TO 300
0159 IF(CASE.EQ.'LMT')GO TO 300
0160 STOP

C.... LONGITUDINAL CASE
0161 300 CONTINUE
0162 310 FORMAT(' ENTER CDU* 0 OR 1. ')
0163 TYPE 310
0164 IF(CASE.EQ.'CDU')GO TO 300
0165 STOP

C.... CONTINUE CASE
0166 311 FORMAT(' ( 1 IF THIS IS A VARIABLE! 0 OTHERWISE )')
0167 TYPE 311
0168 IF(CASE.EQ.'CDU')GO TO 300
0169 STOP

C.... ACCEPT CDU
0170 312 FORMAT(' ENTER CXTU. ') Type 312
0171 ACCEPT $0CDU
0172 ACCEPT $0CXTU
0173 330 FORMAT(' ENTER CDA* 0 OR 1. ')
0174 TYPE 330
0175 ACCEPT $0CDA

135
0179  340 FORMAT( ' ENTER COUP 0 OR 1. ' )
0180  TYPE 340
0181  ACCEPT 60*COUP*BB(2,1)
0182  350 FORMAT( ' ENTER CLU 0 OR 1. ' )
0183  TYPE 350
0184  ACCEPT 60*CLU*AA(3,2)
0185  360 FORMAT( ' ENTER CLA 0 OR 1. ' )
0186  TYPE 360
0187  ACCEPT 60*CLA*AA(3,3)
0188  370 FORMAT( ' ENTER CLDE 0 OR 1. ' )
0189  TYPE 370
0190  ACCEPT 60*CLDE*BB(3,1)
0191  380 FORMAT( ' ENTER CMAD. ' )
0192  TYPE 380
0193  ACCEPT 60*CMAD*AA(1,1)
0194  390 FORMAT( ' ENTER CMG 0 OR 1. ' )
0195  TYPE 390
0196  ACCEPT 60*CMG*AA(1,2)
0197  400 FORMAT( ' ENTER CMU 0 OR 1. ' )
0198  TYPE 400
0199  ACCEPT 60*CMU*BB(1,1)
0200  410 FORMAT( ' ENTER CMTU. ' )
0201  TYPE 410
0202  ACCEPT 60*CMTU
0203  420 FORMAT( ' ENTER CMA 0 OR 1. ' )
0204  TYPE 420
0205  ACCEPT 60*CMA*AA(1,3)
0206  430 FORMAT( ' ENTER CMTA. ' )
0207  TYPE 430
0208  ACCEPT 60*CMTA
0209  440 FORMAT( ' ENTER CMDE 0 OR 1. ' )
0210  TYPE 440
0211  ACCEPT 60*CMDE*BB(1,1)

C.... DEFINE DIMENSIONAL DERIVATIVES
0212  Q1  = RNOSUL1/U1/2.0
0213  XU  = Q1*SS*(CXTU+2,2*CXI=CDU-2.*CD1)/(AMSS*U1)
0214  XA  = Q1*SS*(CDA-CL1)/AMSS
0215  XDE = Q1*SS*CDDE/AMSS
0216  ZU  = Q1*SS*(CLU+2,2*CL1)/(AMSS*U1)U1
0217  ZA  = Q1*SS*(CLA+CD1)/(AMSSU1)
0218  ZDE = Q1*SS*CLDE/(AMSSU1)
0219  AMO = Q1*SS*CBAR*CBAR*(CMAD+CMO)/(2.*AIYU1)
0220  AMU = Q1*SS*CBAR*(CMU+CMTU)/(AIYU1)
0221  AMA = Q1*SS*CBAR*(CMA+CMTA)/AIY
0222  AMDE = Q1*SS*CBAR*CMDE/AIY

C.... DEFINE A MATRIX ELEMENTS
0223  A(1,1)=AMO
0224  A(1,2)=AMU
0225  A(1,3)=AMA
0226  A(1,4)=0.0
0227  A(2,1)=0.0
0228  A(2,2)=XU
0229  A(2,3)=XA
0230  A(2,4)=-COSTHA*32.174
0231  A(3,1)=1.0
0232  A(3,2)=ZU
0233  A(3,3)=ZA
0234  A(3,4)=-SINTHA*COSPHI*32.174/U1
0235  A(4,1)=COSPHI
0236  A(4,2)=0.0
0237  A(4,3)=0.0
0238  A(4,4)=0.0

C.... DEFINE B MATRIX ELEMENTS
0239  B(1,1)=AMDE
0240  B(1,2)=0.0
0241  B(1,3)=0.0
0242  B(2,1)=XDE
0243  B(2,2)=0.0
0244  B(2,3)=0.0
0245  B(3,1)=ZDE
0246  B(3,2)=0.0
0247  B(3,3)=0.0

136
C.... ALL ELEMENTS OF THE AA MATRIX ARE DEFINED
C.... DEFINE ADDITIONAL ELEMENTS OF THE BB MATRIX
0251  BB(1,3)=1.0
0252  BB(2,3)=1.0
0253  BB(3,3)=1.0
C.... DEFINE AP MATRIX (ASSUMED ORDER OF THE OBSERVATION VECTOR IS: 0 U ALPHA, THETA, QDD: AX-AXBIAS, AND AN-ANBIAS)
0255  DO 450 I=1,5
0256  DO 460 J=1,4
0257  AP(I,J)=1.0
0258  440 CONTINUE
0259  450 CONTINUE
0260  AP(6,7)=1.0/32.174
0261  AP(7,3)=-1/32.174
C.... DEFINE BP MATRIX (ASSUMED ORDER OF THE CONTROL VECTOR IS: 0 U ALPHA, THETA, QDD: AX-AXBIAS, AND AN-ANBIAS)
0262  DO 470 I=1,5
0263  DO 480 J=1,3
0264  BP(I,J)=1.0
0265  480 CONTINUE
0266  470 CONTINUE
0267  BP(6,1)=1.0/32.174
0268  BP(7,1)=-1/32.174
0269  490 CONTINUE
0270  490 CONTINUE
C.... SKIP LATERAL DIRECTIONAL INPUT CASE
0271  GO TO 700
C.... LATERAL DIRECTIONAL CASE
0272  500 CONTINUE
0273  510 FORMAT(‘ ENTER CLPr 0 OR 1.’)
0274  TYPE 510
0275  511 FORMAT(‘ ( 1 IF THIS VARIES, 0 OTHERWISE )’)
0276  TYPE 511
0277  ACCEPT 60*CLPrAA(lrl)
0278  520 FORMAT(‘ ENTER CLRr 0 OR 1.’)
0279  TYPE 520
0280  ACCEPT 60*CLRrAA(1r2)
0281  530 FORMAT(‘ ENTER CLBr 0 OR 1.’)
0282  TYPE 530
0283  ACCEPT 60*CLBrAA(2r3)
0284  540 FORMAT(‘ ENTER CLDr 0 OR 1.’)
0285  TYPE 540
0286  ACCEPT 60*CLDrAA(2r3)
0287  550 FORMAT(‘ ENTER CNPr 0 OR 1.’)
0288  TYPE 550
0289  ACCEPT 60*CNPrAA(2r3)
0290  560 FORMAT(‘ ENTER CNRr 0 OR 1.’)
0291  TYPE 560
0292  ACCEPT 60*CNRrAA(2r3)
0293  570 FORMAT(‘ ENTER CNBr 0 OR 1.’)
0294  TYPE 570
0295  ACCEPT 60*CNBrAA(2r3)
0296  580 FORMAT(‘ ENTER CNDr 0 OR 1.’)
0297  TYPE 580
0298  ACCEPT 60*CNDrAA(2r3)
0299  590 FORMAT(‘ ENTER CYPr 0 OR 1.’)
0300  TYPE 590
0301  ACCEPT 60*CYPrAA(2r3)
0302  600 FORMAT(‘ ENTER CYRr 0 OR 1.’)
0303  TYPE 600
0304  ACCEPT 60*CYRrAA(2r3)
0305  610 FORMAT(‘ ENTER CYBr 0 OR 1.’)
0306  TYPE 610
0307  ACCEPT 60*CYBrAA(2r3)
0308  620 FORMAT(‘ ENTER CYDr 0 OR 1.’)
0309  TYPE 620
0310  ACCEPT 60*CYDrAA(2r3)
0311  630 FORMAT(‘ ENTER CNPr 0 OR 1.’)
0312  TYPE 630
0313  ACCEPT 60*CNPrAA(2r3)
C.... DEFINE DIMENSIONAL DERIVATIVES

0314  G1 = RH0UIU1/2.0
0315  BL = G18888SPAN$SPANCLP/(2.8AIU1)
0316  BLR = G18888SPAN$SPANCLR/(2.8AIU1)
0317  BLB = G18888SPAN$CLB/AIX
0318  BNP = G18888SPAN$SPANCHP/(2.8AIU1)
0319  BBR = G18888SPAN$SPCHR/(2.8AIU1)
0320  BNB = G18888SPAN$CHB/AIZ
0321  YB = G18888SPAN$YB/(AM*8U1)
0322  LDA = G18888SPAN$CL/$AIX
0323  LDR = G18888SPAN$CLUDR/AIX
0324  BND = G18888SPAN$CND/AIZ
0325  CND = G18888SPAN$CNDR/AIZ
0326  YDA = G18888SPAN$YDA/(AM888U1)
0327  YDR = G18888SPAN$YDR/(AM888U1)

C.... DEFINE A MATRIX ELEMENTS

0328  A(1P1) = BLF
0329  A(1P2) = BLP
0330  A(1P3) = BLR
0331  A(1P4) = BLB
0332  A(2P1) = BNP
0333  A(2P2) = BNR
0334  A(2P3) = BNB
0335  A(2P4) = BNP
0336  A(3P1) = BINALP
0337  A(3P2) = YB
0338  A(3P3) = COSALP
0339  A(3P4) = 32.174*COSTH*KOSPHI/U1
0340  A(4P1) = 1.0
0341  A(4P2) = COSPHI*TANTH
0342  A(4P3) = 1.0
0343  A(4P4) = 1.0

C.... DEFINE B MATRIX ELEMENTS

0344  B(1P1) = BDA
0345  B(1P2) = BLD
0346  B(1P3) = BDR
0347  B(2P1) = BND
0348  B(2P2) = BNR
0349  B(2P3) = BNB
0350  B(3P1) = YDA
0351  B(3P2) = YDR
0352  B(3P3) = 1.0
0353  B(4P1) = 1.0
0354  B(4P2) = 1.0
0355  B(4P3) = 1.0

C.... ALL ELEMENTS OF THE AA MATRIX ARE DEFINED

C.... DEFINE ADDITIONAL ELEMENTS OF THE BB MATRIX

0356  BB(1P3) = 1.0
0357  BB(2P3) = 1.0
0358  BB(3P3) = 1.0
0359  BB(4P3) = 1.0

C.... DEFINE AP MATRIX (ASSUMED ORDER OF THE OBSERVATION VECTOR IS:
0360  P, R, BETA, PHI, PDOT, RDOT, AND AY-AYBIAS)
0361  DO 650 I=1,6
0362  DO 660 J=1,4
0363  AP(I,J)=1.0
0364  660 CONTINUE
0365  650 CONTINUE

C.... DEFINE AP MATRIX (ASSUMED ORDER OF THE CONTROL VECTOR IS:
0366  DA, DR, AND BIAS)
0367  DO 670 I=1,6
0368  DO 680 J=1,3
0369  BP(I,J)=1.0
0370  680 CONTINUE
0371  670 CONTINUE

C.... ECHO DATA BACK

138
0378 710 FORMAT(' AIRPLANE INPUT DATA',/):
  WING AREA (IN FT**2) = 'F12.4'/
  WEIGHT (IN LBS) = 'F12.4'/
  WING SPAN (IN FT) = 'F12.4'/
  CDAR (IN FT) = 'F12.4'/
  AIRSPEED (IN FT/SEC) = 'F12.4'/
  DENSITY (IN SLUG/FT**3) = 'F12.4'/
  ALPHAI (IN RAD) = 'F12.4'/
  THETAI (IN RAD) = 'F12.4'/
  PHI1 (IN RAD) = 'F12.4'/
  IYy (IN SLUG*FT**2) = 'F12.4'/
  IXX (IN SLUG*FT**2) = 'F12.4'/
  IZZ (IN SLUG*FT**2) = 'F12.4'/
  CL1 = 'F12.4'/
  CD1 = 'F12.4'/
  CXT1 = 'F12.4'/

0379 TYPE 710+S:WEIGHT+SPAN+CDAR+UI+RHO+ALP+THA+PHI1+THETAI,
  AI2+CL1+CD1+CXT1

C.... SPLIT FOR CASES
0380 IF(CASE.EQ.'LONG') 100 TO 750
0382 IF(CASE.EQ.'LATR') 100 TO 850
0384 STOP

C.... LONGITUDINAL CASE
C.... WRITE OUTPUT TO DISPLAY, GET WEIGHTING FACTORS, AND FINISH FILE
0385 750 CONTINUE
0386 760 FORMAT(' LONGITUDINAL DERIVATIVES',/):
  CDU = 'F8.4', XU = 'F8.4'/
  CXTU = 'F8.4', XA = 'F8.4'/
  CDA = 'F8.4', XDE = 'F8.4'/
  CDUE = 'F8.4', ZU = 'F8.4'/
  CLU = 'F8.4', ZA = 'F8.4'/
  CLDE = 'F8.4', HU = 'F8.4'/
  CMAD = 'F8.4', MQ = 'F8.4'/
  CMA = 'F8.4', MA = 'F8.4'/
  CMDE = 'F8.4', MDE = 'F8.4'/

0387 TYPE 760:CDU+XU+CXTU+CDA+XA+CDE+XDE+CLU+ZA+CLDE+HU+CMAD+MQ+CMA+MA+CMDE+MDE

C.... GET THE WEIGHTING MATRIX DIAGONAL VALUES
0388 770 FORMAT(' ENTER THE WEIGHTING FACTOR FOR Q'),
0389 TYPE 770
0390 ACCEPT 609D1(1.1)
0391 780 FORMAT(' ENTER THE WEIGHTING FACTOR FOR VELOCITY.'),
0392 TYPE 780
0393 ACCEPT 609D1(2.2)
0394 790 FORMAT(' ENTER THE WEIGHTING FACTOR FOR ALPHA.'),
0395 TYPE 790
0396 ACCEPT 609D1(3.3)
0397 800 FORMAT(' ENTER THE WEIGHTING FACTOR FOR THETA.'),
0398 TYPE 800
0399 ACCEPT 609D1(4.4)
0400 810 FORMAT(' ENTER THE WEIGHTING FACTOR FOR ODOT.'),
0401 TYPE 810
0402 ACCEPT 609D1(5.5)
0403 820 FORMAT(' ENTER THE WEIGHTING FACTOR FOR AX.'),
0404 TYPE 820
0405 ACCEPT 609D1(6.6)
0406 830 FORMAT(' ENTER THE WEIGHTING FACTOR FOR AN.'),
0407 TYPE 830
0408 ACCEPT 609D1(7.7)

C.... SKIP PAST LATERAL DIRECTIONAL CASE
0409 850 CONTINUE

C.... LATERAL DIRECTIONAL CASE
C.... WRITE OUTPUT TO DISPLAY, GET WEIGHTING FACTORS, AND FINISH FILE
0410 850 CONTINUE
0411 860 FORMAT(' LATERAL DIRECTIONAL DERIVATIVES',/):
  CYB = 'F8.4', YB = 'F8.4'/
  CYDA = 'F8.4', YDA = 'F8.4'/
  CYDR = 'F8.4', YDR = 'F8.4'/
  CLB = 'F8.4', LB = 'F8.4'/
  CLR = 'F8.4', LR = 'F8.4'/
  CLDA = 'F8.4', LDA = 'F8.4'
C... GET THE WEIGHTING MATRIX DIAGONAL VALUES

0413 870 FORMAT(' ENTER THE WEIGHTING FACTOR FOR P,' )
0414 TYPE 870
0415 ACCEPT 40 * D1(1:1)
0416 880 FORMAT(' ENTER THE WEIGHTING FACTOR FOR R,' )
0417 TYPE 880
0418 ACCEPT 40 * D1(2:2)
0419 890 FORMAT(' ENTER THE WEIGHTING FACTOR FOR BETA,' )
0420 TYPE 890
0421 ACCEPT 40 * D1(3:3)
0422 900 FORMAT(' ENTER THE WEIGHTING FACTOR FOR PHI,')
0423 TYPE 900
0424 ACCEPT 40 * D1(4:4)
0425 910 FORMAT(' ENTER THE WEIGHTING FACTOR FOR PDOT,' )
0426 TYPE 910
0427 ACCEPT 40 * D1(5:5)
0428 920 FORMAT(' ENTER THE WEIGHTING FACTOR FOR RDOT,' )
0429 TYPE 920
0430 ACCEPT 40 * D1(6:6)
0431 930 FORMAT(' ENTER THE WEIGHTING FACTOR FOR AY,' )
0432 TYPE 930
0433 ACCEPT 40 * D1(7:7)
0434 950 CONTINUE

C... WRITE MATRICES TO FILE

0435 960 FORMAT(2110)
0436 1160 FORMAT(7F12.6)
0437 WRITE(1,960)
0438 DO 970 I=1,4
0439 WRITE(1,1160)(A(I,J),J=1,4)
0440 970 CONTINUE
0441 WRITE(1,960)
0442 DO 980 I=1,4
0443 WRITE(1,1160)(B(I,J),J=1,4)
0444 980 CONTINUE
0445 WRITE(1,960)
0446 DO 990 I=1,4
0447 WRITE(1,1160)(AA(I,J),J=1,4)
0448 990 CONTINUE
0449 WRITE(1,960)
0450 DO 1000 I=1,4
0451 WRITE(1,1160)(BB(I,J),J=1,4)
0452 1000 CONTINUE
0453 WRITE(1,960)
0454 DO 1010 I=1,4
0455 WRITE(1,1160)(AP(I,J),J=1,4)
0456 1010 CONTINUE
0457 WRITE(1,960)
0458 DO 1020 I=1,4
0459 WRITE(1,1160)(BP(I,J),J=1,4)
0460 1020 CONTINUE
0461 WRITE(1,960)
0462 DO 1030 I=1,4
0463 WRITE(1,1160)(D1(I,J),J=1,7)
0464 1030 CONTINUE
0465 WRITE(1,960)
0466 WRITE(1,1160)(ZERO(I),I=1,4)
0467 STOP
0468 END
A.7.2) MAIN MMLE PROGRAMS

Description: The main program of the MMLE BONES routines acts as a controller in calling the subroutines as needed. Initially, it reads the input data for the starting conditions of the test case. If all states are measured a least squares process is used to compute the initial estimate of the derivatives. If the states are not completely measured this feature must be skipped over or errors in the solution of updates to the coefficients will result.

Flowchart:
Program listing:

**0001** PROGRAM MAIN

```
C ::!

C MAIN PROGRAM OF THE MAXIMUM LIKELIHOOD ESTIMATOR
C TECHNIQUE, (MMLE). THIS PROGRAM IS DERIVED FROM
C THE "BONES" PROGRAM THAT WAS ORIGINALLY DEVELOPED
C BY NASA. THE FOLLOWING SUBROUTINES ARE REQUIRED
C FOR THE OPERATION OF THIS PROGRAM :
C GIRL, EAT, CRAMER, SPIT!, REDUCE, MUL!, OUTPUT
C ADD, MAKE, ZOT, LOAD, LOAD1, SPIT, SOLVE, AND
C DIAGIN.
C THE OUTPUT OF THE PROGRAM IS TWO FILES THAT
C CONTAINS THE MATRICES CA AND CB FOR EACH
C ITERATION AND THE ESTIMATED TIME RESPONSES
C RESPECTIVELY. SEE THE SPECIFIC INSTRUCTIONS
C CONCERNING THE INPUT AND OPERATION OF THIS MMLE
C PROGRAM.
C THIS MODIFIED "BONES" PROGRAM WAS WRITTEN BY:
C ALEX KOTSABASIS
C DATE 22-NOV-80
```

**0002** COMMON MAX,MAX,MAX,MAT,Z,U,D2,E1,APHI,DUM,PHI1,DI1,AB,AA,BB,
```
2 BJI,XJI,BUM,FB,XT1,ZERO,DS4,DD4,EX,XTX,CCC,BIAS,
3 IZE:IBIAS,IC,XLA,APR,NAPR,XTA,JKMM,XT5,AP,DP
```
**0003** COMMON HH+ENN+MX,MUX,RB,IA,JKN,EP,SP,DP,1,MUX1,
```
2 TRACE,K,II+TIME,DD5,MPOL,JMM,LM,TT,JKMM1,
3 ALPHA,NN1,MZ,J,KJ+BD,LL+LN,NN,ME,MUX,EN,PP,AA,KN,FAC,
4 ITR+X1,X2,X3,HM1
```
**0004** COMMON/MATAB/ALX,ALX,BLX,BLX,ERX
**0005** COMMON/CONST/ KABC,KCDF
**0006** COMMON/TRNSFR/ XL(300,7)
**0007** DIMENSION ALX(20,10),BLX(20,10),ERX(10,10)
**0008** DIMENSION XTS(25),APR(25)
**0009** DIMENSION AP(8,4), BP(9,3), XT4(4)
**0010** DIMENSION Z(7,3),U(3,3),B2(7),DD4(5,4),BIAS(5),APHI(5,4),
```
2 XT1(7),PHI1(5,4),D1(8,7),A1(5,4),B(5,4),AA(5,4),
3 B1(5,4),BJI(25,4),XJI(25,7),SUM(25,25),PB(25),
4 DUM(25,4),XT2(7),ZERO(5,DS4(5,4),XT3(7)
```
**0011** BYTE IMANE(15),COMN(80,4)
**0012** COMMON/ANSWER/ KBUGO

**0013** BYTE LOG,DIR,ANS
**0014** DATA LOG,DIR,/'L','D'/

**0015** TYPE 3511
**0016** 3511 FORMAT(/// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// /// }
0020 3513 FORMAT(/:///;/10X,'INDICATE TYPE OF RUN1';//
 1 10X,'IF LONITUDINAL TYPE 'L';',','/
 2 10X,'IF LATERAL-DIRECTIONAL TYPE 'D';','/
 3 10X,'SELECT RUN1')
C 0021  READ(5,3514) ANS
0022 3514 FORMAT(A1)
C C 0023 5 CONTINUE
0024 63 CONTINUE
0025 NI = 25
0026 XXXX = 1.0
0027 MAX = 5
0028 MA = 4
0029 NZ = 7
C C ATTACH DATA FILE CONTAINING MATRICES AND INT. CONSTANTS.
0030 TYPE 119
0031 119 FORMAT(/:///;/10X,'ENTER DATA FILE NAME WITH INITIAL CONDITIONS;',/; 1 10X,'AND MATRICES A, B, AA, AP, ETC.')
0032 128 FORMAT(14A1)
0033 ACCEPT 12Gv(INAME(IADC)vlA9C
0034 INAME(15)
0035 OPEN(UNIT=2rNAME=INAMErTYPE=OLD'ACCESS=SEQUENTIAL;',/ 1 READONLY;FORM='FORMATTED';RECORDSIZE=132)
C CLEAR OUT OLD FILE NAME
0036  DO 127PIASC
0037  INAME(IABC)=' '
127 CONTINUE
C 0038 1700 FORMAT(10E12.4)
0039 3777 FORMAT(12X,7I10)
0040 3700 FORMAT(12X,8F10.4)
0041 700 FORMAT(8F10.4)
0042 777 FORMAT(7I10)
0043 1010 FORMAT(10E12.4)
0044 1011 FORMAT(13x9E12.4)
0045 1012 FORMAT(12E12.4)
0046 1048 FACT = 1.0
0047 1049 BLANC = 0.0
C READ COMMENTS FROM INPUT FILE (4 LINES OF 80 CHARACTERS)
0048  READ(2,1301) ((COMNT(IrJ)PI
0049 1301 FORMAT(80A1)
C READ STATMENTS
0050  READ(2,777) HN,ITR,MZ,MAPR
0051  READ(2,700) HH,EPS,TIME,ALPHA,XLA
C LOAD MATRICES
0052  ENN = NN
0053  KCDF = NN
0054  KABC = ITR
0055  CALL LOAD(4,A,B,AA,BB)
0056  MAX = 8
0057  MA = 7
0058  CALL LOAD(3,AP,BP,D1,D1)
C C C
0059  MAX = 5
0060  MA = 4
0061  NNH1 = NN-1
0062  MU = B(MAX+2)+.01
0063  MX = A(MAX+2)+.01
C READ IN ZEROS' AND JIAEB.
C 0064 READ (2,700) (ZERO(IrI),I=1,MX),(BIAS(IA),IA=1+MX)
C C C
0065 C
0066 CLOSE(UNIT=2)
C
0048  MXP1 = MX + 1
0049  MZM1 = MZ - 1
0050  MUX = MU + MX
0051  HUME = HUMX
0052  MUM1 = HUME + 1
0053  YY = 0.0
0054  XX = 1.0
C ADD BIASES AND ZEROS'
0055  DO 49 I=1,MX
0056       XT4(I) = 0.0
0057  DO 3 I=1,MX
0058       XX = XX + ZERO(I) + BIAS(I)
0059  XX = XX
0060  DO 48 J=1,MU
0061       YY = YY + AA(I,J) + BB(I,J)
0062  48  XX = XX + AA(I,J) + BB(I,J)
C
C
0063  49  YY = YY + AA(I,MX)
0064  49  XX = XX + AA(I,MX)
C
0065  JKM = YY + .01
0066  JKM = XX + .01
0067  JKM1 = JKM - 1
0068  SUM(NI+1) = JKM
0069  SUM(NI+2) = JKM
0070  MAX = NI
0071  MA = NI
C
C INITIALIZE MATRICES TO ZERO
C
0072  CALL ZOT(SUM)
C SELECT APRIORI OPTION THRU MAPR.
C
0073  IF(MAPR) 176,178,177
C
C READ IN APRIORI MATRIX
0074  177  DO 261 IB=1,JKM
0075       DO 663 IA=1,JKM
0076       SUM(IDPIA) = 0.0
0077  663  CONTINUE
0078  261  SUM(IS.IA) = SUM(IDPIA)
0079  261  APR(ID) = SUM(ISPID)
0080  261  CONTINUE
0081  176  CONTINUE
C
0082  DO 664 IA=1,JKMM1
0083       APR(IA) = 0.0
0084  664  CONTINUE
0085  DO 263 IA=1,JKM
0086       APR(IA) = APR(IA)*FACT
0087  263  CONTINUE
C
C ENTER NAME OF DATA FILE WITH MEASURED FLIGHT TEST DATA
C
0088  TYPE 139
0089  139  FORMAT(//,10X,'ENTER FILE NAME CONTAINING THE MEASURED DATA')
C
C ATTACH STAMMENT FOR FILE CONTAINING MEASURED DATA
C
0090  ACCEPT 128,INAME(IA9C),IABC=1,14)
0091  OPEN(UNIT=4,NAME=INAME,TYPE='OLD',ACCESS='SEQUENTIAL',
0092   FORM='UNFORMATTED',READONLY)
C
C REWIND TAPE
0093  REWIND 4

144
READ IN DATA AND PRINT OUT INITIAL CONDITIONS

BIASES AND ZERO'S

C

0113 1302 FORMAT('——————',1'r)
0114 1303 FORMAT(24X',',..',.,',..',..',..',..',..',..',..',..',..',..',..',..',..',..'/)
0115 1304 FORMAT(8X,'NUMBER OF DATA POINTS : ',I3',1'r)
0116 1325 FORMAT(8X,'DIAGONAL ELEMENTS OF THE WEIGHTING MATRIX Dill',/)
0117 1326 FORMAT(/10X,'ESTIMATES OF THE CA] AND CS] MATRICES',/)
0118 1305 FORMAT(10X,'INITIAL INPUT MATRICES CA] AND CS] ',/)
0119 1306 FORMAT(/10X,'STABILITY MATRIX CA] '/)
0120 1307 FORMAT(/10X,'CONTROL MATRIX CS] '/)
0121 1309 FORMAT(10X,'ITERATION',I3', 'WAS COMPLETED',/)

PRINT OUT INPUT DATA

0122
0123 DO 1407 J=1,4
0124 PRINT 1421,(COMNT(IPJ),I=1,80)
0125 1421 FORMAT(10X,80A1)
0126 CONTINUE
0127 PRINT 1302

PRINT 1303
0128 PRINT 1304, NN, ITR, HH, TIME, XLA, NZ
0129 PRINT 1302
0130 PRINT 703
0131 703 FORMAT(/10X,'ZEROS AND BIASES',/)
0132 PRINT 1700, (ZERO(IA),IA=1,MAX)
0133 PRINT 1700, (BIAS(IA),IA=1,MAX)
0134 PRINT 1325, (DI(IBC), IBC=1,7)
0135 PRINT 1302

C SET MAX AND MA TO CA] AND CS] DIMENSIONS

0137 MAX=5
0138 MA=4
0139 PRINT 1305
0140 PRINT 1306
0141 CALL SPIT1(A,AA,1)
0142 PRINT 1307
0143 CALL SPIT1(B,AB,1)

C

STARTING ITERATION LOOP

C

0144 TT = TIME - HH
0145 DO 1 LH=1,NH
0146 TT = TT + HH
0147 U(MU01) = 1.0
0148 : CONTINUE
0149 CONTINUE
0150 XTS(IA) = 0.0
0151 272 PBI(A) = 0.0
0152 IZE = 1
0153 DO 276 IA=1,MAX
0154 IF(ZERO(IA))277P276
0155 277 IZE = IZE + 1
0156 276 CONTINUE
MAIN LOOP FOR INT NUMBER OF ITERATIONS

DO 12 LL = 1, ITR

REWIND TAPE FOR EVERY ITERATION STEP

REWIND 4

TYPE 345, LL

FORMAT(//' NUMBER OF ITERATION PRESENTLY COMPUTED ', //)

MAX = 5

MAT = 5

MAX = 4

CALL SPECIAL MATRIX OUTPUT ROUTINE

IF(LL.EQ.1) GO TO 1308

PRINT 1326

PRINT 1306

CALL SPIT1(A, AA, LL)

PRINT 1307

CALL SPIT1(B, BB, LL)

CONTINUE

CALL SPECIAL MATRIX OUTPUT ROUTINE

IF(LL.EQ.1) GO TO 1308

PRINT 1326

PRINT 1306

CALL SPIT1(A, AA, LL)

PRINT 1307

CALL SPIT1(B, BB, LL)

CONTINUE

MAX = 5

MAT = 5

MAX = 4

CALL EAT (A, HH, PH1, APHI, D54, DD4)

U(3, 1) = 1.0

U(3, 2) = 1.0

U(3, 3) = 1.0

XJI(NI, 1) = JKM

XJI(NI, 2) = MX

BJJ(NI, 1) = JKM

BJJ(NI, 2) = MX

SUM(NI, 1) = JKM

SUM(NI, 2) = JKM

MA = NI

INITIALIZE AND READ DATA FROM TAPE

DO 778 IJK = 1, JKM

DO 778 JKL = 1, IJK

SUM(1, JKL) = 0.0

MAX = NI

CALL ZOT(XJI)

READ IN THE FIRST TWO SERIES OF MEASURED DATA

FROM THE DATA TAPE.

XT1(3) = 0.0

XT2(3) = 0.0

XT1(5) = 0.0

XT2(5) = 0.0

XT1(6) = 0.0

XT2(6) = 0.0
C KBUUG = 1
0203 C IF(AMS.EQ.000) GO TO 2012
0205 C READ (4) DXY,DXY,DXY,DXY,DXY,XT1(4),XT1(1),XT1(7),XT1(2),1
0206 C READ (4) DXY,DXY,DXY,DXY,DXY,XT2(4),XT2(1),XT2(7),XT2(2),1
0207 C GO TO 2013
0208 2012 CONTINUE
0209 XT1(2) = 0.0
0210 XT2(2) = 0.0
0211 XT1(3) = 0.0
0212 XT2(3) = 0.0
0213 XT1(5) = 0.0
0214 XT2(5) = 0.0
0215 U(2,1) = 0.0
0216 U(2,2) = 0.0
0217 C KBUUG = 0.0
0218 C READ (4) XT1(4),XT1(1),XT1(7),XT1(6),U(1,1),DXY,DXY,DXY,1
0219 C READ (4) XT2(4),XT2(1),XT2(7),XT2(6),U(1,2),DXY,DXY,DXY,1
0220 2013 CONTINUE
0221 IC = 0.0
0222 DO 51 I = 1, MX
0223 51 XJI(JKM+I) = XT2(I)
0224 IF(LL-1) 64, 65, 64
0225 64 DO 66 IA = 1, MX
0226 IF(ZERO(IA)) 67, 66, 67
0227 67 IC = IC + 1
0228 XT3(IA) = XT3(IA) + PB(JKM+IC)
0229 XT1(IA) = XT1(IA) + XT3(IA)
0230 XJI(JKM+IA) = XJI(JKM+IA) + XT3(IA)
0231 XT2(IA) = XJI(JKM+IA)
0232 66 CONTINUE
0233 IC = 0.0
C ADD BIASES
0234 DO 166 IA = 1, MX
0235 IF(BIAS(IA)) 167, 166, 167
0236 167 IC = IC + 1
0237 XT4(IA) = XT4(IA) + PB(JKM+IC)
0238 XT1(IA) = XT1(IA) - XT4(IA)
0239 XT2(IA) = XT2(IA) - XT4(IA)
0240 XJI(JKM+IA) = XT2(IA)
0241 166 CONTINUE
0242 65 CONTINUE
C MAIN MMLE LOOP
0243 DO 260 IA = 1, JKMM
0244 XT5(IA) = XT5(IA) + PB(IA)
0245 260 CONTINUE
0246 DO 13 IA = 1, MZ
0247 13 CONTINUE
0248 Z(IA+1) = XT1(IA)
0249 Z(IA+2) = XT2(IA)
0250 13 CONTINUE
0251 IC = 0.0
C ZERO SPLIT
0252 DO 62 I = 1, MX
IF(ZERO(I)) IC = IC+1
XJI(JKM-IZE + IC, I) = 1.0
CONTINUE
CALL GSRF
MAX = HI
HA = HI

C OUTPUT OF ITERATION LOOP
--------------------------------------------------------
DO 325 IA=1,JKM
 SUM(IA, IA) = SUM(IA, IA)*XLA
CALL SPIT(SUM)
SUM(NI, 1) = JKM-1
SUM(NI, 2) = JKM-1
PRINT 1309, LL
PRINT 1302
IF(LL-ITR) 269, 240, 260
SUM(NI,2) = JKM-1
PRINT 1302
STOP

C--------------------------------------------------------
CONTINUE
CALL SOLVE(SUM, PB)
NB = SUM(NI, 1) + 0.01
IJ = 0.0
DO 18 I=1, MX
DO 21 J=1, NU
IF(B(I, J)) 22, 21, 22
IJ = IJ + 1
B(I, J) = B(I, J) + PB(IJ)
CONTINUE
DO 18 J=1, MX
IF(AA(I, J)) 19, 18, 19
IJ = IJ + 1
A(I, J) = A(I, J) + PB(IJ)
CONTINUE
DO 12 CONTINUE
GO TO 83
RETURN
END
Subroutine GIRL

Description: Subroutine GIRL performs the parameter identification.

Important variables:

SUM Contains the second gradient in lower triangular and diagonal locations and off-diagonal a priori weighting in upper triangular. Diagonal a priori weightings are stored in APR. The first gradient appears as an extra column in SUM (the JKH column)

$X_{JI} = \nabla (z_i - y_i)$

$\Phi_1 = e^A\Delta t$

$\Delta t = \int e^At dt$

$Z, U$ measured values of observations and controls

$XT_1, XT_2$ computed values for observations

$XT_3$ variable initial conditions on states

$XT_4$ variable bias on the observations other than states

$XT_5$ difference between estimated coefficients and the a priori values

$PB$ solution vector for the change in the estimates of the coefficients

$MX$ number of states

$MZ$ number of observations

Subroutine listing;
C TIME LOOP
0010  TT = TIME + HM
0011  DO 41 I=2,NNH1
0020   TT = TT + HM
0021  DO 28 JK=1,JKM
0022  DO 28 J=KX1,HZ
0023  28 XJI(JK,J) = 0.0
0024  DO 170 IA = 1,MX
0025  170 XJI(JKM,IA) = XT2(IA)
C
C READ MEASURED DATA FROM DATA FILE

C
C
0026  IF(KBUUG.EQ.0.0) GO TO 3071
C
0028  READ(4) DXY,DXY,DXY,DXY,DXY,Z(4,3),Z(1,3),Z(7,3),Z(2,3),
   1 U(1,3),U(2,3)
0029  Z(3,3) = 0.0
0030  Z(5,3) = 0.0
0031  Z(6,3) = 0.0
C
0032  GO TO 3011
0033  3071 CONTINUE
C
C LONGITUDINAL
0034  READ(4) Z(4,3),Z(1,3),Z(7,3),Z(6,3),U(1,3),DXY,DXY,DXY,DXY,
   1 DXY,DXY
C
0035  Z(2,3) = 0.0
0036  Z(5,3) = 0.0
0037  Z(3,3) = 0.0
0038  U(2,3) = 0.0
C
0039  3011 CONTINUE
C
C
0040  DO 171 IA=1,MX
0041  171 Z(IAM,3) = Z(IAM,3)-XT4(IA)
0042  MAX = NI
0043  MA = 4
0044  CALL ZOT(BJI)
0045  JK = 0
0046  DO 44 J=1,MX
0047  44 K=1,NU
0048  BJI(JKM,J) = BJI(JKM,J)+B(J,K)##(U(K,3)+U(K,2))##0.5
0049  IF(BB(J,K))45,43,45
0050  45 JK = JK +1
0051  XJI(JK,J+MX) = U(K,2)##BP(J+MX,K)
0052  BJI(JK,J) = 0.5##(U(K,2)+U(K,1))
0053  43 CONTINUE
0054  DO 44 K = 1,MX
0055  IF(MA(J,K))46,44,46
0056  46 JK = JK +1
0057  IF(LL-1)4,4,4
0058  4 CONTINUE
0059  BJI(JK,J) = 0.5##(Z(K,2)+Z(K,1))
0060  XJI(JK,J+MX) = Z(K,2)##AP(J+MX,K)
0061  GO TO 44
0062  4 CONTINUE
0063  BJI(JK,J) = (XT2(K) + XTI(K))##0.5
0064  XJI(JK,J+MX) = XT2(K)##AP(J+MX,K)
0065  44 CONTINUE
0066  MAX = NI
0067  MA = 4
0068  NAM = 4
0069  HAT = 5
0070  XJI(N1,2) = M.
0071  CALL MULT(XJI,PHI1,XJI,DUM)
0072  CALL MULT(BJI,APHI,DUM,DUM)
0073  CALL ADD(1.0,DUM,1.0,XJI,XJI)

150
XJI(NM, 2) = P/Z

IBIAS = 0.0
DO 142 IA = 1, MX
IF(IBIAS(IA))143, 142, 163

IBIAS = IBIAS + 1
DO 175 IB = 1, MZ
XJI(JKMM+IBIAS, IB) = 0.0
XJI(JKMM+IBIAS, IA) = 1.0
CONTINUE

JKMM = JKM - 1
DO 7 JK = 1, JKM
DO 7 L = MXP1, MZ
DO 7 K = 1, MZ
XJI(JK, L) = XJI(JK, L) + A(L-MX, K)*XJI(JK, K)*AP(L, K)
CONTINUE
DO 9 L = MXP1, MZ
XJI(JKM+L) = XJI(JKM+L) + B(L-MX, K)*U(K, 3)*BP(L, K)
CONTINUE
DO 8 K = 1, MZ
XJI(JKM+L) = XJI(JKM+L) + A(L-MX, K)*XJI(JKMPK)*AP(L, K)
CONTINUE
DO 3 J = 1, MZ
XT2(J) = XJI(JKMPJ)
XJI(JKMPJ) = Z(JP3) - XT2(J)
CONTINUE
DO 27 K = 1, MZ
D2(K) = D2(K) + XJI(JKMPK)**2
CONTINUE
MAX = NI
MA = 7
CONTINUE
PRINT OUT TIME HISTORIES
TYPE 606. (XT2(IA)) IA = 1, 7, TT
IF(LL.LT.ITR) GO TO 80
DO 1013 IK = 1, 7
XL(IK) = XT2(IK)
CONTINUE
DO 91 I4 = JPKM
DO 92 K = 1, MZ
SUM(149J) = SUM(I4PKJ) + XJI(I4, K)*DI"KtK)*XJI(JPK)
CONTINUE
DO 69 IA = 1, MZ
Z(IA-1) = Z(IA)
Z(IA-2) = Z(IA)
U(I1,1) = U(I1, 2)
U(I2,1) = U(I2, 2)
U(I1,2) = U(I1, 3)
U(I2,2) = U(I2, 3)
CONTINUE
PRINT 607, SUM(JKM, JKM)
FORMAT(//'WEIGHTED ERROR SUM = ', F12.4)
TYPE 606, SUM(JKM, JKM)
MAX = 8
MA = 7
CALL SPIT(D1)
PRINT 608
FORMAT(//'WEIGHTED ERRORS:', /)
PRINT 606, (D2(IA)*D1(IA)) IA = 1, MZ)
TYPE 606, (D2(IA)*D1(IA)) IA = 1, MZ)
DO 2101 IA = 1, MZ
ERX(IA-LL) = D2(IA)
CONTINUE
DO 888 IJK = 1, JKM
SUM(JJK, JKM) = SUM(JKM, IJK)
IF(MAPR) 180, 181, 180
DO 182 IB = 1, JKM

0142 SUM(3B;JKM) = -XT5(3B)+APR(3B)+SUM(3B;JKM)
0143 SUM(3B;IB) = SUM(3B;IB)+APR(3B)
0144 IB? = IB+1
0145 DO 192 IA = 1,IB?1
0146 192 SUM(3B;IA) = SUM(3B;IA) + SUM(IA;IB) + SUM(IA;IB)
0147 CONTINUE
0148 FORMAT///' END OF ITERATION '///' END)
0149 RETURN
0150 END

Subroutine EAT

Description: Subroutine EAT computes $e^{A\Delta t}$ and $\int_0^{\Delta t} e^{At} dt$ using the Taylor series expansion to ten terms. These are returned as PHI and APHI respectively.

Subroutine listing:

```plaintext
0001 SUBROUTINE EAT (ArTrPHIrAPHIrA2rA3)
0002 COMMON MAX*MAX1+MIX1+MIX
0003 DIMENSION A(1),PHI(1),A2(1),APHI(:,A3(1))
0004 MAX2 = MAX*2
0005 II = A(MAX)
0006 JJ = A(MAX2)
0007 PHI(MAX) = A(MAX)
0008 PHI(MAX2) = A(MAX)
0009 CALL ZOT(PHI)
0010 CALL MAKE(APHI,PHI)
0011 CALL MAKE(A3,PHI)
0012 MI = -MAX
0013 DO 1 I=1,II
0014 MI = MI+MAX
0015 MII = MI+I
0016 PHI(MII+I) = 1.0
0017 1 CONTINUE
0018 CALL MAKE(A2,PHI)
0019 G = 1.0
0020 DO 2 I=1,10
0021 BB = I
0022 G = G+BB
0023 CALL ADD(G,APHI,G,A2,APHI)
0024 CALL MULT(A,A2,A2,A3)
0025 CALL ADD(G,PHI,G,A2,PHI)
0026 2 CONTINUE
0027 DO 10 I=1,II
0028 10 J=1,II

```
Subroutine ZOT

Description: Subroutine ZOT initializes the elements of a matrix to zero.

Subroutine listing:

```fortran
SUBROUTINE ZOT(X)
    COMMON MAXVMAXIvMIXIvMIX
    DIMENSION X(1)
    MAX2 = MAX * 2
    IIM1 = X(MAX) - 1.0
    JJM1 = X(MAX2) - 1.0
    LEND = JJM1 * MAX + 1
    DO 1 K = L - JJM1
        DO 1 L = IIM1
            X(K) = 0.0
        1 CONTINUE
    RETURN
END
```

Subroutine LOAD

Description: Subroutine LOAD loads matrices from the input file.

Subroutine listing:

```fortran
SUBROUTINE LOAD (NP,APBPCPD)
    REAL A(1),PB(1),PC(1),PD(1)
    CALL LOAD1(A)
    IF(N .LT. 2) RETURN
    CALL LOAD1(B)
    IF(N .LT. 3) RETURN
    CALL LOAD1(C)
    IF(N .LT. 4) RETURN
    CALL LOAD1(D)
    RETURN
END
```
Subroutine LOAD1

Description: Subroutine LOAD1 actually loads the matrix from the input file.

Subroutine listing:

0001 SUBROUTINE LOAD1(A)
C ROUTINE CALLED BY LOAD LOADS MATRIX A FROM FILE
C
0002 COMMON MAX
0003 REAL A(1)
0004 READ(2,100) II,JJ
0005 100 FORMAT(BXr129110)
0006 KE = (JJ-1)*MAX
0007 DO 10 I=1,II
0008 KEND = I+KE
0009 10 READ(2,1001) (A(K)PKOIPKENDPMAX)
0010 A(MAX) a 11
0011 A(MAX*2) a JJ
0012 1001 FORMAT(8F12.6)
0013 RETURN
0014 END

Subroutine ADD

Description: Subroutine ADD adds scalar multiples of two matrices, Z = g X + h Y.

Subroutine listing:

0001 SUBROUTINE ADD (G,X,H,Y,Z)
C THIS SUBROUTINE ADDS SCALAR MULTIPLES OF TWO
C MATRICES AS FOLLOWS:
C (Z) = G*(X) + H*(Y) WITH \: G = 1.0
C (NO CHECKING IS MADE FOR MATRIX COMPATIBILITY)
C
0002 COMMON MAX,MAX1,MIX1,MIX
0003 DIMENSION X(1),Y(1),Z(1)
C
0004 MAX2 = MAX % 2
0005 II = X(MAX)
0006 JJ = X(MAX2)
0007 JEND = (JJ-1)*MAX+1
0008 IIIM1 = II-1
0009 DO 53 J=1,JEND,MAX
0010 KEND = J+IIIM1
0011 DO 53 K=J,KEND
0012 Z(K) = X(K)*H+Y(K)
0013 Z(MAX) = X(MAX)
0014 Z(MAX2) = X(MAX2)
0015 RETURN
0016 END

154
Subroutine MAKE

Description: Subroutine MAKE moves a copy of the matrix Y into X.

Subroutine listing:

```fortran
0001 SUBROUTINE MAKE(X, Y)
C
C- - - - - - - - - - - - - - - - - - - - - - -
C
C  THIS SUBROUTINE GENERATES A MATRIX X THAT IS
C  A COPY OF MATRIX Y.
C
C  X : NEW MATRIX, COPY OF Y
C  Y : MATRIX TO BE COPIED
C- - - - - - - - - - - - - - - - - - - - - - -
C
0002 COMMON MAX, MAX1, MIX1, MIX
0003 DIMENSION X(1), Y(1)
C
0004 MAX2 = MAX#2
0005 IIM1 = Y(MAX) - 1.
0006 JJM1 = Y(MAX2) - 1.
0007 LEND = J JM1 + MAX
0008 DO 1 L = I, LEND, MAX
0009 KEND = L + IIM1
0010 DO 1 K = L, KEND
0011 1 X(K) = Y(K)
0012 X(MAX) = Y(MAX)
0013 X(MAX2) = Y(MAX2)
0014 RETURN
0015 END
```

Subroutine MULT

Description: Subroutine MULT computes the matrix product C = A B. The matrix C can not be the same as matrix A or B.

Subroutine listing:

```fortran
0001 SUBROUTINE MULT(A, B, C, D)
C
C MULTIPLIES A AND B AND PUTS THE PRODUCT
C IN C AND D (USING SUBMAKE)
C
0002 COMMON MAX, MAX1, MIX1, MIX
0003 DIMENSION A(1), B(1), C(1), D(1)
C
0004 MAX2 = MAX#2
0005 MIX2 = MIX#2
0006 I = A(MAX)
0007 JJ = A(MAX2)
0008 KK = B(MIX2)
0009 JE = (JJ-1)*MAX
0010 KE = (KK-1)*MAX
0011 DO 20 I = 1, JJ
0012 KEND = KE + I
0013 JEND = JE + I
0014 L = 1
0015 DO 20 K = L, KEND, MAX
0016 D(K) = 0.0
0017 JB = L
C
20 C = C + A(I)*B(J)*(D(K) - 0.0)
0018 DO 10 J = I, JEND, MAX
0019 D(K) = A(J)*B(JB) + D(K)
```

155
Subroutine SPIT

Description: Subroutine SPIT prints out a matrix.

Subroutine listing:

```fortran
0001 SUBROUTINE SPIT(X)
  C
  C
  SUBROUTINE USED FOR THE PRINTOUT OF MATRICES
  C
  COMMON MAX, MAX1, MIX, MIX1, MIX
  0002 DIMENSION X(1)
  0003 100 FORMAT(13X, 'DIMENSION ', 8X, I3, ' BY ', I3)
  0005 101 FORMAT(12X, 10E12.4)

  MAX2 = MAX*2
  0007 II = X(MAX)
  0008 JJ = X(MAX2)
  0009 PRINT 100, II, JJ
  0010 KE = (JJ-1)*MAX
  0011 DO 1 I = 1, II
  0012 KE = (JJ-1)*MAX
  0013 1 PRINT 101, (X(K), K=1, KEND, MAX)

  RETURN
  0015 END
```

Subroutine SPIT1

Description: Subroutine SPIT1 prints out the A and B matrices with "*" 's to show which of the parameters have been allowed to vary.

Subroutine listing:

```fortran
0001 SUBROUTINE SPIT1(X, XX, KI)
  C
  C
  SUBROUTINE USED FOR THE PRINTOUT OF MATRICES
  C
  COMMON MAX, MAX1, MIX, MIX1, MIX
  0002 DIMENSION X(1), XX(1)
  0004 BYTE CHAR(4)
  0005 100 FORMAT(10X, 'DIMENSION ', I3, ' BY ', I3)
  0006 101 FORMAT(10X, 5PE12.4, A1)

  MAX2 = MAX*2
  0007 II = X(MAX)
  0008 JJ = X(MAX2)
  0009 PRINT 100, II, JJ
  0010 KE = (JJ-1)*MAX
  0011 DO 1 I = 1, II
```

156
Subroutine SOLVE

Description: Subroutine SOLVE solves the system of linear equations, $A x = b$ where $A$ is symmetrical. Only the lower triangular and diagonal elements of $A$ are used. The $b$ vector is assumed to be stored in the $N+1$ column of $A$, where $N$ is the dimension of the system.

Subroutine listing:

0001   SUBROUTINE SOLVE(A, X)
C
C SOLVES SYSTEM AX = B WHERE A SYMMETRIC MATRIX
C AND B A MATRIX IN N+1 COLUMN OF A
C
0002   REAL A(25, 25), X(25)
0003   CALL REDUCE(A)
0004   N = A(25, 1)
0005   NM1 = N - 1
0006   NP1 = N + 1
C MULTIPLY MATRICES, (L) * (B) . . .
0007   DO 70 I = 2, N
0008      X(I) = A(I, NP1)
0009      IM1 = I - 1
0010   DO 70 J = 1, IM1
0011      70 X(I) = X(I) + A(I, J) * A(J, NP1)
C MULTIPLY BY (DI)
0012   DO 80 I = 2, N
0013      A(I, NP1) = A(I, NP1) / A(I, 1)
0014   DO 80 J = 1, N
0015      A(I, NP1) = X(I) / A(I, 1)
C MULTIPLY BY (L*) TO FORM (L*)*(DI)*L*(B)
0016   DO 90 I = 1, NM1
0017      X(I) = A(I, NP1)
0018      IP1 = I + 1
0019   DO 90 J = IP1, N
0020      X(I) = X(I) + A(J, I) * A(J, NP1)
0021   X(N) = A(N, NP1)
C
0021   RETURN
0022   END
Subroutine DIAGIN

Description: Subroutine DIAGIN obtains the diagonal elements of the inverse of a symmetric matrix.

Subroutine listing:

```fortran
0001    SUBROUTINE DIAGIN(A)
0002       C
0003       C FIND DIAGONAL ELEMENTS OF A INVERSE FOR SYMMETRIC A
0004       C
0005       REAL A(25,25)
0006       CALL REDUCE(A)
0007       N = A(25,1)
0008       NM1 = N-1
0009       DO 90 I=1,NM1
0010          A(I,I) = 1.0/A(I,I)
0011       RETURN
0012       END
```

Subroutine REDUCE

Description: Subroutine REDUCE factors a symmetric matrix A by Cholesky's matrix decomposition.

Subroutine listing:

```fortran
0001    SUBROUTINE REDUCE(A)
0002       C REDUCES SYMMETRIC MATRIX A STORED IN LOWER TRIANGULAR LOCATIONS TO THE FORM (LI)*(D)*(LI*) WHERE L IS A LOWER TRINGULAR MATRIX WITH UNITY DIAGONAL TERMS, D IS A DIGONAL MATRIX, I DENOTES INVERSE AND * TRANSPOSE.
0003       C
0004       REAL A(25,25)
0005       N = A(25,1)
0006       NM1 = N-1
0007       DO 20 K=1,NM1
0008          AKKI = 1.0/A(K,K)
0009          DO 10 I=K+1,N
0010             A(I,K) = A(I,K)-AKKIK*A(K,K)
0011          DO 10 J=1,N
0012             A(J,I) = A(J,I)-AKKIK*A(J,K)
0013          IF(KM1.EQ.0.0) 00 TO 20
0014          DO 15 J=1,N
0015             A(I,J) = A(I,J)-AKKIK*A(K,J)
0016          CONTINUE
0017       RETURN
0018       END
```
Subroutine CRAMER

Description: Subroutine CRAMER computes the confidence levels of the estimated derivatives.

Subroutine listing:

```fortran
0001 SUBROUTINE CRAMER(MU, MX, MZ, NI)

   THIS SUBROUTINE COMPUTES THE CRAMER-RAO BOUNDS
   ALSO KNOWN AS THE CONFIDENCE LEVELS OF THE
   ESTIMATED DERIVATIVES.
   MU = NUMBER OF CONTROL INPUTS
   MX = NUMBER OF STATES
   MZ = NUMBER OF OBSERVATIONS
   NI = MAX. NUMBER OF UNKNOWNS (25)

   normally the apriori contribution to hessian is subtracted
   for this computation but this routine assumes no apriori
   options are being used and hence there are no contributions

0002 COMMON MAX, MU, MM, MAT, Z, U, D2, D1, A, B, AA, BB,
       BJ, X1, SUM, PB, XT1, ZERO, D54, D44, E, X1, CCC, LA, SB.
0003 COMMON /EQDATA/ ANPT
0004 DIMENSION AC(5, 4), BC(5, 4)
0005 DIMENSION XT5(25), APR(25)
0006 DIMENSION AP(8, 4), BP(8, 3), XT4(4)
0007 DIMENSION ZT73), U(3, 3), D27, DDO4, 5, 4), BIAS(5), APHI(5, 4),
       XT1, PII(5, 4), D187, A(5, 4), B(5, 4), AA(5, 4),
       BB(5, 4), BJ1(25, 4), XI1(25, 7), SUM(25, 25), PB(25),
       DUM(25, 4), XT27, ZERO(5), 54, SP4), XT3(7)

   store weighted error sum in ERRSUM
   ERRSUM = SUM(JKMM1), JKMM1/ANPT

0008 AC(5, 1) = MX
0009 AC(5, 2) = MX
0010 BC(5, 1) = MX
0011 BC(5, 2) = MU
0012 JKMM1 = SUM(NI, 1) + 1.01

   obtain diagonal elements of inverse

0013 CALL DIAGIN(SUM)

   compute cramer-rao bounds

0015 WTS = 0.0
0016 DO 1 I = 1, MZ
0017 IF(DI(I, I) .NE. 0.0) WTS = WTS + 1.0
0019 1 CONTINUE
0020 COE.F = ERRSUM/WTS
0021 PRINT 10, ERRSUM, COEFF, WTS
0022 10 FORMAT(' ERRSUM = ', F12.4, ' COEFF = ', F12.4, ' WTS = ', F12.4)
0023 L = 0
0024 DO 2 I = 1, MX
0025 DO 3 J = 1, HU
0026 SC(I, J) = 0.0
0027 IF(BB(I, J) .NE. 1.1) GO TO 3
0029 L = L + 1
0030 SC(I, J) = SORT(ABS(SUM(L, L)) * COEFF)
0031 3 CONTINUE
```

159
Subroutine OUTPUT

Description: Subroutine OUTPUT provides the output of time histories and matrices to user defined files for later plotting.

Subroutine listing:

```
0001 SUBROUTINE OUTPUT
0002 COMMON/MATA/ ALX,BLX
0003 COMMON/TRANSF/ XL
0004 COMMON/CONST/ ITR,NN
0005 BYTE INAME(15)
0006 DIMENSION ALX(20,10),BLX(20,10)
0007 DIMENSION XL(300,7)
0008 TYPE 10
0009 10 FORMAT(/' ENTER FILE NAME FOR OUTPUT OF MMLE MATRICES'/,'/'
0010 ACCEPT 11(INAME(IA8),IAB=1,14)
0011 11 FORMAT(1441)
0012 INAME(15)=0
0013 OPEN(UNIT=2,NAME=INAME,TYPE='NEW',ACCESS='SEQUENTIAL',
0014 1 FORM='FORMATTED',BUFFERCOUNT=2)
0015 35 FORMAT(10E12.4)
0016 40 FORMAT(' ### MATRIX A ### '/)
0017 50 FORMAT(' ### MATRIX B ### '/)
0018 60 FORMAT(' ITERATION ... 'I2,/')
0019 70 FORMAT(/' ESTIMATED TIME RESPONSES'//)
0020 80 DO 130 I=1,ITR
0021 90 WRITE(2,60) I
0022 100 WRITE(2,40)
0023 110 DO 30 J=1,14,4
0024 120 WRITE(2,35) (ALX(J-1+K,I),K=1,4)
0025 130 CONTINUE
0026 140 WRITE(2,50)
0027 150 DO 131 J=1,14,4
0028 160 WRITE(2,35) (BLX(J-1+K,I),K=1,3)
0029 170 CONTINUE
0030 CALL CLOSE(2)
```
C
0031 TYPE 80
0032 FORMAT(/' TYPE IN FILE THAT WILL CONTAIN',
1       ' LAST INERATION TIME RESPONSES.'/
0033 ACCEPT INAME(IA!),IAB=1,14
0034 OPEN(UNIT=3,NAME=INAME,TYPE='NEW',ACCESS='SEQUENTIAL',
1       FORM='UNFORMATTED',BUFFERCOUNT=2)
C
C
0035 DO 330 N=2,NN
0036 WRITE(3) (XL(N,I),I=1,7)
0037 330 CONTINUE
0038 WRITE(3)
0039 RETURN
0040 END
A.7.3) MMLE OUTPUT FORMAT

Following is an example and description of the MMLE output.

Longitudinal:

KU FRL BONES MMLE RESULTS
CCESSHA 172 LONGITUDINAL CAPE
3000. FT ALT. AT 176. FPS AIRSPEED
FLIGHT 19/10/80 RUN 23

NUMEBR OF DATA POINTS : 240
MAXIMUM NUMBER OF ITERATIONS : 9
DATA SAMPLING INTERVAL : 0.0001
FIRST DATA POINT AT TIME : 0.0000
DIAGONAL MULTIPLYING FACTOR : 1.0000
NUMBER OF STATES : 7

ZEROS AND DIABETES
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
DIAGONAL ELEMENTS OF THE WEIGHTING MATRIX Di:
100.000 0.000 0.000 700.000 0.000 40.000 5.000

INITIAL INPUT MATRICES [A] AND [B].
A STAR (*) FOLLOWING THE VALUE OF A MATRIX
ELEMENT INDICATES THAT THE RESPECTIVE DERIVATIVE
IS NOT ESTIMATED BY THE MMLE METHOD.

STABILITY MATRIX [A]
DIMENSION 4 BY 4
-0.586E+01 0.0000E+09 -0.2130E+02 0.0000E+00
0.0000E+00 0.0000E+00 0.1936E+02 -0.316E+02
0.1000E+01 0.0000E+00 -0.2575E+01 -0.454E-02
0.9995E+00 0.0000E+00 0.0000E+00 0.0000E+00

CONTROL MATRIX [B]
DIMENSION 4 BY 3
-0.201E+02 0.0000E+00 0.0000E+00
-0.341E+01 0.0000E+00 0.0000E+00
-0.2409E+00 0.0000E+00 0.0000E+00
0.0000E+00 0.0000E+00 0.0000E+00

WEIGHTED ERROR SUM = 2197421.2500
WEIGHTED ERRORS:
0.1401E+04 0.0000E+00 0.0000E+00 0.2130E+07 0.0000E+00 0.1147E+02 0.2500E+04
ITERATION 1 WAS COMPLETED

\[
\begin{bmatrix}
M'_{q} & M'_{u} & M'_{a} & M'_{}\theta \\
0 & X'_u & X'_a & -s \cos (\phi_1) \\
2 & \frac{z'_a + u'_1}{u'_1 - z'_a} & \frac{z'_a}{u'_1 - z'_a} & -s \cos (\phi_1) \sin (\phi_1) \\
3 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
M'_{qE} & M'_{qc} & M'_{}\theta \\
X'_q & X'_c & X'_\theta \\
2 & \frac{z'_E + u'_1}{u'_1 - z'_a} & \frac{z'_E}{u'_1 - z'_a} & -s \cos (\phi_1) \sin (\phi_1) \\
3 & 0 & 0 & 0 \\
\end{bmatrix}
\]
Lateral:

KU FAL BONES HALE RESULTS
CESSNA 172 LATERAL-DIRECTIONAL CASE
3000 FT. ALT. AT 176. FPS AIRSPEED
FLIGHT 19/10/80 RUN 48

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

INITIAL CONDITIONS

NUMBER OF DATA POINTS 1 140
MAXIMUM NUMBER OF ITERATIONS 9
DATA SAMPLING INTERVAL 0.0000
FIRST DATA POINT AT TIME 0.0000
DIAGONAL MULTIPLYING FACTOR 1.0000
NUMBER OF STATES 7

ZEROS AND BIASES
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
DIAGONAL ELEMENTS OF THE WEIGHTING MATRIX DI:

INITIAL INPUT MATRICES C(A) AND C(B).
A STAR (*) FOLLOWING THE VALUE OF A MATRIX
ELEMENT INDICATES THAT THE RESPECTIVE DERIVATIVE
IS NOT ESTIMATED BY THE HALE METHOD.

STABILITY MATRIX (A)
DIMENSION 4 BY 4

-0.9746E+01 0.1041E+01 -0.1815E+02 0.0000E+00
-0.3238E+00 -0.7214E+00 0.0000E+00
0.3934E-01 -0.9972E+00 0.1827E+00
0.1000E+01 0.3946E-01 0.0000E+00

CONTROL MATRIX (B)
DIMENSION 4 BY 3

0.3330E+02 0.2998E+01 0.0000E+00
-0.5982E+01 -0.6729E+01 0.0000E+00
0.0000E+00 0.3750E+01 0.0000E+00
0.0000E+00 0.0000E+00 0.0000E+00

WEIGHTED ERROR SUM = 52015.3281
WEIGHTED ERRORS:
0.2587E+03 0.0627E+03 0.0000E+00 0.1237E+05 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
ITERATION 1 WAS COMPLETED

\[
\begin{bmatrix}
L_p' \\
L_x' \\
L_y' \\
0.0
\end{bmatrix}
+ \begin{bmatrix}
N_p' \\
N_x' \\
N_y' \\
0.0
\end{bmatrix}
+ \begin{bmatrix}
\sin(a_1) \\
-\cos(a_1)
\end{bmatrix}
+ \begin{bmatrix}
V_y' \\
\frac{\sin(\alpha_1)\cos(\theta_1)}{U_1}
\end{bmatrix}
+ \begin{bmatrix}
1.0 \\
\cos(\theta_1)\tan(\theta_2)
\end{bmatrix}
= \begin{bmatrix}
0.0 \\
0.0
\end{bmatrix}
\]

SIM1 (p, q, r, \beta, \phi, \delta, \zeta, \alpha)
A.8) **TIME HISTORY PLOTTING**

Description: The PLOT03 program collects the time histories of both the actual inflight measurements, and the predicted states of the MMLE BONES program. These are then plotted on the graphics CRT terminal. This process allows the user to observe the fit to the measured states. For hard copy plots the time histories are transferred to the KU-FRL's Hewlett Packard computer.

Program listing:

```plaintext
C *:*t*tt**t*********t*
C
C THIS PROGRAM CAN BE USED TO PLOT
C MEASURED AND ESTIMATED DATA.
C DATE 18-NOV-80
C THE REQUIRED SUBROUTINES ARE:
C PLOT55rINIT AND ORAPH. LINK AS
C FOLLOWS : PLOT=MAIN,PLOT55,INIT, ORAPH.
C
COMMON/STATUS/ISTAT(16)
DIMENSION IARRAY(512),HH(9),RDATA(240,9),EDATA(240,9)
COMMON/FIVEA/ DATA,GAIN
BYTE YES,NO,ANSI,LOG,DIR,ANS2,NAME(15)
DATA LOG-DIR /*L*/,'D'/
DATA YES,NO /*Y*/,'N'/
DATA ISTAT/16*0/
DATA GAIN1/114.6,114.6,143.2,47.7,57.3,57.3,100.,143.2,143.2/
DATA GAIN2/114.6,2.5,143.2,95.5,57.3,100.,25.,143.2,0./

0001 COMMON/STATUS/ISTAT(16)
0002 DIMENSION IARRAY(512),HH(9),RDATA(240,9),EDATA(240,9)
0003 DIMENSION GAIN(9),GAIN1(9),GAIN2(9)
0004 COMMON/FIVEA/ DATA,GAIN
0005 BYTE YES,NO,ANSI,LOG,DIR,ANS2,NAME(15)
0006 DATA LOG-DIR /*L*/,'D'/
0007 DATA YES,NO /*Y*/,'N'/
0008 DATA ISTAT/16*0/
0009 DATA GAIN1/114.6,114.6,143.2,47.7,57.3,57.3,100.,143.2,143.2/
0010 DATA GAIN2/114.6,2.5,143.2,95.5,57.3,100.,25.,143.2,0./

0011 TYPE 1
0012 1 FORMAT(//,'HELLOOOOOO.......',/)
1' THIS IS A PLOTTING PROGRAM USED TO PLOT/
2' MEASURED VERSUS ESTIMATED TIME HISTORIES'/
3' INSTRUCTIONS: '/
4' 1. TYPE IN TYPE OF MANOEUVRE.'/
5' 2. TYPE IN NUMBER OF TIME POINTS N (0-250)/
6' 3. INDICATE NAME OF FILE CONTAINING MEASURED DATA.'/
7' 4. INDICATE NAME OF FILE CONTAINING ESTIMATED DATA.'/
8' 5. WHEN DATA IS PLOTTED HIT 'CR' TO CONTINUE.'/
9' 6. GOOD LUCK....... ALEX'/

003 TYPE 301
0014 301 FORMAT('INDICATE TYPE OF MANOEUVRE'/
1' 'L' FOR LONGITUDINAL'/
2' 'D' FOR LATERAL DIRECTIONAL'/
0015 READ(5,302) ANSI
0016 302 FORMAT(A1)
0017 READ IN NUMBER OF TIME POINTS

165
FORMAT(I3)
READ(5) N
READ DATA FILE WITH MEASURED DATA

TYPE 30
FORMAT(' TYPE IN NAME OF DATA FILE WITH MEASURED DATA'/)

NAME(I) = 0
ACCEPT 31, (NAME(I), I=1,14)
OPEN(UNIT=2,NAME=TYPE='OL',ACCESS='SEQUENTIAL',
  READONLY=FORM='UNFORMATTED')
DO 32 I=1,14
NAME(I) =''
CONTINUE

READ(2) DMY, DMY, DMY, DMY, HH, HH, HH, HH, HH
  DO 31 J=1,7
   EDATA(I+J-1) = HH(J)
   CONTINUE

READ(2) DMY, DMY, DMY, DMY
  DO 30 J=1,7
   EDATA(I+J-1) = 0.0
   CONTINUE

CONTINUE
DO 50 J=1,9
READ('') (HH(J), J=1,7)
CONTINUE

READ(1) K=1,14
CONTINUE

ACCEPT 31, (NAME(I), I=1,14)
OPEN(UNIT=3,NAME=TYPE='OL',ACCESS='SEQUENTIAL',
  READONLY=FORM='UNFORMATTED')
DC 81 K=1,14
NAME(I) =''
CONTINUE

FORMAT(12E12.4)
DO 50 I=1,N
IF(ANS1.EQ.L00) GO TO 310

READ(2) DMY, DMY, DMY, DMY, HH, HH, HH, HH, HH
  DO 31 J=1,7
   EDATA(K+J-1) = HH(J)
   CONTINUE

READ(1) J=1,7
READ(2) HH, HH, HH, HH, HH, HH, HH, HH, HH, HH
  DO 30 J=1,7
   EDATA(J) = HH(J)
   CONTINUE

CONTINUE
DO 50 J=1,7
READ(1) J=1,7
CONTINUE

CONTINUE
DO 50 J=1,7
READ(1) J=1,7
CONTINUE

CONTINUE
DO 50 J=1,7
READ(1) J=1,7
CONTINUE
0064  EDTA(2,J) = 0.0
0065  CONTINUE
0066  C
0067  DO 500 I=1,N
0068  EDTA(I,B)=0.0
0069  EDTA(I,9)=0.0
0070  CONTINUE
0071  IARRAY(1) = 0
0072  IARRAY(3) = 0
0073  CONTINUE
0074  TYPE 100
0075  IF(ANS1.EQ.DIR) GO TO 102
0076  TYPE 90
0077  CONTINUE
0078  FORMAT(/' VARIABLE RANGE +/-',//
0079       1' 1. PITCH RATE - 50 DEG/SEC',/
0080       2' 2. AIRSPEED - 20 FT/SEC',/
0081       3' 3. ANGLE OF ATTACK - 20 DEG',/
0082       4' 4. PITCH ATTITUDE - 30 DEG',/
0083       5' 5. PITCH RATE ACCEL. - 50 DEG/SEC**2',/
0084       6' 6. Longitudinal ACCEL. - 5 G',/
0085       7' 7. Normal ACCEL. - 2 G',/
0086       8' 8. ELEVATOR PSN. - 20 DEG',/
0087       9' 9. * X * (BLANK)',//)
C
C
C

0079  CONTINUE
0080  IF(ANS1.EQ.L00) GO TO 103
0081  TYPE 91
0082  CONTINUE
0083  FORMAT(/' VARIABLE RANGE +/-',//
0084       1' 1. ROLL RATE - 25 DEG/SEC',/
0085       2' 2. YAW RATE - 25 DEG/SEC',/
0086       3' 3. SIDESLIP ANGLE - 20 DEG',/
0087       4' 4. BANK ANGLE - 60 DEG',/
0088       5' 5. ROLL RATE ACCEL. - 50 DEG/SEC**2',/
0089       6' 6. YAW RATE ACCEL. - 50 DEG/SEC**2',/
0090       7' 7. Longitudinal ACCEL. - 5 G',/
0091       8' 8. AILERON DEFLECTION - 20 DEG',/
0092       9' 9. Rudder DEFLECTION - 20 DEG',/
C
C
C

0084  CONTINUE
0085  TYPE 11
0086  FORMAT(' INDICATE VARIABLE NO. FOR TOP PLOT',//)
0087  ACCEPT #KT
0088  TYPE 12
0089  CONTINUE
0090  FORMAT(' INDICATE VARIABLE NO. FOR BOTTOM PLOT',//)
0091  ACCEPT #KB
0092  CONTINUE
0093  IF(ANS1.EQ.L00) GO TO 411
0094  DO 421 I=1,9
0095  GAIN(I) = GAIN1(I)
0096  CONTINUE
0097  CONTINUE
0098  IF(ANS1.EQ.DIR) GO TO 402
0100  DO 422 I=1,9
0101  GAIN(I) = GAIN2(I)
0102  CONTINUE
0103  CONTINUE
C
C
C CLEAR CRT AND FORM GRID FOR PLOTTING

0104 CALL INIT
0105 CALL PLOT55(2,1+2+32+64+128+ISTAT)
0106 DIG 110 K=1,235+50
0107 CALL PLOT55(4+1*K+1,ISTAT)
0108 110 CONTINUE
0109 CALL PLOT55(4,1,229,ISTAT)
0110 CALL PLOT55(5,0+1,ISTAT)

C

C FORM THE MEASURED AND ESTIMATED RESPONSES

0111 DO 130 I=1,N
0112 IARRAY(2*I) = RDATA(I,KT)*GAIN(KT)+150
0113 IARRAY(2*I-1) = RDATA(I,KB)*GAIN(KB)+50
0114 130 CONTINUE
0115 CALL PLOT55(9,20,2,ISTAT)
0116 CALL PLOT55(12,*,*,* TIME HISTORIES *,*,*,*,ISTAT)
0117 CALL PLOT55(9,50,4,ISTAT)
0118 CALL PLOT55(12,*,* MEASURED DATA *,*,*,*ISTAT)
0119 CALL GRAPH(2*N,IARRAY)
0120 DO 140 I=1,N
0121 IARRAY(2*I) = EDATA(I,KT)*GAIN(KT)+150
0122 IARRAY(2*I-1) = EDATA(I,KB)*GAIN(KB)+50
0123 140 CONTINUE
0124 CALL PLOT55(9,50,4,ISTAT)
0125 CALL PLOT55(12,*,* ESTIMATED DATA *,*,*,*ISTAT)
0126 CALL GRAPH(2*N,IARRAY)

C

0127 CALL PLOT55(9,50,4,ISTAT)
0128 CALL PLOT55(12,*,* ISTAT)
0129 KFLAG = 1

C

0130 IF(ANS1.EQ.DIR) GO TO 699
0132 KFLAG1 = 2

C LONGITUDINAL LABELS

0133 CALL PLOT55(9,50,6,ISTAT)
0134 IF(KT.EQ.1) GO TO 601
0136 IF(KT.EQ.2) GO TO 602
0138 IF(KT.EQ.3) GO TO 603
0140 IF(KT.EQ.4) GO TO 604
0142 IF(KT.EQ.5) GO TO 605
0144 IF(KT.EQ.6) GO TO 606
0146 IF(KT.EQ.7) GO TO 607
0148 IF(KT.EQ.8) GO TO 608

C

0150 610 CONTINUE
0151 KFLAG1 = 1

C

0152 CALL PLOT55(9,50,14,ISTAT)
0153 IF(KB.EQ.1) GO TO 601
0155 IF(KB.EQ.2) GO TO 602
0157 IF(KB.EQ.3) GO TO 603
0159 IF(KB.EQ.4) GO TO 604
0161 IF(KB.EQ.5) GO TO 605
0163 IF(KB.EQ.6) GO TO 606
0165 IF(KB.EQ.7) GO TO 607
0167 IF(KB.EQ.8) GO TO 608

C

0169 601 CALL PLOT55(12,*,0 +/- 25 DEG/SEC *,*,*,*ISTAT)
0170 GO TO 640
0171 602 CALL PLOT55(12," V +/- 20 FEET/SEC 'rISTAT)
0172 603 CALL PLOT55(12," ALPHA +/- 20 DEG 'rISTAT)
0174 604 CALL PLOT55(12," THETA +/- 30 DEG 'rISTAT)
0176 605 CALL PLOT55(12," 0 DOT +/- 50 DEG/SEC**2 'rISTAT)
0178 606 CALL PLOT55(12," AX +/- .5 G 'rISTAT)
0180 607 CALL PLOT55(12," AN +/- 2 G 'rISTAT)
0182 608 CALL PLOT55(12," BE +/- 20 DEG 'rISTAT)

0184 640 CONTINUE
0185 IF(KFLAG1.EQ.0) GO TO 610

0187 699 CONTINUE

C LATERAL DIRECTIONAL LABELS

C KFLAG02 = 0

0190 CALL PLOT55(9,?.50,6;ISTAT)
0191 IF(KT.EQ.1) GO TO 701
0193 IF(KT.EQ.2) GO TO 702
0195 IF(KT.EQ.3) GO TO 703
0197 IF(KT.EQ.4) GO TO 704
0199 IF(KT.EQ.5) GO TO 705
0201 IF(KT.EQ.6) GO TO 706
0203 IF(KT.EQ.7) GO TO 707
0205 IF(KT.EQ.8) GO TO 708
0207 IF(KT.EQ.9) GO TO 709

0209 710 CONTINUE

C KFLAG02 = 1

0211 CALL PLOT55(9,?.50,16;ISTAT)
0212 IF(KB.EQ.1) GO TO 701
0214 IF(KB.EQ.2) GO TO 702
0216 IF(KB.EQ.3) GO TO 703
0218 IF(KB.EQ.4) GO TO 704
0220 IF(KB.EQ.5) GO TO 705
0222 IF(KB.EQ.6) GO TO 706
0224 IF(KB.EQ.7) GO TO 707
0226 IF(KB.EQ.8) GO TO 708
0228 IF(KB.EQ.9) GO TO 709

C

0230 701 CALL PLOT55(12," P +/- 25 DEG/SEC 'rISTAT)
0231 GO TO 740
0232 702 CALL PLOT55(12," R +/- 25 DEG/SEC 'rISTAT)
0233 GO TO 740
0234 703 CALL PLOT55(12," BETA +/- 20 DEG 'rISTAT)
0235 GO TO 740
0236 704 CALL PLOT55(12," PHI +/- 60 DEG 'rISTAT)
0237 GO TO 740
0238 705 CALL PLOT55(12," P DOT +/- 50 DEG/SEC**2 'rISTAT)
0239 GO TO 740
0240 706 CALL PLOT55(12," R DOT +/- 50 DEG/SEC**2 'rISTAT)
0241 GO TO 740
0242 707 CALL PLOT55(12," AY +/- .5 G 'rISTAT)
0243 GO TO 740
0244 708 CALL PLOT55(12," DA +/- 20 DEG 'rISTAT)
0245 GO TO 740
0246 709 CALL PLOT55(12," DR +/- 20 DEG 'rISTAT)
CONTINUE

IF(KFLAG.EQ.0) GO TO 710

LOGIC FOR GENERATING NEW PLOTS END TERMINATING

CONTINUE

READ(S,180) KR

FORMAT(12)

CALL INIT

TYPE 210

DO YOU WANT TO REPLACE TOP PLOT? (Y/N)'/

READ(5,220) ANS2

IF(ANS2.EQ.NO) KFLAG = 0

IF(ANS2.EQ.NO) GO TO 230

TYPE 100

IF(ANS1.EQ.LOG) TYPE 90

IF(ANS1.EQ.DIR) TYPE 91

TYPE 240

IF(ANS2.EQ.NO) GO TO 230

FORMAT(' INDICATE NEW VARIABLE NUMBER '/)

ACCEPT *rKT

CONTINUE

TYPE 309

DO YOU WANT TO REPLACE BOTTOM PLOT? (Y/N)'/

READ(5,220) ANS2

IF(ANS2.EQ.NO) GO TO 400

TYPE 320

INDICATE NEW VARIABLE NUMBER'/

ACCEPT *KB

GO TO 105

CONTINUE

IF(KFLAG.EQ.0.) GO TO 410

GO TO 105

CONTINUE

CALL PLOT55(2*512r1+2*32rISTAT)

CALL PLOT55(3*72rrISTAT)

CALL PLOT55(2r1+512rNUMEB+1)*ISTAT)

CALL PLOT55(1+512,ISTAT)

RETURN

END

SUBROUTINE INIT

COMMON/STATUS/ISTAT(16)

DATA ISTAT/16*0/

CALL PLOT55(13,72,ISTAT)

CALL PLOT55(13,74,ISTAT)

CALL PLOT55(2,1+512,ISTAT)

CALL PLOT55(0,-1,0,ISTAT)

RETURN

END

SUBROUTINE GRAPH(N,ARRAY)

COMMON/STATUS/ISTAT(16)

DIMENSION IARRAY(512)

NUMBER=ISTAT(8)/8

CALL PLOT55(7,0,ISTAT)

CALL PLOT55(8,512,ISTAT)

CALL PLOT55(2,1+(NUMBER+1)*2,(NUMBER+1)*10,ISTAT)

CALL PLOT55(3,-N,ARRAY,ISTAT)

CALL PLOT55(1+NUMBER,ISTAT)

CALL PLOT55(9,10,1,ISTAT)

END

170
A.9) CESSNA PROGRAMS

This appendix contains listings of the programs used in the Cessna spin test program.

A.9.1) DATA ACQUISITION

Description: This is an assembly language program for the AIM 65. The program is essentially the same as the one of Appendix A.1, the differences being
- no start and end data is taken;
- Channels 0-14 are sampled continuously every 0.1 secs when the "RUN/STBY" switch is on RUN;
- data is output to the TEAC tape drive every 0.5 secs.

Program listing:

```assembly
; DATA ACQUISITION
RNCNT=0  ; RCNT=0
BLKCNT=2 ;
BUFCTN=4
BUF=5
ORUF=7
CNT=9
BUF1=$200
BUF2=$300
KDRRA2=$A481
KDRRB2=$A483
KDRRA2=$A480
KDRRB2=$A482
DBE=$9008
WDC=$9009
CDR=$900A
MDR=$900B
CSR=$900C
ESR=$900D
ISR=$900E
MDR1=$900F
WDT=$C1
WTC=$C2
ERA=$C3
SLE=$C9
REN=$CA
MDRV=$110
FRT=$04
DA=$20
DBN=$40
CGE=$80
UDB=$A000
UDCB=$A000
UIER=$A00E
UTIL=$A004
UTCH=$A005
UTL=$A00B
UTD=$A009
UIFR=$A00D
BITS=$20
LOADK=$8F
RECK=$8F
CLOSEK=$DF
TIMEL=$527
TIMEIL=$510

; $0400
LDA #$92
STA MDR0
LDA #1
STA RNCNT
LDA #0
STA RNCNT+1
LDA #$FF
STA KDRRA2
LDA #0
STA KDRRB2
LDA #$C0
STA UACR
LDA #<INT
CDR=$900A
START STA $A404
MDRO=$9008
LDA #$12
LDA #$CO
CSR=$9000
STA MDRO
STA UIER
ESR=$900D
LDA #F.EW LDA #<$C34E
ISR=$900E JSR COMD STA UT1L
MDR1=$900F JSR COMD REC2
CCE=$80 MAIN2 JSR GKEY
UDKB=$A000 JSR GKEY
CMP #PECK BNE RECX
UACR=$A0CB CMP #BECK BNE RECX
UIER=$A00E BEQ RECORD
UTIL=$A004 JSR GKEY
UTE=$A005 CMP #RECK
UTL=$A00B JSR GKEY
CMP #CLOSEK
UTD=$A009 BEQ RECORD
UIFR=$A00D CMP #WM
BITS=$20 STA #WM
LOADK=$8F LDA #TW
RECK=$8F JSR CMD
CLOSEK=$DF LDA #WC
TIMEL=$527 JSR CMD
TIMEIL=$510 LDA #WM

171
A.9.2) DATA READBACK

Description: This program is used to read the data off the AIM 65 system's tape drive to be sent to the ground-based system. The program is similar to the one of Appendix A.2, the differences being

- no error checking is done by the AIM 65;
- all 12 BITS of recorded data are transferred just as they have been recorded.
Program listing:

; DATA RECOVERY

RNCNT=0
BLKNT=2
VRUN=4
VBIL=5
CNT=6
BUFI=200
DBR=90008
JDC=90009
CDR=9000A
MDRO=9000B
CSR=9000C
ESR=9000D
ISR=9000E
MDRL=9000F
SLP=SCB
RDL=SCA
RH=SCA
RNY=10
TDR=10
R=90D
SCR=90006
SDR=90007
LOADC=54C
READC=552
CLOSEC=43
INALL=9993
NUMA=9446
READC=993C
OUTALL=99BC
OUTPUT=997A
UTIL=5004
UTICH=5005
UACR=5006

--------

#300
CCE
..BYTE $80
DA
..BYTE $20
NO
..BYTE CR, 'TAPE ERROR', 50
MBUN
..BYTE CR, 'WHICH RUN NUMBER', 50
MBLK
..BYTE CR, 'HOW MANY BLOCKS', 50
MEND
..BYTE CR, 'LAST BLOCK THIS RU', 50
MINV
..BYTE CR, 'INVALID COMMAND', 50
MERR1
..BYTE CR, 'FILE MARK FOUND', 50
MRNCNT
..BYTE CR, 'RUN NUMBER', 50

--------

#300
RESETB
LDA 992
STA MDRO
LDA 5CO
STA UACR
LDA $68 ;$68=300BAUD
; $3A=600
; $1A=1200
; $0D=2400

JMP UTIL
LDA #0

STA UTICh
LDA #$11

STA SCR

MAIN
JSR GCCOM
CMP #0
BEQ MAIN2
JSR INVALID
JSR MAIN2
LDA #$12
STA MDRO
LDA #REW
JSR COMDA
LDA #$LP
JSR COMDA
MAIN
JSR GCCOM
CMP #REW
BEQ READ
JSR CLOSEC
BEQ CLOSE
JSR INVALID
JSR MAIN

CLOSE
LDA #REW
JSR COMDA
LDA #REW
JSR COMDA
JMP MAIN

READ
LDY #MBUN-MO
JSR MESS
JSR GCNT
CMP #0
BEQ CLOSE
STA VRUN
READ1
JSR RBLK
BCS CLOSE
LDA BLKNT
ORA BLKNT+1
BNE READ3
LDY #MRNCNT-MO
JSR MESS

LDA RNCNT
JSR NUMA
READ3
LDA RNCNT
CMP VRUN
BCC READ1
BEQ READ2
LDA #$REW
JSR COMDA
LDA #$LP
JSR COMDA

; CHANGE TO NOPs
; TO TRANSMIT COUNTS

SEND1
JMP SENDB1
LDA RNCNT
JSR SEND
LDA RNCNT+1
JSR SEND
LDA BLKNT
JSR SEND
LDA BLKNT+1
JSR SEND
SEND2
LDA BUFL
JSR SEND
INT
CPY #160
BNE SEND2
LDA BLKNT
CPR #R1
BNE SEND2

CCE JSR COMDA
SENDB3
LDA RNCNT
SENDB4
LDY #0
SEND4
LDA BUFL
JSR SEND
INT
CPY #90
BNE SEND4
LDA BLKNT
CPR #R1
BNE SEND4

CBX RBLK
BCS SENDB
CLOSE1
JMP CLOSEB
END
LDY #MEND-MO
JSR MESS
JMP MAIN

JMP READ1
READ2
LDY #MBUN-MO
JSR MESS
JSR GCNT
CMP #0
BEQ CLOSE
STA VBLK
SEND5
JSR RBLK
BCC SENDB
CLOSE1
JMP CLOSEB
END
LDY #MEND-MO
JSR MESS
JMP MAIN3

; CHANGE TO NOPS
; TO TRANSMIT COUNTS

SENDB1
JMP SENDB1
LDA RNCNT
JSR SEND
LDA RNCNT+1
JSR SEND
LDA BLKNT
JSR SEND
LDA BLKNT+1
JSR SEND
SEND2
LDA BUFL
JSR SEND
INT
CPY #160
BNE SEND2
LDA BLKNT
CPR #R1
BNE SEND2

CCE JSR COMDA
SENDB3
LDA RNCNT
SENDB4
LDY #0
SEND4
LDA BUFL
JSR SEND
INT
CPY #90
BNE SEND4
LDA BLKNT
CPR #R1
BNE SEND4

CBX RBLK
BCS SENDB
CLOSE1
JMP CLOSEB
END
LDY #MEND-MO
JSR MESS
JMP MAIN

JMP READ1
READ2
LDY #MBUN-MO
JSR MESS
JSR GCNT
CMP #0
BEQ CLOSE
STA VBLK
SEND5
JSR RBLK
BCC SENDB
CLOSE1
JMP CLOSEB
END
LDY #MEND-MO
JSR MESS
JMP MAIN3
MESS
LDA #0,Y
PHA
AND #$7F
JSR OUTPUT
INY
PLA
BPL MESS
RTS

GCOM
JSR READM
JSR OUTALL
RTS

INVAL
LDY #MINV-MO
JSR MESS
RTS

GCT1
LDA #0
STA CNT
GCT1
JSR INALL
JSR DPACK
BCC GCT1
LDA CNT
RTS

DPACK
CMP '#0'
BCC RSPAC
CMP #53A
BCC RSPAC
AND #$5F
PHA
LDA CNT
ASL A
ASL A
CLC
ADC CNT
ASL A
STA CNT
PLA
CLC
ADC CNT
STA CNT
CLC
RTS
RSPAC
SEC
RTS

SEND
PHA
SEND1
LDA SCR
AND #$DRE
BNE SEND1
PLA
EOR #$FF
STA SDR
RTS

RBLK
LDA CSR
AND #$HRY
BNE RBLK
LDA #164
STA WDC
LDA #ADL
STA CDR
JSR RWORD
BCS RBLK2
STA RFCNT
JSR RWORD
BCS RBLK2
STA RFCNT+1
JSR RWORD
BCS RBLK2
STA BLKCNT
JSR RWORD
BCS RBLK2
STA BLKCNT+1
LDT #0
RBLK1
JSR RWORD
BCS RBLK2
STA BUF1,Y
IN:
CPY #160
BNE RBLK1
JMP COMDA2
RBLK2
JMP COMDA4

COMDA
PHA
LDA ESR
COMDA3
LDA CSR
AND #$MRY
BNE COMDA1
PLA
STA CDR
COMDA2
LDA ISR
AND CCE
BEO COMDA2
COMDA4
LDA CSR
PHA
AND #2
BEO COMDA5
LDY #$ERR1-MO
JSR MESS
PLA
SEC
RTS
COMDA5
PLA
AND #$81
BNE COMDA3
CLC
RTS
COMDA3
LDY #$MO-MO
PHA
JSR MESS
PLA
JSR NUMA
LDA ESR
JSR NUMA
CLC
RTS
COMDA3
LDY #$MO-MO
PHA
JSR MESS
PLA
JSR NUMA
LDA ESR
JSR NUMA
CLC
RTS

RWORD
LDA ISR
BIT CCE
BNE RWORD2
BIT DA
BEO RWORD
LDA DBR
CLC
RTS
RWORD2
SEC
RTS
END
A.9.3) DATA RECEIVE

Description: This program is written on the Hewlett Packard 9825 of Cessna Aircraft Company. The program receives data from its RS232 port. The program is the same as the one of Appendix A.3 of Reference 2.

Program listing:

```
0: "COMP. TERMINAL WITH CONTROL KEYS AND AUTO FUNCTIONS trk0;file5":
1: fmt 1,c1,x;1+1
2: sfg 2
3: 11+Q
4: wtc Q,1
5: wtb Q,37
6: 0+J;1+K
7: wtc Q,0
8: dim L$[106],C$(0:255),D$(300,30),A$(80),Q$(100)
9: if H=1;"="16
10: qsb "string"
11: oni Q,"in"
12: eir Q,4
13: buf "in",L$,1
14: tfl Q,"in",1
15: on key "KEY"
16: "15":if flg7;trk 0;scfg 7;scfg 6
17: if flg6;for L=1 to 203;for N=1 to 70;num(D$(L,N,1)+P);wtb Q,P
18: if flg5:char(P)+A$(Z12,Z)
19: if flg4;disp A$(max(1,1,len(A$)-31),max(32,len(A$)));next N;O+Z;wtb Q,13
20: if flg6;next L;scfg 6;scfg 7
21: gto 16
22: "KEY":key=C;if C=0;kret
23: if C=66 or C=194;scfg 9;0+C;kret
24: if flg9 and C[C]=64 and C[C]<90;wtb Q,C[C]=64;scfg 9;kret
25: if flg9;disp "NOT A VALID CONTROL KEY",C[C];cfg 9;0+C;kret
26: if Z=99;13+C
27: if C[C]=1000;scfg 7;0+C;kret
28: if C[C]=1001;scfg 2;0+C;4+J;1+K;kret
29: if C[C]=1002;scfg 2;0+C;kret
30: wtb Q,C[C]=P;char(P)+A$(Z12,Z)
31: disp A$(max(1,1,len(A$)-31),max(32,len(A$)))
32: if C[C]=13;"="+A$;0+Z;disp A$
33: kret
34: "in"
35: if flg2;L$[1,1]+Q[S,K];K+1+K;disp (J+1)/5
36: if flg2 and K=31;L$[1,30]+D$(J+1);J+1+J;K;"="+Q$
37: buf "in"
38: if C=7 or C=135;gub "break"
39: eir Q,4
40: tfl Q,;"in",1
41: kret
42: "string":
43: for I=1 to 56
44: I+C[I]
45: next I
46: for I=78 to 87
47: I-30+C[I]
48: next I
49: for I=88 to 96
50: 32+C[I]
51: next I
52: for I=97 to 122
53: I+C[I]
54: next I
55: for I=123 to 175
```

175
A.9.4) DATA PLOTTING

Description: This program is used by the Hewlett Packard 9825 computer of Cessna Aircraft Company to convert and plot out the flight test results. (Sample plots are presented as Figures 7.6.)

Program listing:

```
0: "FNG UNITS CON (quick look data plt) trkl;file 6":goto "START":0+Y+Z
1: "CON":Z+1+Z
2: for K=1 to 29 by 2:num(D$[K]=U)num(D$[K+1]=U
3: (K-1)/2+1
4: shf(U,-4)+U;band(U,15)+U;ior(U,V)=U
5: if bit(11,U);ior(U,-4096)=U
6: .06526=.06229793+M1H
7: next K;ret
8: "START":706+R
9: dim F[2],D[30],M[15]
11: ent "FIRST POSITION OF DYNAMIC DATA",r5,"LAST POSITION OF DYNAMIC",r6
12: 105/2+1+F1[1];106/2+F2[1];idx 1
13: ent "TRK @ TEMP DATA ";T;ent f1g13;goto +4
14: ent "FILE @?",F
15: disp "tape CONTINUE";step
16: trk T;ldf F,D
17: "C":0+D+R+B;cfg 1.2
```

176
APPENDIX B
TRANSFORMATION OF AXES SYSTEMS

This appendix shows the correlation between several axes systems.

Much information contained in this section is taken directly from Reference 28, which deals in depth with the problem of the different axes systems used in airplane analysis.
There are primarily five axes systems used in airplane analysis. These are described here.

1) Body Axes

"The orthogonal body-axes system is fixed within the vehicle with the X-axis along the longitudinal center line of the body, the Y-axis normal to the plane of symmetry, and the Z-axis in the plane of symmetry. This is the axes system about which aircraft instruments are usually mounted. Its main advantage in motion calculations is that vehicle moments of inertia about the axes are constant, so that the terms can be omitted from the equations of motion. It is the logical system to which to refer velocities, accelerations, and stability and control parameters in the study of aircraft handling qualities because the pilot's orientation with respect to this frame is fixed."* (This is the axes system used in this report.)

2) Principal Axes

"The principal axes are an orthogonal body-fixed system for which the products of inertia are zero. The X and Z principal axes lie in the plane of symmetry; the angle between the X body axis and the X principal axes is usually small so that in many cases the body axes can be assumed to coincide with the principal axes."*

*From Reference 28.
3) Flight Stability Axes

"The flight stability axes (sometimes referred to as vehicle stability axes) are an orthogonal body-axes system fixed to the vehicle, the X-axes of which is alined with the relative wind vector when the vehicle is in a steady-state trim condition but then rotates with the vehicle after a disturbance as the vehicle changes angle of attack. This system is preferred in many stability studies because, as with other body-fixed axes, the moments of inertia about the axes remain constant and also because the motions defined are prime-ily those about the flight path rather than about body reference lines."*

(This is the axes system used in Reference 22.)

4) Wind-Tunnel Stability Axes

"The wind-tunnel stability axes are the system about which most wind-tunnel data are obtained. For this system the X-axis is in the same horizontal plane as the relative wind at all times . . . . The angle $\alpha$ between the X-axis of this system and the X-body axes is variable. (It is a constant $\alpha_0$ for the flight stability axes.) This means that vehicle moments of inertia about the X-axis change. It also means that additional terms are required in the transformation equations for static-stability derivatives and for $u,v,w$ derivatives when data are transferred to or from the wind axes or the wind-tunnel stability axes."*

*From Reference 28.
5) Wind Axes

"The wind axes are the system generally used in calculating motions of the vehicle as a point mass. The X-axis for this system is aligned with the relative wind at all times so that vehicle moments of inertia about this axis change. As with the wind-tunnel stability axes, additional terms . . . are required in the transformation to or from the wind axes and either the body, principal, or flight stability axes, since the angle . . . between the X wind axis and the X-axis of either of these systems is variable. Also, since the lateral angle . . . between the X-axes is variable, there are additional terms . . . required in the transformations for some of the lateral derivatives between the wind axes and either of the other axes systems."*

The correlation between these axes systems is perhaps best summarized by Table B.1.

Table B.1 Designation of Force and Moment Coefficients for Different Axes Systems*

<table>
<thead>
<tr>
<th>Component</th>
<th>Coefficients for axes system</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Body or principal</td>
</tr>
<tr>
<td>X-axis force</td>
<td>$C_X$ or $-C_A$</td>
</tr>
<tr>
<td>Y-axis force</td>
<td>$C_Y$</td>
</tr>
<tr>
<td>Z-axis force</td>
<td>$C_Z$ or $-C_N$</td>
</tr>
<tr>
<td>X-axis moment (roll)</td>
<td>$C_{\ell}$</td>
</tr>
<tr>
<td>Y-axis moment (pitch)</td>
<td>$C_m$</td>
</tr>
<tr>
<td>Z-axis moment (yaw)</td>
<td>$C_n$</td>
</tr>
</tbody>
</table>

*From Reference 28.
Transformation from the flight stability axes (as used in Reference 22) to the body axes used in this report involves accounting for the steady-state angle of attack ($\alpha_1$). The following equation takes care of this by correcting the inertias. This is the only change required.

\[
\begin{bmatrix}
I_{xx,s} & I_{yy,s} & I_{xz,s} \\
I_{yz,s} & I_{zz,s} & I_{y} \\
I_{zx,s} & I_{xy,s} & I_{x} \\
\end{bmatrix} =
\begin{bmatrix}
\cos^2\alpha_1 & \sin^2\alpha_1 & (-)\sin^2\alpha_1 & 0 \\
\sin^2\alpha_1 & \cos^2\alpha_1 & \sin^2\alpha_1 & 0 \\
\frac{1}{2}\sin^2\alpha_1 & (-)\frac{1}{2}\sin^2\alpha_1 & \cos^2\alpha_1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
I_{xx} \\
I_{zz} \\
I_{xz} \\
I_{yy}
\end{bmatrix}
\]

[B.1]

NOTE: "s" denotes stability axes; no subscript denotes body axes.

NASA Langley (Reference 30) and NASA Dryden (References 12-16) both use the body axes system. They both, however, use different designations. NASA Langley uses the X, Y, Z, \(i\), \(m\), n designation; NASA Dryden, the A, Y, N, \(i\), \(m\), n designation. The parameters will be presented in the X, Y, Z, \(i\), \(m\), n system in this report. Table B.2 shows the correlation between both these systems.

The symbols (i.e., $Z_{a'}$, etc.) in the definition column of Table B.2 are those as predicted by the MILE "BONES" program. For conversion from normal stability parameters (as per Reference 22) to these state vector derivatives, the reader is referred back to Tables 5.2 and 5.3.

For rigorous conversion between the various axes systems, the reader is referred to Reference 28.
### Table B.2 Comparison of Non-Dimensional Derivatives

<table>
<thead>
<tr>
<th><strong>LONGITUDINAL</strong></th>
<th><strong>LATERAL</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>KU-FRL</strong></td>
<td><strong>KU-FRL</strong></td>
</tr>
<tr>
<td>NASA Langley</td>
<td>NASA Langley/</td>
</tr>
<tr>
<td>designation</td>
<td>Dryden</td>
</tr>
<tr>
<td>designation</td>
<td>designation</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;2&lt;/sub&gt;</td>
<td><strong>L</strong>&lt;sub&gt;p&lt;/sub&gt;</td>
</tr>
<tr>
<td>−&lt;sub&gt;C&lt;/sub&gt;</td>
<td>&lt;sub&gt;C&lt;/sub&gt;</td>
</tr>
<tr>
<td>Z&lt;sup&gt;2&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>L&lt;sub&gt;p&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;</td>
<td><strong>L</strong>&lt;sub&gt;r&lt;/sub&gt;</td>
</tr>
<tr>
<td>−&lt;sub&gt;C&lt;/sub&gt;</td>
<td>&lt;sub&gt;C&lt;/sub&gt;</td>
</tr>
<tr>
<td>Z&lt;sup&gt;u&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>L&lt;sub&gt;r&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;z&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
<td><strong>L</strong>&lt;sub&gt;6&lt;/sub&gt;</td>
</tr>
<tr>
<td>−&lt;sub&gt;C&lt;/sub&gt;</td>
<td>&lt;sub&gt;C&lt;/sub&gt;</td>
</tr>
<tr>
<td>Z&lt;sup&gt;6&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>L&lt;sub&gt;6&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;z&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>L</strong>&lt;sub&gt;6&lt;/sub&gt;</td>
</tr>
<tr>
<td>−&lt;sub&gt;C&lt;/sub&gt;</td>
<td>&lt;sub&gt;C&lt;/sub&gt;</td>
</tr>
<tr>
<td>Z&lt;sup&gt;0&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>L&lt;sub&gt;6&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;q&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;q&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;q&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;q&lt;/sub&gt;</td>
</tr>
<tr>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;r&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;r&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;r&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;r&lt;/sub&gt;</td>
</tr>
<tr>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
</tr>
<tr>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
<td>M&lt;sub&gt;u&lt;/sub&gt; U&lt;sub&gt;y&lt;/sub&gt;y</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;3&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;3&lt;/sub&gt;</td>
</tr>
<tr>
<td>X&lt;sup&gt;3&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>X&lt;sup&gt;3&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td>X&lt;sup&gt;0&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>X&lt;sup&gt;0&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;E&lt;/sub&gt;</td>
</tr>
<tr>
<td>X&lt;sup&gt;E&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>X&lt;sup&gt;E&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
<td><strong>C</strong>&lt;sub&gt;u&lt;/sub&gt;&lt;sub&gt;0&lt;/sub&gt;</td>
</tr>
<tr>
<td>X&lt;sup&gt;0&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
<td>X&lt;sup&gt;0&lt;/sup&gt;mul&lt;sub&gt;1&lt;/sub&gt;</td>
</tr>
<tr>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
<td>q&lt;sub&gt;1&lt;/sub&gt; S</td>
</tr>
</tbody>
</table>