



TEST RISK ASSESSMENT PROGRAM (TRAP): VERSION 2.1

by

Roger P. Bligh
Assistant Research Engineer

Hayes E. Ross, Jr.
Research Engineer

Brian Epperson
Capsher Technology, Inc.

and

Kevin Sherry
Capsher Technology, Inc.

Contract No. DTFH61-94-X-00011
Research Report No. 472540-F

Sponsored by
U.S. Department of Transportation
Federal Highway Administration

JANUARY 2000

**TEXAS TRANSPORTATION INSTITUTE
THE TEXAS A & M UNIVERSITY SYSTEM
COLLEGE STATION, TEXAS 77843-3135**

Test Risk Assessment Program (TRAP):

Version 2.1

by

Roger P. Bligh and Hayes E. Ross, Jr.
Texas Transportation Institute
The Texas A&M University System
College Station, Texas
and
Brian Epperson and Kevin Sherry
Capsher Technology, Inc.
Bryan, Texas

Final Report

Agreement No. DTFH61-94-X-00011
Federal Highway Administration

Texas A&M Research Foundation
Research Report No. 472540-F

January 2000

| | | | |
|--|---|--|--|
| 1. Report No. FHWA-RD-99-182 | 2. Government Accession No. | 3. Recipient's Catalog No. | |
| 4. Title and Subtitle Test Risk Assessment Program (TRAP): Version 2.1 | | 5. Report Date January 31, 2000 | 6. Performing Organization Code |
| 7. Author(s) Roger P. Bligh and Hayes E. Ross, Jr., TTI, and Brian Epperson, and Kevin Sherry, Capsher Technology, Inc. | | 8. Performing Organization Report No. Report No. 472540-F | |
| 9. Performing Organization Name and Address Texas Transportation Institute The Texas A&M University System College Station, Texas 77843-3135 and Capsher Technology, Inc. Bryan, Texas 77802 | | 10. Work Unit No. (TRAIS) | 11. Contract or Grant No. Agreement No. DTFH61-94-X-00011 Project No. RF 472540 |
| 12. Sponsoring Agency Name and Address U. S. Department of Transportation Federal Highway Administration 400 Seventh Street S.W. Washington, D.C. 20590 | | 13. Type of Report and Period Covered Final Oct. 1994 – Oct. 1999 | 14. Sponsoring Agency Code |
| 15. Supplementary Notes Project Managers: Charles F. McDevitt, FHWA and Mort Oskard, FHWA Research Project Title: "Standard Software for Analyzing Crash Test Data" | | | |
| 16. Abstract A user-friendly, Windows based computer program, Test Risk Assessment Program (TRAP) was developed to compute and output occupant risk parameters used in evaluating crash test results of highway safety features. Risk parameters are those contained in the National Cooperative Highway Research Program (NCHRP) Report 350, "Recommended Procedures for the Safety Performance Evaluation of Highway Features, dated 1993, and in the Committee for European Normalization (CEN) standard EN 1317-1, "Road Restraint Systems – Part 1: Terminology and General Criteria for Test Methods," dated April 1998. This report includes a User's Manual and a Programmer's Guide to facilitate use, maintenance, and updating of the program. | | | |
| 17. Key Words CEN, computer, crash, manual, NCHRP, occupant, programmer, risk, software, standard, test, user | | 18. Distribution Statement No restrictions. This document is available to the public through NTIS: National Technical Information Service 5285 Port Royal Road Springfield, Virginia 22161 | |
| 19. Security Classif.(of this report) Unclassified | 20. Security Classif.(of this page) Unclassified | 21. No. of Pages 60 | 22. Price |

Table of Contents

| | |
|---|------------|
| STANDARD SOFTWARE FOR ANALYZING CRASH TEST DATA | 1 |
| INTRODUCTION | 1 |
| OBJECTIVES | 1 |
| APPROACH | 2 |
| PRODUCT | 3 |
| APPENDIX A: TRAP USER'S MANUAL | APPENDIX A |
| TABLE OF CONTENTS | A-v |
| APPENDIX B: TRAP PROGRAMMER'S GUIDE | APPENDIX B |
| TABLE OF CONTENTS | B-v |

STANDARD SOFTWARE FOR ANALYZING CRASH TEST DATA

Introduction

Guidelines for evaluating the impact performance of a highway safety feature in the United States (US) are contained in National Cooperative Highway Research Program (NCHRP) Report 350, "Recommended Procedures for the Safety Performance Evaluation of Highway Features," dated 1993. Standards for evaluating the impact performance of a highway safety feature in the European Union (EU) are contained in Committee for European Normalization (CEN) standard EN 1317-1, "Road Restraint Systems - Part 1: Terminology and General Criteria for Test Methods," dated April 1998. As part of the evaluation process for a given crash test, "occupant risk" parameters are calculated and reported in accordance with Report 350 and standard EN 1317-1 requirements.

An effort is being made to move toward harmonized testing and evaluation of highway safety features between the US, the EU, and other countries. One area this can be accomplished in is the evaluation of occupant risk from a crash test. In fact, there are some distinct similarities between the occupant risk criteria used in Report 350 and standard 1317-1, namely in the computation of occupant impact velocity (OIV) and ridetop acceleration (RA) in Report 350, and in the computation of the theoretical head impact velocity (THIV) and the post-head deceleration (PHD) of standard 1317-1. Most testing agencies in the US and in the EU are computing and reporting both sets of occupant risk criteria.

Computational techniques used to determine occupant risk criteria of Report 350 and standard 1317-1 can vary among testing agencies. The way the results are reported can also vary. A standardized computational and reporting methodology is therefore needed for use among the various US and EU testing agencies. To this end, the US Department of Transportation, Federal Highway Administration (FHWA), contracted with the Texas Transportation Institute, Texas A&M University System, through a Cooperative Agreement to develop a standardized software package that computes and reports both sets of occupant risk criteria.

Objectives

The objective of this cooperative agreement was to develop a standardized code for computing occupant risk that can:

- (a) compute occupant risk measures in accordance with the recommendations in NCHRP Report 350,
- (b) compute occupant risk measures in accordance with CEN standard EN 1317-1, and
- (c) launch standardized codes developed by the National Highway Transportation Safety Administration (NHTSA) and by FHWA to prepare tapes and diskettes of crash test electronic data.

Approach

Initially an effort was made to write the required code around an existing, commercially available software package called DADiSP, a code used by some of the testing agencies. Protracted studies were made to adapt DADiSP for the intended purpose and functionality. When this was unsuccessful, an attempt was made to write a user friendly, Windows based, code for inputting necessary data to DADiSP. However, the functionality provided by DADiSP version 4.0 for Windows was still limited. Specifically, features needed to implement the windows oriented "user friendly" interface developed to support the standard software package were not robust enough to permit implementation within the DADiSP environment. After considerable time and effort, it was determined that the best course of action was to abandon the DADiSP software platform and develop an independent, Windows-based application that would possess all of the desired functionality. This new approach necessitated reprogramming of the occupant risk algorithms, development of independent plotting routines to permit graphical output to both the screen and printer, and the creation of new documentation.

The user-friendly software utilizes the Windows 95/98 or NT operating system, and requires no special proprietary software. A presentation of the Alpha version was made at the 1998 Annual TRB meeting, to demonstrate the program, its use, and the output options. The program was well received by the testing community, both domestically and internationally. Some comments were received at that time. In addition, an internal review revealed the need to incorporate additional features, as follows:

- 1) Help screens were added.
- 2) Algorithms were added to calculate and report the maximum 50 msec average accelerations and the period over which they occur.
- 3) A setup code to install program was added.

Distribution of the beta version of the program "TRAP" (Test Risk Assessment Program) was made during the summer of 1998 to interested test laboratories and to the consultant to TTI, Dr. Malcolm Ray, for review. Also, an overview of the program was presented to attendees at the summer meeting of A2A04, Subcommittee A2A04 (02), International Research Activities, held in Seattle, Washington, August 2-5, 1998. Disk copies of the program and a draft user's manual were distributed to interested parties.

As a result of the internal and external reviews, further refinements to TRAP were made, as follows:

- 1) Algorithms were added to calculate vehicular roll, pitch, and yaw angles as functions of time, given roll, pitch, and yaw rates.

- 1) Algorithms were added to calculate vehicular roll, pitch, and yaw angles as functions of time, given roll, pitch, and yaw rates.
- 2) Reporting of the maximum roll, pitch, and yaw angles and time of their occurrence was added to the Output Report summary page.
- 3) Graphical output was added to provide for plotting of roll, pitch, and yaw angles and roll, pitch, and yaw rates.

The resulting version of the TRAP software (revision 2.0) was distributed on CD ROM during the summer of 1999 to domestic and international test laboratories and other interested parties. An overview of the program was presented to attendees of the summer meeting of TRB Committee A2A04, Subcommittee A2A04(2), International Research Activities, held in Estes Park, Colorado, August 1-4, 1999.

Product

The end product of this agreement consists of the TRAP software code, a TRAP User's Manual, and a TRAP Programmer's Guide. The code is available on disk, and it includes a setup code for installing the program. The User's Manual describes the manner in which TRAP is used. It includes a description of input requirements and output options. The User's Manual is attached hereto, as Appendix A. The Programmer's Guide is intended for programmers needing to maintain and/or extend TRAP's capabilities. It is attached hereto as Appendix B.

NOTE: **IT IS ANTICIPATED THAT UPDATES TO TRAP WILL BE MADE FROM TIME TO TIME. THE USER SHOULD REFER TO THE "README" FILE IN THE ELECTRONIC VERSION OF TRAP FOR THE LATEST CHANGES.**

APPENDIX A

TRAP USER'S MANUAL

User's Manual



Test Risk Assessment Program (TRAP) Version 2.1

January 2000

Copyright ©Texas Transportation Institute 1997
Developed by CAPSHER Technology, Inc.
Info@capsher.com

Table of Contents

| | |
|---|------|
| INTRODUCTION..... | A-1 |
| USING TRAP | A-2 |
| DATA INPUT..... | A-4 |
| DATA INPUT - DESCRIPTION PAGE..... | A-5 |
| DATA INPUT - SETTINGS PAGE..... | A-6 |
| DATA INPUT - ACCELERATION AT CG PAGE..... | A-7 |
| DATA INPUT - ACCELERATION 1 PAGE..... | A-8 |
| DATA INPUT - ACCELERATION 2 PAGE..... | A-9 |
| DATA INPUT - ANGULAR RATES PAGE..... | A-10 |
| DATA INPUT - YAW ANGLE PAGE..... | A-11 |
| ASCII DATA FILES AND BINARY DATA FILES..... | A-12 |
| PERFORMING CALCULATIONS | A-13 |
| RESULTS | A-14 |
| GRAPHS..... | A-15 |
| GRAPH PARAMETERS FORM - GRAPH PAGE..... | A-16 |
| GRAPHS PARAMETERS FORM - PLOTS PAGE..... | A-17 |
| GRAPH PARAMETERS FORM - AXES PAGE..... | A-18 |
| TABLES..... | A-19 |
| MENU AND TOOLBAR REFERENCE..... | A-20 |
| FILE MENU..... | A-20 |
| EDIT MENU | A-21 |
| DATA MENU | A-21 |
| GRAPH MENU..... | A-22 |
| TABLE MENU | A-22 |
| WINDOW MENU | A-23 |
| TOOLBAR | A-24 |

intervals through an interaction formula using defined limits for the x, y, and z accelerations. The maximum value thus obtained is reported as the ASI.

Also computed and reported are maximum 50 ms average accelerations in the x, y, and z directions. Although not required in either the NCHRP Report 350 or the CEN evaluations, some agencies use these values as a measure of performance.

Also, if the combination of roll, pitch, and yaw rates are input, the program computes roll, pitch, and yaw angles with time in accordance with coupled Euler equations. Maximum roll, pitch, and yaw angles are then calculated and output on the "Results" page. The sequence for determining vehicular orientation is as follows: yaw, then pitch, then roll. The user may also plot roll, pitch, and yaw angles with time. To be noted is that in accordance with CEN requirements, the yaw angle data used in calculating THIV is determined based on the assumption that yaw angle is uncoupled (independent) from the roll and pitch angles. Hence, differences may occur between yaw angles computed by the two methods

For further details, see NCHRP report 350 and CEN document CEN/TC 266 N 185 E, PrEN1317-1 for detailed information on testing procedures, data acquisition requirements, and computing the standardized occupant risk factors.

Using TRAP

A good way to get familiar with the capabilities of **TRAP** is to open the sample **TRAP** data file provided. The installation program for **TRAP** automatically copies the SAMPLE.TRP file to a Samples subdirectory under the main **TRAP** installation directory (C:\Program Files\Trap by default).

Opening an existing **TRAP** data file and viewing the results consists of the following steps:

1. Opening an existing **TRAP** data file

Select the Open command from the File menu. This will bring up the standard File Open dialog box. Navigate to the directory where the **TRAP** data file exists, select the appropriate file, and click the OK button. NOTE: the sample file provided with **TRAP** (SAMPLE.TRP) file was installed in C:\Program Files\Trap\Samples by default

2. Running the calculations

After opening the SAMPLE.TRP file, **TRAP** will automatically run the calculations.

3. Viewing the results

After the calculations have run, the Test Summary report view will automatically come up showing a summary of the occupant risk factors calculated. You can also view graphs and tables generated by **TRAP** by selecting which graph or table you wish to see from the Graphs or Tables menu.

4. Viewing and/or modifying the Input Data (optional)

After opening an existing **TRAP** data file, you can view and/or modify the input data by selecting the Input Data command from the Data menu. This will bring up the Data Input form. If you modify any of the input data and click the OK button, **TRAP** will automatically rerun the calculations. Clicking the Cancel button on the Data Input form will discard any modifications you have made to the input data and will not rerun the calculations.

Introduction

Test Risk Assessment Program (**TRAP**) is a program designed to determine the effectiveness of a roadside safety feature by analyzing data from a vehicle crash test of the feature and calculating standardized occupant risk factors.

TRAP calculates occupant risk factors in accordance with the National Cooperative Highway Research Program (NCHRP) guidelines and the European Committee for Standardization (CEN).

NCHRP Report 350 recommends two risk factors to be computed: (1) the lateral and longitudinal components of the Occupant Impact Velocity (OIV) and (2) the highest lateral and longitudinal component of resultant vehicular acceleration averaged over any 10-ms interval subsequent to occupant impact also known as the Occupant Ridedown Acceleration (ORA).

These factors are determined by use of a "flail space" model, details of which were first presented in NCHRP Report 230. The occupant is idealized as a lumped mass, constrained within an idealized occupant compartment. Planar motion in the horizontal plane is assumed. When vehicular impact occurs the occupant moves freely within the compartment until contact is made with either the side or the front of the compartment. After contact, whether with the sides or front of the compartment, the occupant is assumed to stick and remain in contact with the compartment during the remainder of the event. As such, the occupant then experiences the same acceleration as does the vehicle, and these are referred to as "ridgedown" accelerations.

Contact with the compartment occurs when the occupant moves left or right 0.3 m (y direction) or forward 0.6 m (x direction). The following procedure is used to calculate the risk factors of NCHRP Report 350:

For Criterion H and I, Table 5.1:

Vehicular accelerations in the x and y directions are double integrated, independently, to determine the time of first contact, either with the side of the compartment (0.3 m) or the front of the compartment (0.6m), whichever happens first. OIV in both the x and y directions are then computed at this time and reported. The larger value of the two is the OIV. Subsequent to contact, 10 ms moving average occupant ridetdown accelerations (ORA) in the x and y directions are computed and the largest values are reported. The larger value of the two is the ORA. A "linear acceleration," or "trapezoidal rule" integration procedure is used.

For Criterion L:

The above procedure is repeated, except the OIV in the x direction is selected at the time of contact (whether contact is with the side or front of the compartment), and the maximum ORA in the x direction subsequent to contact is selected.

CEN specifies three risk factors: (1) Theoretical Head Impact Velocity (THIV), (2) Post Impact Head Deceleration (PHD), and (3) the Acceleration Severity Index (ASI).

The following procedure is used to calculate the CEN risk factors:

CEN calculation procedures for THIV and PHD are similar to those used for OIV and ORA except that THIV equals the resultant of OIV components in the x and y directions at initial occupant contact and PHD equals the resultant of the ORA components in the x and y directions subsequent to initial occupant contact. There are two other differences: (a) vehicular yaw motions are accounted for in the CEN procedures but are neglected in the Report 350 procedures, and (b) the head (occupant) in the CEN procedures can be positioned forward of the vehicle's center of gravity prior to vehicular impact whereas the occupant is always assumed to be located at the vehicle's center of mass prior to vehicular impact in the Report 350 procedures.

Computation of the ASI involves the determination of moving 50 ms averages of vehicular accelerations in the x, y, and z directions. ASI is computed at each of these moving 50 ms

You can also use TRAP to analyze data from a new crash test.
Analyzing data from a new crash test with TRAP consists of the following steps:

1. Gathering the required data

In order for TRAP to compute all the results it is capable of calculating, it needs X, Y and Z vehicular accelerations and roll, pitch and yaw rates of the vehicle.

TRAP is capable of handling situations where all the data it needs is not available. In this case it will calculate what it can with the available data. See Performing Calculations for detailed information.

The acceleration data required consists of 3 individual components of acceleration (x,y, and z) measured by a triaxial accelerometer mounted at or near the vehicle's center of gravity.

Sometimes the triaxial accelerometer cannot be mounted at or near the vehicle's center of gravity, in which case two triaxial accelerometers must be mounted along the vehicle's longitudinal centerline and two sets of acceleration data are required as well as the distances the two accelerometers are from the vehicle's center of gravity.

In either case, all acceleration data must be recorded at the same uniform frequency and should have the same time duration.

The yaw data can be in the form of yaw rate recorded at the same frequency as the acceleration data, or yaw angle vs. time where the time must be explicitly specified and does not have to correspond to the points in time that accelerations are measured.

It is recommend that yaw rate be used if available.

The data recorded during the crash test can exist in one or more files. For example the x, y, and z components of acceleration could be stored as individual columns within the same file, or each could be stored in separate files. See ASCII data files and Binary data files for more information on the required format of the files.

2. Creating a new TRAP data file

Run the TRAP program and select the New command from the File menu.

This will bring up the Data Input form.

3. Entering the required data using the Data Input form

Fill in the fields on the Data Input form and then click OK.

4. Running the calculations

TRAP automatically runs the calculations after the OK button is clicked on the Data Input form. If some data has not been entered then some or all of the items shown on the report may be unavailable (marked N/A). In addition some or all the graphs and tables may be unavailable.

5. Reviewing the results

TRAP calculates the standardized occupant risk factors and displays them in the results report view.

TRAP can also display certain information in graphical or tabular format.

See Results for further information.

6. Saving the TRAP file.

You should save the TRAP file by selecting the Save command from the File menu before exiting the program.

Data Input

The data necessary for TRAP is entered by the user on the Data Input form. This form is a multipage form consisting of the following pages: Description Page, Settings Page, Acceleration Page(s), Angular Rates Page, and Yaw Angle Page

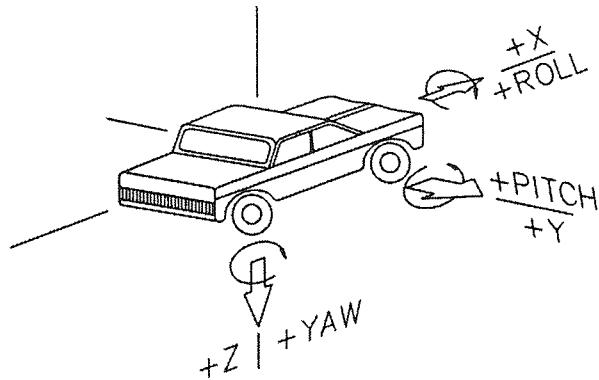
In general, you should step through each page entering the required data.

Once you have entered the data on all the pages, click OK

TRAP will automatically run the calculations.

See Running the Calculations and Viewing Results for more information.

TRAP uses the sign convention documented in NCHRP report 350 as shown in the figure below. It is very important that all input data (accelerations, angular rates and / or angles) uses this same sign convention.



Sign Convention used by TRAP

Data Input - Description Page

The screenshot shows a Windows-style dialog box titled "Data Input". The "Description" tab is selected. The form contains several sections with input fields:

- Test agency:** Texas Transportation Institute
- Test number:** 472480-1 **Test date:** 10/20/1997
- Test article:** G4(1S) W-beam Guardrail
- Test vehicle:**
 - Description:** 1992 Geo Tracker
 - Test inertial mass:** 982 kg
 - Gross static mass:** 1060 kg
- Impact conditions:**
 - Speed:** 99.7 km/h
 - Angle:** 19.74 degrees
- Distance of theoretical head from center-of-gravity:**
 - Along x-axis:** 0 mm

At the bottom right are buttons for **OK**, **Cancel**, and **Help**.

The Description page allows you to enter general descriptive information about the current test you are analyzing. Most of the data on this page is for reference only and not used in the calculations.

Test agency, Test number, Test date, Test article:

These fields contain descriptive information about the test and are for reference only. The values entered here will be shown on the Results report view.

Test vehicle:

The fields in this box are for reference only. They are not used in any calculations. The values entered here will be shown on the Results report view.

Impact conditions:

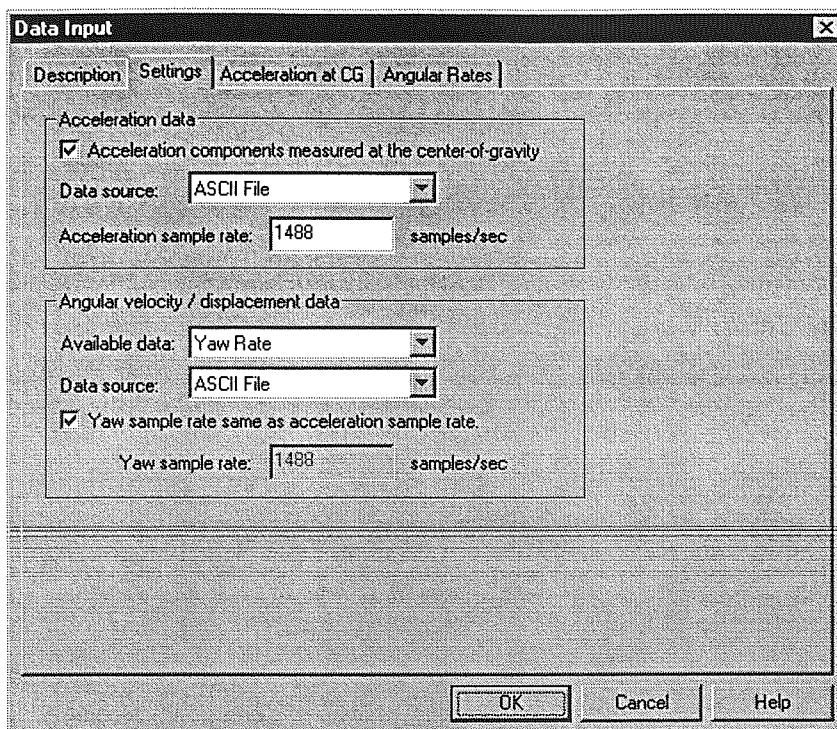
Specify the speed and angle of the impact. These values are for reference only. They are not used in any calculations. The calculations are all based on the relative motion of the occupant with respect to the vehicle's interior and are not effected by the impact speed or impact angle. The values entered here will be shown on the Results report view.

Distance of theoretical head from center-of-gravity:

The theoretical head used in determining the THIV and the PHD is assumed to be located at the vehicle's center of mass. Specify a non-zero value here only if the theoretical head is positioned forward of the vehicle's center of gravity. This value is used by the calculations.

Once you have filled the fields on this page, go on to the Settings page.

Data Input - Settings Page



The Settings page allows you to specify information about the files that contain data from the test.

Accelerations are measured at the C.G.

Check this box if the triaxial accelerometer recording the accelerations is mounted at or very near the center of gravity of the test vehicle.

If the accelerometer cannot be mounted at the vehicle's center of gravity, then two triaxial accelerometers must be used and mounted along the vehicle's longitudinal centerline. Note that two sets of X,Y, and Z acceleration data will be required in this case.

Data source:

Specify whether the input file is ASCII or Binary.

Acceleration sample rate

Specify the number of samples per second at which the acceleration data was recorded.

Available data:

Specify whether the angular data is Yaw Angle or Yaw Rate. Yaw rate is preferred. It is obtained by onboard instrumentation measuring yaw rate.

If you do not have yaw rate data available you can enter yaw angle vs. time instead and TRAP can calculate yaw rate. Yaw angle vs. time is sometimes obtained by measuring yaw angle from individual frames recorded by an overhead camera.

Data source:

Specify whether the input file is ASCII or Binary.

Yaw sample rate same as acceleration sample rate

Check this box if the yaw sample rate is the same as the sample rate used for the acceleration data.

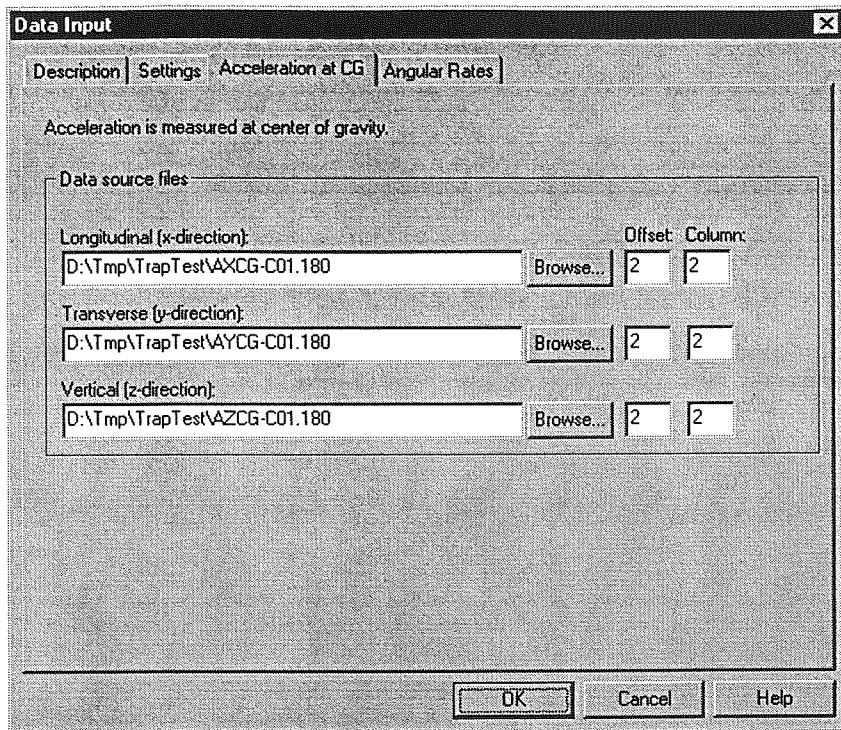
Yaw sample rate:

Specify the number of samples per second at which the yaw rate data were recorded.

Note: If roll and pitch rates are also input, the number of samples per second at which the roll and pitch rate data were recorded must equal that for the yaw rate data.

Once you have filled the fields on this page, go on to the Acceleration page(s).

Data Input - Acceleration at CG Page



This page allows you to specify information about the data source file(s) containing the vehicular acceleration data. This page will be shown if you specified that accelerations were measured at or near the vehicle's center of mass on the Settings page, otherwise the Acceleration -1 and Acceleration - 2 pages will be shown and you will have to enter in two sets of acceleration data.

Data source files:

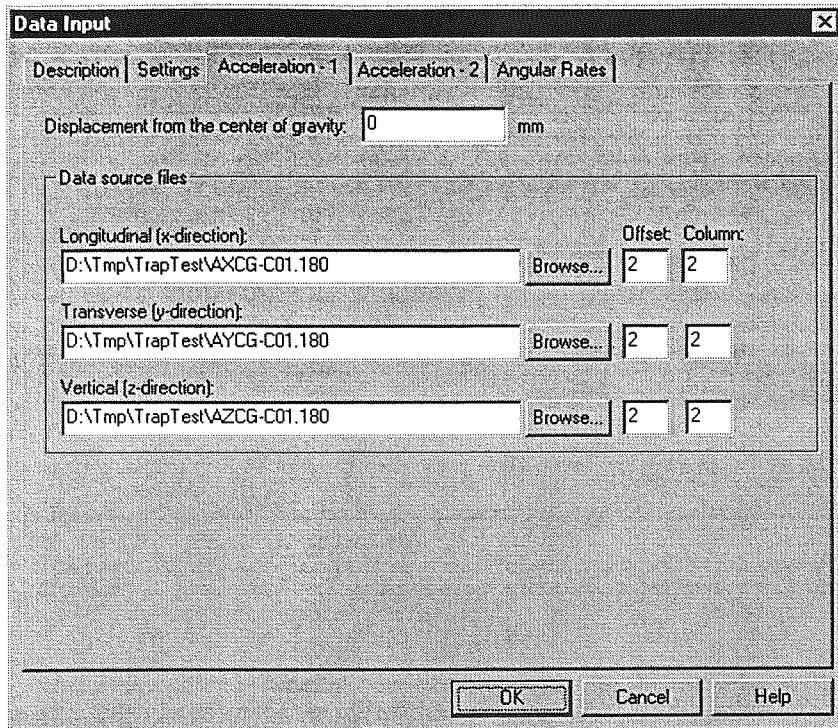
Specify file names for X, Y, and Z data files. The **Browse** button will allow you to browse for the files. NOTE: the acceleration data must be in G's.

Offset, Column

These fields allow you to specify the column number and the number of rows to skip (offset). See ASCII data files and Binary data files for more information on file formats.

Once you have filled the fields on this page, go on to the Angular Rates or the Yaw Angle Page.

Data Input - Acceleration 1 Page



This page allows you to specify information about the data source file(s) containing the vehicular acceleration data. This page will be shown if you specified that accelerations were not measured at or near the vehicle's center of mass on the Settings page, otherwise the Acceleration at CG page will be shown and you will only have to enter in one set of acceleration data.

Displacement from center of gravity:

Enter the displacement from the center of gravity for this measurement.

The value entered is the distance along the longitudinal axis of the vehicle (x-axis) from the vehicle's center of gravity to the accelerometer. Distances forward of the center of gravity are positive, and distances aft of the center of gravity are negative. The distance is required to be in millimeters.

NOTE: This field is only shown if the acceleration data was not measured at or near the vehicle's center of gravity.

Data source files:

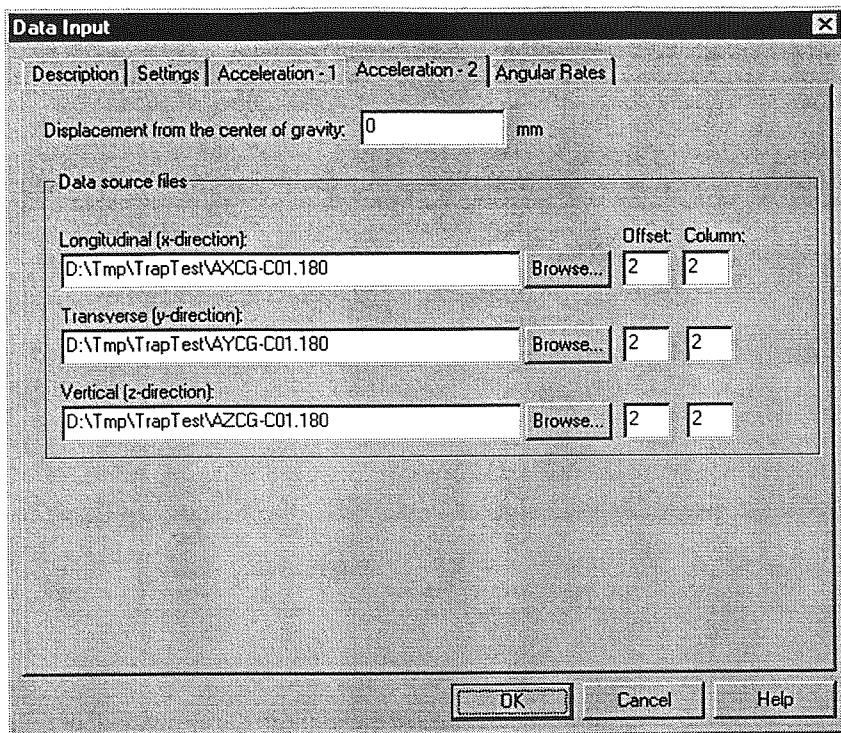
Specify file names for X, Y, and Z data files. The **Browse** button will allow you to browse for the files. NOTE: the acceleration data must be in G's.

Offset, Column

These fields allow you to specify the column number and the number of rows to skip (offset). See ASCII data files and Binary data files for more information on file formats.

Once you have filled the fields on this page, go on to the Acceleration - 2 Page.

Data Input - Acceleration 2 Page



This page allows you to specify information about the data source file(s) containing the vehicular acceleration data. This page will be shown if you specified that accelerations were not measured at or near the vehicle's center of mass on the Settings page, otherwise the Acceleration at CG page will be shown and you will only have to enter in one set of acceleration data.

Displacement from center of gravity:

Enter the displacement from the center of gravity for this measurement.

The value entered is the distance along the longitudinal axis of the vehicle (x-axis) from the vehicle's center of gravity to the accelerometer. Distances forward of the center of gravity are positive, and distances aft of the center of gravity are negative. The distance is required to be in millimeters.

NOTE: This field is only shown if the acceleration data was not measured at or near the vehicle's center of gravity.

Data source files:

Specify file names for X, Y, and Z data files. The **Browse** button will allow you to browse for the files. NOTE: the acceleration data must be in G's.

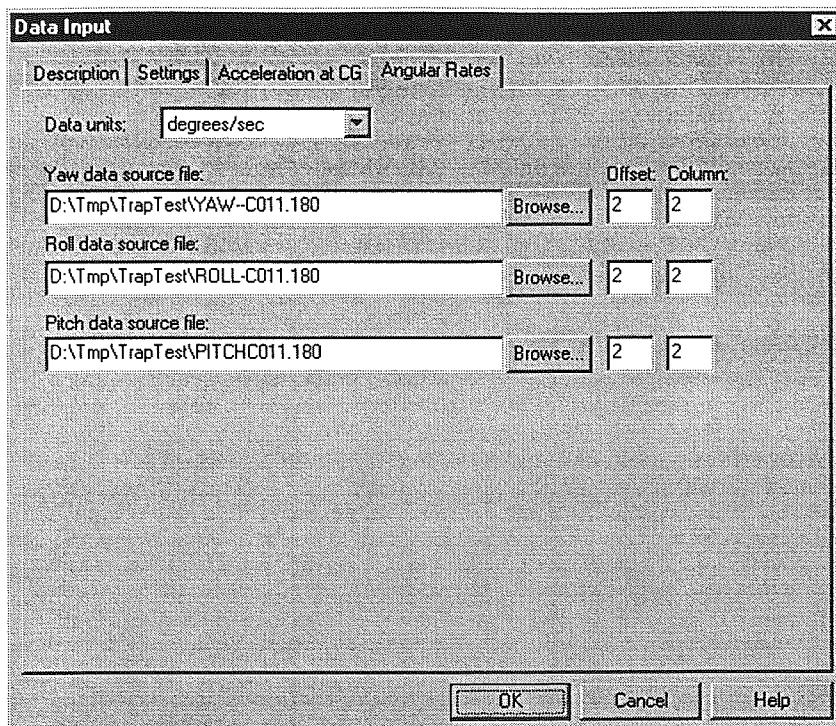
Offset, Column

These fields allow you to specify the column number and the number of rows to skip (offset).

See ASCII data files and Binary data files for more information on file formats.

Once you have filled the fields on this page, go on to the Angular Rates or the Yaw Angle Page.

Data Input - Angular Rates Page



The Angular Rates page allows you to enter yaw rate data of the vehicle vs. time. As an option, you may also enter roll and pitch rate data.

This page is only shown if you specified yaw rate on the Settings page. The Yaw Angle page is shown if you specified yaw angle on the Settings page.

Data units

Enter the units used for the angular rate data (degrees/sec or radians/sec)

Data source file:

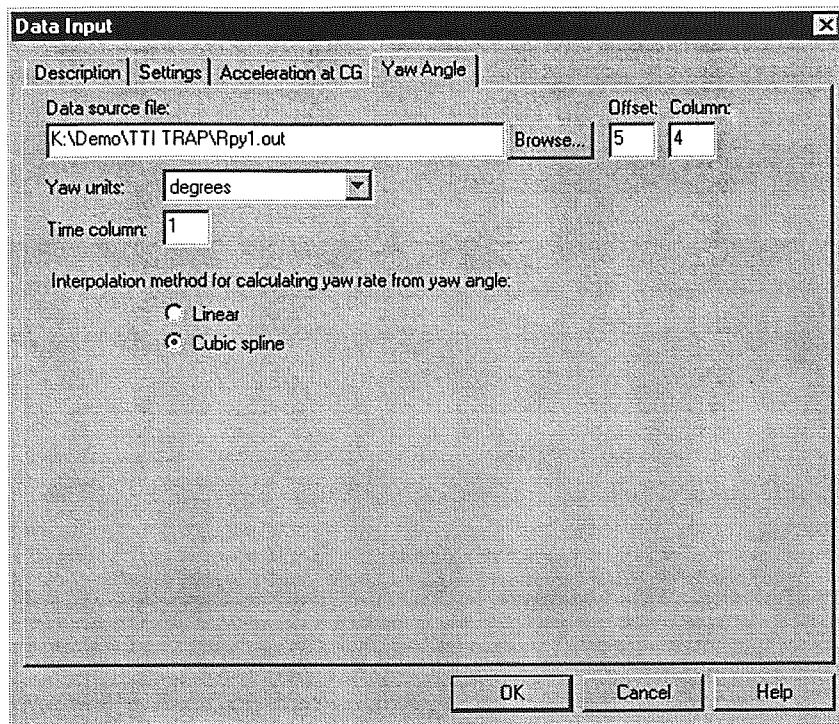
Specify file name for the file containing yaw rate data for the test you are analyzing. The **Browse** button will allow you to browse for the file. If roll and pitch rate data are input, specify the name for the file containing roll and pitch rate data for the test you are analyzing.

Offset, Column

These fields allow you to specify the column number and the number of rows to skip (offset). See ASCII data files and Binary data files for more information on file formats.

Once you have filled the fields on this page and the other pages of the Input Data and clicked the OK button, the program will ask if you would like to run the calculations now. Go on to Performing Calculations .

Data Input - Yaw Angle Page



The Yaw Angle page allows you to enter information about the yaw angle of the vehicle vs. time. This page is only shown if you specified yaw angle on the Settings page. The Yaw Rate page is shown if you specified yaw rate on the Settings page.

Data source file:

Specify file name for the file containing Yaw angle data for the test you are analyzing. The **Browse** button will allow you to browse for the file.

Offset, Column

These fields allow you to specify the column number and the number of rows to skip (offset). See ASCII data files and Binary data files for more information on file formats.

Yaw units

Enter the units used in the yaw angle data (degrees or radians)

Time column:

Enter the column containing the time parameter. NOTE: the time must be specified in seconds.

Interpolation method:

The calculation engine will interpolate values between data points using either Linear or Cubic Spline methods. The cubic spline method will generally provide greater accuracy.

Once you have filled the fields on this page and the other pages of the Input Data and clicked the OK button, the program will ask if you would like to run the calculations now. Go on to Performing Calculations .

ASCII Data Files And Binary Data Files

The acceleration and yaw data files can be in two formats: ASCII or binary. The file format you choose depends upon the data file produced by your data acquisition equipment.

An ASCII data file contains numbers represented as text. Values are arranged in rows and columns. In the following example, the ASCII data file is composed of two columns of data. The first two rows (lines) consist of header information are not part of the data. Therefore the program must be told that the real data starts at an offset of 2.

```
AC 1489 6.720E-04 0.000E+00 G's 0.000E+00 CLASS180
      "AXCG   "    0     20    -11    654    -0.6768721
0.0000      0.0615338
0.0007      0.0307669
0.0013      0.0923007
0.0020      0.1538346
0.0027      0.1846015
0.0034      0.0923007
0.0040      -0.0615338
0.0047      -0.1846015
0.0054      -0.2153684
```

Binary data files contain data in 32-bit float or 64-bit double, IEEE format. Since binary files have no new-line character, it must be told how many columns are in the file. For example, a data file containing 36 numbers could be 6 columns by 6 rows, 4 by 9, or 9 by 4. An offset can still be specified for a binary file if there is some information at the beginning of the file you wish to skip over. For binary files the offset is specified in bytes rather than rows(lines).

Performing Calculations

The program will automatically run the calculations if you have changed any information on the Input Data pages, or if the calculations have not been run yet.

Once the calculations have run, the Results will appear on screen. Some results may not be available if any of the required data needed to compute them is missing from the input data. Unavailable items will be marked N/A in the Test Summary Report. TRAP will compute as much as it can with the data it is given. The following table summarizes the data required to compute each item in the Test Summary Report.

| Computed Value | X Accel. | Y Accel. | Z Accel. | Yaw Rate | Pitch Rate | Roll Rate |
|--------------------|----------|----------|----------|----------|------------|-----------|
| OIV - X | X | | | | | |
| OIV - Y | | X | | | | |
| THIV | X | X | | X* | | |
| Ridedown Accel. X | X | | | | | |
| Ridedown Accel. Y | | X | | | | |
| PHD | X | X | | | | |
| ASI | X | X | X | | | |
| Max 50-msec Avg. X | X | | | | | |
| Max 50-msec Avg. Y | | X | | | | |
| Max 50-msec Avg. Z | | | X | | | |
| Max Roll angle | | | | X | X | X |
| Max Pitch angle | | | | X | X | X |
| Max Yaw angle | | | | X | X | X |

*As an alternative, yaw angle versus time may be input.

Results

Once the calculations have been performed, the Results window will show a Test Summary Report. To view this window later select Results from the Data Menu.

| Test Summary Report | | |
|--|--------------------------------|--|
| General Information | | |
| Test Agency: | Texas Transportation Institute | |
| Test Number: | 472480-1 | |
| Test Date: | 10/20/1997 | |
| Test Article: | G4(1S) W-beam Guardrail | |
| Test Vehicle | | |
| Description: | 1992 Geo Tracker | |
| Test Inertial Mass: | 982 kg | |
| Gross Static Mass: | 1060 kg | |
| Impact Conditions | | |
| Speed: | 99.7 km/hr | |
| Angle: | 19.7 degrees | |
| Occupant Risk Factors | | |
| Impact Velocity (m/s) | | at 0.1210 seconds on left side of interior |
| x-direction | 4.1 | |
| y-direction | -6.2 | |
| THIV (km/hr): | 22.2 | at 0.1159 seconds on left side of interior |
| THIV (m/s): | 6.2 | |
| Ridedown Accelerations (g's) | | |
| x-direction | -5.0 | (0.2193 - 0.2293 seconds) |
| y-direction | 9.9 | (0.2198 - 0.2298 seconds) |
| PHD (g's): | 11.3 | (0.1155 - 0.1255 seconds) |
| ASI: | 0.90 | (0.0988 - 0.1488 seconds) |
| Max. 50msec Moving Avg. Accelerations (g's) | | |
| x-direction | -4.7 | (0.0728 - 0.1228 seconds) |
| y-direction | 7.6 | (0.0995 - 0.1495 seconds) |
| z-direction | 1.7 | (0.1751 - 0.2251 seconds) |
| Max Roll, Pitch, and Yaw Angles (degrees) | | |
| Roll | -6.8 | (0.2510 seconds) |
| Pitch | 2.8 | (0.7500 seconds) |
| Yaw | 22.2 | (0.7500 seconds) |

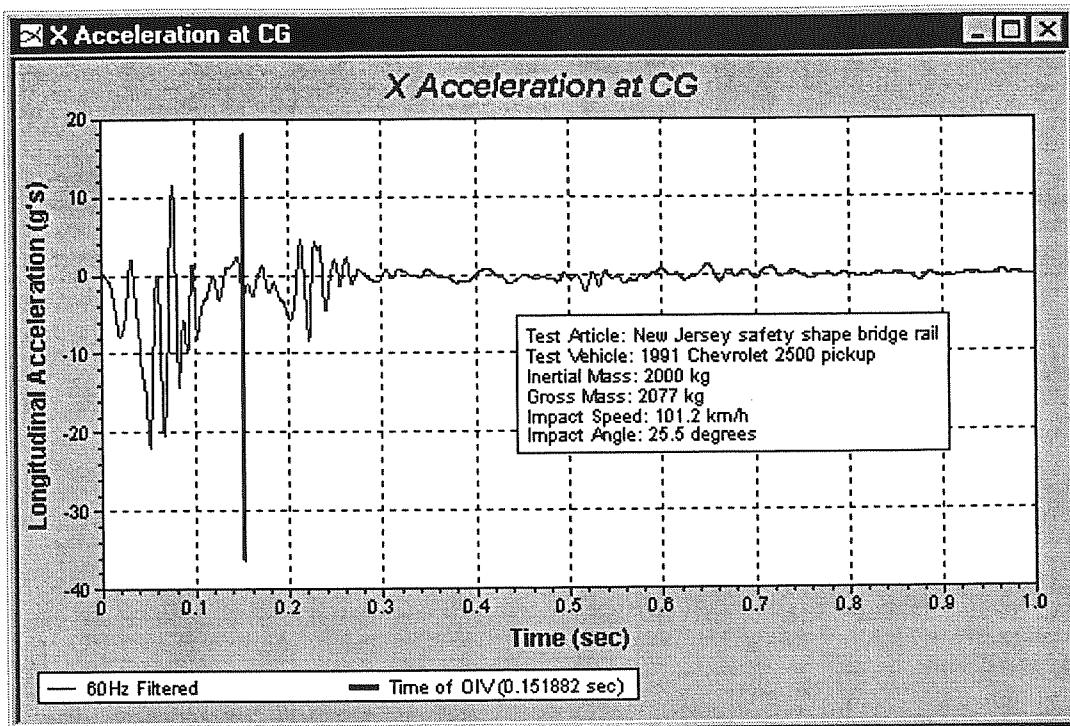
You can print these results by pressing the Print button on the toolbar, or by selecting Print from the File menu .

The Graphs menu allows you to generate several different graphs.

The Tables menu allows you to generate several different tables showing the same data that is available in the graphs, but in tabular format.

See Graphs and Tables for more information.

Graphs



To view the graphs, select **Graph All** from the Graph menu. You may also draw one graph at a time, by selecting from the choices in the Graph menu.

You can edit the properties of an individual graph. To edit a graph, select Properties from the Edit menu , or double-click on a graph.

Each graph has three sets of properties:

Graph : Controls general settings

Plots : Which plots to show on a certain graph

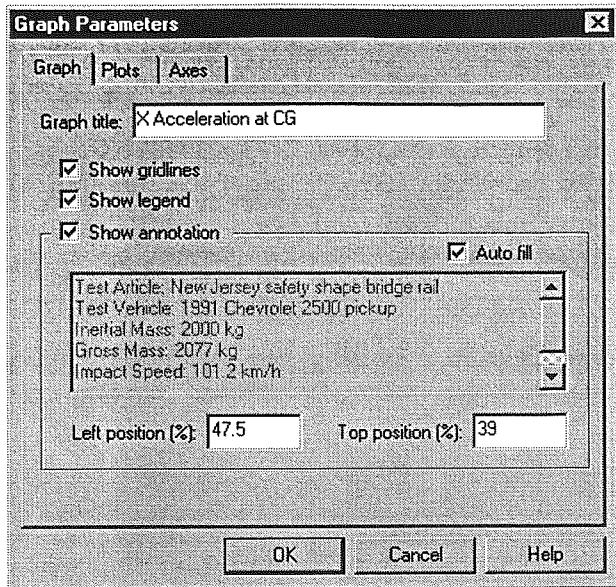
Axes : Handles axes scale settings

You may also arrange the positions of the annotation and legend in the graph by clicking on them and dragging them to the desired location.

Once you have a suitable graph, you may print it by pressing the Print button on the toolbar, or by selecting **Print** from the File menu . You may also copy an image of the graph to the clipboard by selecting the **Copy** command from the Edit menu .

NOTE: When copying graphs to the clipboard for pasting into other applications, you will get the best results by maximizing the size of the graph before performing the copy operation.

Graph Parameters Form - Graph Page



The Graph page of the Graph Parameters form has several options allowing you to control the appearance of a graph.

Graph title

Type in the title for the graph here.

Show gridlines

Check this box if you want gridlines displayed on the graph.

Show legend

Check this box if you want the legend displayed on the graph.

Show annotation

Check this box if you want the annotation to be displayed on the graph.

Auto fill

Check this box if you want TRAP to automatically fill in the annotation for you.

NOTE: This field is disabled unless the Show annotation box is checked.

Text box

If the Auto fill box is not checked, you can type in the text you want.

NOTE: This field is disabled unless the Show annotation box is checked.

If the Auto fill box is checked, this box displays the default text that will be shown in the annotation on the graph.

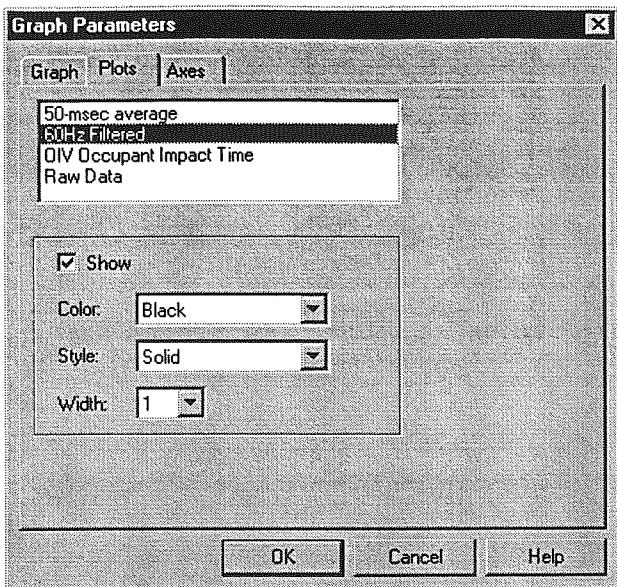
Left position, Right position

These fields allow you to control the positioning of the annotation

You may also drag the annotation by mouse when the Graph Settings window is closed.

NOTE: These fields are disabled unless the Show annotation box is checked.

Graphs Parameters Form - Plots Page



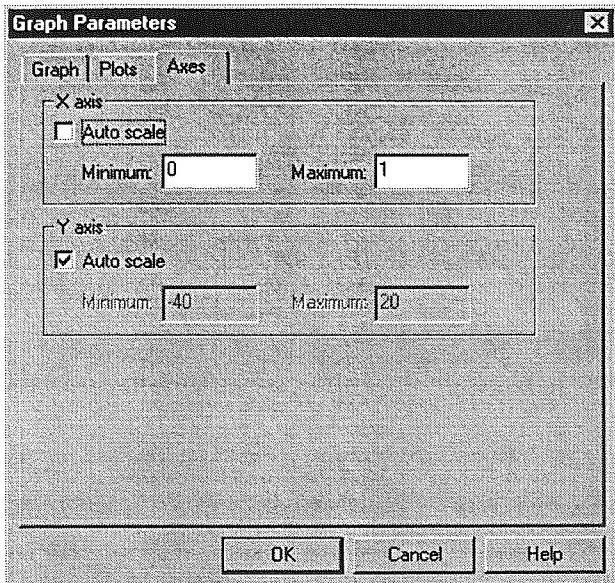
For some graphs, more than one plot(curve) may be shown on the same graph. This page allows you to control which plots are shown on a graph and the properties (line color, style, and width) of those plots.

The list box at the top of the page shows all plots available for the specific graph.

After selecting a plot in the list of plots, you may make it visible on the graph by checking the **Show** checkbox. Once a plot is visible, you may adjust the line color, style, and width.

NOTE: the Color, Style, and Width fields will all be disabled if the **Show** checkbox is not checked.

Graph Parameters Form - Axes Page



This page allows you to control the limits of the scale associated with each axis on a graph.

For each of the X and Y axes, you may use auto scale, or type in your own minimum and maximum values.

Tables

| Time (sec) | Raw (G's) | 60Hz Filtered (G's) | 50 msec Avg (G's) |
|---------------|--------------|------------------------|----------------------|
| 0.0000000 | 0.0615338 | 0.0370095 | -1.7401768 |
| 0.0006720 | 0.0307669 | 0.0381394 | -1.8509377 |
| 0.0013441 | 0.0923007 | 0.0319651 | -1.9539043 |
| 0.0020161 | 0.1538346 | 0.0154066 | -2.0511277 |
| 0.0026882 | 0.1846015 | -0.0141964 | -2.1278399 |
| 0.0033602 | 0.0923007 | -0.0587751 | -2.1672216 |
| 0.0040323 | -0.0615338 | -0.1195005 | -2.1655807 |
| 0.0047043 | -0.1846015 | -0.1970568 | -2.1286603 |
| 0.0053763 | -0.2153684 | -0.2919406 | -2.0683572 |
| 0.0060484 | -0.2461353 | -0.4045882 | -1.9973882 |
| 0.0067204 | -0.3692030 | -0.5352985 | -1.9264192 |
| 0.0073925 | -0.5230376 | -0.6840978 | -1.8657058 |
| 0.0080645 | -0.7384060 | -0.8506290 | -1.8242730 |
| 0.0087366 | -0.9230075 | -1.0344045 | -1.8041719 |
| 0.0094086 | -1.1691430 | -1.2354441 | -1.8013004 |
| 0.0100806 | -1.4460450 | -1.4551768 | -1.8123765 |

Select the table you wish to view from the list of available tables on the Table Menu.

Tables allow you to see data that is available in the graphs in a tabular text format.

Each graph that you can display with TRAP has a corresponding table.

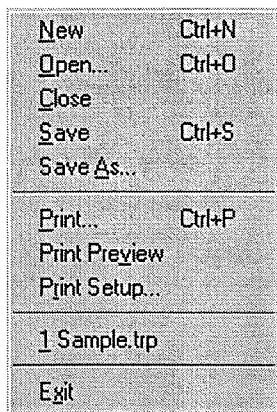
You can display a table by selecting which table you wish to see from the menu

There are no formatting options for tables.

You can copy the entire contents of a table window to the Clipboard by selecting the Copy command from the Edit menu. You cannot select only portions of the table to copy - the Copy command always copies the entire contents of the table window to the Clipboard where it can be pasted in to other applications.

Menu and Toolbar Reference

File Menu



New - Creates a new **TRAP** data file

Open... - Opens an existing **TRAP** data file

Close - Closes the currently open **TRAP** data file

Save - Saves the currently open **TRAP** data file

Save As... - Saves the currently open **TRAP** data file with a new name

Print... - Prints the active window

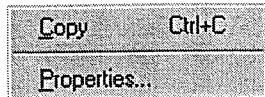
Print Preview - Previews how the active window will appear when printed

Print Setup... - Brings up the Print Setup dialog to allow changing printer settings

1,2,3,4 - Displays a list of the most recently opened **TRAP** data files

Exit - Exits **TRAP**

Edit Menu



This menu can be used to copy graphics and text to the Clipboard for pasting into other applications.

Copy

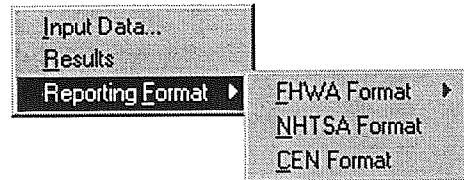
Use this command to copy the contents of the active window to the Clipboard. This option is only available for Graphs, Tables, and the Summary report

Properties....

Use this command to display the Graph Parameters Form for the currently active Graph window. This command is only available when a Graph window is the active window

The Properties item on the Edit menu is only available for a graph window. See Graphs for information on modifying graphs.

Data Menu



Input Data...

Brings up the Input Data form

Calculate

Runs the calculations

Results

Brings up the Results report view

Reporting Format

Runs external data formatting programs

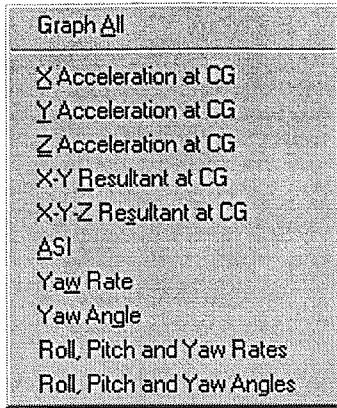
These programs are separate programs independently developed from TRAP. These programs are all DOS based and run in their own separate DOS window.

TRAP only provides a means to launch these other programs. There is no communication of data between TRAP and these other programs. These programs generate standard formatted files specified by the respective agency or organization.

NOTES:

1. FHWA has two programs, one to generate a header, and one to generate the main body of the report.
2. There is no CEN formatting program available at this time, selecting the CEN option will launch a batch file that does nothing. When a CEN formatting program becomes available in the future, it can be called from the batch file, and thus launched from TRAP.

Graph Menu



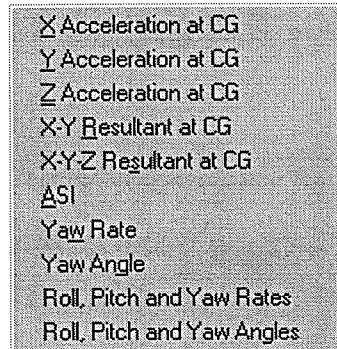
To view the graphs, select **Graph All** from the Graph menu. You may also draw one graph at a time, by selecting from the choices in the Graph menu.

Once one or more graphs are on the screen, you may arrange the windows. See the Windows menu for more information.

See Graphs for information on modifying the graphs.

Note: A plot produced by the "Yaw Angle" option is based on the assumption that yaw angle is uncoupled or independent from roll and pitch angles. Plots produced by the "Roll, Pitch, and Yaw Angles" option are determined by use of the Euler coupled equations (see Introduction for further discussion). Differences may therefore occur between the yaw angles thus plotted.

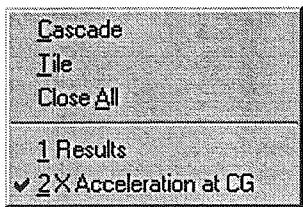
Table Menu



Select the table you wish to view from the list of available tables.

See Tables for more information.

Window Menu



Cascade

Arranges the currently open windows in an overlapping fashion

Tile

Arranges all open windows in a non-overlapping fashion.

Close All

Closes all the open windows. This does not close the open TRAP data file, just the open windows.

1,2,3, etc.

This list shows the currently open windows. Select any window from this list to bring it to the forefront.

Toolbar



The toolbar is displayed across the top of the application window, below the menu bar. The toolbar provides quick mouse access to many tools used in TRAP.

To hide or display the Toolbar, choose Toolbar from the View menu (ALT, V, T).

| Click | To |
|-------|---|
| | Create a new TRAP data file. |
| | Open an existing Trap data file. TRAP displays the Open dialog box, in which you can locate and open the desired file. |
| | Save the current TRAP data file with its current name. If you have not named the project, TRAP displays the Save As dialog box. |
| | Print the active window. |
| | Display the Data Input Form |
| | View Results |

APPENDIX B

TRAP PROGRAMMER'S GUIDE

Programmer's Guide



Test Risk Assessment Program (TRAP) Version 2.1

January 2000

Copyright ©Texas Transportation Institute 1997
Developed by CAPSHER Technology, Inc.
Info@capsher.com

Table of Contents

| | |
|----------------------------------|------|
| COMPILER | B-1 |
| THIRD PARTY LIBRARIES..... | B-1 |
| CURVE FITTING..... | B-1 |
| LINEAR INTERPOLATION | B-2 |
| CUBIC SPLINE | B-2 |
| SIGN CONVENTION | B-3 |
| MATHEMATICAL COMPUTATIONS | B-3 |
| INTEGRATION | B-3 |
| DIFFERNTIATION..... | B-3 |
| TRAP SPECIFIC CALCULATIONS | B-3 |
| PROCESSING STEP..... | B-4 |
| CALCULATION STEP..... | B-4 |
| FILTERING..... | B-5 |
| POST-CALCULATION STEP..... | B-6 |
| EXECUTING THE CALCULATIONS..... | B-6 |
| SUBROUTINES | B-6 |
| SDMVI..... | B-8 |
| DEFINITIONS | B-8 |
| CINPUT DATA VARIABLES | B-9 |
| CCARUTILDOC VARIABLES | B-10 |
| FLOW CHART | B-14 |
| TRAP-PROGRAM STATE DIAGRAM | B-14 |

APPENDIX B

TRAP Programmer's Guide

This Programmer's Guide is intended for programmers needing to maintain and/or extend the capabilities of TRAP (Test Risk Assessment Program). TRAP calculates several occupant risk parameters as detailed in National Cooperative Highway Research Program (NCHRP) Report 350 (1993), and in Committee for European Normalization (CEN) document EN 1317-1(April 1998). Report 350 parameters implemented by TRAP are detailed in Appendix A (sections A4 and A5) as well as Appendix F of Report 350. CEN parameters implemented by TRAP are detailed in Sections 6 and 7 of the CEN document. See the NCHRP and CEN documents for detailed information on testing procedures, data acquisition requirements, and computing the standardized occupant risk parameters.

Compiler

TRAP is written in C++ using Microsoft Foundation Classes (MFC). The compiler used is Microsoft Visual C++ Version 5.0

Third Party Libraries

The only third party software used in TRAP is the Quinn-Curtis graphing package. The following classes are derived from classes in the Quinn-Curtis package:

CGraphWnd : public QCGraphWnd
CGraphView: public QCPageView
CGraphPrintWnd: public QCPageWnd

CCurve uses the class QCStaticData

Curve Fitting

Curve fitting and interpolation are sometimes necessary when preprocessing the raw input data in preparations for the calculations. The calculations expect the data to all be on the same time interval. When the raw data are on different time intervals, interpolation is used to make all the arrays on a common time interval. This is explained in more detail in a later section in the Programmers Documentation named "Preprocessing Step". Both linear interpolation and cubic spline techniques are used.

Linear Interpolation

Of the two types of curve fitting used in TRAP linear interpolation is the least accurate but simplest to implement. The linear interpolation algorithm finds y_i given x_i and two arrays KnownX[] and KnownY[].

Find j such that $\text{KnownX}[j] \leq x_i < \text{KnownX}[j+1]$ using the `hunt()` routine from Numerical Recipes.

```
dy = ( KnownY[j+1] - KnownY[j] ) / (KnownX[j+1] - KnownX[j])
yi = KnownY [j] + (xi - KnownX [j]) * (dy)
```

Cubic Spline

The spline interpolation routines used in TRAP are taken from Numerical Recipes in C Second Edition pages 113-115. These pages contain the code for the `spline()` and `splint()` functions and explain the process much better than this document could. The only modification made to the routines from Numerical Recipes used in TRAP is in the `splint()` function. The version used in TRAP takes an extra parameter to return the first derivative as well as the value of the spline function at the desired x point.

The differentiation calculation within the **Cubic Spline** routine is from the equation for calculating the derivative given in Numerical Recipes as eq. (3.3.5) on page 95 as follows.

$$\begin{aligned} dy/dx &= (y[j+1] - y[j]) / (x[j+1] - x[j]) - \\ &\quad (3A^2 - 1)/6 * (x[j+1] - x[j]) d2y[j] + \\ &\quad (3B^2 - 1)/6 * (x[j+1] - x[j]) d2y[j+1] \end{aligned}$$

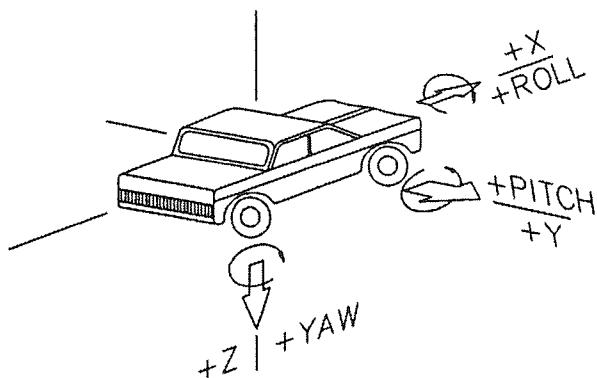
Where A, B, and the second derivative ($d2y$) are already calculated in the spline interpolation routine.

The following table lists all the routines within TRAP that use any integration, differentiation, interpolation, or cubic spline routines.

| Routine | Uses |
|--|---|
| CalculateResults() in calculate.cpp | Integrate() in calculate.cpp Filter() in calculate.cpp ButterworthFilter() in calculate.cpp |
| SetupCalcYawRates() in CarUtilDoc.cpp | Splint() in CarUtilDoc.cpp LinearInterp() in CarUtilDoc.cpp |
| SetupYawAccDiffRates() in CarUtilDoc.cpp | Splint() in CarUtilDoc.cpp |

Sign Convention

TRAP uses the sign convention documented in NCHRP report 350 as shown in the figure below. It is very important that all input data (accelerations, angular rates and / or angles) uses this same sign convention.



Sign Convention used by TRAP

Mathematical Computations

Integration

There is a single integration routine in TRAP that uses the Trapezoidal Rule to compute the integral of the function provided.

Differentiation

There are several places in TRAP where the derivative of a curve is computed. The derivative computed is equivalent to the slope of the piecewise linear function between provided data points.

TRAP Specific Calculations

There are several steps involved in the TRAP calculations. The user must first specify what types of data are available. The program then attempts to read in the data specified by the user setting flags indicating the availability of data in the process. There is an availability flag for each of the data sets needed for the calculations (X acceleration, Y acceleration, Z acceleration Yaw, Roll and Pitch data). The application then calculates as much as it can given the availability of data. Certain values cannot be calculated if all data sets are not available.

Preprocessing step

After the data is read in and the application knows what data is truly available, all data must be converted to the same time interval. The assumption is made that the first value in each data set occurred at the same point in time but there is the possibility that the different sets of data were sampled at different rates. If the data has been sampled at different rates the application must first determine which of the sampling rates is highest (gives the smallest change in time between data points). This highest sampling rate will be referred to as the *calculation sampling rate* and the multiplicative inverse of the *calculation sampling rate* will be referred to as the *calculation delta time*. The application must then determine which set of data limits the number of data points that can be used for the calculations.

The data used for the calculations in TRAP is to be interpolated not extrapolated. The application must determine, from the sampling rates and number of data points for the individual data sets, the earliest time at which one of the data set ends. This time will be referred to as the *max data time*.

The application must then calculate the greatest time value that is both a multiple of the *calculation delta time* and less than or equal to the *max data time*. This time will be referred to as the *calculation max time*. The number of data points to be used for the calculations is then known to be the *calculation max time* divided by the *calculation delta time*. The data sets with a different sampling rate than the *calculation sampling rate* are then resampled at the *calculation sampling rate* using the curve fitting technique selected by the user. The default curve fitting technique is cubic spline but linear interpolation is also available to the user.

Calculation step

At this point the application knows which data sets are available and has access to data sets with the same sampling rate and the same number of data points. The applications will attempt to calculate the following values:

Occupant Impact Velocity (OIV)

- Time of impact
- Interior side of vehicle impacted
- Impact Velocity in x-direction
- Impact Velocity in y-direction

Theoretical Head Impact Velocity (THIV)

- Time of impact
- Interior side of vehicle impacted
- Resultant Impact Velocity

Ridedown Accelerations

- Max. 10-msec moving average acceleration in x-direction after occupant impact
- Max. 10-msec moving average acceleration in y-direction after occupant impact

PHD

Max. 10-msec moving average resultant acceleration in x and y directions after occupant impact

ASI

Max. 50-msec moving average accelerations

x-direction

y-direction

z-direction

xy resultant acceleration

xyz resultant acceleration

Max roll, pitch, and yaw angles.

Each of these values is dependent on the availability of one or more data sets. Therefore, the calculation routine contains many conditional statements to check the availability of data before a calculation is actually performed. The following table details the calculation dependencies.

| Computed Value | X Accel. | Y Accel. | Z Accel. | Yaw Rate | Pitch Rate | Roll Rate |
|-----------------------|---------------------|---------------------|---------------------|---------------------|-----------------------|----------------------|
| OIV - X | X | | | | | |
| OIV - Y | | X | | | | |
| THIV | X | X | | | X* | |
| Ridedown Accel. X | X | | | | | |
| Ridedown Accel. Y | | X | | | | |
| PHD | X | X | | | | |
| ASI | X | X | | X | | |
| Max 50-msec Avg. X | X | | | | | |
| Max 50-msec Avg. Y | | X | | | | |
| Max 50-msec Avg. Z | | | X | | | |
| Max Roll angle | | | | X | X | X |
| Max Pitch angle | | | | X | X | X |
| Max Yaw angle | | | | X | X | X |

*As an alternative, yaw angle versus time may be input.

Filtering

For plotting purposes, NHCRP Report 350 recommends acceleration data be filtered using a CFC (Channel Frequency Class) of 60 Hz. TRAP uses a Butterworth 4-pole phaseless digital filter as documented in SAE J211. The filtering routine used by TRAP is a direct implementation of the algorithm detailed in SAE J211.

Post-calculation step

If all data sets are not available the data to be displayed in some curves, graphs or tables may not be available. Each curve, graph and table object contains an availability flag that is set after the calculations based on the availability of data sets. If the data to be displayed by a curve or a table could not be calculated then the availability flag for that curve or table is set to false. If none of the curves to be displayed by a graph are available then the availability flag for the graph is set to false.

Executing the calculations

The calculations are run automatically when a project file is opened and when any changes are made to the input data.

Subroutines

The following table lists all the classes used in TRAP with a description of how each is used. Note: TRAP was originally called CarUtil, so many of the classes retain that name. Also, when used below, QC means Quinn-Curtis

| Class | Type | Purpose |
|---------------------|----------------|--|
| CAboutDlg | Dialog | Popup dialog with program info shown when user selects "About TRAP..." from the Help menu |
| CAccelerationPage | Property Page | "Acceleration..." tab property page used to enter acceleration data |
| CCarUtilApp | App | Main Application class |
| CCarUtilDoc | Doc | TRAP's doc class. This stores all the data used in TRAP |
| CCarUtilView | View | Default view for the application |
| CChildFrame | Frame | Application's CMDIChildWnd |
| CColumn | | Stores a single column of data, including values and titles, for use by CTable |
| CCurve | | Stores settings for a single curve for use by CGraph |
| CDataInputSheet | Property Sheet | Property sheet for editing input data. Contains 4 pages for Description, Settings, Acceleration, and Yaw data |
| CDescriptionPage | Property Page | "Description" tab property page used to enter general description data |
| CErrorView | View | Displays list of errors and/or warnings in a text window |
| CGraph | | Stores all setup info for a graph. Contains a list of curves (CCurve) |
| CGraphAxesPage | Property Page | "Axes" tab property page used to enter axis settings for a graph |
| CgraphCurvesPage | Property Page | "Plot" tab property page used to enter curve settings for a graph |
| CGraphFrame | Frame | MDI frame to display a Graph View in |
| CgraphGeneralPage | Property Page | "Graph" tab property page used to enter general settings for a graph |
| CGraphPrintWnd | | QC override. Special window used for drawing graph into when it is being printed |
| CgraphPropertySheet | Property Sheet | Property sheet for editing graph settings. Contains 3 pages for General, Curves, and Axis settings |
| CGraphView | View | QC override. View (window) used to display graphs |
| CGraphWnd | | QC override. Used to override QC graph functionality. |
| CIinputData | | Stores all input data entered by user |
| CMainFrame | Frame | MDI main frame window |
| CReportView | View | Displays results as text report in a view |
| CSettingsPage | Property Page | "Settings" tab property page used to select type of raw data to read in |
| CTable | | Stores information to display a table of data. Contains a list of columns (CColumn). The table is displayed using CTableView |
| CTableView | View | Display a table in a text window. |
| CYawPage | Property Page | "Yaw.." tab property page used to enter yaw/angular data |
| FILTER | | Helper class used to set up parameters for Filter() function |

SDMVI

There are some additional classes that were used as a layer between the standard MFC Doc/View classes and our application.

The default Doc/View models available in MFC (i.e. SDI, MDI) were not adequate for TRAP. TRAP has a hybrid Doc/View architecture which allows only a single document to be open at a time, but allows multiple views on the doc. We call this model "Single Document – Multiple View Interface" or SDMVI for short. The special behavior includes such things as updating the Window menu properly and insuring that only one instance of each of our application's windows are shown at a time. This special behavior has been implemented by sub-classing off of some of the standard Doc/View related MFC classes. Some of our SDMVI classes were in turn sub-classes off of some of the standard application classes, others were used directly. This is shown in the following table.

| MFC class | SDMVI class derived from the MFC class | Application class derived from the SDMVI class |
|-------------------|--|--|
| CMDIChildWnd | CSDMVIChildWnd | CGraphFrame |
| CMultiDocTemplate | CSDMVIDocTemplate | |
| CDocument | CSDMVIDocument | CCarUtilDoc |
| CMDIFrameWnd | CSDMVIFrameWnd | CMainFrame |
| CWinApp | CSDMVIWinApp | CCarUtilApp |

Definitions

The most important variables in the program are in two particular classes. Those are CCarUtilDoc and CInputData. The variables are commented in the header (.h) files for these two classes. For convenience those sections of the file are shown here.

CinputData variables

```
***** Description
CString m_strTestAgency ;           // Test agency
CString m_strTestNumber ;          // Test number
CString m_strTestDate ;            // Test date
CString m_strTestArticle ;         // Test article
CString m_strVehicleDescription ; // Vehicle description
double m_dInertiaMass ;           // Test inertia mass (kg)
double m_dGrossMass ;             // Gross static mass (kg)
double m_dImpactSpeed ;           // Impact speed (km/h)
double m_dImpactAngle ;           // Impact angle (degrees)
double m_dTheorHeadDistanceX ;    // Distance of theoretical head from CG (mm)
***** Settings
double m_dSampleRate ;           // Acceleration sample rate (samples/sec)
BOOL   m_bYawSampleRateSame;      // Flag indicating if Yaw data is sampled at same rate as accel data
double m_dYawSampleRate;          // Yaw sample rate (samples/sec)
BOOL   m_bAccelMeasuredAtCG;     // Flag indicating if Acceleration components were measured at CG
int    m_nAccelDataSource ;       // Acceleration file type (ASCII, Binary-float, Binary-double)
int    m_nYawDataType ;           // Yaw data type (Angle or Rate)
int    m_nYawDataSource ;         // Yaw data file type (ASCII, Binary-float, Binary-double)
***** Acceleration 1
double m_dAccelDisplacementFromCG[2]; // Displacement from CG (mm)

CString m_strAccelXFilePath[2];     // X Acceleration data file pathname
int    m_nAccelXFileOffset[2];       // X Acceleration data file offset
int    m_nAccelXFileNcols[2];        // X Acceleration data file column count
int    m_nAccelXFileCol[2];          // which column contains X Acceleration data

CString m_strAccelYFilePath[2];     // Y Acceleration data file pathname
int    m_nAccelYFileOffset[2];       // Y Acceleration data file offset
int    m_nAccelYFileNcols[2];        // Y Acceleration data file column count
int    m_nAccelYFileCol[2];          // which column contains Y Acceleration data

CString m_strAccelZFilePath[2];     // Z Acceleration data file pathname
int    m_nAccelZFileOffset[2];       // Z Acceleration data file offset
int    m_nAccelZFileNcols[2];        // Z Acceleration data file column count
int    m_nAccelZFileCol[2];          // which column contains Z Acceleration data

***** Yaw
CString m_strYawUnits ;           // Units for Yaw data (deg/sec, rad/sec)
CString m_strYawFilePath ;         // Yaw data file pathname
int    m_nYawFileOffset ;          // Yaw data file offset
int    m_nYawFileNcols ;           // Yaw file column count
int    m_nYawFileCol ;             // which column contains Yaw data
int    m_nYawFileTimeCol ;          // which column contains Time data
int    m_nYawInterpMethod ;         // Interpolation method for getting rate from angle (linear, cubic spline)

***** Roll
CString m_strRollFilePath ;        // Roll data file pathname
int    m_nRollFileOffset ;          // Roll data file offset
int    m_nRollFileNcols ;           // Roll file column count
int    m_nRollFileCol ;             // which column contains Roll data

***** Pitch
CString m_strPitchFilePath ;       // Pitch data file pathname
int    m_nPitchFileOffset ;          // Pitch data file offset
int    m_nPitchFileNcols ;           // Pitch file column count
int    m_nPitchFileCol ;             // which column contains Pitch data
```

CcarUtilDoc variables

```
CString m_strWarnings ; // stores warning messages for display by CErrorView
CString m_strErrors ; // stores error messages for display by CErrorView
CString m_strReport ; // stores results report for display by CReportView
CString m_strAnnotation ; // used on graph settings form to build graph annotation
BOOL m_bAnnotationAutoFill ; // indicates if annotation will be entered by user or created automatically by application

CInputData m_InputData ; // contains all input data entered by user
BOOL m_bNeedToRecalc ; // flag indicating if calculations need to be done
int m_nErrors ; // count of errors that exist
int m_nWarnings ; // count of warnings that exist

CGraph *m_pGraph1 ; // Setup information for Graph 1
CGraph *m_pGraph2 ; // Setup information for Graph 2
CGraph *m_pGraph3 ; // Setup information for Graph 3
CGraph *m_pGraph4 ; // Setup information for Graph 4
CGraph *m_pGraph5 ; // Setup information for Graph 5
CGraph *m_pGraph6 ; // Setup information for Graph 6
CGraph *m_pGraph7 ; // Setup information for Graph 7
CGraph *m_pGraph8 ; // Setup information for Graph 8
CGraph *m_pGraph9 ; // Setup information for Graph 9
CGraph *m_pGraph10 ; // Setup information for Graph 10

CTable *m_pCurrentTable ; // Used to transfer the table pointer into the TableView

CTable *m_pTable1 ; // Setup information for Table 1
CTable *m_pTable2 ; // Setup information for Table 2
CTable *m_pTable3 ; // Setup information for Table 3
CTable *m_pTable4 ; // Setup information for Table 4
CTable *m_pTable5 ; // Setup information for Table 5
CTable *m_pTable6 ; // Setup information for Table 6
CTable *m_pTable7 ; // Setup information for Table 7
CTable *m_pTable8 ; // Setup information for Table 8
CTable *m_pTable9 ; // Setup information for Table 9
CTable *m_pTable10 ; // Setup information for Table 10

// curve data arrays for OIV Occupant Impact Time marker shown on X and Y acceleration graphs
double *m_adOccupantImpactTimeOIV_X1 ;
double *m_adOccupantImpactTimeOIV_Y1 ;
double *m_adOccupantImpactTimeOIV_X2 ;
double *m_adOccupantImpactTimeOIV_Y2 ;

// curve data arrays for THIV Occupant Impact Time marker shown on X-Y resultant graph
double *m_adOccupantImpactTimeTHIV_X ;
double *m_adOccupantImpactTimeTHIV_Y ;
int m_nOccupantImpactTimeCount ;

// Curves showing Occupant Impact
CCurve *m_pOccupantImpactCurve1 ;
CCurve *m_pOccupantImpactCurve2 ;
CCurve *m_pOccupantImpactCurve4 ;

// Curves for Filtered Data
CCurve *m_pFilterCurve1 ;
CCurve *m_pFilterCurve2 ;
CCurve *m_pFilterCurve3 ;
CCurve *m_pFilterCurve4 ;
CCurve *m_pFilterCurve5 ;
CCurve *m_pFilterCurve6 ;
```

```

CCurve *m_pFilterCurve7 ;
CCurve *m_pFilterCurve8 ;

// Curves for Raw Data
CCurve *m_pRawCurve1 ;
CCurve *m_pRawCurve2 ;
CCurve *m_pRawCurve3 ;
CCurve *m_pRawCurve4 ;
CCurve *m_pRawCurve5 ;

// Curves for 50-msec average
CCurve *m_pAverageCurve1 ;
CCurve *m_pAverageCurve2 ;
CCurve *m_pAverageCurve3 ;
CCurve *m_pAverageCurve4 ;
CCurve *m_pAverageCurve5 ;
CCurve *m_pAverageCurve6 ;
CCurve *m_pAverageCurve7 ;
CCurve *m_pAverageCurve8 ;

// Curves for 10-msec average
CCurve *m_p10msMoveAvgCurve1 ;
CCurve *m_p10msMoveAvgCurve2 ;
CCurve *m_p10msMoveAvgCurve3 ;
CCurve *m_p10msMoveAvgCurve4 ;
CCurve *m_p10msMoveAvgCurve5 ;

// Curves for Roll, Pitch and Yaw Rate graph
CCurve *m_pCurveCoupledYawRate ;
CCurve *m_pCurveRollRate ;
CCurve *m_pCurvePitchRate ;

// Curves for Roll, Pitch and Yaw Angle graph
CCurve *m_pCurveCoupledYawAngle ;
CCurve *m_pCurveRollAngle ;
CCurve *m_pCurvePitchAngle ;

// Variables for reading in raw Yaw Angle data from file
int      m_nFileYawAngleArraySize ; // number of Yaw Angle data points read in from file
double   *m_pdFileYawAngleTime ;           // Time corresponding to Yaw Angle value from file
double   *m_pdFileYawAngle ;              // Yaw Angle from file

// Variables for reading in raw data from file all at same time intervals
int      m_nFileArraySize ;      // number of points
double   *m_pdFileTime ;          // times (shared by all arrays)
double   *m_pdFileAccelX1 ;       // X component of acceleration 1 from file
double   *m_pdFileAccelY1 ;       // Y component of acceleration 1 from file
double   *m_pdFileAccelZ1 ;       // Z component of acceleration 1 from file
double   *m_pdFileAccelX2 ;       // X component of acceleration 2 from file
double   *m_pdFileAccelY2 ;       // Y component of acceleration 2 from file
double   *m_pdFileAccelZ2 ;       // Z component of acceleration 2 from file
double   *m_pdFileYawRate ;        // Yaw Rate from file
double   *m_pdFileRollRate ;       // Roll Rate from file
double   *m_pdFilePitchRate ;      // Pitch Rate from file

int      m_nAccelArraySize;
int      m_nYawRateArraySize;
int      m_nRollArraySize ;
int      m_nPitchArraySize ;

BOOL    m_bXAccelDataAvailable;    // indicates if X acceleration data is available
BOOL    m_bYAccelDataAvailable;    // indicates if Y acceleration data is available
BOOL    m_bZAccelDataAvailable;    // indicates if Z acceleration data is available

```

```

BOOL  m_bYawDataAvailable;      // indicates if yaw data is available
BOOL  m_bRollDataAvailable;    // indicates if roll data is available
BOOL  m_bPitchDataAvailable;   // indicates if pitch data is available

// Calculated roll, pitch and yaw arrays
int   m_nRollPitchYawSize ;    // Number of points in roll, pitch and yaw calculated arrays
double *m_pdCalcRPYTime ;      // Time, common for all roll, pitch and yaw calculated arrays
double *m_pdCalcCoupledYawRate ;
double *m_pdCalcCoupledYawAngle ;
double *m_pdCalcRollRate ;
double *m_pdCalcRollAngle ;
double *m_pdCalcPitchRate ;
double *m_pdCalcPitchAngle ;

// Calculated arrays
int   m_nCalcArraySize ;       // Number of points in calculated arrays
double *m_pdCalcTime ;         // Time, common for all calculated arrays
double *m_pdCalcAccelXCG ;     // Calculated X Acceleration at CG
double *m_pdCalcAccelYCG ;     // Calculated Y Acceleration at CG
double *m_pdCalcAccelZCG ;     // Calculated Z Acceleration at CG
double *m_pdCalcYawRate ;      // Calculated yaw rate
double *m_pdCalcYawAngle ;     // Calculated yaw angle

double *m_pdCalcPHD ;          // Calculated PHD
double *m_pdCalcResultantXY ;   // Calculated X-Y resultant acceleration
double *m_pdCalcResultantXYZ ;  // Calculated X-Y-Z resultant acceleration
double *m_pdCalcASI ;          // Calculated ASI

// Data storage for 10-msec average curves
double *m_pdCalcResultantXYFiltered ;
double *m_pdCalcResultantXYZFiltered ;
double *m_pdCalcAccelIXFiltered ;
double *m_pdCalcAccelYFiltered ;
double *m_pdCalcAccelZFiltered ;
double *m_pdCalcYawRateFiltered ;
double *m_pdCalcYawAngleFiltered ;

// Data storage for 50-msec average curves
double *m_pdCalcAccelXAveraged ;
double *m_pdCalcAccelYAveraged ;
double *m_pdCalcAccelZAveraged ;
double *m_pdCalcResultantXYAveraged ;
double *m_pdCalcResultantXYZAveraged ;

// Data storage for 10-msec average curves
double *m_pdCalcX10msecMoveAvg ;
double *m_pdCalcY10msecMoveAvg ;
double *m_pdCalcZ10msecMoveAvg ;
double *m_pdCalcXY10msecMoveAvg ;
double *m_pdCalcXYZ10msecMoveAvg ;

double m_dCalcOccupantImpactTime ; // CEN impact time
int   m_nImpactIndex ;           // CEN impact time index

double m_dNCHRPOccupantImpactTime ; // NCHRP impact time
int   m_nNCHRPImpactIndex ;        // NCHRP impact time index

double m_dMaxPHD ;   // max Post-impact Head Decceleration
int   m_nIndexMaxPHD ; // time index of max PHD
double m_dTHIV ;     // Theoretical Head Impact Velocity

double m_dOIVx ; // Occupant Impact Velocity x

```

```

double m_dOIVy ; // Occupant Impact Velocity y

double m_dMaxRDAx ; // max ridedown acceleration x
double m_dMaxRDAy ; // max ridedown acceleration y

int m_nMaxRDAxIndex ; // index of max ridedown acceleration x
int m_nMaxRDAyIndex ; // index of max ridedown acceleration y

double m_dMaxRollAngle ; // max. Roll angle
double m_dMaxPitchAngle ; // max. Pitch angle
double m_dMaxYawAngle ; // max. Yaw angle

int m_nMaxRollAngleIndex ; // index of max Roll angle
int m_nMaxPitchAngleIndex ; // index of max Pitch angle
int m_nMaxYawAngleIndex ; // index of max Yaw angle

double m_dMaxASI ; // max Acceleration Severity Index
int m_nIndexMaxASI ; // time index of max ASI

int m_nOIVImpactSurface ; // impact surface for OIV calc
int m_nTHIVImpactSurface ; // impact surface for THIV calc

BOOL m_bOIVImpact ; // occupant impact OIV calc
BOOL m_bTHIVImpact ; // occupant impact THIV calc

double m_dMaxAccelXAveraged ; // maximum 50msec moving avg. Ax
double m_dMaxAccelYAveraged ; // maximum 50msec moving avg. Ay
double m_dMaxAccelZAveraged ; // maximum 50msec moving avg. Az

int m_nMaxAccelXAvgIndex ; // index of max. 50msec moving avg. Ax
int m_nMaxAccelYAvgIndex ; // index of max. 50msec moving avg. Ay
int m_nMaxAccelZAvgIndex ; // index of max. 50msec moving avg. Az

```

Flow Chart

Following is a state diagram for TRAP. In TRAP there are three states: (1) no file opened yet, (2) file opened but not being edited, and (3) file being edited. The diagram shows how the program transitions between the different states based on interaction from the user. Note that the calculations are run implicitly by exiting the Edit Input screen using the "OK" button.

