VEHICLE DYNAMICS MEASUREMENT SYSTEM FOR EVALUATING OLDER DRIVER PERFORMANCE

By

ETHAN J. DAVIS

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING

UNIVERSITY OF FLORIDA

2006

This document is dedicated to my grandfather Robert ‘Sport’ Davis Sr.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Niezrecki for assisting me with this research and providing useful knowledge and guidance. I would also like to thank Dr. John Schueller for helping me with many on-campus issues and Dr. Carl Crane for serving on my supervisory committee. Furthermore, I would like to express gratitude to everyone at the Occupational Therapy Department at the College of Public Health and Health Professions. I would also like to thank Richard Phillips for helping me with parts of my research and providing some helpful tips along the way.

Finally, I would like to thank my parents and the rest of my family for their continued support.

TABLE OF CONTENTS

page

ACKNOWLEDGMENTS iv

LIST OF TABLES viii

LIST OF FIGURES ix

ABSTRACT xi

CHAPTER

1 INTRODUCTION 1

1.1 Background 1
1.2 Older Driver Statistics 2
1.3 Purpose of FHWA Research 3
1.4 Objectives 5

2 LITERATURE REVIEW 9

2.1 Previous Research Studies 9
2.2 Devices 13
2.2.1 DriveCam 13
2.2.2 Vericom 14
2.2.3 Dewetron 16

3 SYSTEM REQUIREMENTS 18

3.1 Performance 18
3.2 Transducers 19
3.3 GPS 20
3.4 Video 21
3.5 Ease of Use 21
3.6 Cost 21
3.7 Flexibility 22

4 SYSTEM DESIGN 23

4.1 Vehicle Description 23

4.2 Selection of Components 26
4.2.1 Data Acquisition System 26
4.2.2 Transducers 28
4.2.3 Modules 31
4.2.4 Power Supplies 32
4.2.5 GPS 36
4.2.6 Video 36
4.2.7 Software 38
4.3 Installation 39
4.4 Calibration 49
4.5 Operation 49

5 SYSTEM VERIFICATTON 51

5.1 Interpretation of Data 51
5.2 Analysis of Acceleration Test Maneuver 54
5.3 Analysis of Deceleration Test Maneuver 56
5.4 Analysis of Right Turn Test Maneuver 58
5.5 Analysis of Left Turn Test Maneuver 60

6 FHWA RESEARCH ANALYSIS 63

6.1 Description of *main.m* (Road Course) 64
6.2 Description of *stats.m* (Road Course) 67
6.3 Description of *sort.m* (Road Course) 69
6.4 Description of *sim.m* and *sim_sort.m* 69
6.5 Description of SPSS Analysis 70

7 RESULTS AND DISCUSSION 73

7.1 FHWA Research Results 73
7.2 Example Analysis of Turning Phase of FHWA Maneuver 73

8 CONCLUSION 78

8.1 Future Research Studies 78
8.2 Recommendations and Improvements 79

APPENDIX

A INTERSECTION DIAGRAMS 80

B COMPONENT TECHNICAL SPECIFICATIONS 92

C CUSTOM PULLEY DRAWINGS 95

D MATLAB CODES 97

E EXAMPLE OF SPSS OUTPUT FILE FOR MAXIMUM YAW DURING THE TURN PHASE OF MANEUVER 1 118

LIST OF REFERENCES 120

BIOGRAPHICAL SKETCH 122

LIST OF TABLES

Table page

4-1 Technical Specifications for Signal Conditioning Analog Input Modules. 32

4-2 Transducer Output Ranges and Module Input Ranges..32

5-1 Measurement Abbreviations and Respective Units for All Plots.............................53

7-1 Summary of Statistics for Turning Phase of FHWA Road Course Maneuver 2......74

LIST OF FIGURES

Figure page

1-1 Street Map of The FHWA Road Course in Gainseville, FL. 4

1-2 Picture of NODRTC Simulator showing vehicle cab and projection screens...........7

2-1 Display Output from ARGOS Instrumented Vehicle (Rizzo et al., 292)................11

2-2 Picture of DriveCam...14

2-3 Picture of Vericom VC3000...15

2-4 Picture of Dewetron DEWE5000..16

4-1 Exterior View of 2005 Buick Century Sedan NODRTC Research Vehicle. 23

4-2 Picture of Left Foot Accelerator Pedal Adapter...24

4-3 Picture of Hand Controls, Turn Signal Extender, and Steering Wheel Receptacles...25

4-4 Picture of Auxiliary Brake Pedal. 25

4-5 Wiring Diagram for Data Acquisition System. 27

4-6 Picture of National Instruments Signal Conditioning Carrier (SC-2345) and Modules...27

4-7 Picture of Kistler Type 5210 Capacitive Accelerometer Power Supply. 33

4-8 Picture of PCB Model 480E09 Power Supply for ICP Laser Tachometer. 34

4-9 Wiring Diagram of Angular Rate Sensor Power Supply. 35

4-10 Picture of Equipment Installed in Trunk. 40

4-11 Trunk Diagram Displaying the Approximate Location of Each Component. 40

4-12 Picture of Laser Tachometer, Mounting Bracket, and Drive Shaft.........................42

4-13 Picture of Installed Sting Potentiometer and Custom Pulley Assembly. 43

4-14 Picture of Lateral Accelerometer, Longitudinal Accelerometer, and Angular Rate Sensor Installed Underneath the Front Passenger Seat. 44

4-15 Picture of Miniature Video Camera and Installation Bracket. 46

4-16 Wiring Diagram of All System Components. 47

4-17 Wiring Diagram Indicating the Location of All System Components in the Vehicle. 48

5-1 Example Plot of a Typical Left Turn. 54

5-2 Plot of a Typical Normal Acceleration Test Maneuver. 55

5-3 Plot of a Typical Aggressive Acceleration Test Maneuver. 56

5-4 Plot of a Typical Normal Deceleration Test Maneuver. 57

5-5 Plot of a Typical Aggressive Deceleration Test Maneuver. 57

5-6 Plot of a Typical Normal Right Turn Test Maneuver. 58

5-7 Plot of a Typical Aggressive Right Turn Test Maneuver. 59

5-8 Plot of a Typical Normal Left Turn Test Maneuver. 61

5-9 Plot of a Typical Aggressive Left Turn Test Maneuver. 61

6-1 FHWA Road Course Generated Using GPS Coordinates with Intersections of Interest Highlighted and Labeled. 65

6-2 Example of Maneuver Plot Used to Verify That The Vehicle Dynamics Data and GPS Data Are Synchronized. 66

6-3 Plot of Maneuver *4B* Generated Using *stats.m* With Markers Indicating the Start and End of The Turning Phase. 68

7-1 Picture of Right-Turn Channelization from FHWA Road Course Maneuver 2A. ...75

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Master of Engineering

VEHICLE DYNAMICS MEASUREMENT SYSTEM FOR EVALUATING OLDER DRIVER PERFORMANCE

By

Ethan J. Davis

May 2006

Chair: Christopher Niezrecki
Cochair: John Schueller
Major Department: Mechanical and Aerospace Engineering

In 2003, The Occupational Therapy Department of The College of Public Health and Health Professions at The University of Florida founded the National Older Driver Research and Training Center (NODRTC). The NODRTC has received funding from the Centers for Disease Control (CDC), Federal Highway Administration (FHWA), National Highway and Traffic Safety Administration (NHTSA), and the Florida Department of Transportation (FDOT). The NODRTC offers assessment of driving skills, referrals to health professionals to improve driving skills, and counseling in the use of alternative transportation for former drivers.

This thesis discusses the development and implementation of a system to measure vehicle dynamics for the purpose of evaluating older driver performance. The system will be used for several research studies, although this paper will only discuss its use in an FHWA study. The goal of the study is to evaluate the effectiveness of intersection

improvements recommended by the FHWA. The recommended intersection improvements are designed to make the intersections easier to negotiate, particularly for older drivers.

The vehicle dynamics measurement system consists of a data acquisition system, various sensors, and a GPS antenna. The system is capable of measuring lateral acceleration, longitudinal acceleration, yaw (turning rate), speed, steering wheel position, brake status, and turn signal status. There is also an event trigger that can be activated by an evaluator to mark the data so that the event can be analyzed later. The GPS antenna is used to determine the location of the vehicle while it is driving a pre-determined road course. The data are extracted for the intersections of interest and an algorithm is used to divide the intersection into three distinct phases (approach, turn, and recovery). Several statistics are calculated for each variable and phase of the intersection. These statistics were analyzed using SPSS statistical analysis software.

Overall the system was useful in acquiring vehicle dynamics at particular intersections and inferring differences between old and young drivers. Furthermore, in relation to the FHWA study, the results indicated that the recommended intersection improvements benefited both old and young drivers and were more helpful to older drivers.

CHAPTER 1
INTRODUCTION

This thesis will discuss the development of a vehicle dynamics acquisition system to be used as an aid in driving research. The system will be installed in a 2005 Buick Century Sedan to be used by the National Older Driving Research and Training Center (NODRTC) to conduct research on older drivers. This chapter will discuss the background and motivation for older driver research, specifically the research being conducted by the NODRTC. The goals and objectives of the vehicle dynamics acquisition system will also be presented.

1.1 Background

The Occupational Therapy Department of the College of Public Health and Health Professions at The University of Florida founded the Seniors Institute on Transportation and Communications in 2000. The name was later changed to the National Older Driver Research and Training Center to reflect its natural focus after receiving funding in 2003 from the Centers for Disease Control and Prevention (CDC) and the Federal Highway Administration (FHWA). The NODRTC later received additional funding from the National Highway and Traffic Safety Administration (NHTSA) and the Florida Department of Transportation (FDOT) and is currently conducting three research projects for the CDC, NHTSA, and FHWA. The purpose of the CDC research is to determine the effectiveness of particular screening tests in determining a person’s driving abilities. The goal is to identify which tests can accurately predict a driver to be unfit to drive and prevent them from having to take a potentially dangerous driving test. The NHTSA research is being performed to determine the validity and reliability of guidelines issued by the American Medical Association (AMA) for diagnosing driving ability, specifically The Physician's Guide to Assessing and Counseling Older Drivers. The FHWA research evaluates older driver performance on roadway design features identified by the FHWA as possibly problematic for older drivers. The FHWA has issued guidelines for highway designs that are believed to increase the safe driving ability of older drivers. This research examines differences in driving performance between road designs that either meet or do not meet these guidelines.

The NODRTC has two vehicles dedicated to driving research and evaluations. The design and implementation of the system described in this thesis may eventually be used for all of the research projects previously mentioned. Initially it will be used for the FHWA research and for that reason it will be the focus of the discussion. The system itself will be the same for the different studies. However, the specific goals for acquiring and processing the data will be unique. From this point on, all discussion will be relevant to the FHWA research, particularly the data processing and analysis.

1.2 Older Driver Statistics

The emphasis for evaluating older driver's driving abilities is motivated by the growing number of senior citizens driving, their frequency of collisions, and the severity of their crashes. In 2003, there were 35.9 million people in United States age 65 or older. That number is expected to double by the year 2030. When vehicle fatality rates are compared on the basis of miles traveled annually, the fatality rate for seniors is four times that of the 30-59 year old age group (DOT HS 809 328). An interesting note is that most of the older driver fatalities in 2000 occurred during the daytime (81%), on weekdays (71%), and involved another vehicle (76%) (DOT HS 809 328). All of these statistics suggest that older drivers are some of the most dangerous drivers on the road. Furthermore, with an inevitable increase in the number of older drivers from the baby boom generation, the number of collisions and fatalities as a result of older drivers is likely to increase.

1.3 Purpose of FHWA Research

As previously mentioned, the FHWA research project is focused on assessing older driver performance on roadway conditions identified by the FHWA as potentially problematic for older drivers. The FHWA has proposed certain roadway improvements that are intended to help all drivers negotiate an intersection. It is believed that these improvements will be most beneficial to older drivers. Two examples of these roadway improvements are an offset turning lane and a four foot wide receiving lane. The offset turning lane is the left-most lane when approaching an intersection. Instead of facing the oncoming turning lane, which is the standard for most intersections, the offset turning lane is positioned four or five feet further left. This results in the driver having an improved view of oncoming traffic, thus making it easier and safer to complete the turn. The four foot wide receiving lane is also intended to make navigation of an intersection easier. The four foot wide receiving lane is an extension on the right side of the roadway after completing a left turn at an intersection. The widened receiving lane allows the driver to make a wider turn and avoid having to turn sharply and possibly hit the median in the center of the road.

The NODRTC designed a road course in Gainesville, Florida, to evaluate the effectiveness of these improved roadway safety features. The road course incorporates ten intersections. Five of the intersections feature different recommended roadway improvements and the other five are their similar counterparts without the improvements.

They are all signalized intersections and all but one are left turn maneuvers. A street map of the FHWA road course showing the route is provided in Figure 1-1.

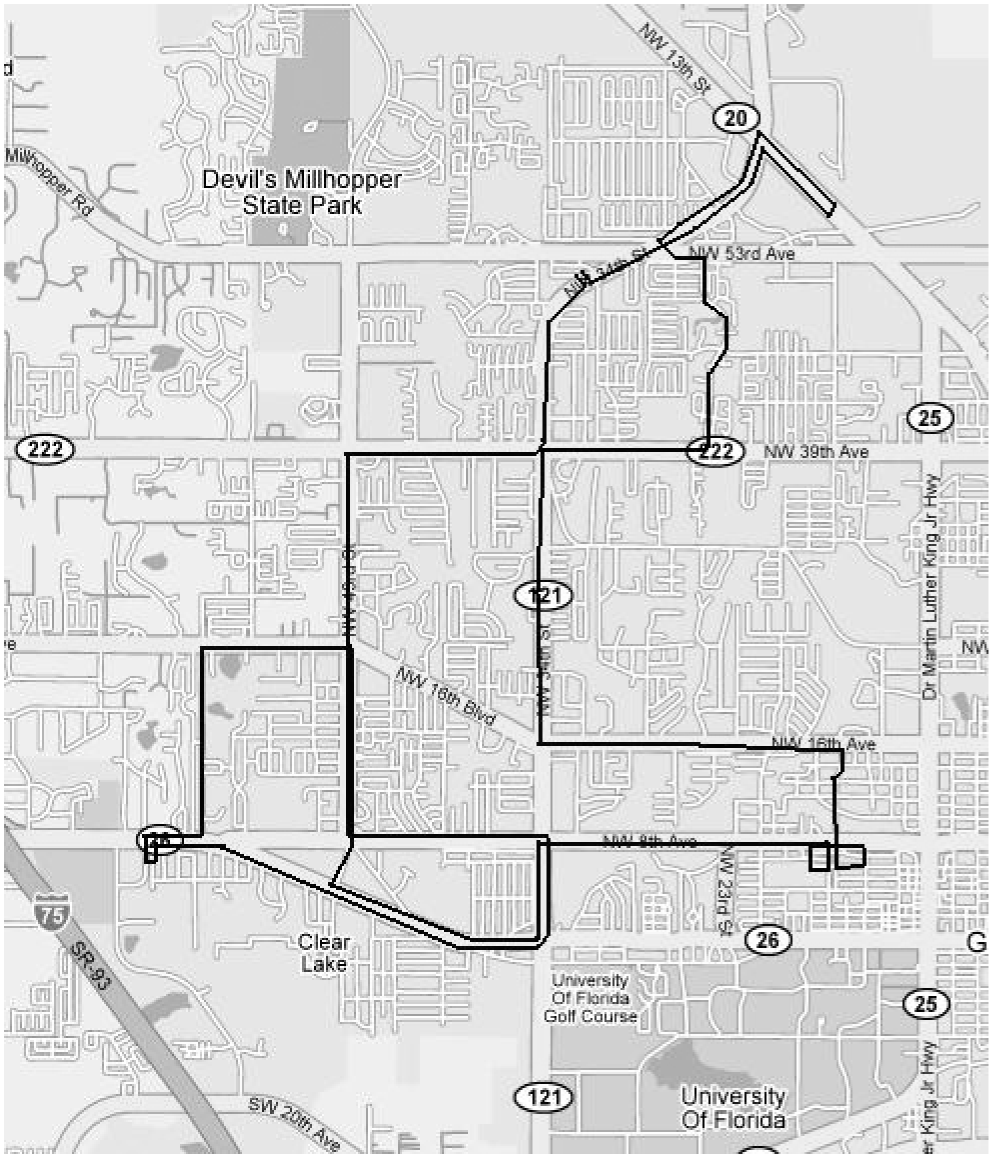


Figure 1-1. Street Map of The FHWA Road Course in Gainseville, FL.

A Driving Rehabilitation Specialist accompanies the test subject as they drive the road course. The Driving Rehabilitation Specialist, also referred to as the evaluator, assesses the driver's performance during the designated intersections. The evaluator uses a scoring sheet that lists all of the possible driving errors for each intersection. As the driver navigates the course, the evaluator pays careful attention to each intersection and scores driving errors including lane maintenance, forward vehicle gap acceptance, and overall stability. The FHWA research calls for at least 100 drivers age twenty-five or older to be evaluated. There are two hypotheses being simultaneously tested in the FHWA study. The first is that all drivers will benefit from the improved safety features and will have fewer driving errors on these particular intersections. The second hypothesis is that older drivers will benefit from the new safety features even more than the rest of the population. The older drivers are expected to have more driving errors per person for the unimproved intersections, but perform similar to the younger drivers for the improved intersections. It is certainly possible that any age driver may exhibit poor driving performance for both the improved and unimproved intersections. This is why a statistical analysis will be performed on the behavioral data to detect any underlying trends.

1.4 Objectives

In addition to the subjective evaluator scoring, the NODRTC wanted an objective way of measuring driver performance. The decision was made to design a system capable of measuring and recording vehicle dynamics such as acceleration and yaw. The system needed to be affordable and easy for an evaluator to operate. The desired measurements were longitudinal acceleration, lateral acceleration, yaw, speed, brake status (on/off), and steering wheel position. Longitude and Latitude provided by the Global Positioning System (GPS) were also needed to determine when the vehicle was at a particular intersection. The vehicle dynamics measurements will ultimately be combined with the behavioral data to determine which kinematic measurements are the best indicators of driving performance. This will be performed by a NODRTC research assistant once all of the subjects have been evaluated.

Due to uncontrollable factors such as traffic and weather, each FHWA participant will encounter different driving scenarios. This will certainly make it more difficult to evaluate driver performance based on the vehicle dynamics. For example, if a subject approached an intersection to make a left turn and had a green left-turn arrow, they would most likely slow down to make the turn and then accelerate up to a safe driving speed. On the other hand, if another subject approached the same intersection but had a red light, they would have to stop and wait for a green light to complete the turn. The longitudinal acceleration profiles for the two subjects would be drastically different. The first subject would most likely decelerate before the turn, while the other subject has to accelerate after being stopped at the red light. Therefore, drivers will need to be compared based on the driving conditions that they encounter. In order to evaluate all of the drivers for the FHWA study on an equal playing field, the NODRTC purchased a full-scale driving simulator. The simulator features a vehicle cab and three large projection screens for 180° viewing and a realistic driving experience. A picture of the simulator is provided in Figure 1-2.

The ten intersections from the road course have been recreated for the simulator using the driving scenario programming language. The simulator intersections were designed to be as similar to their road course counterparts as was possible using the scenario programming language. The simulator has some limitations including only being able to produce perpendicular intersections. Since one of the road course intersections is not perpendicular, it could not be recreated for the simulator. The advantage of evaluating drivers with the simulator is that they can all be scored for the exact same driving conditions. The simulator is capable of calculating all of the vehicle dynamics that are being measured on the road course. Since all drivers will encounter the same scenarios, the simulator should be helpful in determining which measures are good indicators of driving performance.

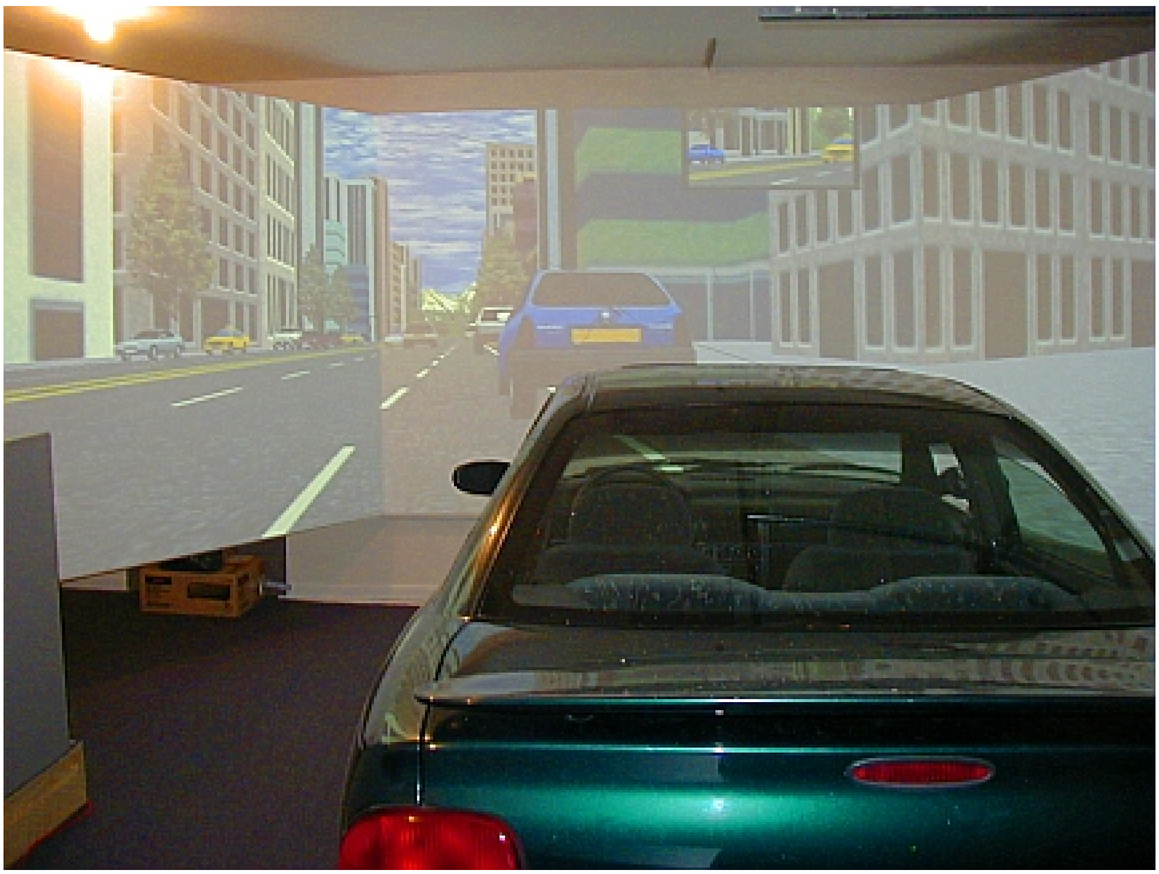


Figure 1-2. Picture of NODRTC Simulator showing vehicle cab and projection screens.

The rest of this paper will focus on the design, selection, installation, and data processing of the road course vehicle dynamic measuring system. A literature review of similar systems and driving research using such systems is provided in Chapter 2. The specific requirements for the system will be discussed in detail in Chapter 3. The overall design of the system and the selection and installation of components will be discussed in

Chapter 4. System verification and sample driving maneuvers will be analyzed in Chapter 5. Chapter 6 will review the processing and analysis of the data as it relates to the FHWA study. The results and discussion will be presented in Chapter 7. Finally, a conclusion is provided in Chapter 8.

CHAPTER 2
LITERATURE REVIEW

A literature review was conducted to gather information for two purposes. The first was to investigate other research studies which in some way used vehicle dynamics and/or a quantitative assessment to evaluate driving performance. The second was to evaluate off-the-shelf or "turn-key" devices which can be used to measure and record vehicle dynamics. It should be noted that the NODRTC conducted an extensive literature review of more than 2,500 articles related to elderly driving, driving rehabilitation, and intersection roadway design. The literature review presented in this chapter is solely concerned with instrumented vehicles, devices for acquiring vehicle dynamics, and relevant research studies.

2.1 Previous Research Studies

Wood and Worringham (2005) used a quantitative assessment to evaluate driving performance of drivers with Parkinson's disease. An accredited professional driving instructor and a trained occupational therapist accompanied each driver as they navigated a 19.4 kilometer open road course. The occupational therapist "scored seven aspects of driving performance at 147 locations along the route (general observation, observation of blind spots, indication (signaling), braking/acceleration, lane position, gap selection, and approach" (177). The occurrence and type of error was recorded for each location. Also, each of the 147 locations "was allocated to one of nine categories, including roundabouts, merging, pulling in/pulling out, traffic light and non-traffic light controlled intersections, reversing and parking, one way and two way traffic, and lane changing" (177). This allowed researchers to determine what types of errors were occurring most frequently and in which type of driving situation. The date was used to give each driver a driver safety rating. The study was used to correlate a driver's driver safety rating with the number of years since the diagnosis of Parkinson's disease. Although this research did not make use of an instrumented vehicle, it showed how an objective quantitative assessment could be used to evaluate driving performance.

Knodler et al. (2005) used a quantitative assessment and a "full-scale dynamic driving simulator as a tool for evaluating driver comprehension by placing drivers in a fully interactive scenario just as if they were actually driving" (270-271). The goal of the study was to determine which type of protected/permissive left-turn (PPLT) signal display was most effective. Again, this study did not make use of an instrumented vehicle however it used a full scale simulator to objectively measure driver behavior.

Another simulator study was conducted by McGehee et al. (2001) to investigate how long it takes for a participant to adapt to the simulator. McGehee et al. objectively measured a participants adaptation by measuring steering wheel reversals. "A steering wheel reversal is operationally defined as a deflection of the steering wheel away from a central or neutral position (i.e., where the wheels are completely straight) followed by a reversal of the steering wheel back to the neutral position" (2). In order to quantify steering wheel reversals a threshold of 6 degrees from neutral position was chosen. Steering wheel reversals were counted during a 25-minute drive and were used to determine how quickly a driver could adapt to the simulator. The study showed that on average "older drivers need about three minutes to adapt and get the "feel" of the simulator. Before this time driving behavior in the simulator may not be representative of actual driving performance" (4).

Rizzo et al. (2002) conducted a study using an instrumented vehicle because "road tests were designed to ensure that novice drivers know and can apply the rules of the road, and not to test older, experienced drivers who may be impaired" (291). Rizzo et al. also states "an instrumented vehicle permits quantitative assessments of driver performance in the field, under actual road conditions. These measurements are not subject to the type of human bias that affects inter-rater reliability on a standard road test" (291-292). The researchers developed an instrumented vehicle called ARGOS (the Automobile for Research in Ergonomics and Safety). "ARGOS was designed to examine objective indices of driving performance in normal and potentially unfit drivers and consists of a mid-seized vehicle with extensive instrumentation and sensors hidden within its infrastructure" (292). ARGOS also utilized video cameras and an example of the display output from ARGOS is shown in Figure 2-1.

Figure 2-1. Display Output from ARGOS Instrumented Vehicle (Rizzo et al., 292)

The four video images were multiplexed onto a single display. The values for the various measurements were superimposed over the video display. Uc et al. (2004) used ARGOS to evaluate route-following and safety errors in drivers with early Alzheimer disease and also provided a more detailed description of ARGOS. “Experimental performance data (e.g.,, steering wheel position, normalized acceleration and vehicle speed) were digitized at 10 Hz and reduced to mean SD (standard deviation), or count. Driver’s lane tracking and visual scanning activity of the environment were recorded by videotape at 30 frames/s using miniature lipstick-sized cameras mounted unobtrusively within the vehicle” (833). It is evident from Figure 2-1 that ARGOS has forward, rear, driver’s face, and driver’s hands cameras. Furthermore, ARGOS was capable of measuring accelerator (gas) pedal and brake pedal depression (measured as a percent of fully depressed). ARGOS was used successfully to quantitatively assess drivers with Alzheimer disease while performing a specific driving task.

Dingus and Hulse (1997) used an instrumented vehicle to determine the effects of age and experience on navigation while driving. “The system consisted of video cameras to record pertinent event and eye movement data, an experimenter control panel to record incident classification data, and sensors for the detection of variations in driving performance and behavior” (185). The system utilized four video cameras with nearly identical views as those used with ARGOS. “The system provided the capability to store a line of numerical data every 0.1 seconds” (185). It utilized accelerometers to measure lateral and longitudinal acceleration forces. The system also measured steering wheel position, vehicle speed, and brake pedal status (on/off). The research focused on using lateral acceleration measurements to indicate driver awareness. It also made use of the video cameras to time the driver's glancing eye movements. Although the study was mainly concerned with driver navigation and awareness rather than driver performance, it was able to use an instrumented vehicle to objectively quantify driver behavior.

Boyce and Geller (2002) used a vehicle with concealed video cameras and transducers to evaluate driver behavior, specifically younger males. The "computer generated dependent measures, in concert with real-time covert video recording of the participants' driving, allowed for unprecedented opportunities to perform a behavior analysis of driving performance in the context of normal driving. Thus, problems associated with truthfulness of self-report, infrequency of occurrence, and error variation due to in-vehicle observers were minimized or avoided altogether" (52). In addition to video recording, the vehicle also measured and recorded driver safety-belt use, number of times a turn signal was used, "vehicle velocity including average speed, velocity changes, and velocity variance, and following distance measured in meters" (53). Again, the video camera configuration was similar to that found in ARGOS. Although not directly related to driver performance, this study made use of an instrumented vehicle to measure driver behavior in the absence of a driving evaluator.

2.2 Devices

This section will offer a brief review of several commercially available products that are used to measure vehicle dynamics. These devices are called turn-key devices because they can be installed and ready to use right out of the box. In general, they are designed to serve a single purpose and usually not expandable or configurable.

2.2.1 DriveCam

The DriveCam is a device that is primarily used for vehicle fleet tracking and maintenance. It is installed behind the central rear view mirror in a vehicle. It features two video cameras to record forward-facing and rear-facing views. In addition, it has a longitudinal and a lateral accelerometer. A picture of the DriveCam is provided in Figure 2-2.

Figure 2-2. Picture of DriveCam.

The DriveCam uses removable digital media and as a result can only store small amounts of data (about 20 seconds). It records continuously and in the case of an event (e.g., hard turn or braking), it will store 10 seconds before and 10 seconds after the event. The measurements from the accelerometers are displayed at the bottom of the screen during video playback. The accelerometers are only accurate to a tenth of a g. The device can also be triggered by the user to save a specific event. Due to the limited number of measurements and lack of expandability, in addition to the short recording time, the DriveCam was not a suitable solution for the FHWA research.

2.2.2 Vericom

Vericom Computers Inc. manufactures the VC3000 which is an on board vehicle dynamometer. It features a two axis accelerometer which is used to measure longitudinal and lateral acceleration. It integrates the longitudinal acceleration to calculate speed and distance traveled. It is a self contained unit and is usually installed with suction cups to the inside of the passenger side windshield. The VC3000 can accept up to 6 additional sensor inputs. These sensors are sold by Vericom and only a limited selection is available. The data collected with the VC3000 can be downloaded to a computer and analyzed using the provided software. Figure 2-3 shows a picture of the Vericom VC3000.

Figure 2-3. Picture of Vericom VC3000.

The VC3000 is primarily used for vehicle performance tests (e.g., horsepower, braking) and traffic accident investigation and reconstruction. These tests are usually of short duration and consequently the VC3000 was designed with only enough flash memory to store approximately 21.8 minutes of data. This was the main reason that the VC3000 could not be used for the FHWA research. Furthermore, it is not completely expandable and the software has limited capabilities.

2.2.3 Dewetron

Dewetron is a worldwide manufacturer of portable data acquisition systems. They offer several different products depending on the requirements of the application. Their products are essentially personal computers combined with data acquisition hardware and Dewetron's own proprietary software. The units are capable of measuring nearly any type of signal including video and GPS. Dewetron offers desktop as well as portable products including some that are battery powered. Figure 2-4 shows a picture of one of the Dewetron units.

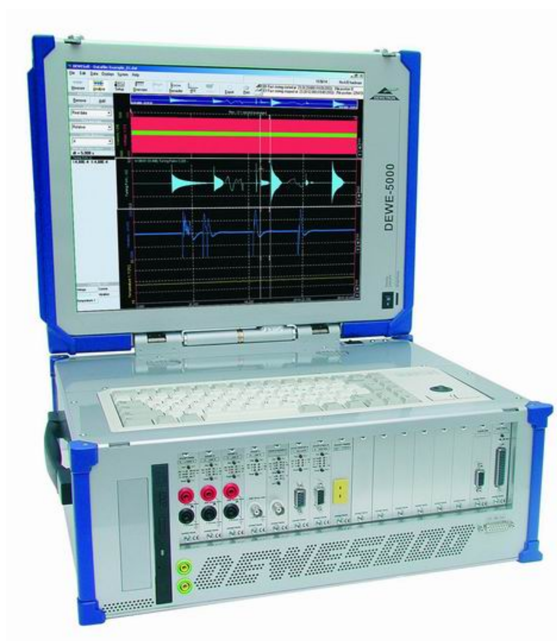


Figure 2-4. Picture of Dewetron DEWE5000.

The Dewetron systems are very sophisticated and the software is specifically designed for evaluating vehicle performance. However, in the context of the FHWA research, the Dewetron system would only be used as a data recorder. The software used to analyze the data from the FHWA road course would still have to be developed. As a result, the Dewetron system would be too expensive to use especially considering that its full functionality is not being utilized.

CHAPTER 3
SYSTEM REQUIREMENTS

After reviewing several off-the-shelf products with their own advantages and disadvantages, a decision was made to design a system from scratch that satisfies all of the requirements. The system must be capable of measuring accelerations, speed, yaw, steering wheel position, longitude, latitude, and brake and turn signal status. Additionally, the system should be affordable, easy to operate, and flexible. Designing a customized system allows it to be configurable and used for a wide variety of applications. This chapter will discuss the specific goals and performance requirements of the system as well as additional considerations including cost, ease of use, and flexibility.

3.1 Performance

The system should offer good performance in the form of accuracy and high sampling rate. This system will likely be used for future studies other than the FHWA research. Therefore, it is necessary that the system offers a high level of performance such that future upgrades will not be necessary. The system should have a resolution of at least 16-bits to provide accurate sampling of each channel. A 16-bit system would allow a voltage resolution of 0.0001526 volts for an input range of 10 volts. In addition to accuracy, sampling rate is another important performance specification. A digital data acquisition system can only sample an analog signal at discrete times. Therefore, a high sampling rate is needed to accurately reproduce the original analog signal. If frequency measurements are desired, the sampling frequency must be at least twice the highest frequency component of the analog signal according to the Nyquist Theorem. This is to prevent aliasing, which occurs when the sampled signal appears to have a different frequency than the original analog signal. Frequency measurements, for example frequency of steering wheel reversals, are not being considered for the FHWA study. However, it will still be necessary to accurately recreate the original signal. Furthermore, frequency measurements (e.g., steering wheel input) may be utilized in future NODRTC research. It was decided that the system should be capable of sampling each channel at a minimum of 10,000 samples per second (10k Hz). This will ensure accurate reconstruction of the analog signal and allow for frequency measurements up to 5000 Hz. This is certainly capable of measuring any input physically generated by a human. Also, it is doubtful that any of the vehicle dynamics will have frequency components of this magnitude.

3.2 Transducers

The accelerometers used for measuring longitudinal and lateral acceleration need to be able to accurately measure at low frequencies (i.e., less than 10 Hz). Measurement of the high frequency vehicle vibration is not important to this study. The changes in acceleration will be gradual and the maximum acceleration in any direction under normal driving conditions is not expected to exceed 1 g.

The transducer that is used to measure the steering wheel position should be non-intrusive. It should be able to accurately provide the position of the steering wheel at any instant. The sensor should not add any extra resistance to the steering wheel that may affect the driver's ability to turn the wheel.

As with the accelerometers, the transducer that measures yaw should be able to measure accurately at low frequencies. Under normal driving conditions, the maximum yaw is not expected to exceed 90°/sec. This would be the equivalent of completing a 90°

turn in one second, which is unrealistic for everyday driving. The yaw sensor should be able to determine the direction and rate that the vehicle is turning.

The transducer used to measure vehicle speed should be non-intrusive. Velocity can be derived from the longitudinal acceleration, but a more direct measurement would be preferable. It may be possible to acquire vehicle speed via the car's on board diagnostic system.

The data acquisition system should be able to determine if the driver is using the brake and the turn signals. Additionally, the evaluators requested that an event trigger be installed in the vehicle cab. The event trigger will be used as a way to mark the data in the case of an adverse event for later review.

3.3 GPS

A global positioning system receiver and antenna is needed to acquire the vehicle's location while driving on the road course. The receiver should be WAAS capable and easily integrated into the data acquisition system. The Wide Area Augmentation System (WAAS) consists of 25 ground reference stations and two geostationary satellites orbiting above North America. The ground reference stations monitor the GPS satellites and correct for errors caused by satellite position error, clock drift, and atmospheric effects (Garmin Ltd.). The corrected signals are then relayed to the geostationary satellites. The correction message can be received from the geostationary satellites using a standard WAAS-capable GPS receiver. A GPS receiver using WAAS has an accuracy of less than 3 meters 95 percent of the time. Differential GPS (DGPS) uses a local reference receiver, whose location is known exactly, to calculate the error in the GPS signals. The error signals are then broadcast, usually via radio, to the other GPS receivers. DGPS requires a reference receiver and radio transmitters and is consequently more costly than a WAAS-capable receiver. As a result, DGPS will not be considered for this project.

3.4 Video

The video system should consist of a forward view, rear view, and a view of the driver. The driver view will be used after a driving assessment to evaluate the driver's eye movements. All of the views should be recorded simultaneously to a single digital medium. Digital video will provide distortion-free frame by frame playback. The media needs to be long enough so that the entire road course will fit on a single recording.

3.5 Ease of Use

The data acquisition system will be primarily operated by Occupational Therapists and therefore the system should be user friendly and able to accommodate users with minimal technical experience. The end user should not have to calibrate or configure the system in any way. Consideration will be taken in the design of the system such that the user can operate it after receiving minimal training.

3.6 Cost

The NODRTC has received substantial funding from the FHWA. In addition to the data acquisition system, the budget must be used for purchasing the road course vehicle, driving simulator equipment, and vehicle cab for the driving simulator. Therefore the costs for the vehicle dynamics acquisition system should be minimized if possible. Regardless of the data acquisition system used, the transducers and sensors will cost the same. The system should be expandable so that the option to use additional sensors for the FHWA or other research is available.

3.7 Flexibility

This system will likely be used for other older driver research at the conclusion of the FHWA study. The system must be flexible and adaptable to other areas of research. For example, the FHWA research is only concerned with 10 specific intersections whereas other research may be interested in driver performance on an entire course. Furthermore, different measurements, possibly physiological, may be desired for future research.

CHAPTER 4
SYSTEM DESIGN

This chapter will discuss the selection, installation, and operation of the entire system including video. The selection of components including the data acquisition system, transducers, GPS, and video equipment will be discussed in detail. A description of the vehicle and its driving aids will also be presented.

4.1 Vehicle Description

The system was installed in a 2005 Buick Century Sedan. A picture of the vehicle is provided in Figure 4-1.

Figure 4-1. Exterior View of 2005 Buick Century Sedan NODRTC Research Vehicle.

The vehicle is equipped with adaptive driving equipment for handicapped drivers. There is a left foot accelerator pedal adapter that mounts to the floor and replaces the standard accelerator pedal. The adapter is secured to a bracket that is permanently attached to the floor using locking pins. This adaptation is for drivers who have limited or no use of their right leg. The adapter features a metal plate to prevent the driver from accidentally pressing the original accelerator with their right leg. Figure 4-2 shows a picture of the left foot accelerator adapter.

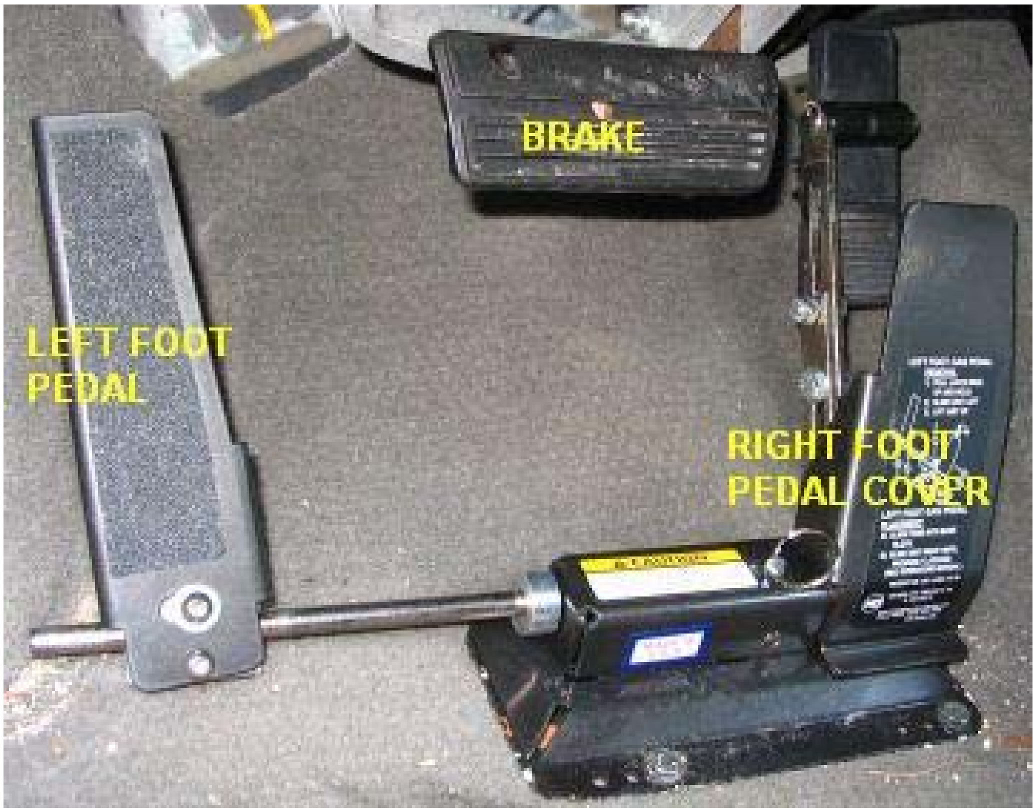


Figure 4-2. Picture of Left Foot Accelerator Pedal Adapter.

For driver's who have limited use of both legs, there are hand controls. The hand control is a lever on the left side of the steering wheel that is mechanically linked to the brake and accelerator pedals. The driver rotates the lever to accelerate and pushes it forward to brake. Since the driver must use their left hand to control the accelerator and brake, there is a turn signal extender that allows the driver to activate the turn signal with their right hand. The car also features three receptacles on the steering wheel for spinner knobs. The knobs enable a driver suffering from arthritis to turn the steering wheel easier. A picture of the hand controls and steering wheel receptacles is shown in Figure 4-3.

Figure 4-3. Picture of Hand Controls, Turn Signal Extender, and Steering Wheel Receptacles.

In addition to the adaptive driving equipment, the car is also equipped with an auxiliary brake pedal for the evaluator. The auxiliary brake pedal is shown in Figure 4-4.

Figure 4-4. Picture of Auxiliary Brake Pedal.

The auxiliary brake pedal is mechanically linked to the standard brake pedal using a steel cable and pulleys. The auxiliary brake is used by the evaluator if a driver exhibits reckless driving behavior. Also, the car was purchased with a bench seat which allows the evaluator to grab the steering wheel and take control of the vehicle if necessary.

4.2 Selection of Components

4.2.1 Data Acquisition System

The data acquisition system is the centerpiece of the design. All of the sensors are connected to it and it is used to record all data. It is also the user interface for the entire system. The system was installed in a car and must be mobile, so a desktop computer would not be practical. The decision was made to use a powerful laptop mated with a data acquisition card. The laptop chosen is a Dell Inspiron with a Pentium 4 2.4 GHz processor and 512 MB of RAM. The data acquisition card is a National Instruments DAQCard-6036E with PCMCIA connectivity. The card is 16-bit and is capable of 200,000 samples per second. This allows sampling rates of 10,000 samples per second for 20 channels. The card is connected to a Signal Conditioning Carrier (SCC) via a 68-pin cable. The SCC is the link between the sensors and the data acquisition card. Each SCC can hold several signal conditioning modules, which are customized inputs depending on the type of sensors used. The modules allow connection of different transducers (e.g., accelerometers, thermocouples, and strain gauges) to a single measuring device. The SCC most appropriate for this application is the SC-2345. The SC-2345 can hold up to 20 modules (16 inputs, 2 outputs, 2 counters) and has 8 digital I/O lines built in. There are three options for powering the SC-2345: 5 volt DC power from a DAQ device, AC power, or a 7 to 42 volt DC power supply. AC power is required to power the laptop, so it will also be used to power the SC-2345. In addition to selecting the right module for each sensor input, the physical connection can also be specified. There are a wide variety of connections including BNC, 9-pin D-sub, SMB, MIL-Spec, and LEMO. These connections are referred to as panelettes and any combination may be used. For this application, all of the panelettes are BNC and have two inputs per panelette. The panelettes have lead wires which are connected to the screw terminals of each module. This allows each sensor to be connected using its native connector. A simple wiring diagram is shown in Figure 4-5 and a picture of the SCC is provided in Figure 4-6.

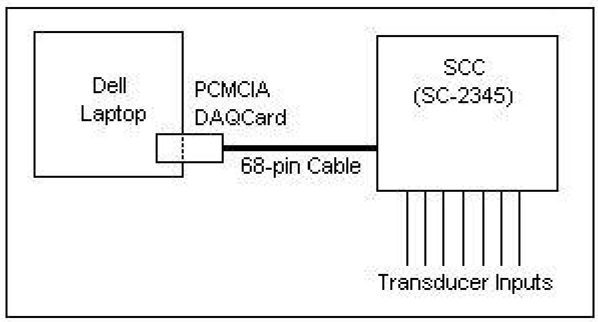


Figure 4-5. Wiring Diagram for Data Acquisition System.

Figure 4-6. Picture of National Instruments Signal Conditioning Carrier (SC-2345) and Modules.

The selection of modules will be discussed following the description of the sensors. The sensors must be selected first so that the proper type and input range of each module can be chosen.

4.2.2 Transducers

The section will discuss the selection of each transducer. The basis for selection and relevant specifications will be provided.

Accelerometers. It was determined that the accelerometers used to measure longitudinal and lateral acceleration need to be capable of measuring at extremely low frequencies. The decision was made to use a pair of static accelerometers manufactured by Kistler. These accelerometers can measure accelerations in the range of 0 to 300 Hertz. The accelerometers utilize a variable capacitance sensing element. A small inertial mass is attached to a flexure element cantilevered between two plates. The mass deflects under acceleration resulting in a change in the capacitance between the two plates. Internal circuitry provides AC excitation and synchronous amplitude demodulation resulting in a voltage output that is proportional to the applied acceleration. There are two models available (8310A2 and 8310A10) with ranges of either ± 2 or ± 10 *g*'s. The 10 *g* model was chosen to measure longitudinal acceleration while the 2 *g* model was chosen for lateral acceleration. The 2 *g* model is used to measure lateral acceleration because even a high-end sports car cannot exceed 1*g* laterally on a skidpad (Road & Track). By choosing the accelerometer with the smaller range, the lateral acceleration can be measured more precisely. The 10 *g* model was chosen to measure longitudinal acceleration. Under normal driving conditions the longitudinal acceleration is not expected to exceed 2 *g*'s in either direction. However, it may be possible to exceed 2 *g*'s under heavy breaking or in a collision. Therefore the 10 *g* accelerometer is used to measure longitudinal acceleration to prevent saturation of the signal at 2 *g*'s. Technical specifications for the accelerometers and all other transducers are provided in Appendix

B. Installation and power requirements for the accelerometers will be discussed later in the chapter.

Angular rate sensor (Yaw). An angular rate sensor was chosen to measure the vehicle yaw (i.e., rate of rotation). The angular rate sensor is manufactured by BEI Technologies Inc., and is called the GyroChip (AQRS-00075-111). It is a compact, rugged micromachined vibrating quartz gyroscope, designed specifically for demanding automotive and commercial applications (BEI 964172 Rev. A). The GyroChip uses a double-ended quart tuning fork made from monocrystalline piezoelectric quartz. As a result of the Coriolis effect, there is a voltage output directly proportional to the rate of rotation. The use of a single piezoelectric crystal makes it less susceptible to variations in temperature and extends its life. The maximum rate of rotation the GyroChip is capable of measuring is 75 degrees per second. It is expected that this rate would not be exceeded under normal driving conditions. If the car were to exceed this rate and sustain it, that would mean it is capable of completing a 90 degree turn in just over a second. This is certainly not realistic, however a strong jerk on the steering wheel may cause the car to approach the maximum rate. Furthermore, the GyroChip is designed especially for use with automotive stability control systems so it is an appropriate choice for measuring vehicle yaw.

String potentiometer (steering wheel position). Measuring the position of the steering wheel accurately and in a non-obtrusive manner proved to be difficult. An absolute rotary encoder was first considered, but the decision was made to use a string potentiometer and custom made pulley. It is not possible to mount a rotary encoder on the steering shaft so it would have to be connected using a pulley or gear. Knowing that a pulley or gear would be needed to connect a sensor to the shaft, a string potentiometer was chosen instead. Most absolute rotary encoders transmit a digital word to the data acquisition system. It is much easier to use a string potentiometer because it outputs a voltage directly proportional to displacement. When attached to a pulley on the steering shaft, it could accurately measure the position of the steering wheel. A PT1DC Cable-Extension Position Transducer (PT1DC-50-BK-Z10-MC4-SG) produced by Celesco was selected. Several aspects of the PT1DC are customizable including cable length, cable exit location, output voltage, and connection type. The PT1DC features internal power conditioning so it does not require a regulated power supply. The cable tension is only 5 ounces which will apply a negligible amount of torque (less than 0.1 ft-lb.) to the steering shaft. This small amount of torque will have negligible effect on the position of the steering wheel or the effort needed to rotate it in either direction. The maximum available range of 50 inches was selected in case the string potentiometer could not be mounted close to the steering shaft. If absolutely necessary, several pulleys could be used to route the cable from the sensor to the pulley on the steering shaft. The design of the custom made pulley for the steering shaft and the installation of the string potentiometer will be discussed later in the chapter.

Laser tachometer (Speed). As with steering wheel position, measuring speed in a non-obtrusive manner proved to be a challenge. Obtaining vehicle speed from the On-Board Diagnostic system was investigated but proved to be too difficult to integrate with the data acquisition system. The LaserTach ICP tachometer from The Modal Shop Inc. was chosen to measure vehicle speed. The LaserTach emits a laser and has a sensor to detect when it is reflected. A retroreflective target is placed on the rotating shaft to reflect the laser. The LaserTach produces a voltage pulse each time the laser is reflected. The frequency of the pulse signal is the rotational frequency of the shaft. Integrated Charge Pre-Amplifier (ICP) sensors have a built-in microelectronic amplifier which results in a low impedance coaxial system. An external ICP power supply or a data acquisition system capable of powering ICP sensors must be used to provide power to the sensor.

Event trigger, brake, and turn signals. The event trigger, brake, and turn signals are essentially identical. They are simply coaxial cables connected to each signal individually. The coaxial cables are wired in parallel with the brake and turn signal lights. The other end of each coaxial cable is connected to the data acquisition system. The event trigger is slightly different in that it must be supplied a voltage. The event trigger is a simple push-button with one terminal connected to +12 volts from the vehicle's electrical system and the other terminal wired to the data acquisition system. The event trigger input to the data acquisition system is coupled with a ground wire to determine when the trigger is pushed. All of these signals will generate approximately 12 volts when activated. A simple logic statement can be implemented in the software such that if the voltage is above a certain threshold the signal will be considered activated.

4.2.3 Modules

As previously mentioned, the input modules for the SCC can be selected to match the output of each sensor. Once all of the transducers were selected, the appropriate modules could be chosen. All of the sensors chosen will require modules with analog inputs. Each analog input module contains two individual inputs and are available with different input voltage ranges. A total of five modules will be needed to accommodate all of the inputs. It was determined that (2) ±5 volt, (1) ±10 volt, and (2) ±20 volt modules would be needed. The specifications for each module are provided in Table 4-1.

Table 4-1. Technical Specifications for Signal Conditioning Analog Input Modules.

Model	Quantity	Channels	Input Range	Bandwidth
SCC-AI02	2	2	± 20 volts	10 kHz
SCC-AI03	1	2	± 10 volts	10 kHz
SCC-AI04	2	2	± 5 volts	10 kHz

Each channel of each analog input module contains an instrumentation amplifier, a lowpass filter, and a potentiometer for calibration. Table 4-1 displays the output range for each transducer and the input range for the selected module.

Table 4-2. Transducer Output Ranges and Module Input Ranges.

Transducer	Output Range (V)	Module	Input Range (V)
Longitudinal Accelerometer	0 – 2	SCC-AI04	± 5
Lateral Accelerometer	0 – 2	SCC-AI04	± 5
Angular Rate Sensor	0.25 – 4.75	SCC-AI04	± 5
String Potentiometer	0 – 10	SCC-AI03	± 10
Laser Tachometer	0 – 1	SCC-AI04	± 5
Brake	0 – 12	SCC-AI02	± 20
Left Turn Signal	0 – 12	SCC-AI02	± 20
Right Turn Signal	0 – 12	SCC-AI02	± 20
Event Trigger	0 – 12	SCC-AI02	± 20

All of the modules will be connected to panelettes with BNC connections. BNC offers a quick and reliable connection and is also the native form of connectivity for all of the sensors.

4.2.4 Power Supplies

The accelerometers, angular rate sensor, and laser tachometer all require specialized power supplies. The string potentiometer power supply does not need to be regulated, nor does the supply voltage for the event trigger. The brake and turn signal connections will be powered by the voltage provided to the respective lights.

The power supplies for the accelerometers are also manufactured by Kistler and run off of a single 9-volt battery. The power supplies are specifically designed for capacitive accelerometers. Each power supply is connected to an accelerometer via a 4-pin Microtech connector. The output of each power supply uses a BNC connector and is easily connected to the data acquisition system. Each power supply provides the appropriate voltage and current to each accelerometer. They feature internal jumpers to adjust the gain and a potentiometer to adjust the output voltage bias. The model is Type 5210 and a picture is provided in Figure 4-7.

Figure 4-7. Picture of Kistler Type 5210 Capacitive Accelerometer Power Supply.

The laser tachometer is an ICP sensor so a standard off-the-shelf ICP power supply may be used. The selected power supply is a Model 480E09 manufactured by PCB. The power supply uses BNC connections for both the input and output. It features an adjustable gain and provides the proper voltage and current for the laser tachometer. The power supply operates off of three 9-volt batteries providing approximately 50 hours of battery life. A picture of the power supply is provided in Figure 4-8.

Figure 4-8. Picture of PCB Model 480E09 Power Supply for ICP Laser Tachometer.

A suitable power supply for the angular rate sensor could not be found so it had to be fabricated. The angular rate sensor requires a regulated power supply of 5 volts DC. The decision was made to build the power supply using a voltage regulator integrated circuit. Although the string potentiometer does not require a regulated power supply, it was incorporated into the angular rate power supply so that both could be controlled using a single DPST (double pole, single throw) switch. This will reduce the number of switches that need to be switched when using the system. A wiring diagram of the power supply, including the DPST switch, is provided in Figure 4-9.

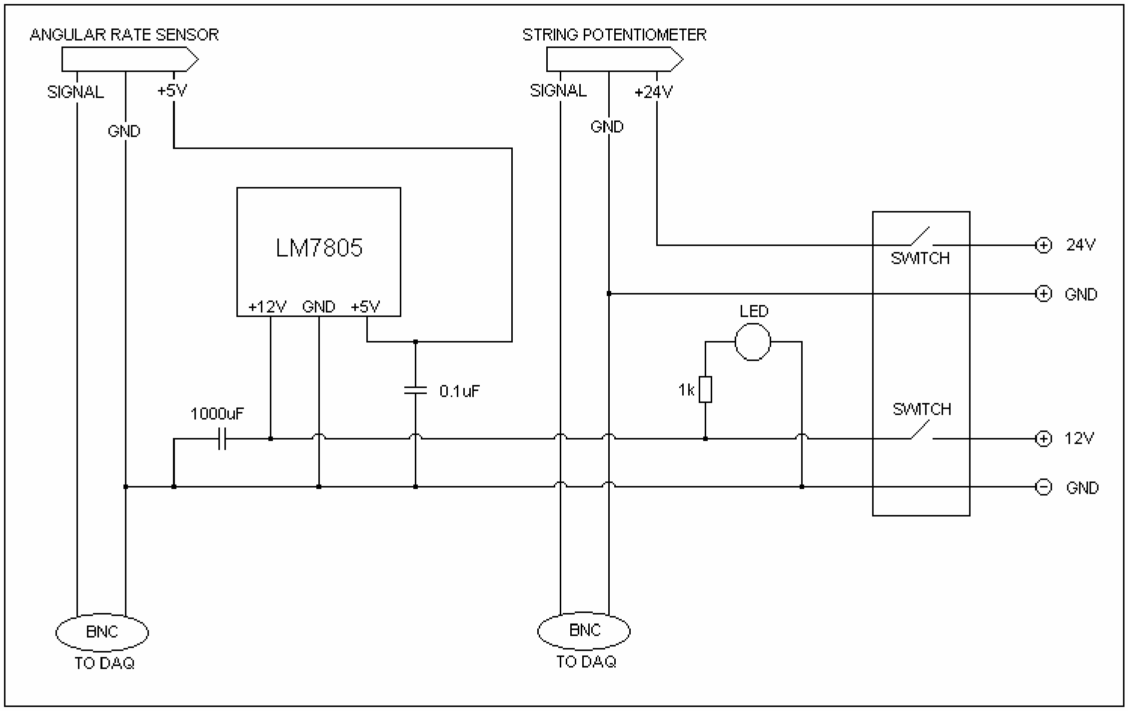


Figure 4-9. Wiring Diagram of Angular Rate Sensor Power Supply.

The power supply accepts +12 volts to provide power to the regulating circuit and + 24 volts to power the string potentiometer. A LM7805 voltage regulator chip is used to provide a constant + 5 volts to the angular rate sensor. The 1000 µF capacitor helps to maintain constant input to the regulator while the 0.1 µF capacitor eliminates any high frequency pulses that may interfere with operation of the regulator. The LM7805 requires an input voltage at least two to three volts greater than the output voltage. The car's electrical system provides approximately + 12 volts to the regulator chip, which satisfies the input voltage criterion. The + 24 volts is provided by a DC-DC converter normally used to supply power for a laptop. The DC-DC converter accepts +12 volts and steps it up to + 24 volts. The + 24 volts does not need to be regulated so a simple switch can be used to turn it on and off. The LED is used to indicate the on/off status of the power supply. Since the power supply receives power from the car's battery, it could drain the battery if left on. A 3-pin connector is used to connect to each sensor to the power supply. BNC connections are used for the outputs of the power supply for a quick a reliable connection to the data acquisition system. All of the circuitry for the power supply is contained in a plastic enclosure. The inputs to the power supply use 2-pin connectors so the entire power supply can be removed or replaced quickly.

AC power is provided by a VectorMaxx 750-watt power inverter. The power inverter is powered by the car's 12 volt electrical system. The output is a modified sine wave with a frequency of 60 Hz and an amplitude of 115 volts (rms), similar to a standard household outlet. The power inverter is connected to a Monster Cable brand power strip for distribution of the AC power. All of the devices that use AC power are connected to the power strip. The power inverter features short-circuit and overload protection circuitry.

4.2.5 GPS

A Garmin GPS 16 was selected for the GPS receiver. The GPS 16 is WAAS capable and uses the NMEA-0183 standard. It is enclosed in a rugged, waterproof enclosure and features a magnetic bottom for quick installation. Since the laptop does not have a serial port, a USB/Serial converter will be used to connect the receiver. The converter also provides power to the receiver which cannot be done with a standard serial port.

4.2.6 Video

The video system consists of four ultra-mini pinhole cameras (CCM630P), a quad multiplexer (QC800), and a portable mini-DV recorder (Sony GVD-1000). The cameras are color video cameras and measure 1.25 inches square. There is a choice of either a 78° or 90° viewing angle. The version with the 90° viewing angle was chosen because it provided greater visibility for each camera. Each camera uses a 1/3-inch high-resolution CCD sensor and features an automatic iris and backlight compensation. The cameras are powered with 12 volts DC so the car's electrical system will be used to power them. A fuse was installed for protection as well as a switch to turn the cameras off when not in operation.

The quad multiplexer is a device most commonly used for security applications. It combines the video signal from up to four cameras into one signal. There are several options for displaying the four views: a large main view with the other three views shown at the bottom, a large main view with a smaller window for a second view, a large main view sequentially shifting between all cameras, and all four views shown at the same time in equally sized windows. The latter option was selected since each view is equally important.

A portable mini-DV recorder (Sony GV-D1000) was chosen for its exceptional quality and ease of use. The recorder features a 4-inch LCD screen to verify the system is working properly and cameras are aligned properly before recording. The mini-DV format is a digital format with up to 500 lines of horizontal resolution. This is better resolution than Hi-8 and S-VHS and it is digital. The digital format allows it to be transferred to a computer easily for backup or editing. Some mini-DV tapes feature memory chips that can be used to save information such as the date and time of the recording. The memory chip can also be used to build a table of contents. The table of contents contains a list of specific points in the recording. These points can be used for easier access during playback or for editing to be done later.

The Quad multiplexer and mini-DV recorder use standard AC power and are powered by the power inverter. Composite (RCA) video cables are used to connect the cameras to the multiplexer and the multiplexer to the recorder. The mini-DV tapes can record 90 minutes in EP mode. This is long enough to record the entire road course on a single tape since the road course takes approximately 1 hour and 20 minutes to complete. The entire video system is completely independent from the data acquisition system. The only device the two systems share is the power inverter. The video will be used to further review a subject's performance if necessary. It may also be used as a training tool for new evaluators.

4.2.7 Software

National Instruments' proprietary software LabVIEW (version 6.1) will be used to operate the data acquisition system. LabVIEW is a graphical programming language designed specifically for interaction with National Instruments hardware. It features built-in measurement and control subroutines, called virtual instruments, which allow for fast and easy program development. A combination of several virtual instruments are wired together to create a working program. A virtual instrument can be changed or created from scratch to perform a custom operation. The user interface for the final program can be configured to hide or show various controls (inputs) and indicators (outputs). The program to be used for this system is fairly simple. It will be required to measure all of the channels from the data acquisition system and record the data to the laptop hard drive. National Instruments also provides the Measurement and Automatic Explorer (MAX) which is used to configure the entire system. The National Instruments hardware including the DAQ-Card, SCC, and SCC modules are entered in to MAX. Additionally, MAX can be used to monitor each channel to ensure that each sensor is working properly.

GPSLog software developed by U.S. Positioning LLC will be used to record the GPS data. GPSLog is a simple program that records the output sentences from any NMEA-0183 compliant GPS receiver. NMEA-0183 is a standard defining signal requirements, data transmission protocols, and sentence formats for 4800-baud serial data buses, which are widely used for GPS receiver communication (National Marine Electronics Association). GPSLog can save either the raw data sentences or just important information such as longitude, latitude, speed, and heading.

The MathWorks MATLAB software will be used to process and analyze the data. MATLAB is a high-level technical computing language which is computationally faster than traditional programming languages like C++ and Fortran (MathWorks Inc.). MATLAB is the ideal application for inputting and analyzing the data. It features toolboxes which contain subroutines for performing data analysis and statistics. MATLAB is a very powerful application however it will be mainly used for simple operations such as matrix manipulation and statistical analysis.

4.3 Installation

The system was installed in the vehicle in the least conspicuous way possible in order to have a minimal impact on the driver. The driver is most likely driving a vehicle different from their own and any equipment in plain view may make them nervous. The majority of the equipment including the power inverter, SCC, power supplies, and video equipment was installed in the trunk of the vehicle. A sheet of medium-density fiberboard (MDF) was cut to fit in the trunk and was used as a mounting platform for the equipment.

A picture of the equipment installed in the trunk and a diagram showing the approximate location of each component is shown in Figures 4-10 and 4-10 respectively.

Figure 4-10. Picture of Equipment Installed in Trunk.

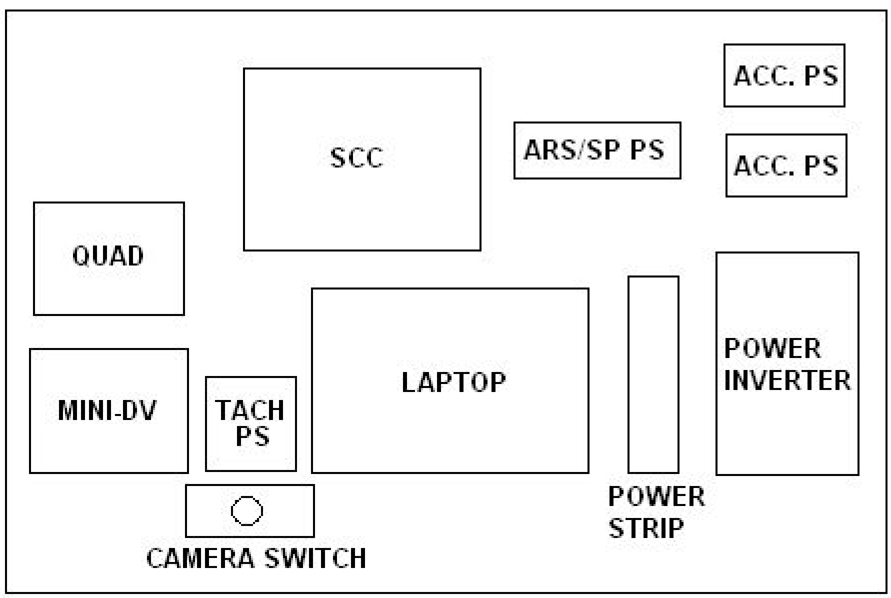


Figure 4-11. Trunk Diagram Displaying the Approximate Location of Each Component. (QUAD → Video Multiplexer, PS → Power Supply, ARS → Angular Rate Sensor, SP → String Potentiometer, Acc. → Accelerometer)

The laptop was originally installed in the trunk however the Occupational Therapists requested it be located in the back seat so that it could be operated from inside the vehicle.

There were few options for the locations of the laser tachometer and string potentiometer. The laser tachometer had to be installed near either one of the front wheels because the vehicle is front-wheel driven and does not have a rear axle. The string potentiometer needed to be installed close to the steering shaft because it must be mechanically attached to the shaft. A bracket included with the laser tachometer was fastened to an existing mounting bracket for one of the car's brake lines. The laser tachometer is threaded and was installed in the bracket using the provided hardware. The front axle was painted flat black and a ¼-inch wide piece of retroreflective tape was installed. A picture of the laser tachometer setup is provided in Figure 4-12.

The tachometer was aimed at the retroreflective tape and the laser beam was focused using the adjustment screw. When the sensor is powered, a LED illuminates to indicate that the laser is being reflected and measured properly.

The string potentiometer is physically attached to the steering shaft via a pulley and therefore had to be installed relatively close to the shaft. It was not possible to remove the steering shaft in order to install an off-the-shelf pulley. The only feasible option was to design a pulley that could be attached to the shaft without having to remove it. The pulley is split in half so that it can be secured to the shaft using socket cap screws. It features a groove to guide the cable from the string potentiometer as it winds around the pulley. The steering shaft is not perfectly round and consequently the inside of the pulley had a unique geometry. The pulley was machined out of a scrap piece of aluminum using an end-mill and lathe. The best location that for the string potentiometer that allowed the cable to reach the pulley unimpeded was on the lower part of the firewall. The string potentiometer was secured to the firewall using J-B Weld two-part epoxy. A picture of the string potentiometer and pulley assembly installed in the vehicle is provided in Figure 4-13.

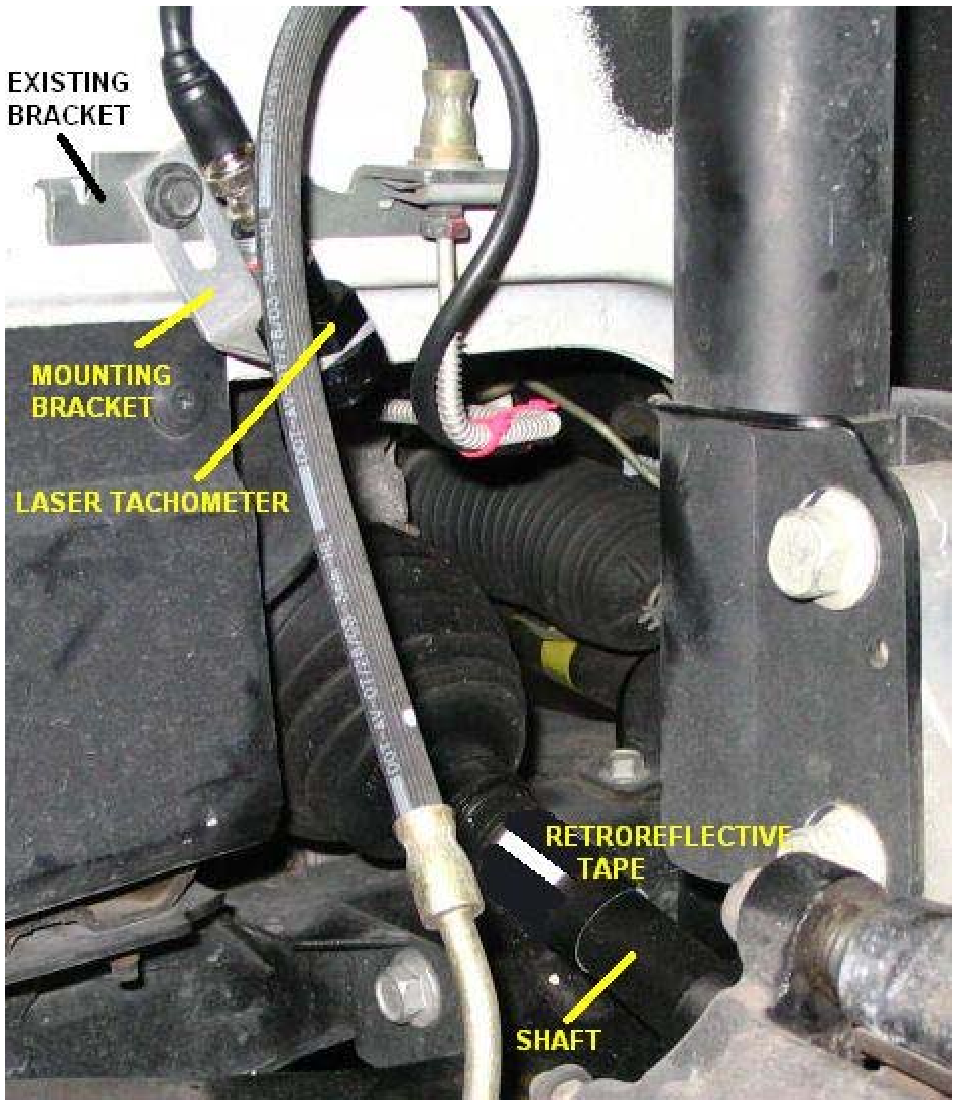


Figure 4-12. Picture of Laser Tachometer, Mounting Bracket, and Drive Shaft.

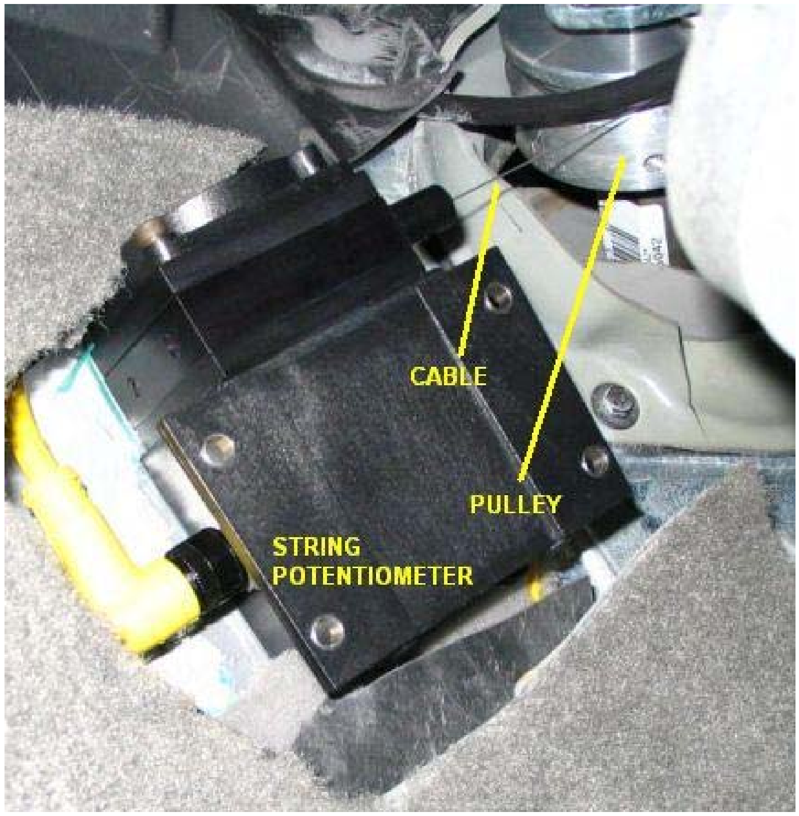


Figure 4-13. Picture of Installed Sting Potentiometer and Custom Pulley Assembly.

A drawing of the custom pulley was generated using Pro Engineer modeling software and can be found in Appendix C.

The accelerometers and angular rate sensor do not need to be located near any particular area of the car. The main requirement is that they are attached securely to a rigid part of the vehicle in order to measure accelerations and yaw accurately. Furthermore, the three sensors should be located as close to the center of the car as is feasibly possible. This is to reduce the amount of measured acceleration that may be a result of suspension movement. The best location that would also allow for the sensors to be out of sight was under either one of the front seats. The driver's seat features power controls and as a result there are several motors underneath it. Therefore, the only option was to install the sensors under the front passenger seat. A piece of carpet and insulating foam was removed to allow access to the structural members of the vehicle. A frame rail that ran laterally across the vehicle was used as a guide to position the longitudinal accelerometer. The longitudinal accelerometer was located along the frame rail such that it measured in a direction perpendicular to the frame rail. The orientation of the angular rate sensor did not matter as long as it was positioned level so that it measured rotation about the vertical axis of the car. It was positioned along the frame rail so that a perpendicular face on the angular rate sensor could be used as a mounting location for the lateral accelerometer. All of the sensors were attached using J-B Weld two-part epoxy. A picture of the sensors installed beneath the passenger seat is provided in Figure 4-14.

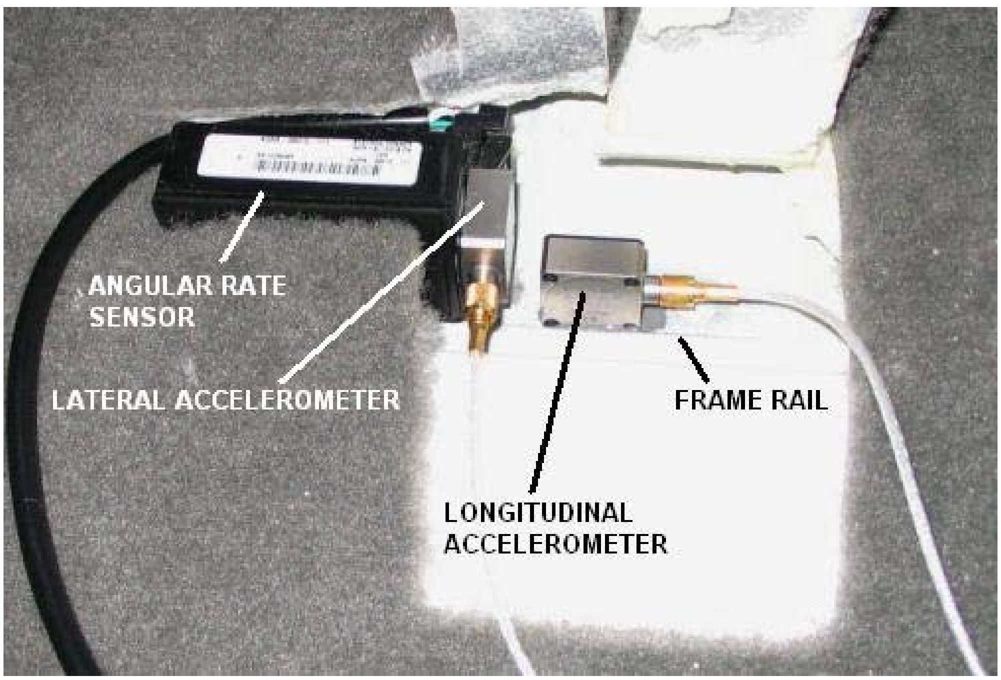


Figure 4-14. Picture of Lateral Accelerometer, Longitudinal Accelerometer, and Angular Rate Sensor Installed Underneath the Front Passenger Seat.

When the carpet is replaced and the passenger seat is in its normal position, the sensors are completely hidden.

The brake and turn signals were measured by wiring the positive and negative (ground) leads of each in parallel with the inputs to their respective modules. All the signals measure approximately 12 volts when in use. A logic statement was implemented in LabVIEW that determines if the signal is above 6 volts. The result is a 1 corresponding to the signal being activated and a 0 for it not being activated. The event trigger was wired in the same was as the other signals except the positive 12 volts had to be supplied. The event trigger is simply a pushbutton connected to the car's positive 12 volts and an input to a module. The other input to the module is connected to the car's frame (ground). The same logic used for the other signals was also used for the event trigger.

The GPS antenna was attached to the top of the trunk using the magnetic mount. It was wired to a serial/USB converter which was connected to the USB port of the laptop.

The positive lead of the power inverter was connected to the positive terminal of the car's battery and the negative lead was connected to the vehicle chassis (ground). A 40-ampere fuse was installed on the positive lead near the battery to protect the inverter and all of the equipment connected to it.

The multiplexer and mini-DV recorder was located in the trunk of the vehicle with the rest of the equipment. The four cameras were installed in the ceiling of the vehicle cab. They came with brackets which allowed them to be installed by screwing the bracket into the rigid foam ceiling. The orientation of the cameras is completely adjustable. Each one was wired with 12 volt DC power and a composite video cable. A picture of the camera which monitors the driver's eye movements is provided in Figure 4-15.

Figure 4-15. Picture of Miniature Video Camera and Installation Bracket.

The video cable from each camera was wired to an input of the multiplexer. The output of the multiplexer was wired to the input of the mini-DV recorder using a composite video cable.

All of the wires for the various sensors, cameras, and power inverter were concealed beneath the door sills, carpet, and headliner. The only evidence inside the vehicle cab that would indicate the presence of special equipment is the four video cameras. All of the cameras are mounted to the ceiling and although they are visible they should not interfere with the driver's field of view. A wiring diagram of the entire system is presented in Figure 4-16. The figure shows how the various components are connected and the cables that were used. Figure 4-17 is also provided to show the location of the components in the vehicle.

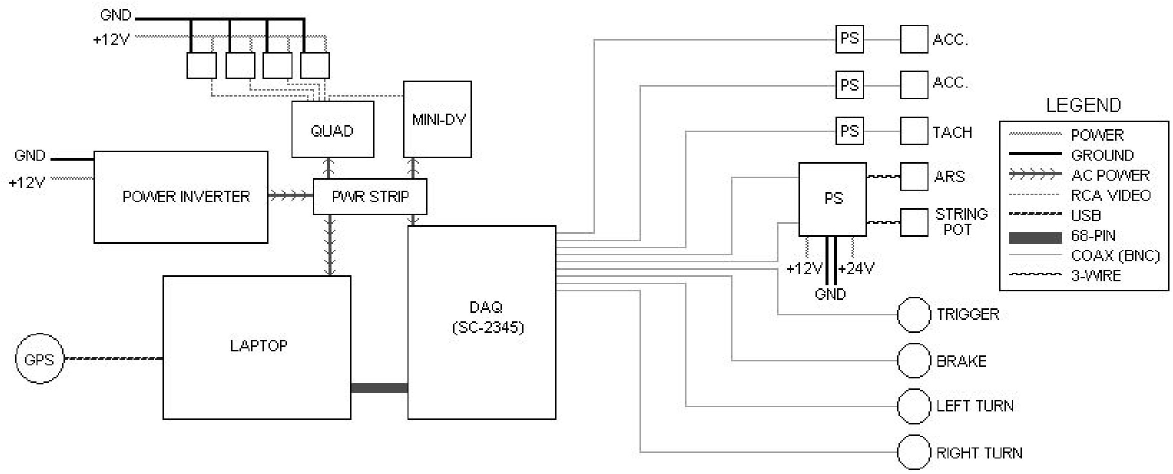


Figure 4-16. Wiring Diagram of All System Components.

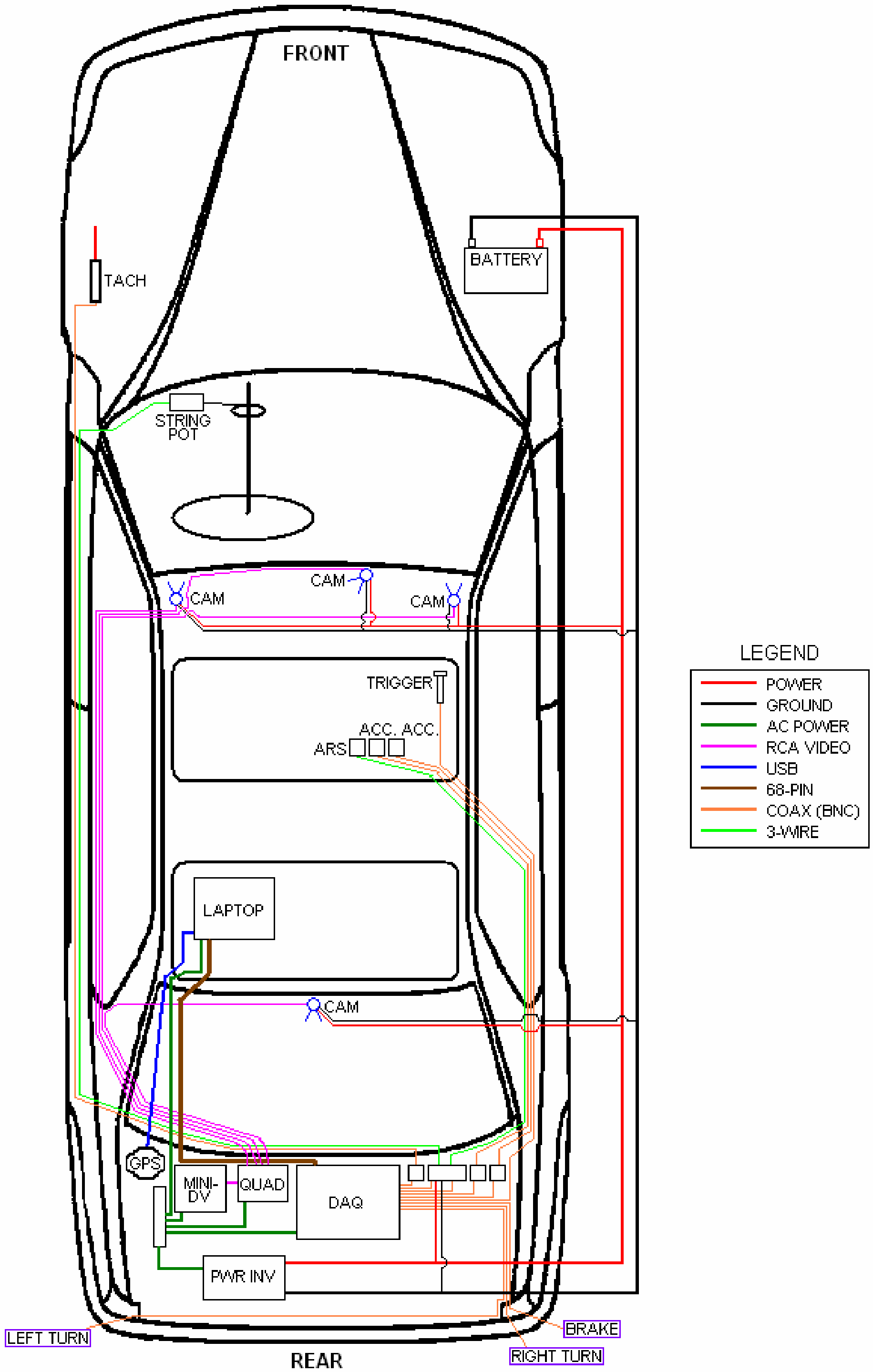


Figure 4-17. Wiring Diagram Indicating the Location of All System Components in the Vehicle.

4.4 Calibration

The entire system except for the GPS receiver, event trigger, and brake and turn signals was setup in a laboratory environment. The LabVIEW program used to operate the system was designed and refined in the laboratory. The sensitivity for the accelerometers provided by the technical specification sheets was verified using the earth's gravity. Since they are static accelerometers, they will measure exactly 1 g when placed on a level surface. There was no direct way of calibrating the angular rate sensor. However, it was verified that the sensor measured 0 radians per second when stationary and provided positive and negative voltage for clockwise and counterclockwise rotations respectively. The laser tachometer and string potentiometer needed to be calibrated after being installed in the vehicle. The laser tachometer was calibrated by driving at a constant speed and comparing the result with the speed acquired by GPS. It was then possible to calculate the factor which relates rotational frequency of the drive shaft in hertz to vehicle speed in miles per hour. The string potentiometer was calibrated by measuring the voltage with the wheel centered, turned one rotation to the left, and turned one rotation to the right. There was no calibration needed for the event trigger, brake and turn signals, or the GPS receiver.

4.5 Operation

A set of instructions was created such that the user could operate the system after receiving minimal training. The first step is to turn on all of the equipment including the power inverter, laptop, power supplies, and video equipment. Next, the user opens the LabVIEW program and GPSLog. The data acquisition system is activated by pressing a start button on the control panel of the LabVIEW program. GPSLog connects to the GPS receiver and then starts logging the data. The video recording starts recording by pressing record on the min-DV recorder. Once the system is running, the driving evaluation can begin. At the end of the evaluation the programs are terminated and the video recording is stopped. Finally, all of the equipment is turned off. The data from a one hour and twenty minute evaluation is approximately 5 megabytes and is easily transferred to another computer using a USB flash drive.

CHAPTER 5
SYSTEM VERIFICATTON

Once the system was installed in the vehicle, it was necessary to make sure that all the components were working correctly and that the data provided was accurate. Before implementing the system in actual research, a series of tests were performed to quantify the capabilities of the system and build a model for analyzing the data. It was decided that the data should be presented in a visual format so that the trends and interactions between measurements could be easily observed. The next section discusses the interpretation of the data and is followed by an analysis of the tests used to verify the system's performance.

5.1 Interpretation of Data

As previously mentioned, the data needs to be represented in a visual format in order to analyze the performance of the system. The independent variable for all the measurements is time and it is plotted along the horizontal axis. The measurements are made using a mixture of units so a conversion is necessary that will allow the full range of each measurement to be graphed on a single plot. The values for acceleration (measured in g's) and yaw (measures in radians/second) were observed to be less than a magnitude of one for most driving situations. Since there are three measurements whose values are in the same range in their native units, the rest of the measurements are normalized to fit in this range. The steering wheel position can be represented in radians, degrees, or revolutions. The decision was made to measure steering wheel position in revolutions because it is on the same order as the accelerations and yaw. Furthermore, measuring it in revolutions allows an investigator to have a direct physical understanding of the wheel position. Whether speed was measured in kilometers per hour or miles per hour, it was too large to fit in the zero to one range. Since the FHWA research is focused on the time before, during, and after an intersection, vehicle speeds will most likely be less than 40 miles per hour. Therefore the units for speed are represented in miles per hour divided by 20 (MPH/20). The brake and turn signals are simply on or off so they can easily be represented in the zero to one range. In order to simplify the plots, the brake and turn signal are only plotted when they are active.

In addition to being represented in different units, the different measurements are also measuring quantities in different directions (e.g., longitudinal and lateral acceleration). Therefore a sign convention must be developed so that the different measurements can be plotted on a single graph and interpreted correctly. The sign convention is based on the assumption that the vehicle is always moving in the forward direction. Longitudinal acceleration will be positive for acceleration and negative for deceleration. Speed will always be measured in a positive sense since it always had a positive value. Lateral acceleration, yaw, and steering wheel position will be positive when the vehicle is moving forward and making a right turn and negative when it is moving forward and making a left turn. The brake status is positive when active and is assigned a value of one simply to keep it out of the way of the other measurements on the plot (i.e., non-zero). The turn signals will have a magnitude of 0.4 when active and will be positive for right turns and negative for left turns. Basically, the measurements related to forward movement (i.e., longitudinal acceleration and speed) will be the same sign regardless of the direction of a turn. The turn-related measurements (i.e., lateral acceleration, yaw, and steering wheel position) will be positive for right turns and negative for left turns. The sign convention used when the vehicle is in reverse will not be discussed here because there are not any reversing maneuvers in the FHWA research study.

The GPS and vehicle dynamics data are recorded using two different programs that start recording at different times. Therefore, the two sets of data must be synchronized. The *xcorr* (cross-correlation) function in MATLAB is used to determine how much the GPS data needs to be shifted in order to synchronize it with the vehicle dynamics data. The laser tachometer has a lower threshold of about 4 miles per hour and cannot measure any speeds lower than this value. Since the speed provided by the laser tachometer and GPS are both sampled once per second, the GPS speed is used for all plots and calculations since it provides the necessary data in the 0 to 4 miles per hour range. An example of a maneuver plot for a left turn is provided in Figure 5-1. The abbreviations used for the measurements and their units as depicted in the figure are listed in Table 5-1.

Table 5-1. Measurement Abbreviations and Respective Units for All Plots.

Measurement	Abbreviation	Units
Lateral Acceleration	LA	*g*
Longitudinal (Forward) Acceleration	FA	*g*
Steering Wheel Position	ST	*revolutions (off of center)*
Yaw	YW	*radians/second*
Speed (Laser Tachometer)	SPT	*MPH/20*
Speed (GPS)	SPG	*MPH/20*
Turn Signal Status	TS	*ON/OFF*
Brake Status	BR	*ON/OFF*

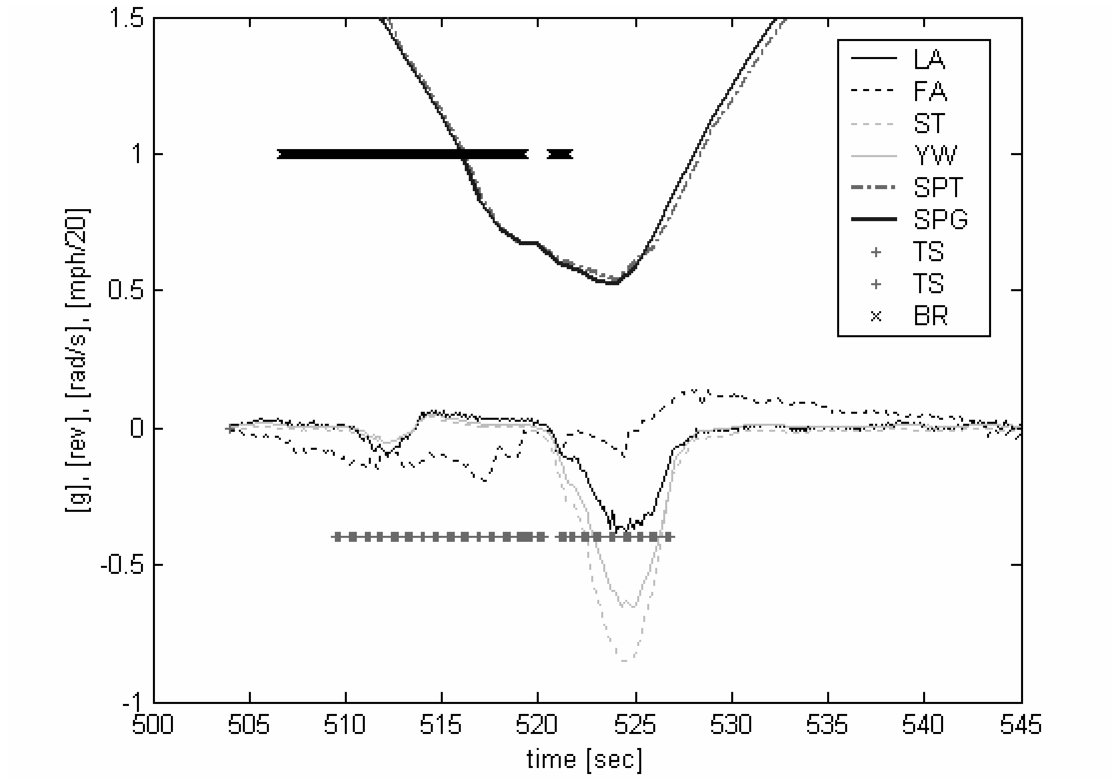


Figure 5-1. Example Plot of a Typical Left Turn.

Figure 5-1 shows all of the measurements represented on a single plot. In order to better understand and interpret the values of the results, several simple driving maneuvers were performed. All of the maneuvers were performed in an empty parking lot with lane markings. The four maneuvers were (1) accelerating from a stop, (2) decelerating to a stop, (3) 90-degree right turn, and a (4) 90-degree left turn. Each maneuver was performed in two distinct manners. The maneuver was performed once in a normal, safe manner. The maneuver was then performed a second time in an aggressive manner, almost pushing the vehicle to its limits.

5.2 Analysis of Acceleration Test Maneuver

For the acceleration test maneuvers, the vehicle was at rest and accelerated until the vehicle reached 25 miles per hour. For the normal driving condition case, the accelerator pedal was gently depressed and the car accelerated gradually. For the aggressive driving condition, the accelerator pedal was abruptly applied and the car was accelerated quickly but not enough for the tires to lose traction. The plots for the normal and aggressive acceleration test maneuvers are provided in Figures 5-2 and 5-3 respectively. For the rest of the analyses in this chapter, all of the plots will use the speed measured with GPS and will not display turn signal data.

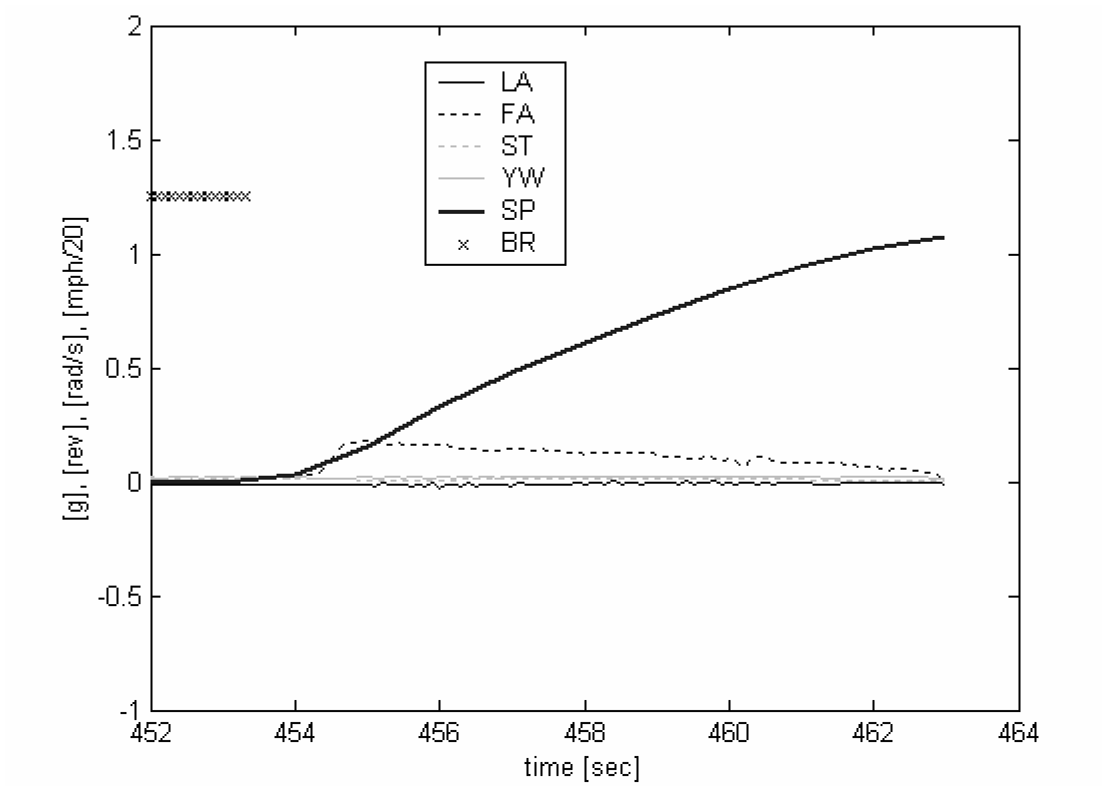


Figure 5-2. Plot of a Typical Normal Acceleration Test Maneuver.

From the plot of the normal acceleration, it can be seen the brake is initially applied and then released when the car begins to accelerate. The lateral acceleration, yaw, and steering wheel position are negligible as expected during a straight-line maneuver. The longitudinal acceleration reaches a maximum value of 0.18 g and then reduces gradually. When comparing the two scenarios, the aggressive case achieves the top speed much quicker and has a higher longitudinal acceleration of 0.40 g. The time needed to reach 25

MPH was roughly 8 seconds for the normal case and 4.5 seconds for the aggressive case. The results from these tests are as expected and help to quantify the car's performance. If a driver were to accelerate with an acceleration of 0.4 *g* then it can be assumed that the driver is driving aggressively and at the car's limits.

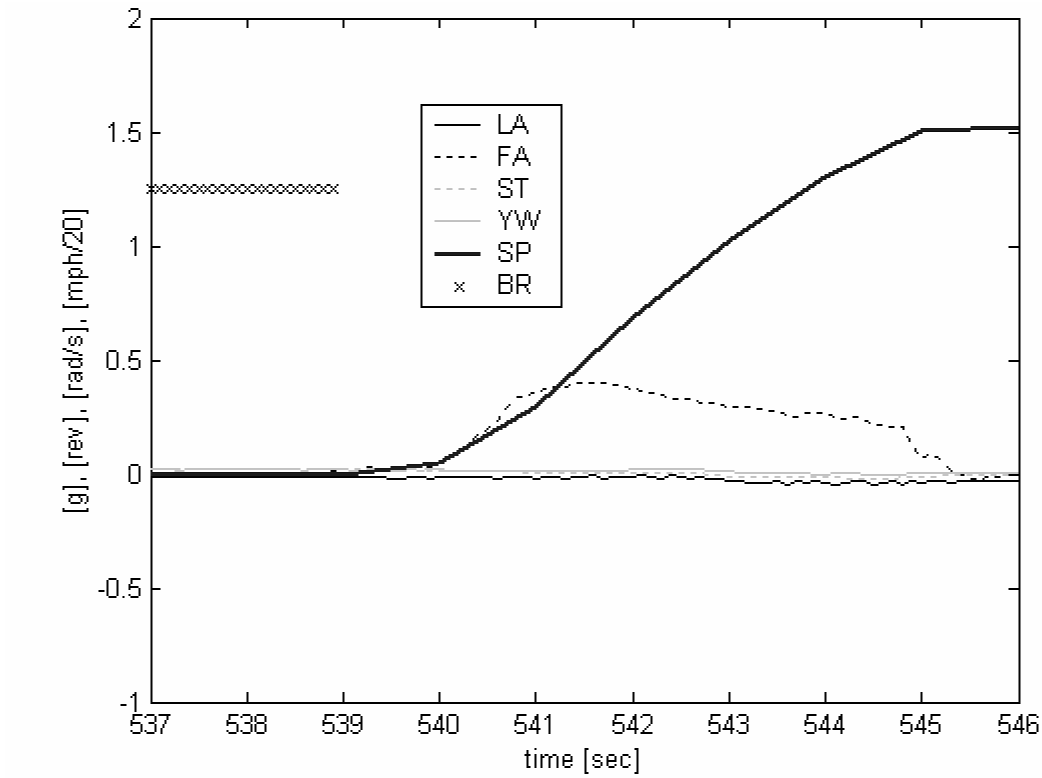


Figure 5-3. Plot of a Typical Aggressive Acceleration Test Maneuver.

5.3 Analysis of Deceleration Test Maneuver

A similar analysis was performed for the deceleration test. The car was decelerated in a straight-line from about 25 MPH. The plots for the deceleration test maneuvers are provided in Figures 5-4 and 5-5. The normal deceleration generated a maximum force of about -0.19 *g* while the aggressive deceleration reached nearly -0.43 *g*. The normal case took approximately 6 seconds to decelerate to a stop from 20 MPH. On the other hand, the aggressive case required only 3.5 seconds. Again, these results are as expected and assist in quantifying the deceleration capabilities of the car.

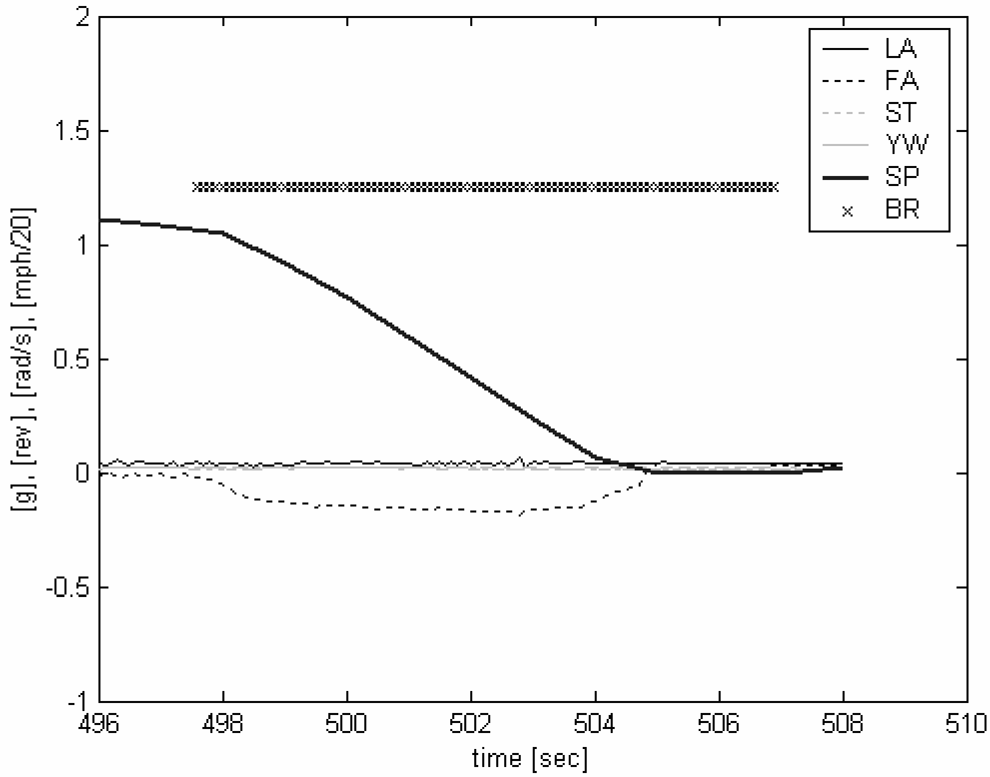


Figure 5-4. Plot of a Typical Normal Deceleration Test Maneuver.

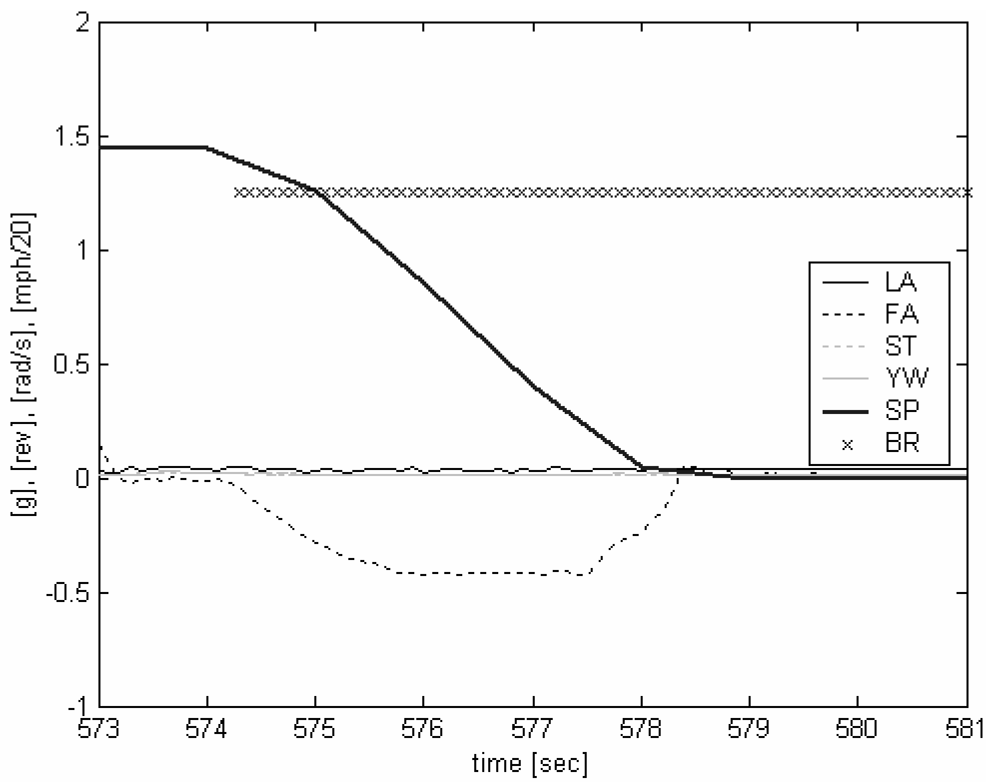


Figure 5-5. Plot of a Typical Aggressive Deceleration Test Maneuver.

5.4 Analysis of Right Turn Test Maneuver

The right and left turn test maneuvers were performed in a similar manner. The same roadway lane markings were used to ensure the geometry of the turns is identical. In all cases, the driver approached the turn, applied the brake, and then completed the turn. The only difference between the normal and aggressive scenarios was the speed at which the turn was taken. The speed during the midpoint of the turn was about 10 MPH for the normal case and 15 MPH for the aggressive case. The turns are tight and resemble those of a residential neighborhood. So, taking one of the turns at 15 MPH was actually close to the car's limits. The plots for the normal and aggressive right turn test maneuvers are provided in Figures 5-6 and 5-7 respectively.

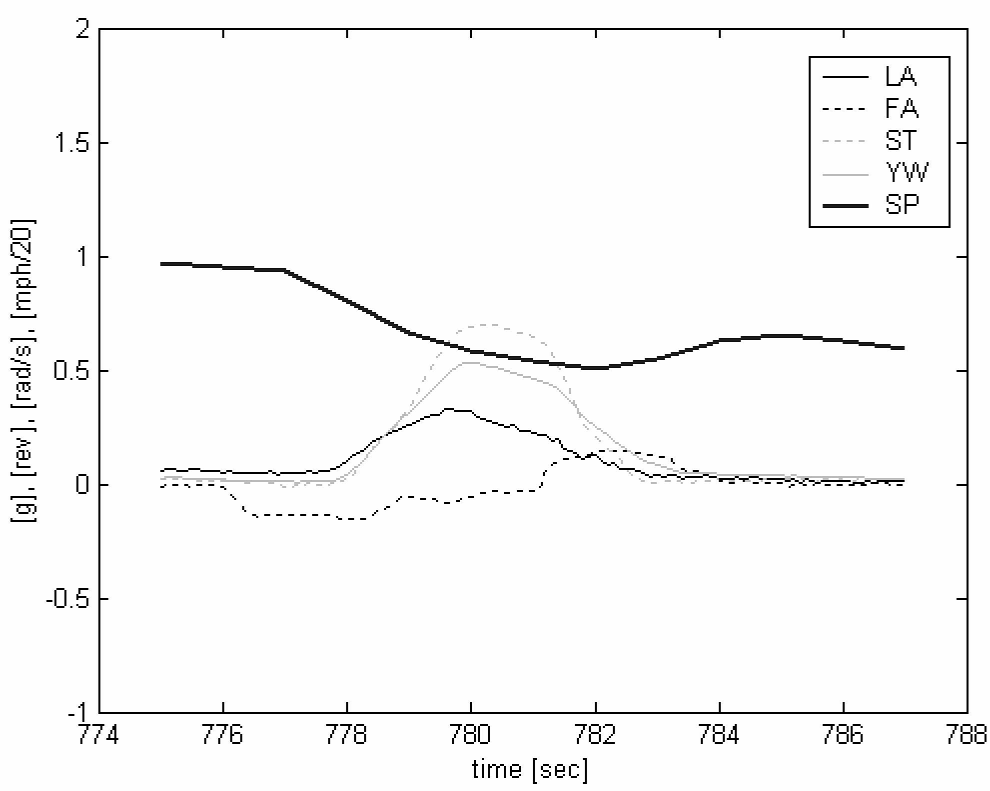


Figure 5-6. Plot of a Typical Normal Right Turn Test Maneuver.

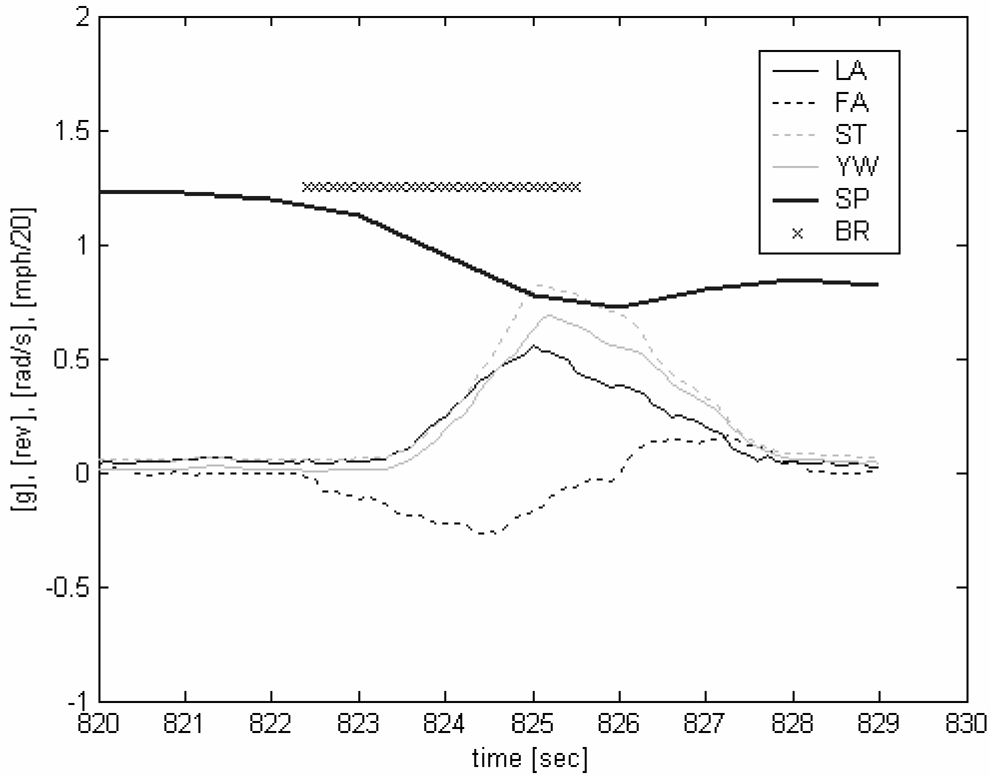


Figure 5-7. Plot of a Typical Aggressive Right Turn Test Maneuver.

In both cases, the longitudinal acceleration and speed relationship is as expected based on the acceleration and deceleration test cases. The speed decreases when the longitudinal acceleration is negative and the brake is applied. The speed begins to rise at the end of the turn when the longitudinal acceleration is positive. The most relevant measurements for the turning maneuvers are lateral acceleration, yaw, and steering wheel position. The profiles for the three measurements are similar however the magnitudes of lateral acceleration and yaw differ significantly. The steering wheel position reaches a maximum value of 0.695 revolutions for the normal case and 0.825 revolution for the aggressive case. These two values should be approximately the same because the geometry of the turn is the same for both cases and consequently the path traveled by the vehicle should also be the same. The reason for the difference in the steering wheel position values can be determined by examining the profiles. For the normal case, the profile looks nearly symmetric and is within 10 percent of the maximum value for 1.6 seconds. For the aggressive case, the steering wheel position reaches its maximum value more abruptly and is within 10 percent of this value for only 0.9 seconds. Therefore, it can be inferred that for the normal case the driver turned the wheel gradually as they entered the turn, then held the wheel in a fixed position during the turn, and finally gradually turned the wheel back to center at the exit of the turn. On the other hand, for the aggressive case, the driver turned the wheel abruptly to begin the turn and almost immediately began returning it to center. Since the aggressive turn was completed at a higher speed, it is expected that there will be higher values for yaw and lateral acceleration. This can be verified by examining the plots. The lateral acceleration reaches a maximum value of 0.332 g for the normal case and 0.561 g for the aggressive case. Yaw reaches a maximum value of 0.531 radians/second for the normal case and 0.685 radians/second for the aggressive case. A similar discussion will now be presented for the left turn test maneuvers.

5.5 Analysis of Left Turn Test Maneuver

The results from the left turn test maneuvers are nearly identical to those from the right turn test maneuvers. The plots for the normal and aggressive left turn test maneuvers are provided in Figures 5-8 and 5-9 respectively. The same trends that were observed for the right turns test maneuvers are also observed for the left turns. The steering wheel position reaches similar maximum values of 0.71 and 0.77 revolutions for the normal and aggressive cases respectively. The maximum lateral acceleration achieved 0.248 g for the normal case and 0.603 g for the aggressive case. Yaw reached a maximum value of 0.423 radians/second for the normal case and 0.682 radians/second for the aggressive case.

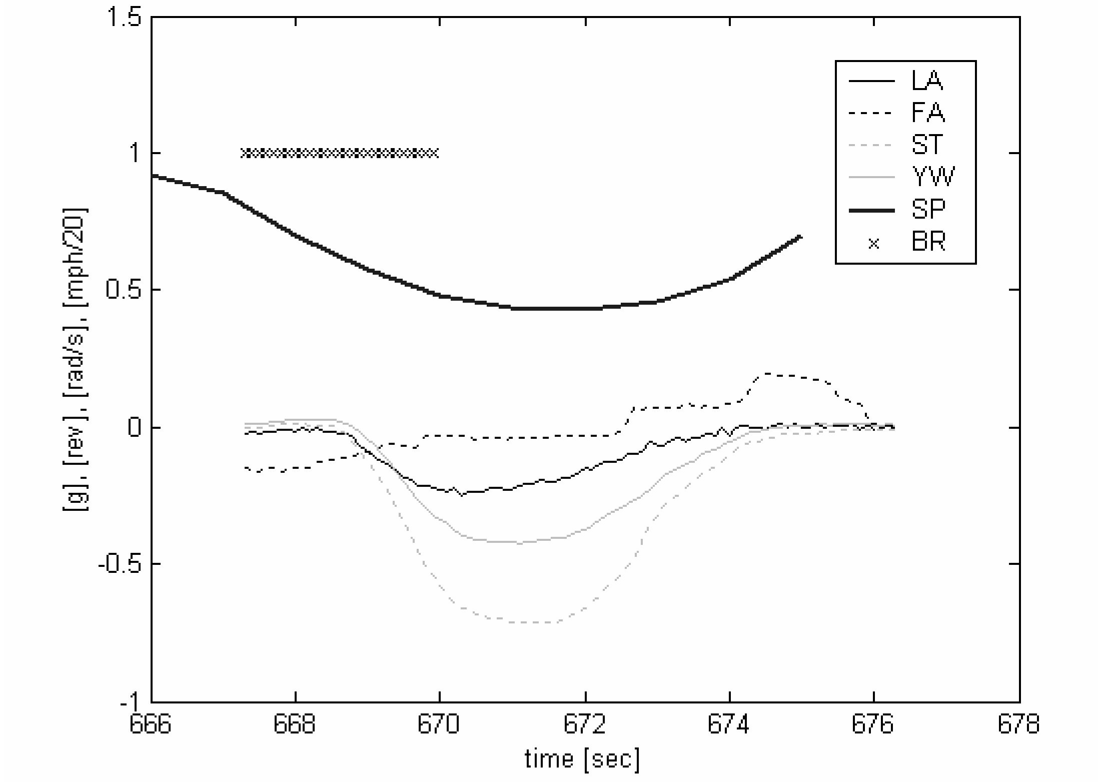


Figure 5-8. Plot of a Typical Normal Left Turn Test Maneuver.

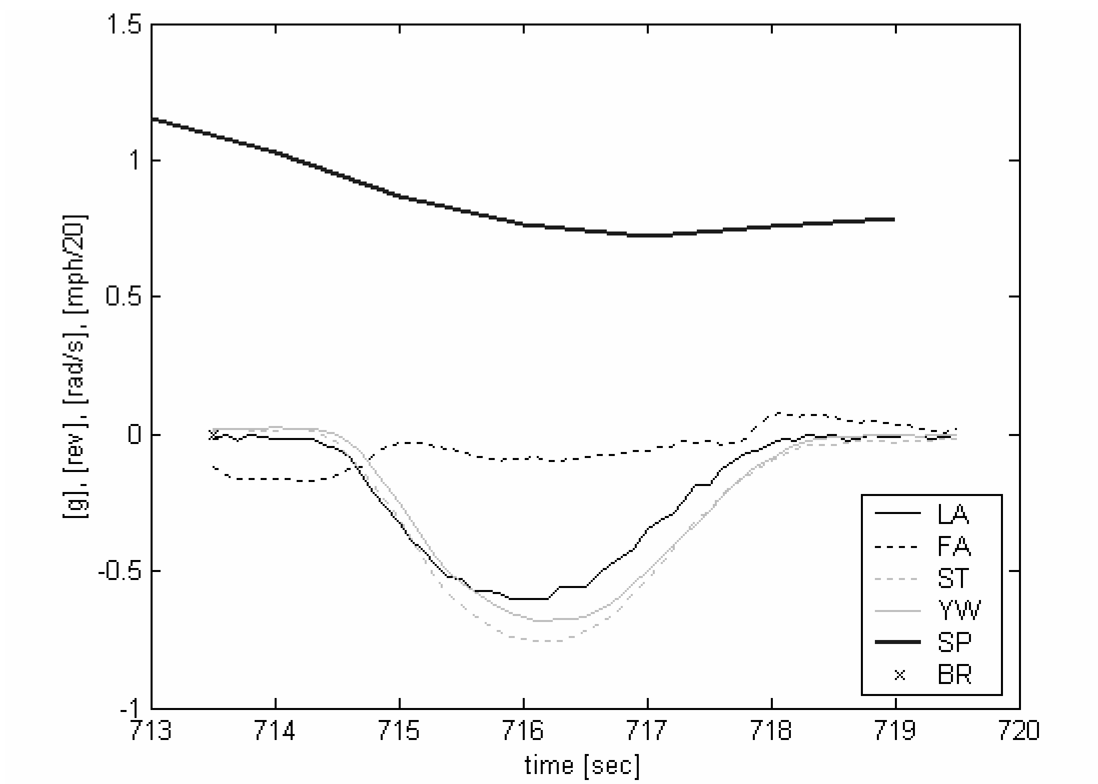


Figure 5-9. Plot of a Typical Aggressive Left Turn Test Maneuver.

These results are as expected and confirm the results from the right turn test maneuvers. It can be assumed that if a driver achieves a lateral acceleration value near 0.6 g for any driving maneuver, they are driving the car at its limits. However, a similar statement cannot be made for the yaw value. This is because yaw depends on the curvature of the turn (i.e., geometry of the intersection). Furthermore, the results for right and left turns measured on a road course will be different because in general right turns will have a smaller radius than left turns.

This analysis provides a basis for understanding the results obtained from the road course evaluations. Most importantly, the capabilities of the test vehicle have been quantified in terms of longitudinal and lateral acceleration. If a driver reaches these limits, then they are most likely driving in an unsafe manner. Although this was a general analysis, a more in-depth analysis could be performed for any particular driving maneuver depending on the hypothesis being tested. For example, if a research study was focused on driving behaviors during lane changes, then several lane changes could be performed at different rates and vehicles speeds for a better understanding of the maneuver.

CHAPTER 6
FHWA RESEARCH ANALYSIS

This chapter discusses the procedures used to analyze the data for the FHWA research study. The purpose of the FHWA research is to study the effectiveness of intersection improvements recommended by the FHWA. Participants navigate a road course and their driving performance is evaluated for ten intersections. The ten intersections are five pairs of similar intersections where one intersection features the recommended improvement and the other one does not. The naming convention for the intersection uses the format *1A, 1B, 2A, 2B,* etc. where the number refers to the intersection pair and *A* and *B* represent improved and unimproved intersections respectively.

The analysis of the road course data is conducted in three separate steps and there is a MATLAB M-file specific to each step. The MATLAB files are named *main*, *stats*, and *sort* respectively. The *main* file is used to locate the ten intersections using the GPS data. It also converts the voltage measurements from each sensor to their desired units and then creates an output file for each maneuver (intersection). The *stats* file is used to separate each maneuver into three phases (approach, turn, and recovery). It calculates statistics for each measurement of each phase of each maneuver. The statistics are compiled and output files are generated for each of the three phases. Lastly, the *sort* file takes all of the statistic files from each subject and sorts the data by maneuver rather than subject. The final result is fifteen files which contain the statistics for each phase of each intersection pair. Finally, SPSS for Windows statistics software is used to analyze the results and compare the improved and unimproved intersections. All of the MATLAB codes for this chapter may be found in Appendix D.

An almost identical analysis is performed on the simulator data. There are ten simulator scenarios that are replicas of the ten road course maneuvers. The simulator creates output files that are similar to the output files generated by the *main* program used to analyze the road course data. Similar versions of the *stats* and *sort* programs are used to analyze and format the data so that is in a form identical to that of the road course data. The exact same procedure is used to analyze the statistics from the road course and simulator using SPSS. A more detailed discussion of each MATLAB file and the SPSS analysis will now be presented.

6.1 Description of *main.m* (Road Course)

Before the file can be run, the data must first be entered into the MATLAB workspace. The vehicle dynamics data and the GPS data are both stored in comma-delimited text files. The GPS data contains a header and text stings (timestamp from GPSLog) so MATLAB's Import Data Wizard is used to add the data to the workspace. The Import Data Wizard recognizes text and automatically sorts out the numerical data to be imported. The vehicle dynamics data is imported using the *dlmread* command. The first step in the program is to plot the route traveled by the vehicle using the GPS coordinates. A simple algorithm is implemented to determine when the vehicle is at the start or end location for a particular intersection. The desired start and end locations were chosen by the driving evaluators. Their GPS coordinates were found by driving the road course and marking each location by using the event trigger. A plot of the entire road course from GPS coordinates is shown in Figure 6-1.

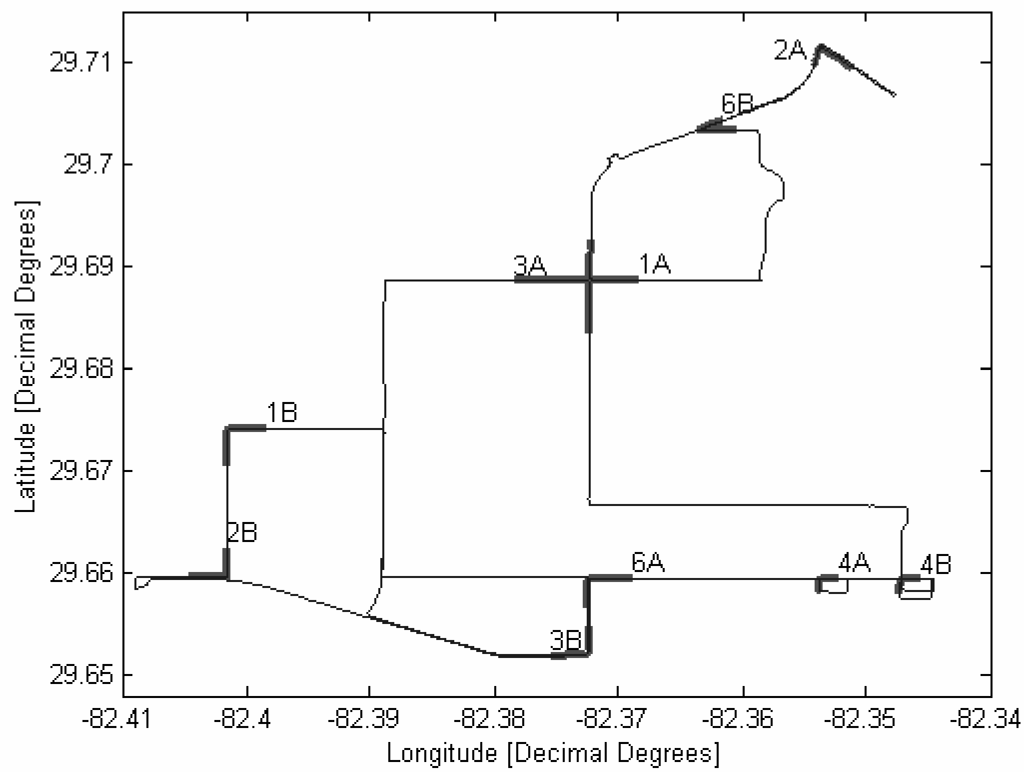


Figure 6-1. FHWA Road Course Generated Using GPS Coordinates with Intersections of Interest Highlighted and Labeled.

The ten intersections for which drivers are evaluated are highlighted and labeled at the start of each maneuver. This plot is generated to ensure that the proper intersections are selected for further analysis.

The vehicle dynamics data and GPS data are recorded using two different programs and consequently the two sets of data are not synchronized. A correlation analysis between the speed provided by GPS and the speed provided by the laser tachometer is performed. The correlation is used to find the time offset between the two sets of data and the GPS data is adjusted accordingly. A plot of each maneuver is generated to verify that the GPS data was adjusted correctly. An example of one of these plots is provided in Figure 6-2.

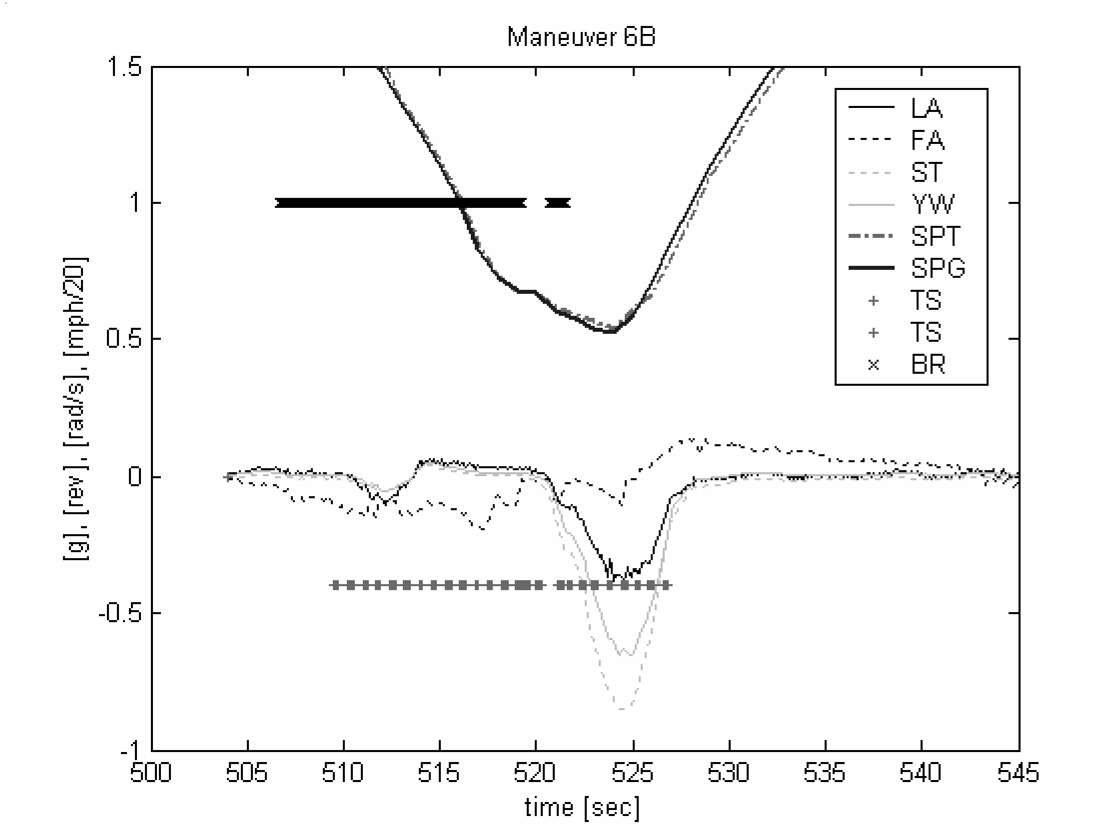


Figure 6-2. Example of Maneuver Plot Used to Verify That The Vehicle Dynamics Data and GPS Data Are Synchronized.

If for some reason the two sets of data are not synchronized exactly, the user has the option to shift the GPS data by the desired number of seconds. The plots are regenerated and the process is repeated until the data is synchronized.

The final step is to create output files for each maneuver. The output files contain all of the measurements in their appropriate units for the entire duration of each maneuver. The files are created using the *dlmwrite* command in MATLAB. The result is a comma-delimited file for each maneuver which can easily be imported into MATLAB for further analysis.

6.2 Description of *stats.m* (Road Course)

The *stats* MATLAB file is responsible for determining the three phases (approach, turn, and recovery) of each maneuver and calculating statistics for each phase. The first step is to import the data by reading the files generated by the *main* file using the *dlmread* command. The next step is to determine the three phases of each maneuver. This is done by implementing an algorithm which uses the heading from GPS and vehicle yaw. The algorithm uses the heading information to determine when the vehicle is in the middle of the turn. It then analyzes the yaw data to find out when the yaw is less than 0.05 radians per second. Since the yaw is near its maximum during the turn, the last point before the turn when the yaw is still less than 0.05 radians per second is considered to be the start of the turn. The first point after the turn when the yaw returns to less than 0.05 radians per second is considered to be the end of the turn. The point immediately before the start of the turn is considered to be the end of the approach phase and the point immediately after the end of the turn is considered to be the start of the recovery phase. The start of the approach phase and the end of the recovery phase are determined by the start and end locations of the maneuver respectively. Although the distances for the approach and recovery phases vary from maneuver to maneuver, they will always be the same for the same maneuver because the GPS locations for the start and end of the maneuver are constant. For comparison with the simulator scenarios, the distances of the approach and recovery phases need to be equal to those of the simulator. This is accomplished by converting the latitude and longitude to feet and then calculating the distance traveled during each phase. The approach and recovery phases can then be truncated accordingly to equal the distances traveled in the respective simulator scenarios.

The *stats* program generates a plot for each maneuver so the user can verify that the turn phase was selected correctly. A large and bold cross is used to mark the start and end of the turning phase. An example of one of these plots is shown in Figure 6-3.

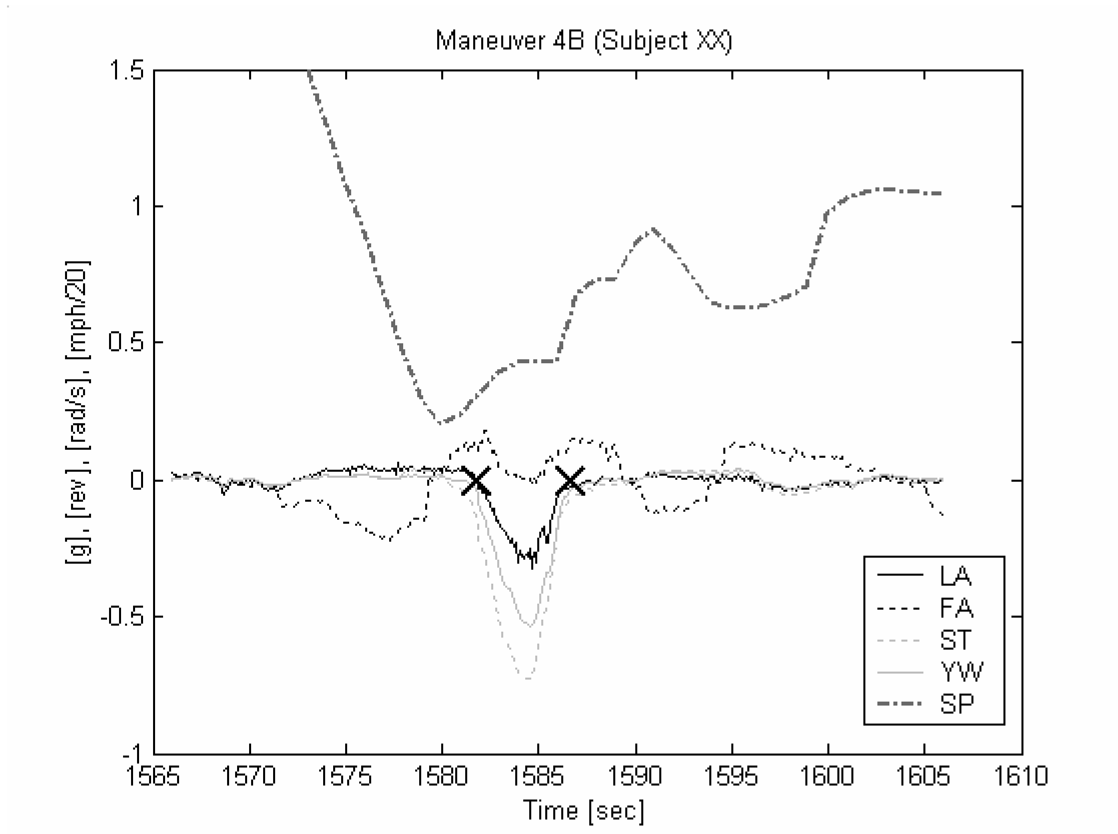


Figure 6-3. Plot of Maneuver *4B* Generated Using *stats.m* With Markers Indicating the Start and End of The Turning Phase.

Once the selection of phases is verified, the program is used to calculate statistics for each phase of each maneuver. The twelve statistics calculated for each measurement are minimum, maximum, mean, root-mean-square, integral, variance, median, range, skewness, kurtosis, sum/time, and sum/distance. The last two statistics listed are the sum of the respective measurement normalized by the time and distance traveled respectively during the phase. These twelve statistics are calculated for lateral acceleration, longitudinal acceleration, steering, yaw, speed, and combined acceleration. Combined acceleration is the absolute magnitude of the overall acceleration of the vehicle and is defined in Equation 6.1

$$com\_accel = \sqrt{lat\_accel^2 + long\_accel^2} \qquad \text{Eq. 6.1}$$

The final result is 72 statistics for each phase. Since there are three phases per maneuver, and a total of ten maneuvers, there are 2,160 statistics calculated for each subject. These statistics are compiled into comma-delimited text files using the *dlmwrite* command in MATLAB. There are three files, one for each phase, that contain the statistics for all ten maneuvers. In addition to these statistics, the user enters the subject's age and gender. The age is used to determine what age group the subject is in depending if the driver is young or old (65 years and older). The age group, gender, and maneuver number is also included in the output files to assist with the SPSS analysis.

6.3 Description of *sort.m* (Road Course)

The purpose of the sorting program is to compile all of the statistic files and sort them by maneuver pairs. The program reads all of the statistic files from each subject and arranges them by maneuver number. The program generates fifteen comma-delimited text files. There is one file for each phase of each maneuver pair. In each file there is a row for each subject. The row contains subject number, age group, gender, and maneuver number. It is then followed by all of the statistics for the improved maneuver (i.e., *A*) which is followed by all of the statistics for the unimproved maneuver (i.e., *B*). The statistics are arranged in this order so that they can easily be analyzed using SPSS.

6.4 Description of *sim.m* and *sim_sort.m*

The simulator software produces files nearly identical to those produced by the *main.m* program for the road course. A program titled *sim.m* is very similar to *stats.m* for the road course and calculates statistics for each phase of each maneuver. There are subtle differences between the two programs. For example, the simulator data is provided in different units compared to those used for the road course, so *sim.m* is used to convert the data to the appropriate units. Also, the algorithm used to determine the phases of each maneuver is slightly different due to the fact that the heading for the simulator always goes from zero to plus (right turn) or minus (left turn) ninety degrees. This is different than the road course heading which is the actual global heading of the vehicle and varies based on the orientation of the intersection. The program generates statistics files which are identical to those generated by the *stats.m* program for the road course.

A sorting program named *sim_sort.m* is used to sort the simulator statistics files in much the same way as *sort.m* did for the road course statistics. There are two minor changes that were made. First, the order of the maneuvers in the statistic files for the simulator is different from the order for the road course, so the indexing used to sort them is changed appropriately. The other change is to account for an extra intersection named *1C*. This intersection is similar to *1B* except that is has a different improvement feature recommended by the FHWA. This improvement could not be found in any intersections in Gainesville, FL and as a result could not be incorporated into the road course. The effect of the improvement wanted to be studied so a simulator scenario was created. The *sim_sort.m* program creates fifteen output files similar to those created for the road course. The only difference is that the files for the maneuver *1* intersections contain the data for three maneuvers instead of two.

6.5 Description of SPSS Analysis

SPSS statistical analysis software is used to analyze the results from the road course vehicle and simulator statistics. A general lineal model (GLM) repeated measures analysis was performed. This provides analysis of variance when the same measurement is used several times on each subject or case. For this analysis, the *within-subject* variable is the improvement of the intersection. The *between-subject* variable is age and it is used to divide the population into two groups (young and old). This general linear model analysis can test the effects of both the *between-subject* factors and the *within-subject* factors as well as the interaction between the two factors. The specific options chosen for the analysis were descriptive statistics, estimates of effect size, and observed power. Additionally, a plot of the estimated marginal means of a particular measure is generated. This plot has a line for each age group, a vertical axis for the mean of the measurement, and a horizontal axis for the improvement status of the intersection. The plot provides a useful too for visually analyzing the effects of each factor and the interaction between them. A sample output file for one phase of one maneuver and five statistics (maximum lateral acceleration, maximum longitudinal acceleration, maximum combine acceleration, maximum yaw, and maximum speed) is provided in Appendix E. Clearly performing and analyzing the results of this analysis for all statistics of all phases of all maneuvers would be extremely time consuming. The analysis calculates the significance of the effect of each factor on a statistic and is used to determine which statistics are different as a result of intersection improvement or age group. Once some of the significant statistics are found, they can be further analyzed to infer about the effect of the intersection improvement or age group. For example, if both age groups had a higher maximum speed for the improved intersection, it could be inferred that the improvement helped by allowing the majority of participants to achieve a higher speed during the turn. This entire procedure reduces the time needed for analysis by analyzing a particular statistic for all subjects at the same time rather than on an individual basis.

CHAPTER 7
RESULTS AND DISCUSSION

7.1 FHWA Research Results

At the time of this thesis submission, the FHWA research study had recently concluded. It is currently in the initial stages of data analysis. The turning phase was the first phase to be analyzed because it captures the dynamics during the intersection which is the main focus of interest. The approach and recovery phases will be analyzed at a later date to determine driver stability before and after the intersections. There are 72 statistics calculated for each phase, but it would be too time consuming to analyze all of the statistics for each maneuver using SPSS. The first statistics to be investigated for the turning phase of all maneuvers were (1) maximum combined acceleration, (2) maximum longitudinal acceleration, (3) maximum lateral acceleration, (4) maximum yaw, and (5) maximum speed. SPSS can be used to determine if these statistics are significantly different for older and younger drivers (age group), improved and unimproved intersections (intersection improvement), and the interaction between age group and intersection improvement. An example analysis of these five statistics for one of the FHWA road course maneuvers will now be presented.

7.2 Example Analysis of Turning Phase of FHWA Maneuver

The five statistics are compiled into tables for each maneuver. The average and standard deviation for each statistic is tabulated for four categories. The four categories are improved-young, improved-old, unimproved-young, and unimproved-old, where improved/unimproved refers to the intersection and young/old refers to the driver's age group. The statistics for Maneuver 2 from the FHWA road course are presented in Table 7-1.

Table 7-1. Summary of Statistics for Turning Phase of FHWA Road Course Maneuver 2.

		Maximum Combined Acceleration (g)		**Maximum Longitudinal Acceleration (g)**		**Maximum Lateral Acceleration (g)**		**Maximum Yaw (radians/sec)**		**Maximum Speed (mph)**	
Descriptive statistics											
Intersection	**Age**	Ave	SD	Ave	SD	Ave	SD	Ave	SD	Ave	SD
Improved	**Young**	0.30	0.05	0.19	0.06	0.25	0.06	0.40	0.06	22.17	2.66
	Old	0.30	0.06	0.19	0.08	0.22	0.06	0.37	0.07	20.75	2.91
Unimproved	Young	0.33	0.05	0.19	0.04	0.31	0.05	0.46	0.05	19.48	4.47
	Old	0.33	0.05	0.18	0.04	0.32	0.06	0.46	0.05	18.44	6.30
Inferential Statistics											
		F	p	F	p	F	p	F	p	F	p
Age Group (Older Versus Younger Drivers)		0.12	0.73	0.11	0.74	0.49	0.49	1.29	0.26	3.03	0.09
Intersection Type (Improved versus Unimproved)		19.5	0.01*	0.21	0.65	71.50	0.01*	59.33	0.01*	11.53	0.01*
Interaction (Age x Intersection)		0.51	0.47	0.17	0.68	4.45	0.04*	2.04	0.16	0.07	0.80

Maneuver 2 of the FHWA road course is a 90-degree right turn onto a multi-lane divided highway. Both the improved and unimproved intersections are signalized. The improved intersection features right-turn channelization. The right-turn channelization is a dedicated lane for all vehicles that are making a right turn at the intersection. It extends all the way into the receiving highway and allows drivers to accelerate up to speed before merging into traffic. A picture of the right-turn channelization is provided in Figure 7-1.

Figure 7-1. Picture of Right-Turn Channelization from FHWA Road Course Maneuver 2A.

An analysis of the statistics for FHWA Road Course Maneuver 2 will now be presented. First, the significance, if any, of each variable is investigated. There are four measurement statistics that are significant ($p \leq 0.05$) for the type of intersection (improved/unimproved). They are maximum combined acceleration, maximum lateral acceleration, maximum yaw, and maximum speed. The improved intersection has lesser values for all of these statistics except maximum speed. This can be explained by examining the equations for lateral acceleration and yaw which are provided in Equations 7-1 and 7-2 respectively.

$$a_l = \frac{v^2}{r} \qquad \text{Eq. 7-1}$$

$$\omega = \frac{v}{r} \qquad \text{Eq. 7-2}$$

where v is the vehicle velocity and r is the radius of curvature.

It should be noted that these equations are derived from their governing differential equations and in their current form are valid only for an arc of constant curvature. However, they can still be used to understand the interaction between speed, lateral acceleration, and yaw. Since the speed is greater for the improved intersection and the yaw and lateral acceleration is less, it can be inferred that the radius of curvature is greater than for the unimproved intersection. This can be easily verified by examining the geometry of each intersection. The results indicate that drivers are able to negotiate the improved intersection at higher speeds while experiencing less lateral forces.

Maximum lateral acceleration is the only variable that is significant for the interaction between age group and intersection type. The older drivers had a greater value of maximum lateral acceleration for the unimproved intersection and a smaller value for the improved intersection. Since the maximum speed is greater for the younger age group for both intersections, we would expect the respective values for maximum yaw to also be greater. The only explanation is that the maximum statistic measures do not indicate the time at which they occurred. If it is assumed that both age groups of drivers achieved maximum lateral acceleration at approximately the same time (midway through the turn). Then it is likely that the younger and older drivers reached their maximum speed at different times. Further analysis of the data could reveal if the older drivers began to decelerate earlier.

Maximum speed is the only variable that approaches significance ($p \leq 0.10$) for the age group. Younger drivers had greater speed for both the improved and unimproved intersections. This is an expected result since older drivers generally drive with more caution and at a lower speed.

This analysis, or one very similar can be performed on the remaining maneuvers. Examination of all 72 measurement statistics can reveal which ones are most significant. The measurement statistics of interest may vary depending on the specific improvement being examined in each maneuver. Furthermore, the statistics can be separated into different cases depending on the driving conditions (i.e., stopping before the turn or signal status). The other phases (approach and recovery) of each maneuver will also be analyzed using a similar process. The approach and recovery phases will be analyzed to determine how each intersection affects the driver's stability before and after the turn. It has been shown that the measurement statistics and their associated significance is useful in analyzing each intersection. These results combined with an understanding of the intersection can be used to infer information about the driver's performances.

CHAPTER 8
CONCLUSION

The vehical dynamics acquisition system developed for the FHWA study was able to successfully measure and record the vehicle dynamics of a Buick sedan. The addition of GPS allowed the system to acquire the vehicle dynamics at specific locations. The system is flexible and can measure a wide variety of inputs. It can easily be expanded with the addition of SCC modules and transducers to measured additional information. The system was first used for the FHWA study and proved to be a useful tool in quantifying driving behavior. It could be used in its current form for a different study concerned with a different driving behavior. The only change would be to the MATLAB code in order to analyze the particular area of interest. The video equipment, although not directly incorporated with the data acquisition and GPS equipment, was a helpful tool for the evaluators. It was used to train new evaluators and to review evaluations to ensure that evaluators were scoring driving errors consistently.

8.1 Future Research Studies

It is likely that this system will be used in the future for other driving behavior studies. If needed, additional sensors can easily be added. For example, a load cell could be used to measure the amount of force the driver is applying to a pedal. The system could even be adapted to measure physiological responses from the driver such as heart rate, or perspiration. Also, the cameras can be remounted to change the field of view. For example, the cameras could be used to monitor vehicle lane maintenance by aiming the cameras toward the lane markings next to the vehicle.

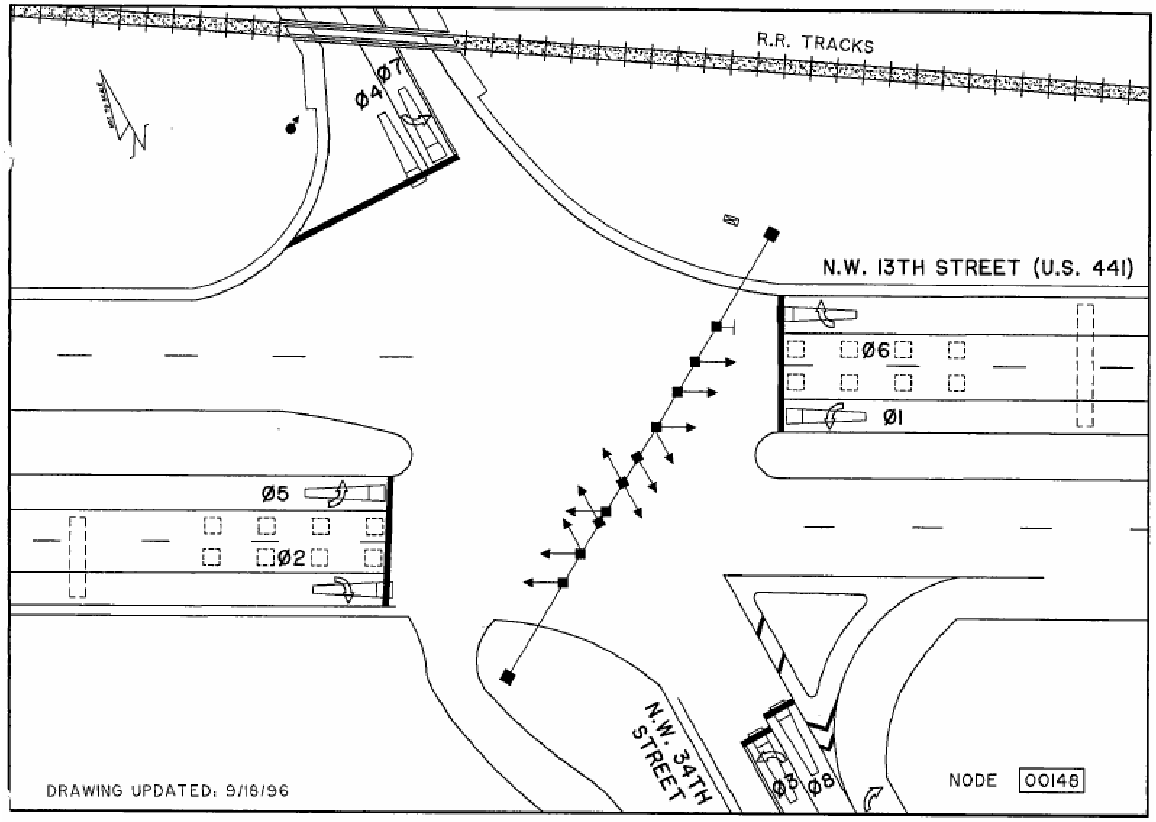
8.2 Recommendations and Improvements

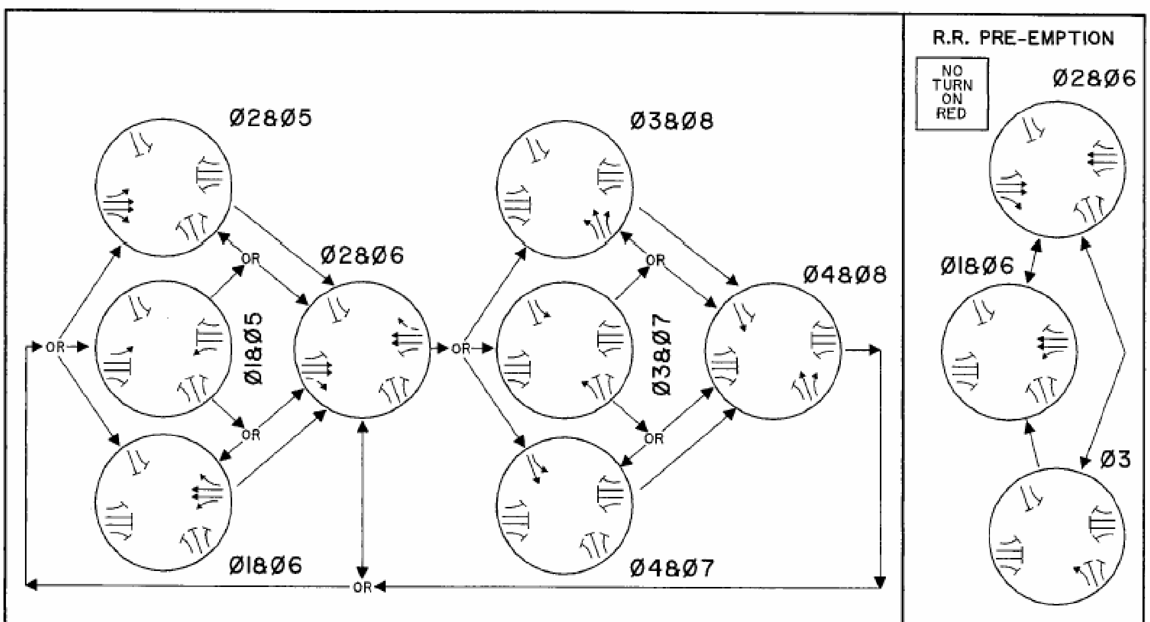
Although the system worked well and as intended, some improvements could be made that would make it easier to use and operate. The first recommendation is to find an alternate way to measure the vehicle speed. The speed obtained from the GPS was only updated once per second and was less accurate when the vehicle was stationary or moving slowly. The speed measured using the laser tachometer was inconsistent due to the laser not being reflected sometimes. Furthermore, the laser tachometer could not measure speed below 5 miles per hour. An infrared ground speed sensor like the Correvit® LF™ (Appendix B) would be able to measure vehicle speed faster and more accurately.

The second recommendation is to use only one program for operating the system. This could be done by incorporating the GPS with the rest of the system via a serial port and LabVIEW. The LabVIEW Virtual Instrument could be programmed to interface with the GPS antenna using the NMEA 0183 Standard.

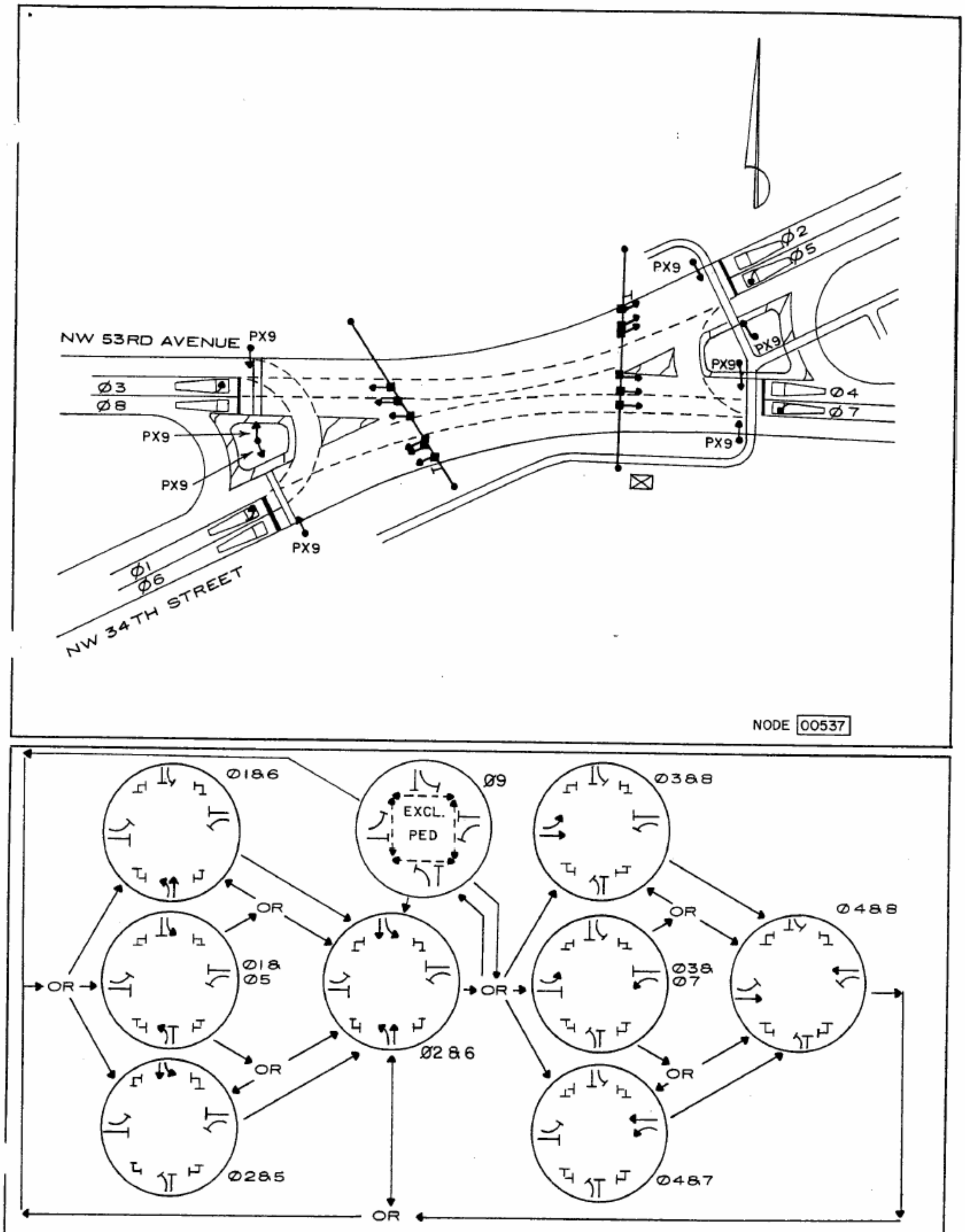
The final recommendation is related to the overall design and installation of the system. If a similar system were to be installed in another vehicle then all of the equipment should be contained in a single container or rack. The rack would contain the power inverter, all power supplies, computer, and data acquisition system. There should be one panel that contains all of the connections for the sensors. Also, a personal computer (PC) could be used instead of a laptop computer. This would allow the hardware to be contained in the rack and the monitor located in the cab of the vehicle. The computer could be controlled using a Bluetooth keyboard and/or mouse. The advantage of designing the system like this is that it would take up less room in the vehicle, be simpler to operate, and easier to install or remove.

APPENDIX A
INTERSECTION DIAGRAMS

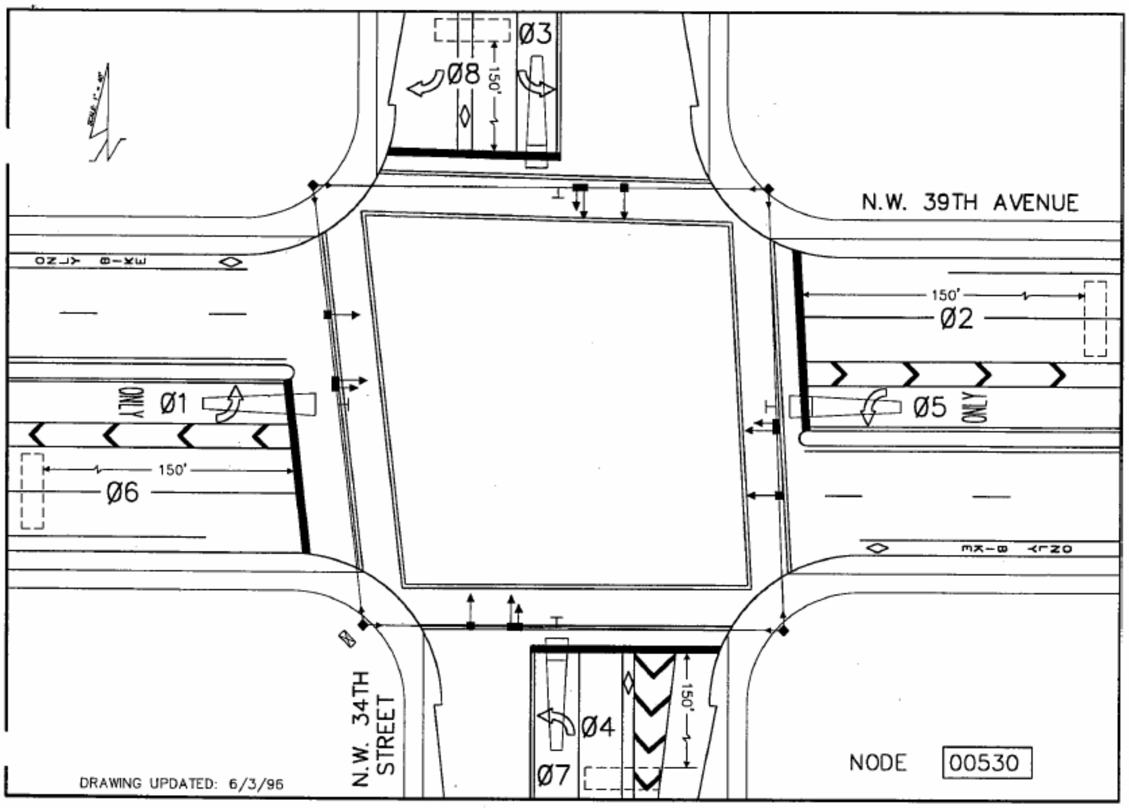


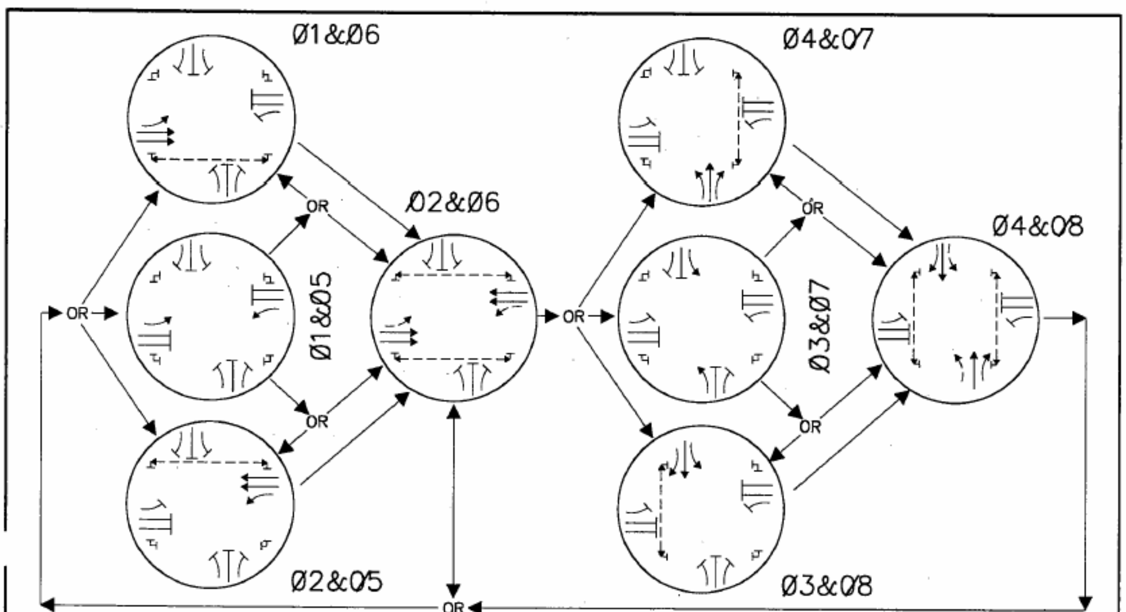


22 DEC. 80	N.W. 13TH STREET & N.W. 34TH STREET	342

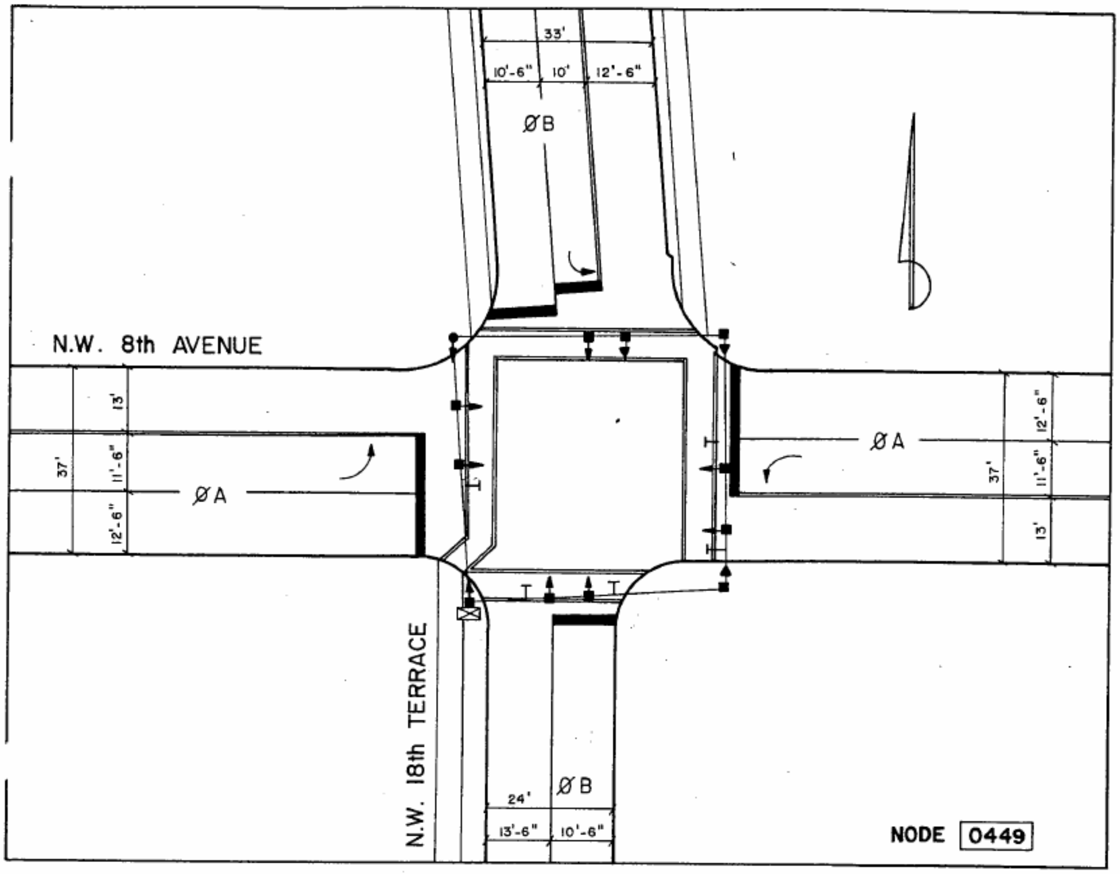


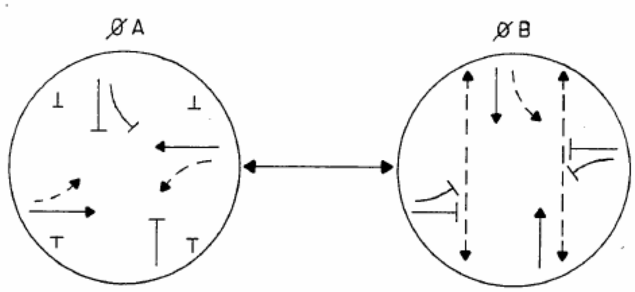
APRIL 13, 1987	NW 53RD AVENUE & 34 TH STREET	345



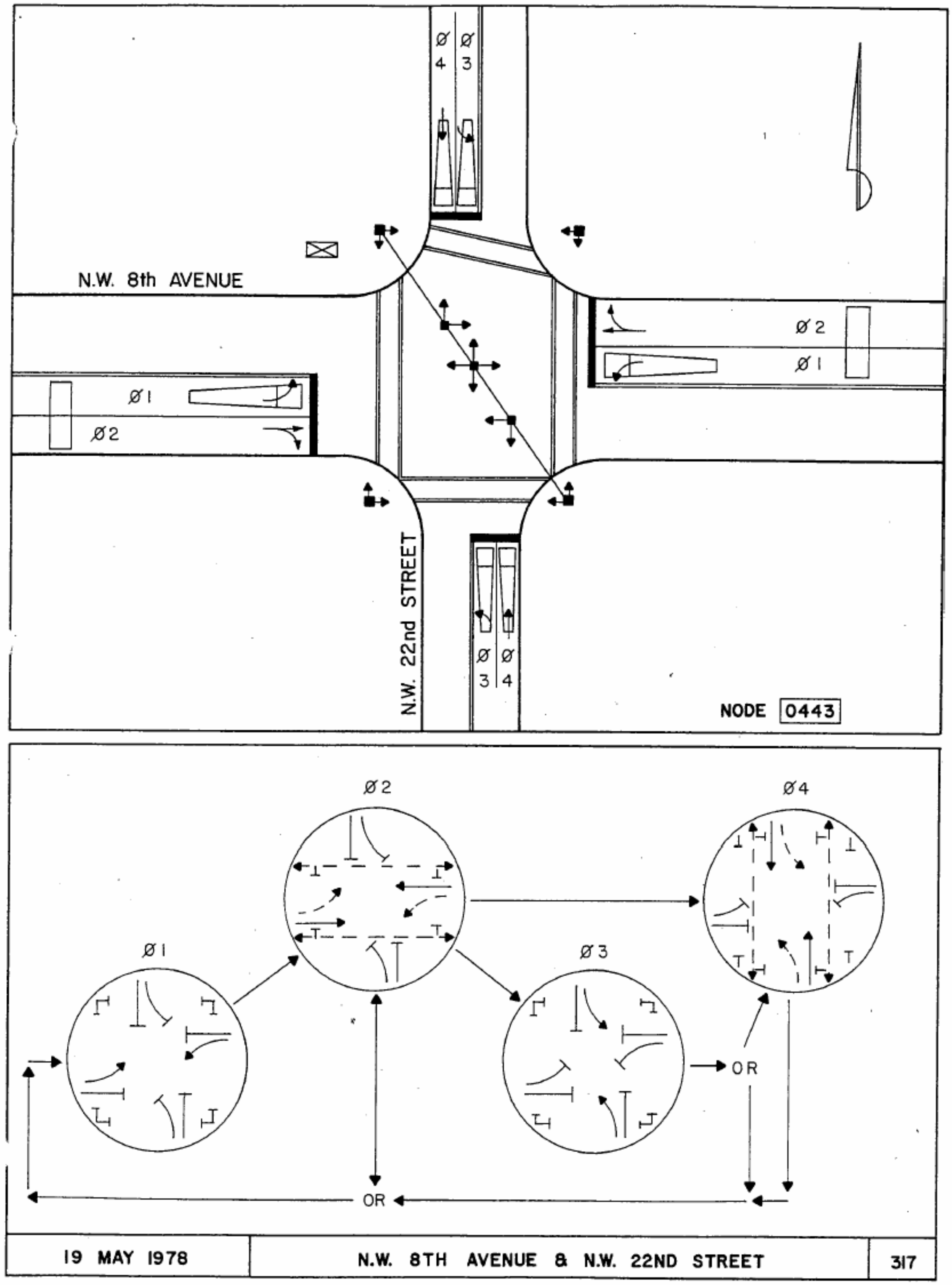


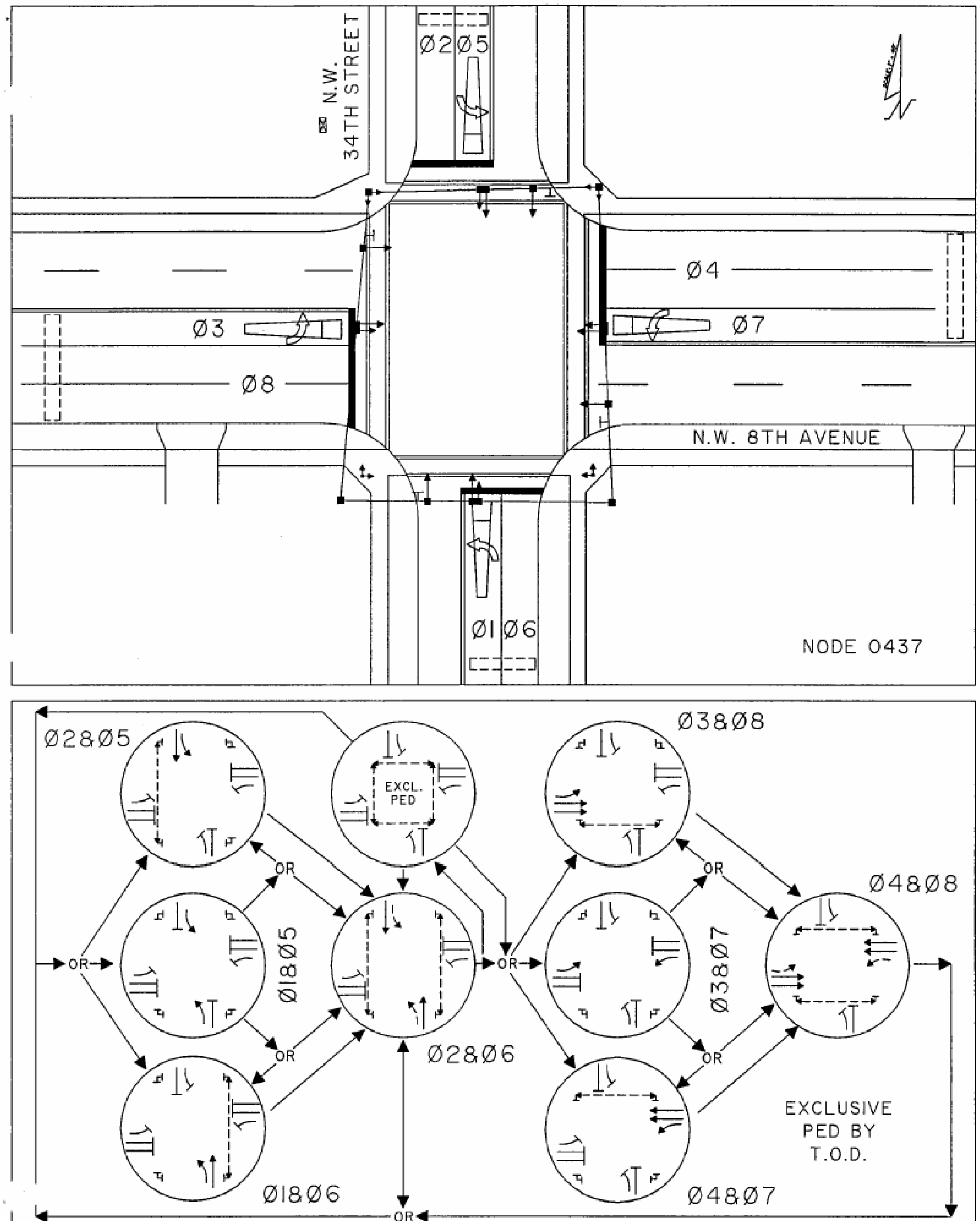
26 FEB. 90	N.W. 34TH STREET & N.W. 39TH AVENUE	340



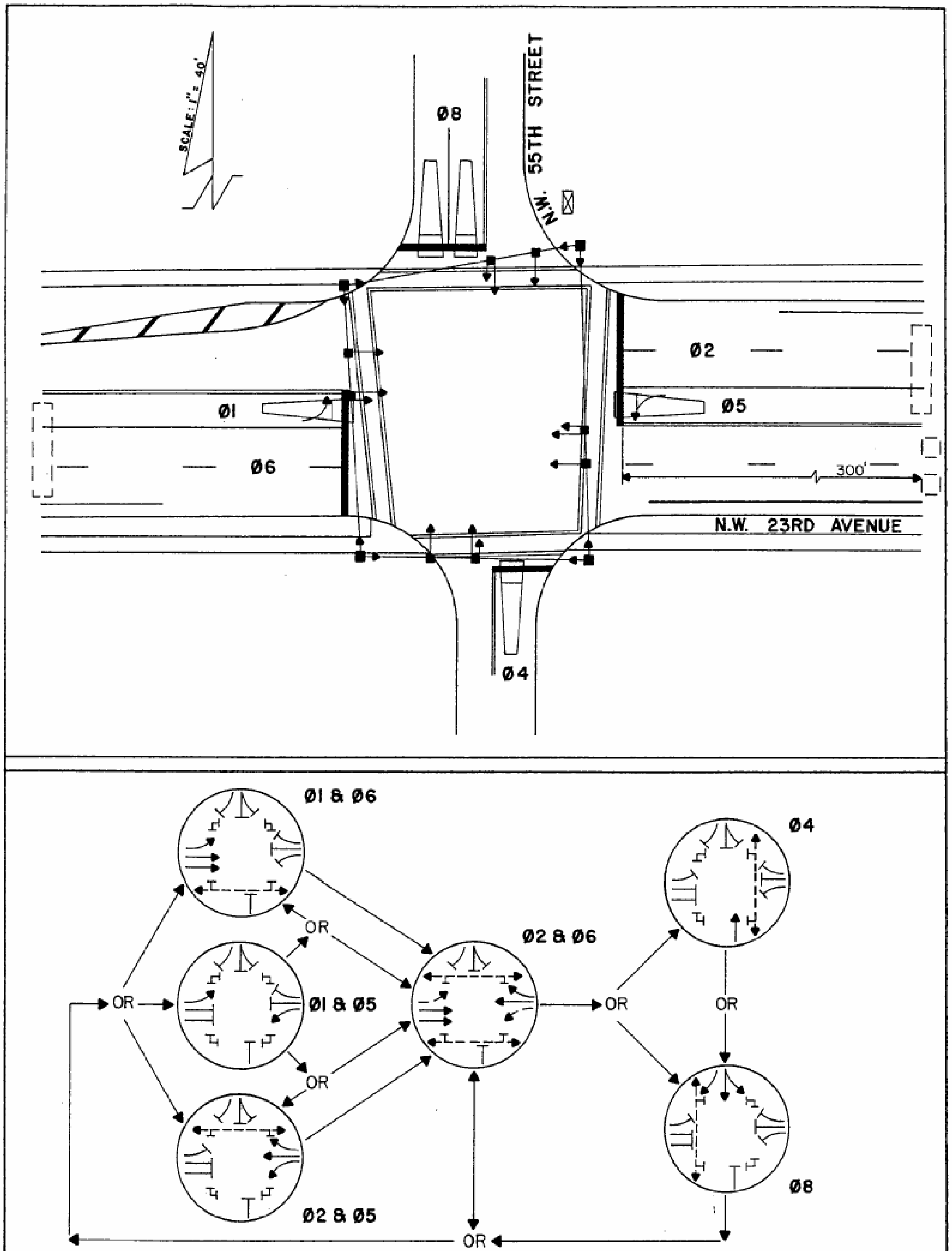


11 APRIL 1978	N.W. 8TH AVENUE & N.W. 18TH TERRACE	316

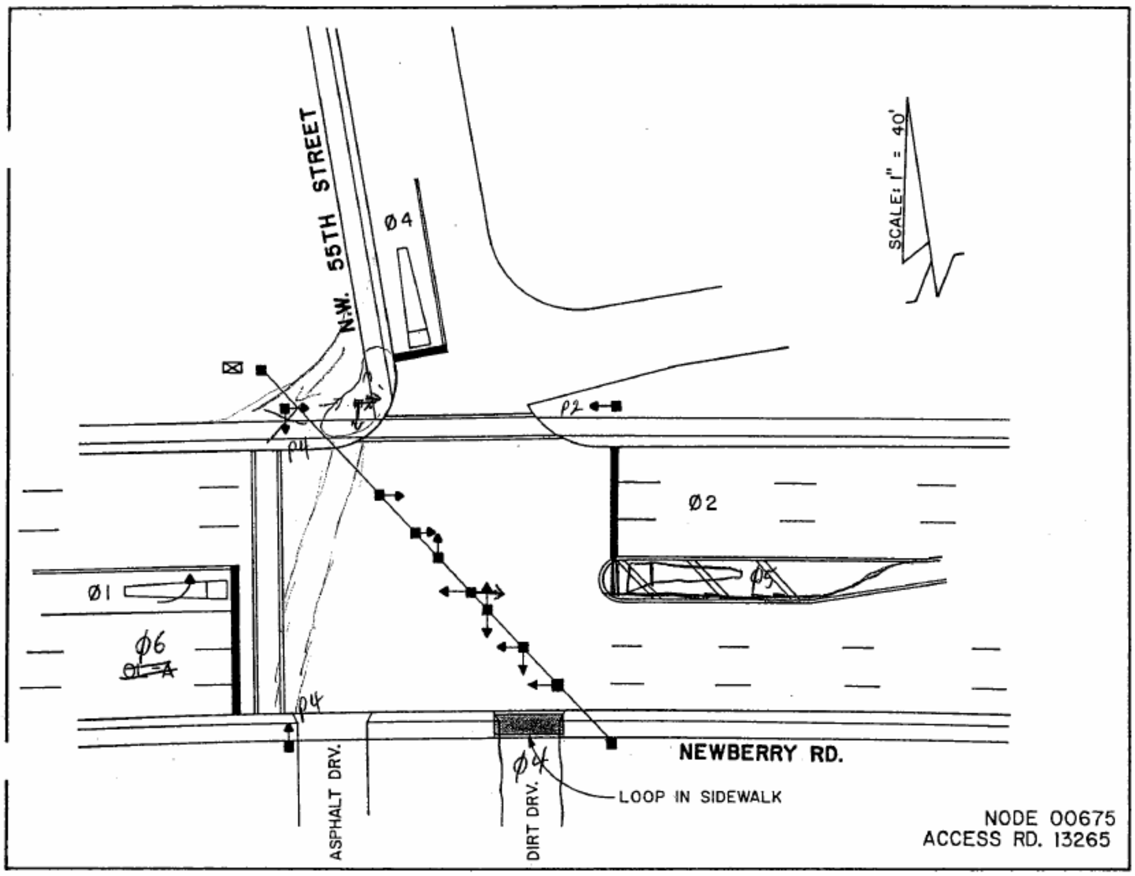

Ø 4
Ø 3
N.W. 8th AVENUE
Ø 2
Ø 1
Ø 1
Ø 2
N.W. 22nd STREET
Ø 3
Ø 4
NODE 0443
Ø 2
Ø 4
Ø 1
Ø 3
OR
OR
19 MAY 1978
N.W. 8TH AVENUE & N.W. 22ND STREET
317

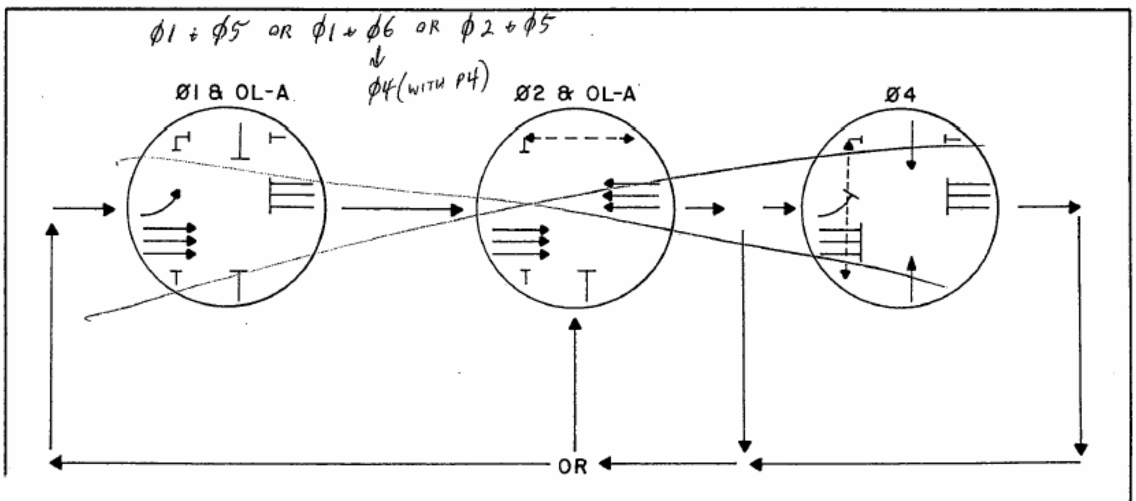


18 SEPT. 95	N.W. 8TH AVENUE & N.W. 34TH STREET	318

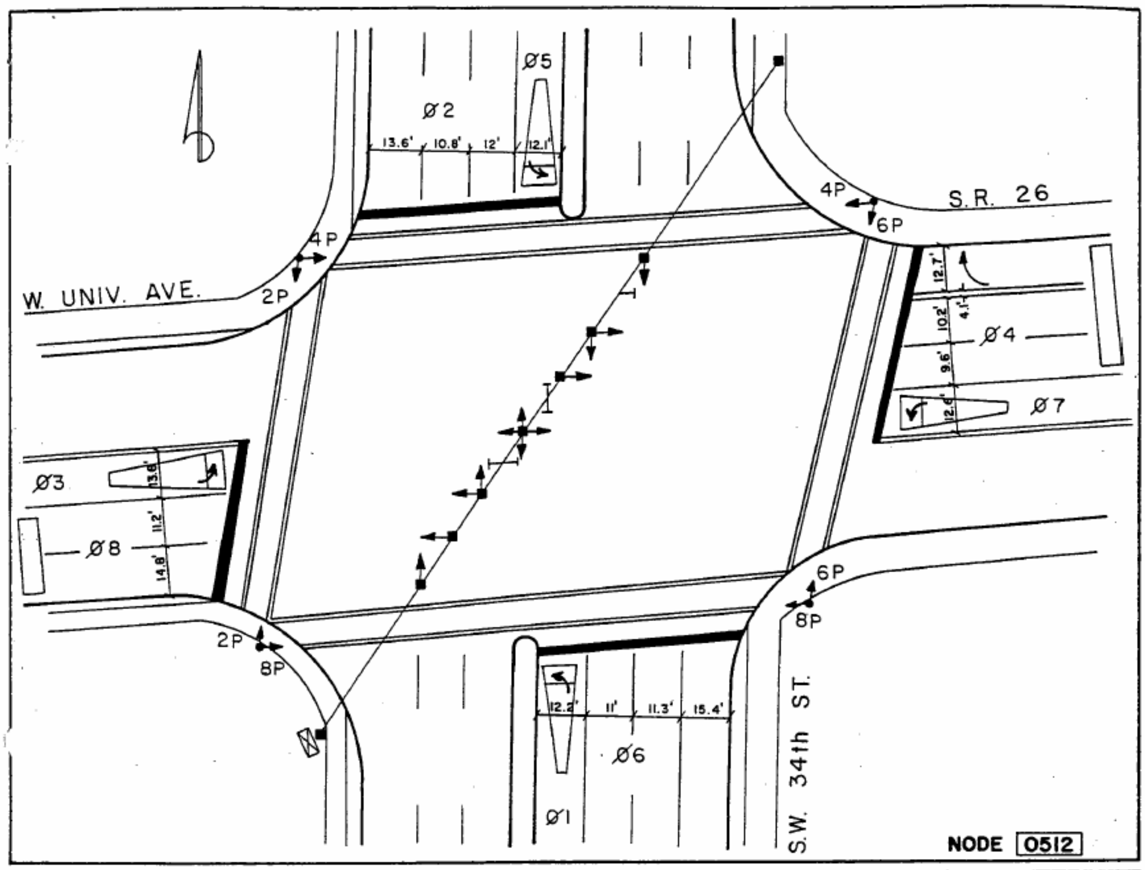


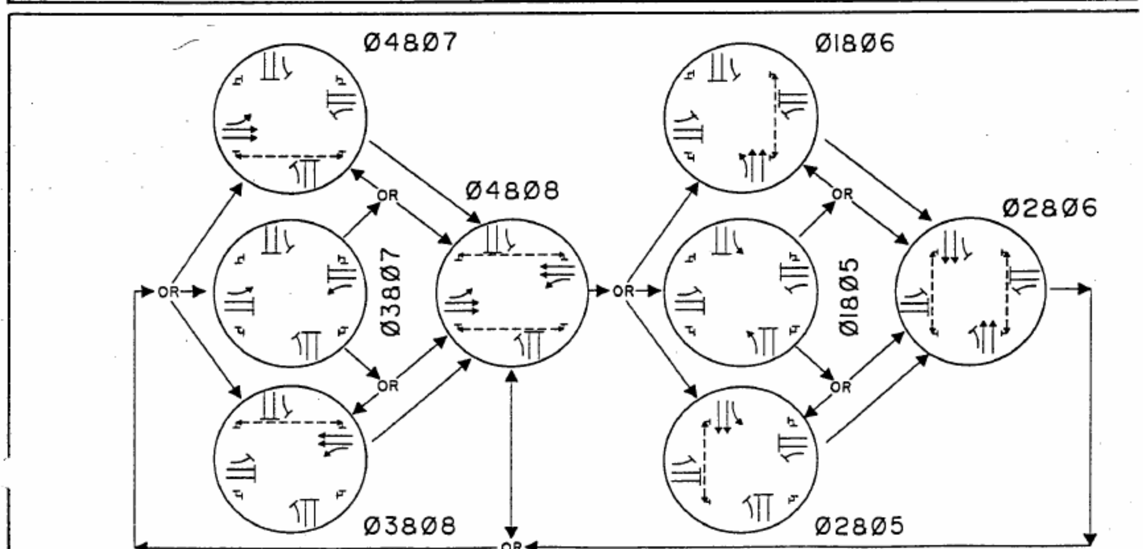
9 JULY 1991	N.W. 23RD AVENUE & N.W. 55TH STREET	331



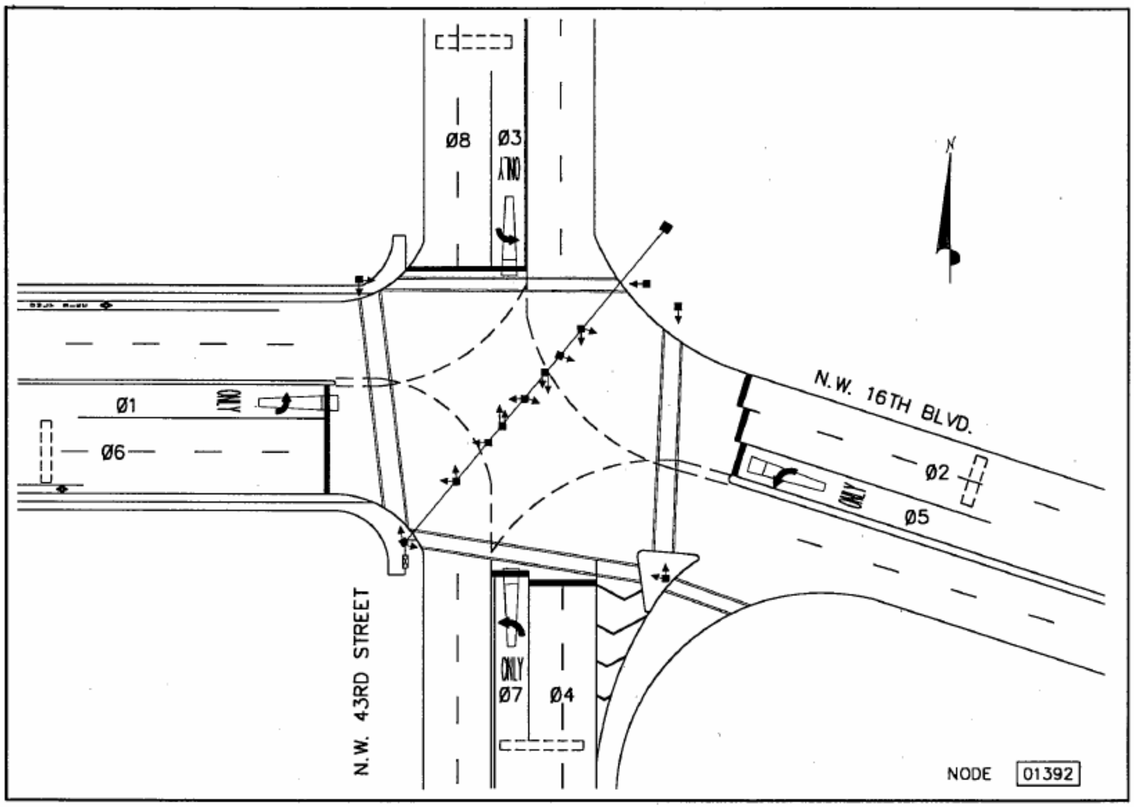


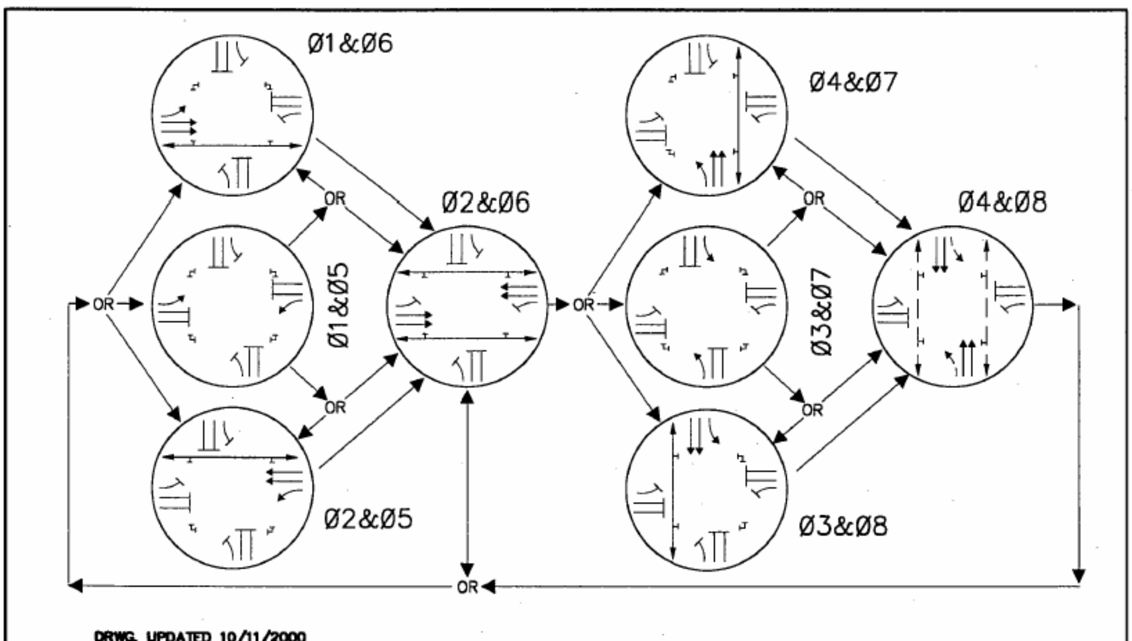
19 DECEMBER '89	NEWBERRY RD. (SR26) & N.W. 55TH STREET	133



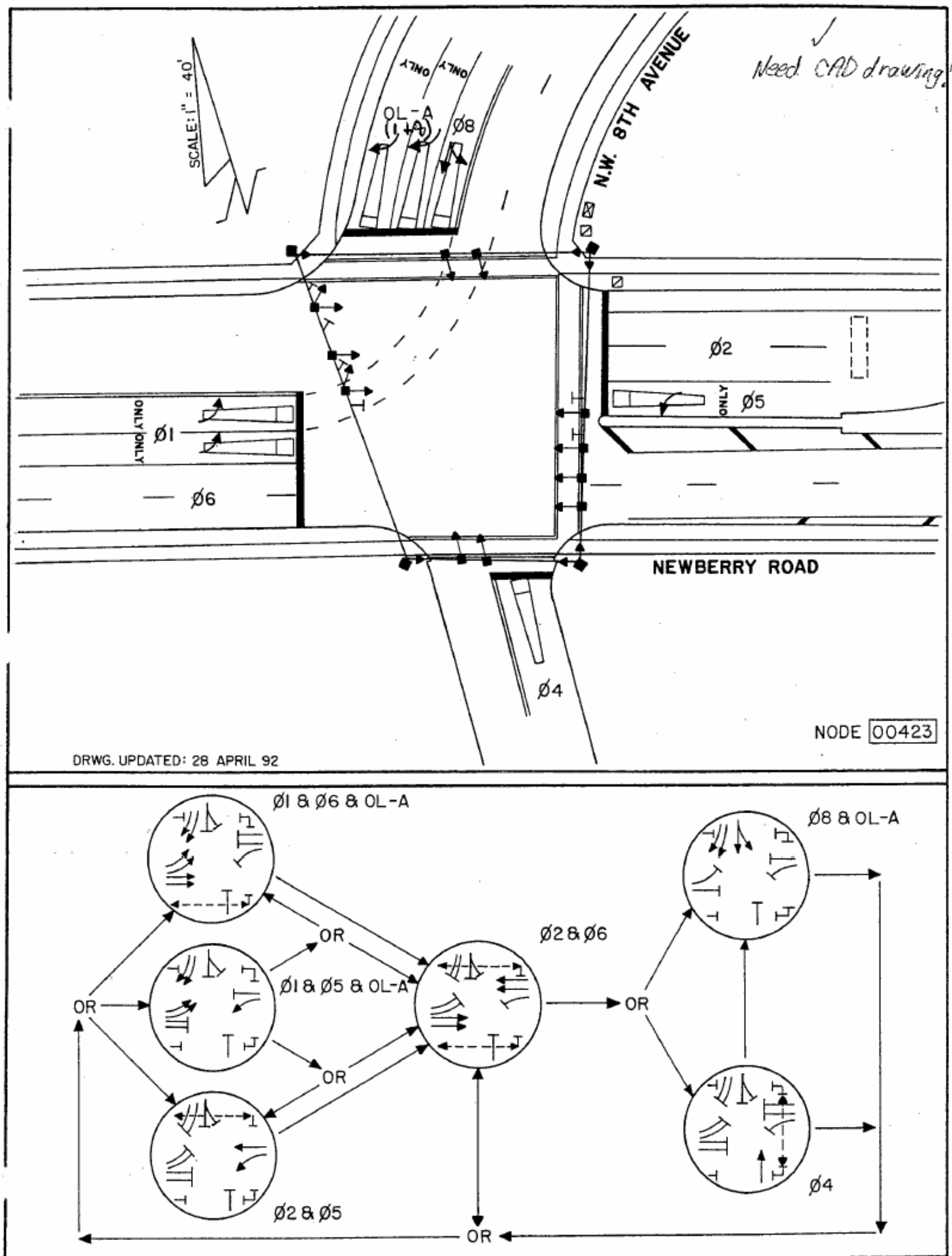


SEPTEMBER 1984	WEST UNIVERSITY AVENUE & 34TH STREET	127





9 FEBRUARY 1990	N.W. 43RD STREET & N.W. 16TH BOULEVARD	330



30 OCTOBER 1989	NEWBERRY ROAD (SR 26) & N.W. 8TH AVENUE	130

APPENDIX B
COMPONENT TECHNICAL SPECIFICATIONS

Kistler Static Accelerometers

Model Number	1. 8310A2	2. 8310A10
3. Range	*4.* ± 2 *g*	5. ± 10 *g*
6. Sensitivity	*7.* 1000 *mV/g*	*8.* 200 *mV/g*
9. Threshold	*10.* 540 μ*g*	11. 2830 μ*g*
12. Frequency Response	13. 0-300 Hz	14. 0-180 Hz
15. Temp. Operating Range	16. -40 – 85 °C	17. -40 – 85 °C
18. Supply Voltage	*19.* 3.8 – 16 *volts*	20. 3.8 – 16 *volts*
21. Supply Current (nom.)	*22.* 1.3 *amps*	23. 1.3 *amps*
24. Housing	25. Titanium, hermetically sealed	26. Titanium, hermetically sealed
27. Connector	28. 4-pin Microtech	29. 4-pin Microtech

BEI GyroChip Angular Rate Sensor

30. Model Number	31. AQRS-00075-111
32. Range	33. ± 75°/sec
34. Sensitivity	35. 33.33 °/volt
36. Input Voltage	37. + 5 volts ± 5%
38. Input Current	39. <20 mA
40. Bias Calibration	41. +2.50 volts (nom.)
42. Temp. Operating Range	43. -40 – 85 °C
44. Connector	45. 3-pin

LaserTach Laser Tachometer

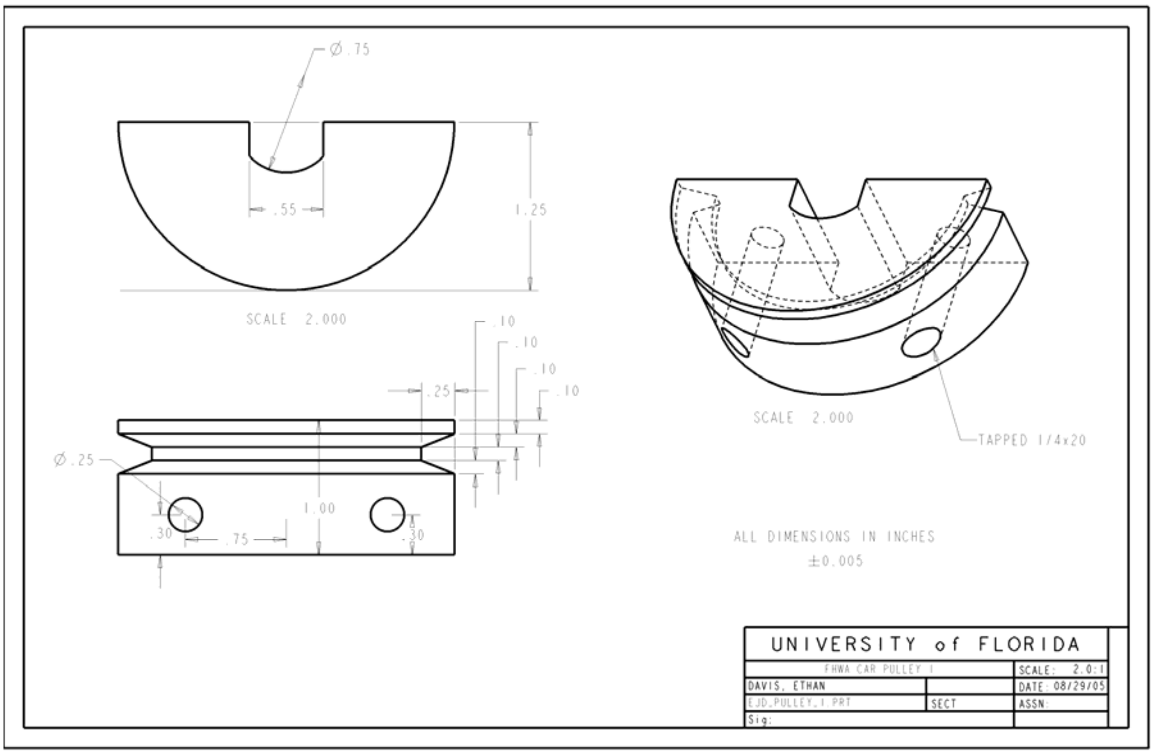
Speed Range	30,000 RPM max.
Input Voltage	18 – 30 volts
Input Current	3 – 20 mA (constant)
Output Type	Square wave
Output Amplitude	1 Vpp @ 4 mA supply current
Operating Range	6 feet (90°) 2 feet (45°)
Temp. Operating Range	-40 – 85 °C
Connector	BNC

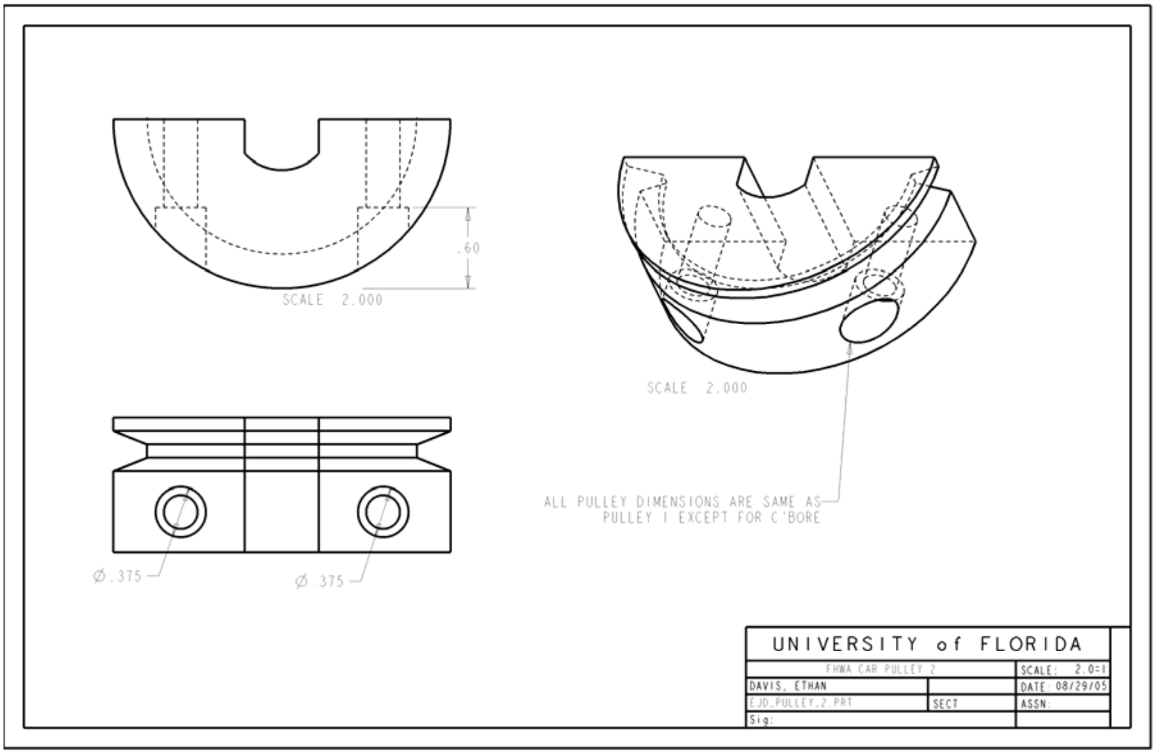
Celesco Cable-Extension Position Transducer (String-Pot)

(* indicates custom selection)

46. Model Number	47. PT1DC-50-BK-Z10-MC4-SG
48. Range*	49. O – 50 inches
50. Accuracy	51. 0.15% (full scale)
52. Resolution	53. infinite
54. Input Voltage	55. 14.5 – 40 volts
56. Input Current	57. 10 mA max.
58. Output Voltage*	59. 0 – 10 volts
60. Cable	61. 0.019-in. dia.nylon-coated stainless steel
62. Cable Exit*	63. back
64. Cable Tension	65. 5 oz.
66. Temp. Operating Range	67. -40 – 200 °F
68. Connector*	69. 4-pin micro-connector (1 not used)

APPENDIX C
CUSTOM PULLEY DRAWINGS




.60
SCALE 2.000
SCALE 2.000
ALL PULLEY DIMENSIONS ARE SAME AS
PULLEY I EXCEPT FOR C'BORE
Ø.375
Ø.375
UNIVERSITY of FLORIDA
FHWA CAR PULLEY 2
SCALE: 2.0=1
DAVIS, ETHAN
DATE: 08/29/05
EJD_PULLEY_2.PRT
SECT
ASSN:
Sig:

APPENDIX D
MATLAB CODES

main.m

```
% ********************************************************************
%
%                    FHWA MAIN (ROAD COURSE)
%
%   University of Florida
%   College of Publich Health and Health Professions
%   Occupational Therapy Department
%   Created by: Ethan J. Davis (Mechanical & Aerospace Engineering)
%   Date: 06-27-05
%
%   ------------------------ INSTRUCTIONS -----------------------------
%   Place this M-File in the same folder with "data.txt" and "GPSLog.txt"
%   Open Matlab, Open this M-File
%   In main MATLAB menu, Click on "File" --> "Import Data..."
%   Locate the Folder and select "GPSLog.txt", press "Next" then "Finish"
%   Run this M-File (F5)
%
% ********************************************************************
format short g

% sort GPS data
data = GPSLog;
lat = data(:,1);                    % latitude
long = data(:,2);                   % longitude
sp_gps = data(:,3)*1.15077945;      % speed (GPS)/convert from knots to mph
head = data(:,4);                   % heading
dt = 0.1;                           % time step (0.1 sec = 10 Hz)
A = ['2A6B1A4B4A6A1B2B3B3A'];       % text string of maneuver labels

% start/end GPS locations and headings for each maneuver
locs = [29.70983498891190, -82.35401500066121, 13;      % Start 2a
        29.70910663604740, -82.35062166849770, 134;     % End   2a
        29.70412502288820, -82.36198666890461, 244;     % S     6b
        29.7033,           -82.36,             90;      % E     6b
        29.68863665262860, -82.36857500076290, 271;     % S     1a
        29.683,            -82.3725,           180;     % E     1a
        29.65938498179120, -82.34601332346600, 271;     % S     4b
        29.6575,           -82.34730332692460, 182;     % E     4b
        29.65943164825440, -82.35261001586910, 270;     % S     4a
        29.6578,           -82.35378999710080, 180;     % E     4a
        29.65947335561120, -82.36929833094280, 270;     % S     6a
        29.656,            -82.37249832153320, 180;     % E     6a
        29.67413164774580, -82.39865334828690, 270;     % S     1b
        29.670,            -82.4015,           180;     % E     1b
        29.662,            -82.40154666900639, 180;     % S     2b
        29.6596,           -82.405,            272;     % E     2b
        29.65189666748050, -82.37493000030520, 88;      % S     3b
        29.65512332916260, -82.37237332661950, 360;     % E     3b
        29.6886,           -82.3779,           90;      % S     3a
        29.693,            -82.3723,           1];      % E     3a

lo_tol = 0.0005 ;       % low tolerance for finding maneuver locations
hi_tol = 0.0005;        % high tolerance "     "        "        "
hd_tol = 4;             % heading tolerance "  "        "        "
f = 0;clear('f');f = 0;x = 1;y = 1;null = 0;
% for each start or end location...
% find when vehicle is within prescribed range of location
% and properly oriented
for r = 1:length(locs)
    clear('f2')
    f2 = find(lat < locs(r,1)+hi_tol & lat > locs(r,1)-lo_tol & long < locs(r,2)+lo_tol &
long > locs(r,2)-hi_tol & head < locs(r,3)+hd_tol & head > locs(r,3)-hd_tol);
    if size(f2) < 1
        null(x) = r;    % list of locations not found
```

```
        x = x+1;
    else
        f3 = find(f2 > max(f));
        f(r) = min(f2(f3(1)):f2(f3(length(f3))));
        c(y) = r;       % list of locations found
        y = y+1;
    end
end

% the following steps are in the event that the computer crashed and the date
% files are incomplete. it ensures that each maneuver has a start and end
% location, otherwise they are not included.
% if first found location is not a start location, remove it from list
if rem(c(1),2) ~= 0
    w = c;
else
    w = c(2:length(c));
end
clear('c');
c = w;
% if last location is not an end location, remove it from list
if rem(c(length(c)),2) ~= 0
    w = c(1:length(c)-1);
else
    w = c;
end

% plot maneuvers and road course using GPS data
figure(1);clf
for h = w(1):2:w(length(w))-1   % plot each maneuver in list
    lo = (f(h));
    hi = (f(h+1));
    plot((long(lo:hi)),(lat(lo:hi)),'Color',[1 0 0],'LineWidth',3);
    text_name = [A(h:h+1)];
    if h == 1
        long_factor = -0.003;
    else
        long_factor = 0;
    end
    % add maneuver labels to plot
    text((long(lo))+long_factor,(lat(lo))+.0015,text_name);hold on
end
plot(long,lat)                % plot entire road course
xlim([-82.41 -82.34])
ylim([29.648 29.715])
ylabel('Latitude [Decimal Degrees]')
xlabel('Longitude [Decimal Degrees]')

% Prompt user if maneuvers were selected correctly
In_1 = input('Are the selected Maneuvers correct ? (1 = yes, 2 = no) ');
if In_1 == 1
    adjust = 0;
    adjust_index = 0;
    adjust_index_2 = 0;
    secs = 0;
    secs_index = 0;
    % open and read kinematic data file
    data2 = dlmread('data.txt',',');
    % sort kinematic data
    al = data2(:,4);     % acceleration, lateral
    af = data2(:,5);     % acceleration, forward
    sp = data2(:,1);     % speed
    sw = data2(:,7);     % steering wheel
    br = data2(:,10);    % brake
    ar = data2(:,6);     % angular rate
    lt = data2(:,2);     % left turn signal
    rt = data2(:,3);     % right turn signal
    trig = data2(:,9);   % event trigger
    % remove DC offsets
    al = al - mean(al);
    af = af - mean(af);
    ar = ar - mean(ar);
    % conversions
    al = al/1.007;          % convert to g
    af = -af/0.198;         % convert to g
    sw = -(sw+2.18)/2.92;   % convert to revolutions
    ar = ar*.581776;        % converr to rad/s
    % speed conversion
    for s = 1:length(sp)
        if sp(s) < 0.77
            sp(s) = 0;
```

```
        elseif sp(s) > 14
            sp(s) = 0;
        else
            sp(s) = sp(s)*4.5575424; % convert to mph
        end
    end
    % create new vector with every tenth sample of speed
    for i = 1:fix(length(sp)/10)
        sp2(i) = sp(i*10);
    end
    % boolean expressions for determining brake and turn signal status
    for k = 1:length(al)
        if br(k) == 1 & rt(k) == 1 & lt(k) == 0
            lt(k) = 1;rt(k) = 0;
        elseif br(k) == 1 & rt(k) == 0 & lt(k) == 1
            lt(k) = 0;rt(k) = 1;
        elseif br(k) == 1 & rt(k) == 1 & lt(k) == 1
            lt(k) = 0;rt(k) = 0;
        end
    end

    t = dt:dt:length(br);       % create time array [10 samp/sec]
    t2 = 0:1:length(br);        % create time array for speed [1 samp/sec]
    lti = 0;rti = 0;bri = 0;    % initialize left turn, right turn, and brake indices

    % correlate GPS speed and tachometer speed
    [y,lags] = xcorr(sp_gps(1:length(sp_gps)),sp2(1:length(sp2)));
    [c,I] = max(y);
    nlags = length(lags);
    nzero = nlags - (nlags-1)/2;
    tau = I - nzero;
    f = f - tau;
    % do until speeds are correlated
    while adjust_index < 1
        tau = tau + adjust;
        if tau < 0
            sp_gps(1:-tau) = 0;
            k2 = -tau + 1;
        else
            k2 = 1;
        end
        temp_sp_gps = sp_gps;
        temp_lat = lat;
        temp_long = long;
        temp_head = head;
        for k = k2:length(sp_gps)-tau
            sp_gps(k) = temp_sp_gps(k+tau);
            lat(k) = temp_lat(k+tau);
            long(k) = temp_long(k+tau);
            head(k) = temp_head(k+tau);
        end
        tau = 0;

        for q = w(1):2:w(length(w)) % one maneuver at a time
            rangelo = (f(q))/dt;        % start of maneuver
            rangehi = (f(q+1)+secs)/dt; % end of maneuver
            v = 1;u = 1;y = 1;
            % create indices for plotting turn signal and brake
            for z = rangelo:rangehi
                if rt(z) == 1
                    rt2(v) = 0.4;rti(v) = z*dt;v = v + 1;
                end
                if lt(z) == 1
                    lt2(u) = -0.4;lti(u) = z*dt;u = u + 1;
                end
                if br(z) == 1
                    br2(y) = 1;bri(y) = z*dt;y = y + 1;
                end
            end
            % plot the data for for each maneuver seperately from start location to end 
location
            figure(((q+1)/2)+1);clf
            plot(t(rangelo:rangehi),al(rangelo:rangehi),'Color',[0 0 0],'LineStyle','-
');hold on
            plot(t(rangelo:rangehi),af(rangelo:rangehi),'Color',[0 0 0],'LineStyle',':');
            plot(t(rangelo:rangehi),sw(rangelo:rangehi),'Color',[0.5 1 
0],'LineStyle',':');
            plot(t(rangelo:rangehi),ar(rangelo:rangehi),'Color',[0.5 1 0],'LineStyle','-
');
            plot(t2(rangelo/10:rangehi/10),sp2(rangelo/10:rangehi/10)/20,'Color',[0 0.5 
1],'LineStyle','-.','LineWidth',2);
```

```
            plot(t2(rangelo/10:rangehi/10),sp_gps(rangelo/10:rangehi/10)/20,'Color',[0 0
1],'LineStyle','-','LineWidth',2);
            ylim([-1 1.5])
            if lti ~= 0

plot(lti,lt2,'LineStyle','none','Marker','+','MarkerSize',3,'Color',[1,0,1]);
            else

plot(rangelo*dt,0,'LineStyle','none','Marker','+','MarkerSize',3,'Color',[1,0,1])
            end
            if rti ~= 0

plot(rti,rt2,'LineStyle','none','Marker','+','MarkerSize',3,'Color',[1,0,1]);
            else

plot(rangelo*dt,0,'LineStyle','none','Marker','+','MarkerSize',3,'Color',[1,0,1])
            end
            if bri ~= 0

plot(bri,br2,'LineStyle','none','Marker','x','MarkerSize',6,'Color','black');
            else

plot(rangelo*dt,0,'LineStyle','none','Marker','x','MarkerSize',6,'Color','black')
            end
            hold off
            legend('LA','FA','ST','YW','SPT','SPG','TS','TS','BR',0);
            xlabel('time [sec]')
            ylabel('[g], [rev], [rad/s], [mph/20]')
            Maneuver = ['Maneuver '];
            TitleName = [Maneuver, A(q:q+1)];
            title(TitleName)
            clear('rt2','rti','lt2','lti','br2','bri')
            rt2 = 0;rti = 0;lt2 = 0;lti = 0;br2 = 0;bri = 0;
        end
        % ask user if GPS speed and tachometer speed are correlated
        if adjust_index == 1
        else
            In_2 = input('Are Tach Speed (purple) and GPS Speed (yellow) less than 1
second apart ? (1 = yes, 2 = no) ');
            In_2 = 1;
        end
        % if not, which way does GPS speed need to shift
        if In_2 == 1
            adjust_index = 1;
            adjust = 0;
        else
            In_3 = input('Is the Tach speed (purple) to the right of the GPS speed
(yellow) ? (1 = yes, 2 = no) ');
            if In_3 == 1
                sign = -1;
            else
                sign = 1;
            end
            adjust = input('What is the difference in seconds ? ');
            adjust = adjust*sign;
        end
        % ask user if they would like to create output files
        In_5 = input('Would you like to create Output Files ? (1 = yes, 2 = no) ');
        if In_5 == 1
            status = 'Please Wait...'       % can take a few minutes
            for q = w(1):2:w(length(w))
                lo = (f(q))/dt;
                hi = (f(q+1)+secs)/dt;
                % create new array with GPS data every 1/10 second
                for i = 10:10:length(lat)*10
                    lat3(i-9:i) = lat(i/10);
                    sp_gps3(i-9:i) = sp_gps(i/10);
                    long3(i-9:i) = long(i/10);
                    head3(i-9:i) = head(i/10);
                end
                format long
                out =
[t(lo:hi)',af(lo:hi),al(lo:hi),sw(lo:hi),ar(lo:hi),br(lo:hi),lt(lo:hi),rt(lo:hi),(sp_gps3
(lo:hi))',lat3(lo:hi)',long3(lo:hi)',head3(lo:hi)'];
                TitleName = [Maneuver, A(q:q+1)];
                dlmwrite(TitleName,out)     % create output file
            end
        end
    end % end while statement
end % end if statement
status = 'Finished'
```

stats.m

```
% *********************************************************************
%
%                    FHWA Statisitics (ROAD COURSE)
%
%   University of Florida
%   College of Publich Health and Health Professions
%   Occupational Therapy Department
%   Created by: Ethan J. Davis (Mechanical & Aerospace Engineering)
%   Date: 08-13-05
%
%   -------------------------- INSTRUCTIONS --------------------------
%   Place this M-File in the same folder with the 10 Maneuver Files
%   Open Matlab, Open this M-File
%   Run this M-File (F5)
%
% *********************************************************************
clear all
format short g
heading = [33 224 241 241 240 240 200 68 335 70]; % heading during the middle of each
turn
A = ['2A6B1A4B6A1B2B3B4C3A'];   % text string of maneuver labels
Maneuver = ['Maneuver '];       % maneuver string
app = 0;
turn = 0;
rec = 0;
ttm = 0;
sub = input('What is the Subject Number ? ');
age = input('How old is the subject? ');
gen = input('What gender is the subject? (0 = MALE, 1 = FEMALE) ');
% determine age groups
if age < 65
    age_1 = 0;
    age_2 = 0;
elseif age > 64 & age < 76
    age_1 = 1;
    age_2 = 1;
elseif age > 75 & age < 86
    age_1 = 2;
    age_2 = 1;
else
    age_1 = 3;
    age_2 = 1;
end
% process one maneuver at a time
for k = 1:2:19
    rec_flag = 0;
    TitleName = [Maneuver, A(k:k+1)];

clear('f','f2','f3','f4','f5','data','t','af','al','sw','ar','br','lt','rt','sp','lat','l
ong','lat_ft','lonf_ft','head','t2','sp2','head2')
    data = dlmread(TitleName,',');

    if A(k+1) == 'A'
        imp = 1;
    elseif A(k+1) == 'B'
        imp = 2;
    elseif A(k+1) == 'C'
        imp = 3;
    else
        status = 'ERROR'
    end

    t = data(:,1);      % time
    af = data(:,2);     % forward accel
    al = data(:,3);     % lateral accel
    sw = data(:,4);     % steering
    ar = data(:,5);     % yaw
    br = data(:,6);     % brake
    lt = data(:,7);     % left turn
    rt = data(:,8);     % right turn
    sp = data(:,9)*10;  % speed
    lat = data(:,10);   % latitude
    long = data(:,11);  % longitude
    head = data(:,12);  % heading

    lat_ft = lat*364500;        % convert to feet
    long_ft = long*316800;      % convert to feet
```

```
    % Create new arrays for time and speed by taking every 10th sample.
    % This is to remove the stair-step appearance on the plots.
    for i = 1:length(t)/10
        t2(i) = t(i*10);
        sp2(i) = sp(i*10);
        head2(i) = head(i*10);
    end
    % find when vehicle is in middle of turn
    if k == 1 | k == 13 % i.e. right turns
        f = find(head > heading((k+1)/2) - 5 & head < heading((k+1)/2) + 50);
    else                % i.e. left turns
        f = find(head > heading((k+1)/2) - 50 & head < heading((k+1)/2) + 0);
    end

    f2 = find(abs(ar) < 0.05);
    f3 = find(f2 < min(f));
    tsi = f2(max(f3));       % turn start index
    if k == 1 | k == 13
        if k == 1
            f6 = find(head > 100);
        else
            f6 = find(head > 230);
        end
        f7 = find(f6 > tsi);
        f4 = find(f2 > f6(min(f7)));
    else
        f4 = find(f2 > min(f));
    end
    tei = f2(min(f4));       % turn end index
    rsi = tei + 1;           % recovery start index
    rei = length(ar);        % recovery end index
    asi = 1;                 % approach start index
    aei = tsi - 1;           % approach end index
    f5 = find(sp == 0);      % determine if driver stopped
    if length(f5) > 0
        stop = 1;
    else
        stop = 0;
    end

    % calculate distance traveled during approach phase
    app_dis = 0;
    for i = asi:aei-1
        app_dis = app_dis + sqrt((lat_ft(i)-lat_ft(i+1))^2+(long_ft(i)-long_ft(i+1))^2);
    end

    % calculate distance traveled during turning phase
    turn_dis = 0;
    for i = tsi:tei-1
        turn_dis = turn_dis + sqrt((lat_ft(i)-lat_ft(i+1))^2+(long_ft(i)-
long_ft(i+1))^2);
    end

    % calculate distance traveled during recovery phase (stop at 500 feet)
    rec_dis = 0;
    while rec_flag < 1
        for i = rsi:rei-1
            if rec_dis > 500
                rei = i;
                rec_flag = 1;
            else
                rec_dis = rec_dis + sqrt((lat_ft(i)-lat_ft(i+1))^2+(long_ft(i)-
long_ft(i+1))^2);
            end
        end
            rec_flag = 1;
    end

    app_time = t(aei) - t(asi);     % time to complete approach phase
    turn_time = t(tei) - t(tsi);    % time to complete turning phase
    rec_time = t(rei) - t(rsi);     % time to complete recovery phase
    com = 0;
    for i = 1:length(al)
        com(i) = sqrt(af(i)^2+al(i)^2);    % combined acceleration
    end
    % calculate statistics for each phase
    % approach
    if k == 1 | k == 13             % right turns
        factor = 1;
        app_yaw_min = min(ar(asi:aei));
```

```
    app_yaw_max = max(ar(asi:aei));
    app_lat_min = min(al(asi:aei));
    app_lat_max = max(al(asi:aei));
    app_str_min = min(sw(asi:aei));
    app_str_max = max(sw(asi:aei));
else
    factor = -1;
    app_yaw_min = -max(ar(asi:aei));
    app_yaw_max = -min(ar(asi:aei));
    app_lat_min = -max(al(asi:aei));
    app_lat_max = -min(al(asi:aei));
    app_str_min = -max(sw(asi:aei));
    app_str_max = -min(sw(asi:aei));
end
app_fwd_min = min(af(asi:aei));
app_fwd_max = max(af(asi:aei));
app_spd_min = min(sp(asi:aei));
app_spd_max = max(sp(asi:aei));
app_com_min = min(com(asi:aei));
app_com_max = max(com(asi:aei));

app_yaw_avg = mean(factor.*ar(asi:aei));
app_yaw_rms = norm(factor.*ar(asi:aei))/sqrt(length(ar(asi:aei)));
app_yaw_int = sum(factor.*ar(asi:aei))*0.1*length(ar(asi:aei));
app_yaw_var = var(factor.*ar(asi:aei),1);
app_yaw_med = median(factor.*ar(asi:aei));
app_yaw_ran = app_yaw_max - app_yaw_min;
app_yaw_skw = skewness(factor.*ar(asi:aei),0);
app_yaw_kur = kurtosis(factor.*ar(asi:aei),0);
app_yaw_s_t = sum(factor.*ar(asi:aei))/app_time;
app_yaw_s_d = sum(factor.*ar(asi:aei))/app_dis;

app_lat_avg = mean(factor.*al(asi:aei));
app_lat_rms = norm(factor.*al(asi:aei))/sqrt(length(al(asi:aei)));
app_lat_int = sum(factor.*al(asi:aei))*0.1*length(al(asi:aei));
app_lat_var = var(factor.*al(asi:aei),1);
app_lat_med = median(factor.*al(asi:aei));
app_lat_ran = app_lat_max - app_lat_min;
app_lat_skw = skewness(factor.*al(asi:aei),0);
app_lat_kur = kurtosis(factor.*al(asi:aei),0);
app_lat_s_t = sum(factor.*al(asi:aei))/app_time;
app_lat_s_d = sum(factor.*al(asi:aei))/app_dis;

app_fwd_avg = mean(af(asi:aei));
app_fwd_rms = norm(af(asi:aei))/sqrt(length(af(asi:aei)));
app_fwd_int = sum(af(asi:aei))*0.1*length(af(asi:aei));
app_fwd_var = var(af(asi:aei),1);
app_fwd_med = median(af(asi:aei));
app_fwd_ran = app_fwd_max - app_fwd_min;
app_fwd_skw = skewness(af(asi:aei),0);
app_fwd_kur = kurtosis(af(asi:aei),0);
app_fwd_s_t = sum(af(asi:aei))/app_time;
app_fwd_s_d = sum(af(asi:aei))/app_dis;

app_str_avg = mean(factor.*sw(asi:aei));
app_str_rms = norm(factor.*sw(asi:aei))/sqrt(length(sw(asi:aei)));
app_str_int = sum(factor.*sw(asi:aei))*0.1*length(sw(asi:aei));
app_str_var = var(factor.*sw(asi:aei),1);
app_str_med = median(factor.*sw(asi:aei));
app_str_ran = app_str_max - app_str_min;
app_str_skw = skewness(factor.*sw(asi:aei),0);
app_str_kur = kurtosis(factor.*sw(asi:aei),0);
app_str_s_t = sum(factor.*sw(asi:aei))/app_time;
app_str_s_d = sum(factor.*sw(asi:aei))/app_dis;

app_spd_avg = mean(sp(asi:aei));
app_spd_rms = norm(sp(asi:aei))/sqrt(length(sp(asi:aei)));
app_spd_int = sum(sp(asi:aei))*0.1*length(sp(asi:aei));
app_spd_var = var(sp(asi:aei),1);
app_spd_med = median(sp(asi:aei));
app_spd_ran = app_spd_max - app_spd_min;
app_spd_skw = skewness(sp(asi:aei),0);
app_spd_kur = kurtosis(sp(asi:aei),0);
app_spd_s_t = sum(sp(asi:aei))/app_time;
app_spd_s_d = sum(sp(asi:aei))/app_dis;

app_com_avg = mean(com(asi:aei));
app_com_rms = norm(com(asi:aei))/sqrt(length(com(asi:aei)));
app_com_int = sum(com(asi:aei))*0.1*length(com(asi:aei));
app_com_var = var(com(asi:aei),1);
app_com_med = median(com(asi:aei));
```

```
app_com_ran = app_com_max - app_com_min;
app_com_skw = skewness(com(asi:aei),0);
app_com_kur = kurtosis(com(asi:aei),0);
app_com_s_t = sum(com(asi:aei))/app_time;
app_com_s_d = sum(com(asi:aei))/app_dis;

% turn
if k == 1 | k == 13
    factor = 1;
    turn_yaw_min = min(ar(tsi:tei));
    turn_yaw_max = max(ar(tsi:tei));
    turn_lat_min = min(al(tsi:tei));
    turn_lat_max = max(al(tsi:tei));
    turn_str_min = min(sw(tsi:tei));
    turn_str_max = max(sw(tsi:tei));
else
    factor = -1;
    turn_yaw_min = -max(ar(tsi:tei));
    turn_yaw_max = -min(ar(tsi:tei));
    turn_lat_min = -max(al(tsi:tei));
    turn_lat_max = -min(al(tsi:tei));
    turn_str_min = -max(sw(tsi:tei));
    turn_str_max = -min(sw(tsi:tei));
end
turn_fwd_min = min(af(tsi:tei));
turn_fwd_max = max(af(tsi:tei));
turn_spd_min = min(sp(tsi:tei));
turn_spd_max = max(sp(tsi:tei));
turn_com_min = min(com(tsi:tei));
turn_com_max = max(com(tsi:tei));

turn_yaw_avg = mean(factor.*ar(tsi:tei));
turn_yaw_rms = norm(factor.*ar(tsi:tei))/sqrt(length(ar(tsi:tei)));
turn_yaw_int = sum(factor.*ar(tsi:tei))*0.1*length(ar(tsi:tei));
turn_yaw_var = var(factor.*ar(tsi:tei),1);
turn_yaw_med = median(factor.*ar(tsi:tei));
turn_yaw_ran = turn_yaw_max - turn_yaw_min;
turn_yaw_skw = skewness(factor.*ar(tsi:tei),0);
turn_yaw_kur = kurtosis(factor.*ar(tsi:tei),0);
turn_yaw_s_t = sum(factor.*ar(tsi:tei))/turn_time;
turn_yaw_s_d = sum(factor.*ar(tsi:tei))/turn_dis;

turn_lat_avg = mean(factor.*al(tsi:tei));
turn_lat_rms = norm(factor.*al(tsi:tei))/sqrt(length(al(tsi:tei)));
turn_lat_int = sum(factor.*al(tsi:tei))*0.1*length(al(tsi:tei));
turn_lat_var = var(factor.*al(tsi:tei),1);
turn_lat_med = median(factor.*al(tsi:tei));
turn_lat_ran = turn_lat_max - turn_lat_min;
turn_lat_skw = skewness(factor.*al(tsi:tei),0);
turn_lat_kur = kurtosis(factor.*al(tsi:tei),0);
turn_lat_s_t = sum(factor.*al(tsi:tei))/turn_time;
turn_lat_s_d = sum(factor.*al(tsi:tei))/turn_dis;

turn_fwd_avg = mean(af(tsi:tei));
turn_fwd_rms = norm(af(tsi:tei))/sqrt(length(af(tsi:tei)));
turn_fwd_int = sum(af(tsi:tei))*0.1*length(af(tsi:tei));
turn_fwd_var = var(af(tsi:tei),1);
turn_fwd_med = median(af(tsi:tei));
turn_fwd_ran = turn_fwd_max - turn_fwd_min;
turn_fwd_skw = skewness(af(tsi:tei),0);
turn_fwd_kur = kurtosis(af(tsi:tei),0);
turn_fwd_s_t = sum(af(tsi:tei))/turn_time;
turn_fwd_s_d = sum(af(tsi:tei))/turn_dis;

turn_str_avg = mean(factor.*sw(tsi:tei));
turn_str_rms = norm(factor.*sw(tsi:tei))/sqrt(length(sw(tsi:tei)));
turn_str_int = sum(factor.*sw(tsi:tei))*0.1*length(sw(tsi:tei));
turn_str_var = var(factor.*sw(tsi:tei),1);
turn_str_med = median(factor.*sw(tsi:tei));
turn_str_ran = turn_str_max - turn_str_min;
turn_str_skw = skewness(factor.*sw(tsi:tei),0);
turn_str_kur = kurtosis(factor.*sw(tsi:tei),0);
turn_str_s_t = sum(factor.*sw(tsi:tei))/turn_time;
turn_str_s_d = sum(factor.*sw(tsi:tei))/turn_dis;

turn_spd_avg = mean(sp(tsi:tei));
turn_spd_rms = norm(sp(tsi:tei))/sqrt(length(sp(tsi:tei)));
turn_spd_int = sum(sp(tsi:tei))*0.1*length(sp(tsi:tei));
turn_spd_var = var(sp(tsi:tei),1);
turn_spd_med = median(sp(tsi:tei));
turn_spd_ran = turn_spd_max - turn_spd_min;
```

```
turn_spd_skw = skewness(sp(tsi:tei),0);
turn_spd_kur = kurtosis(sp(tsi:tei),0);
turn_spd_s_t = sum(sp(tsi:tei))/turn_time;
turn_spd_s_d = sum(sp(tsi:tei))/turn_dis;

turn_com_avg = mean(com(tsi:tei));
turn_com_rms = norm(com(tsi:tei))/sqrt(length(com(tsi:tei)));
turn_com_int = sum(com(tsi:tei))*0.1*length(com(tsi:tei));
turn_com_var = var(com(tsi:tei),1);
turn_com_med = median(com(tsi:tei));
turn_com_ran = turn_com_max - turn_com_min;
turn_com_skw = skewness(com(tsi:tei),0);
turn_com_kur = kurtosis(com(tsi:tei),0);
turn_com_s_t = sum(com(tsi:tei))/turn_time;
turn_com_s_d = sum(com(tsi:tei))/turn_dis;

% recovery
if k == 1 | k == 13
    factor = 1;
    rec_yaw_min = min(ar(rsi:rei));
    rec_yaw_max = max(ar(rsi:rei));
    rec_lat_min = min(al(rsi:rei));
    rec_lat_max = max(al(rsi:rei));
    rec_str_min = min(sw(rsi:rei));
    rec_str_max = max(sw(rsi:rei));
else
    factor = -1;
    rec_yaw_min = -max(ar(rsi:rei));
    rec_yaw_max = -min(ar(rsi:rei));
    rec_lat_min = -max(al(rsi:rei));
    rec_lat_max = -min(al(rsi:rei));
    rec_str_min = -max(sw(rsi:rei));
    rec_str_max = -min(sw(rsi:rei));
end
rec_fwd_min = min(af(rsi:rei));
rec_fwd_max = max(af(rsi:rei));
rec_spd_min = min(sp(rsi:rei));
rec_spd_max = max(sp(rsi:rei));
rec_com_min = min(com(rsi:rei));
rec_com_max = max(com(rsi:rei));

rec_yaw_avg = mean(factor.*ar(rsi:rei));
rec_yaw_rms = norm(factor.*ar(rsi:rei))/sqrt(length(ar(rsi:rei)));
rec_yaw_int = sum(factor.*ar(rsi:rei))*0.1*length(ar(rsi:rei));
rec_yaw_var = var(factor.*ar(rsi:rei),1);
rec_yaw_med = median(factor.*ar(rsi:rei));
rec_yaw_ran = rec_yaw_max - rec_yaw_min;
rec_yaw_skw = skewness(factor.*ar(rsi:rei),0);
rec_yaw_kur = kurtosis(factor.*ar(rsi:rei),0);
rec_yaw_s_t = sum(factor.*ar(rsi:rei))/rec_time;
rec_yaw_s_d = sum(factor.*ar(rsi:rei))/rec_dis;

rec_lat_avg = mean(factor.*al(rsi:rei));
rec_lat_rms = norm(factor.*al(rsi:rei))/sqrt(length(al(rsi:rei)));
rec_lat_int = sum(factor.*al(rsi:rei))*0.1*length(al(rsi:rei));
rec_lat_var = var(factor.*al(rsi:rei),1);
rec_lat_med = median(factor.*al(rsi:rei));
rec_lat_ran = rec_lat_max - rec_lat_min;
rec_lat_skw = skewness(factor.*al(rsi:rei),0);
rec_lat_kur = kurtosis(factor.*al(rsi:rei),0);
rec_lat_s_t = sum(factor.*al(rsi:rei))/rec_time;
rec_lat_s_d = sum(factor.*al(rsi:rei))/rec_dis;

rec_fwd_avg = mean(af(rsi:rei));
rec_fwd_rms = norm(af(rsi:rei))/sqrt(length(af(rsi:rei)));
rec_fwd_int = sum(af(rsi:rei))*0.1*length(af(rsi:rei));
rec_fwd_var = var(af(rsi:rei),1);
rec_fwd_med = median(af(rsi:rei));
rec_fwd_ran = rec_fwd_max - rec_fwd_min;
rec_fwd_skw = skewness(af(rsi:rei),0);
rec_fwd_kur = kurtosis(af(rsi:rei),0);
rec_fwd_s_t = sum(af(rsi:rei))/rec_time;
rec_fwd_s_d = sum(af(rsi:rei))/rec_dis;

rec_str_avg = mean(factor.*sw(rsi:rei));
rec_str_rms = norm(factor.*sw(rsi:rei))/sqrt(length(sw(rsi:rei)));
rec_str_int = sum(factor.*sw(rsi:rei))*0.1*length(sw(rsi:rei));
rec_str_var = var(factor.*sw(rsi:rei),1);
rec_str_med = median(factor.*sw(rsi:rei));
rec_str_ran = rec_str_max - rec_str_min;
rec_str_skw = skewness(factor.*sw(rsi:rei),0);
```

```
    rec_str_kur = kurtosis(factor.*sw(rsi:rei),0);
    rec_str_s_t = sum(factor.*sw(rsi:rei))/rec_time;
    rec_str_s_d = sum(factor.*sw(rsi:rei))/rec_dis;

    rec_spd_avg = mean(sp(rsi:rei));
    rec_spd_rms = norm(sp(rsi:rei))/sqrt(length(sp(rsi:rei)));
    rec_spd_int = sum(sp(rsi:rei))*0.1*length(sp(rsi:rei));
    rec_spd_var = var(sp(rsi:rei),1);
    rec_spd_med = median(sp(rsi:rei));
    rec_spd_ran = rec_spd_max - rec_spd_min;
    rec_spd_skw = skewness(sp(rsi:rei),0);
    rec_spd_kur = kurtosis(sp(rsi:rei),0);
    rec_spd_s_t = sum(sp(rsi:rei))/rec_time;
    rec_spd_s_d = sum(sp(rsi:rei))/rec_dis;

    rec_com_avg = mean(com(rsi:rei));
    rec_com_rms = norm(com(rsi:rei))/sqrt(length(com(rsi:rei)));
    rec_com_int = sum(com(rsi:rei))*0.1*length(com(rsi:rei));
    rec_com_var = var(com(rsi:rei),1);
    rec_com_med = median(com(rsi:rei));
    rec_com_ran = rec_com_max - rec_com_min;
    rec_com_skw = skewness(com(rsi:rei),0);
    rec_com_kur = kurtosis(com(rsi:rei),0);
    rec_com_s_t = sum(com(rsi:rei))/rec_time;
    rec_com_s_d = sum(com(rsi:rei))/rec_dis;

    % compile statistics
    man = str2num(A(k));
    app_temp =
[sub,age_1,age_2,gen,man,imp,stop,app_time,app_dis,app_yaw_min,app_yaw_max,app_yaw_avg,ap
p_yaw_rms,app_yaw_int,app_yaw_var,app_yaw_med,app_yaw_ran,app_yaw_skw,app_yaw_kur,app_yaw
_s_t,app_yaw_s_d,app_lat_min,app_lat_max,app_lat_avg,app_lat_rms,app_lat_int,app_lat_var,
app_lat_med,app_lat_ran,app_lat_skw,app_lat_kur,app_lat_s_t,app_lat_s_d,app_fwd_min,app_f
wd_max,app_fwd_avg,app_fwd_rms,app_fwd_int,app_fwd_var,app_fwd_med,app_fwd_ran,app_fwd_sk
w,app_fwd_kur,app_fwd_s_t,app_fwd_s_d,app_str_min,app_str_max,app_str_avg,app_str_rms,app
_str_int,app_str_var,app_str_med,app_str_ran,app_str_skw,app_str_kur,app_str_s_t,app_str_
s_d,app_spd_min,app_spd_max,app_spd_avg,app_spd_rms,app_spd_int,app_spd_var,app_spd_med,a
pp_spd_ran,app_spd_skw,app_spd_kur,app_spd_s_t,app_spd_s_d,app_com_min,app_com_max,app_co
m_avg,app_com_rms,app_com_int,app_com_var,app_com_med,app_com_ran,app_com_skw,app_com_kur
,app_com_s_t,app_com_s_d];
    if length(app) > 1
        app = [app;app_temp];   % approach stats
    else
        app = [app_temp];
    end

    turn_temp =
[sub,age_1,age_2,gen,man,imp,stop,turn_time,turn_dis,turn_yaw_min,turn_yaw_max,turn_yaw_a
vg,turn_yaw_rms,turn_yaw_int,turn_yaw_var,turn_yaw_med,turn_yaw_ran,turn_yaw_skw,turn_yaw
_kur,turn_yaw_s_t,turn_yaw_s_d,turn_lat_min,turn_lat_max,turn_lat_avg,turn_lat_rms,turn_l
at_int,turn_lat_var,turn_lat_med,turn_lat_ran,turn_lat_skw,turn_lat_kur,turn_lat_s_t,turn
_lat_s_d,turn_fwd_min,turn_fwd_max,turn_fwd_avg,turn_fwd_rms,turn_fwd_int,turn_fwd_var,tu
rn_fwd_med,turn_fwd_ran,turn_fwd_skw,turn_fwd_kur,turn_fwd_s_t,turn_fwd_s_d,turn_str_min,
turn_str_max,turn_str_avg,turn_str_rms,turn_str_int,turn_str_var,turn_str_med,turn_str_ra
n,turn_str_skw,turn_str_kur,turn_str_s_t,turn_str_s_d,turn_spd_min,turn_spd_max,turn_spd_
avg,turn_spd_rms,turn_spd_int,turn_spd_var,turn_spd_med,turn_spd_ran,turn_spd_skw,turn_sp
d_kur,turn_spd_s_t,turn_spd_s_d,turn_com_min,turn_com_max,turn_com_avg,turn_com_rms,turn_
com_int,turn_com_var,turn_com_med,turn_com_ran,turn_com_skw,turn_com_kur,turn_com_s_t,tur
n_com_s_d];
    if length(turn) > 1
        turn = [turn;turn_temp];% turn stats
    else
        turn = [turn_temp];
    end

    rec_temp =
[sub,age_1,age_2,gen,man,imp,stop,rec_time,rec_dis,rec_yaw_min,rec_yaw_max,rec_yaw_avg,re
c_yaw_rms,rec_yaw_int,rec_yaw_var,rec_yaw_med,rec_yaw_ran,rec_yaw_skw,rec_yaw_kur,rec_yaw
_s_t,rec_yaw_s_d,rec_lat_min,rec_lat_max,rec_lat_avg,rec_lat_rms,rec_lat_int,rec_lat_var,
rec_lat_med,rec_lat_ran,rec_lat_skw,rec_lat_kur,rec_lat_s_t,rec_lat_s_d,rec_fwd_min,rec_f
wd_max,rec_fwd_avg,rec_fwd_rms,rec_fwd_int,rec_fwd_var,rec_fwd_med,rec_fwd_ran,rec_fwd_sk
w,rec_fwd_kur,rec_fwd_s_t,rec_fwd_s_d,rec_str_min,rec_str_max,rec_str_avg,rec_str_rms,rec
_str_int,rec_str_var,rec_str_med,rec_str_ran,rec_str_skw,rec_str_kur,rec_str_s_t,rec_str_
s_d,rec_spd_min,rec_spd_max,rec_spd_avg,rec_spd_rms,rec_spd_int,rec_spd_var,rec_spd_med,r
ec_spd_ran,rec_spd_skw,rec_spd_kur,rec_spd_s_t,rec_spd_s_d,rec_com_min,rec_com_max,rec_co
m_avg,rec_com_rms,rec_com_int,rec_com_var,rec_com_med,rec_com_ran,rec_com_skw,rec_com_kur
,rec_com_s_t,rec_com_s_d];
    if length(rec) > 1
        rec = [rec;rec_temp];   % recovery stats
    else
        rec = [rec_temp];
```

```
    end

    % plot each maneuver
    figure((k+1)/2);clf
    plot(t,al,'Color',[0 0 0],'LineStyle','-');hold on
    plot(t,af,'Color',[0 0 0],'LineStyle',':');
    plot(t,sw,'Color',[0.5 1 0],'LineStyle',':');
    plot(t,ar,'Color',[0.5 1 0],'LineStyle','-');
    plot(t2,sp2/20,'Color',[0 0.5 1],'LineStyle','-.','LineWidth',2);
    plot(t(tsi),0,'Marker','x','LineWidth',2,'MarkerSize',16,'Color',[0,0,0]);
    plot(t(tei),0,'Marker','x','LineWidth',2,'MarkerSize',16,'Color',[0,0,0]);hold off
    st = ['(Subject )'];TitleName2 = [TitleName,' ',st(1:9),num2str(sub),st(10)];
    ylim([-1 1.5])
    title(TitleName2)
    legend('LA','FA','ST','YW','SP',0);
    xlabel('time [sec]')
    ylabel('g, rev, rad/s, mph/20')
end

% create statistic output files
appstr = ['app_'];
subject = ['subject_'];
AppName = [appstr,subject,num2str(sub),'.txt'];
dlmwrite(AppName,turn)

turnstr = ['turn_'];
TurnName = [turnstr,subject,num2str(sub),'.txt'];
dlmwrite(TurnName,turn)

recstr = ['recovery_'];
RecName = [recstr,subject,num2str(sub),'.txt'];
dlmwrite(RecName,rec)

status = 'Finished'
```

sort.m

```
% *********************************************************************
%
%            FHWA - Sorting Program (organizes stats by maneuver)
%
%   University of Florida
%   College of Publich Health and Health Professions
%   Occupational Therapy Department
%   Created by: Ethan J. Davis (Mechanical & Aerospace Engineering)
%   Date: 09-15-05
%
%   ------------------------- INSTRUCTIONS --------------------------
%   Place this M-File in the same folder with all of the Statistic Files
%   (i.e. "turn_subject_23.txt" )
%   Open Matlab, Open this M-File
%   Run this M-File (F5)
%
%  --------------------------------------------------------------------
%   *** NOTE: In order to process a given subject, the subject's number
%   must be added to the "subs" vector on line 25 (see below) ***
%  --------------------------------------------------------------------
%
% *********************************************************************

clear all
subs = [1,2,3];    % <<---- ADD SUBJECT NUMBERS HERE
app_str = ['app_'];
turn_str = ['turn_'];
rec_str = ['recovery_'];
sub_str = ['subject_'];
% initialize arrays
man_1_app = 0;
man_2_app = 0;
man_3_app = 0;
man_4_app = 0;
man_6_app = 0;
man_1_turn = 0;
man_2_turn = 0;
man_3_turn = 0;
man_4_turn = 0;
man_6_turn = 0;
man_1_rec = 0;
man_2_rec = 0;
```

```
man_3_rec = 0;
man_4_rec = 0;
man_6_rec = 0;

% process 1 subject at a time
for cnt = 1:length(subs)
    app_file_name = [app_str,sub_str,num2str(subs(cnt)),'.txt'];
    app_data = dlmread(app_file_name);  % read approach file
    turn_file_name = [turn_str,sub_str,num2str(subs(cnt)),'.txt'];
    turn_data = dlmread(turn_file_name);% read turn file
    rec_file_name = [rec_str,sub_str,num2str(subs(cnt)),'.txt'];
    rec_data = dlmread(rec_file_name);  % read recovery file

        % sort each file by maneuver
        man_1A_temp_app = app_data(3,:);
        man_1B_temp_app = app_data(7,:);
        man_2A_temp_app = app_data(1,:);
        man_2B_temp_app = app_data(8,:);
        man_3A_temp_app = app_data(10,:);
        man_3B_temp_app = app_data(9,:);
        man_4A_temp_app = app_data(5,:);
        man_4B_temp_app = app_data(4,:);
        man_6A_temp_app = app_data(6,:);
        man_6B_temp_app = app_data(2,:);

        man_1A_temp_turn = turn_data(3,:);
        man_1B_temp_turn = turn_data(7,:);
        man_2A_temp_turn = turn_data(1,:);
        man_2B_temp_turn = turn_data(8,:);
        man_3A_temp_turn = turn_data(10,:);
        man_3B_temp_turn = turn_data(9,:);
        man_4A_temp_turn = turn_data(5,:);
        man_4B_temp_turn = turn_data(4,:);
        man_6A_temp_turn = turn_data(6,:);
        man_6B_temp_turn = turn_data(2,:);

        man_1A_temp_rec = rec_data(3,:);
        man_1B_temp_rec = rec_data(7,:);
        man_2A_temp_rec = rec_data(1,:);
        man_2B_temp_rec = rec_data(8,:);
        man_3A_temp_rec = rec_data(10,:);
        man_3B_temp_rec = rec_data(9,:);
        man_4A_temp_rec = rec_data(5,:);
        man_4B_temp_rec = rec_data(4,:);
        man_6A_temp_rec = rec_data(6,:);
        man_6B_temp_rec = rec_data(2,:);

    % create statistic files compiled by maneuver
    if length(man_1_turn) > 1
        man_1_app = [man_1_app;man_1A_temp_app,man_1B_temp_app];
        man_2_app = [man_2_app;man_2A_temp_app,man_2B_temp_app];
        man_3_app = [man_3_app;man_3A_temp_app,man_3B_temp_app];
        man_4_app = [man_4_app;man_4A_temp_app,man_4B_temp_app];
        man_6_app = [man_6_app;man_6A_temp_app,man_6B_temp_app];
        man_1_turn = [man_1_turn;man_1A_temp_turn,man_1B_temp_turn];
        man_2_turn = [man_2_turn;man_2A_temp_turn,man_2B_temp_turn];
        man_3_turn = [man_3_turn;man_3A_temp_turn,man_3B_temp_turn];
        man_4_turn = [man_4_turn;man_4A_temp_turn,man_4B_temp_turn];
        man_6_turn = [man_6_turn;man_6A_temp_turn,man_6B_temp_turn];
        man_1_rec = [man_1_rec;man_1A_temp_rec,man_1B_temp_rec];
        man_2_rec = [man_2_rec;man_2A_temp_rec,man_2B_temp_rec];
        man_3_rec = [man_3_rec;man_3A_temp_rec,man_3B_temp_rec];
        man_4_rec = [man_4_rec;man_4A_temp_rec,man_4B_temp_rec];
        man_6_rec = [man_6_rec;man_6A_temp_rec,man_6B_temp_rec];
    else
        man_1_app = [man_1A_temp_app,man_1B_temp_app];
        man_2_app = [man_2A_temp_app,man_2B_temp_app];
        man_3_app = [man_3A_temp_app,man_3B_temp_app];
        man_4_app = [man_4A_temp_app,man_4B_temp_app];
        man_6_app = [man_6A_temp_app,man_6B_temp_app];
        man_1_turn = [man_1A_temp_turn,man_1B_temp_turn];
        man_2_turn = [man_2A_temp_turn,man_2B_temp_turn];
        man_3_turn = [man_3A_temp_turn,man_3B_temp_turn];
        man_4_turn = [man_4A_temp_turn,man_4B_temp_turn];
        man_6_turn = [man_6A_temp_turn,man_6B_temp_turn];
        man_1_rec = [man_1A_temp_rec,man_1B_temp_rec];
        man_2_rec = [man_2A_temp_rec,man_2B_temp_rec];
        man_3_rec = [man_3A_temp_rec,man_3B_temp_rec];
        man_4_rec = [man_4A_temp_rec,man_4B_temp_rec];
        man_6_rec = [man_6A_temp_rec,man_6B_temp_rec];
    end
```

```
end

man_str = ['Maneuver_'];
sta_str = ['_Stats'];
app_str = ['_Approach'];
turn_str = ['_Turn'];
rec_str = ['_Recovery'];

% create output files
man_1_fn_app = [man_str,num2str(1),app_str,sta_str,'.txt'];
man_2_fn_app = [man_str,num2str(2),app_str,sta_str,'.txt'];
man_3_fn_app = [man_str,num2str(3),app_str,sta_str,'.txt'];
man_4_fn_app = [man_str,num2str(4),app_str,sta_str,'.txt'];
man_6_fn_app = [man_str,num2str(6),app_str,sta_str,'.txt'];
dlmwrite(man_1_fn_app,man_1_app)
dlmwrite(man_2_fn_app,man_2_app)
dlmwrite(man_3_fn_app,man_3_app)
dlmwrite(man_4_fn_app,man_4_app)
dlmwrite(man_6_fn_app,man_6_app)

man_1_fn_turn = [man_str,num2str(1),turn_str,sta_str,'.txt'];
man_2_fn_turn = [man_str,num2str(2),turn_str,sta_str,'.txt'];
man_3_fn_turn = [man_str,num2str(3),turn_str,sta_str,'.txt'];
man_4_fn_turn = [man_str,num2str(4),turn_str,sta_str,'.txt'];
man_6_fn_turn = [man_str,num2str(6),turn_str,sta_str,'.txt'];
dlmwrite(man_1_fn_turn,man_1_turn)
dlmwrite(man_2_fn_turn,man_2_turn)
dlmwrite(man_3_fn_turn,man_3_turn)
dlmwrite(man_4_fn_turn,man_4_turn)
dlmwrite(man_6_fn_turn,man_6_turn)

man_1_fn_rec = [man_str,num2str(1),rec_str,sta_str,'.txt'];
man_2_fn_rec = [man_str,num2str(2),rec_str,sta_str,'.txt'];
man_3_fn_rec = [man_str,num2str(3),rec_str,sta_str,'.txt'];
man_4_fn_rec = [man_str,num2str(4),rec_str,sta_str,'.txt'];
man_6_fn_rec = [man_str,num2str(6),rec_str,sta_str,'.txt'];
dlmwrite(man_1_fn_rec,man_1_rec)
dlmwrite(man_2_fn_rec,man_2_rec)
dlmwrite(man_3_fn_rec,man_3_rec)
dlmwrite(man_4_fn_rec,man_4_rec)
dlmwrite(man_6_fn_rec,man_6_rec)

status = 'Finished'
```

sim.m

```
% ********************************************************************
%
%                    Simulator Scenario Analysis
%
%   University of Florida
%   College of Publich Health and Health Professions
%   Occupational Therapy Department
%   Created by: Ethan J. Davis (Mechanical & Aerospace Engineering)
%   Date: 8-25-05
%
%   -------------------------- INSTRUCTIONS ---------------------------
%   Place this M-File in the same folder with all of the Scenario
%   Files (i.e. "scenario_1a.csv" )
%   Open Matlab, Open this M-File
%   Run this M-File (F5)
%
% ********************************************************************
clear all
format short g

sub = input('What is the Subject Number ? ');
age = input('How old is the subject? ');
gen = input('What gender is the subject? (0 = MALE, 1 = FEMALE) ');
% determine age groups
if age < 65
    age_1 = 0;
    age_2 = 0;
elseif age > 64 & age < 76
    age_1 = 1;
    age_2 = 1;
elseif age > 75 & age < 86
    age_1 = 2;
    age_2 = 1;
```

```
else
    age_1 = 3;
    age_2 = 1;
end

man_labels = ['1a1b1c2a2b3a3b4a4b5a5b'];    % maneuver labels
str_1 = ['fhwa'];
str_2 = ['scenario_'];
turn = 0;
rec = 0;
app = 0;

% process one maneuver at a time
for k = 1:2:21
    filename = [str_2,man_labels(k:k+1),'.csv'];
    data = dlmread(filename);   % read file
    % convert improvement letters to numbers
    if man_labels(k+1) == 'a'
        imp = 1;
    elseif man_labels(k+1) == 'b'
        imp = 2;
    elseif man_labels(k+1) == 'c'
        imp = 3;
    else
        status = 'ERROR'
    end

    af = data(:,2)/32.174;      % forward accel
    al = data(:,3)/32.174;      % lateral accel
    sw = data(:,7)/360;         % steering
    ar = data(:,9);             % yaw
    sp = data(:,8);             % speed
    head = data(:,6);           % heading
    t = 0.1:0.1:length(head)/10;% time

    if k == 7 | k == 9          % right turns
        f = find(head > 45);    % find middle of turn
    else
        f = find(head < -45);   % find middle of turn
    end

    f2 = find(abs(ar) < 0.05);
    f3 = find(f2 < min(f));
    f4 = find(f2 > min(f));
    tsi = f2(max(f3));      % turn start index
    tei = f2(min(f4));      % turn end index
    rsi = tei + 1;          % recovery start index
    rei = length(af);       % recovery end index
    asi = 1;                % approach start index
    aei = tsi - 1;          % approach end index

    % plot maneuver
    figure((k+1)/2);clf
    plot(t,al(rangelo:rangehi),'Color',[0 0 0],'LineStyle','-');hold on
    plot(t,af(rangelo:rangehi),'Color',[0 0 0],'LineStyle',':');
    plot(t,sw(rangelo:rangehi),'Color',[0.5 1 0],'LineStyle',':');
    plot(t,ar(rangelo:rangehi),'Color',[0.5 1 0],'LineStyle','-');
    plot(t,sp/20,'Color',[0 0.5 1],'LineStyle','-.','LineWidth',2);
    plot(t(tsi),0,'Marker','x','MarkerSize',12,'Color',[0,0,0]);
    plot(t(tei),0,'Marker','x','MarkerSize',12,'Color',[0,0,0]);hold
    ylim([-2 2])
    legend('LA','FA','ST','YW','SP',0)
    st = ['(Subject )'];
    title_str = ['Maneuver ',man_labels(k:k+1),' ',st(1:9),num2str(sub),st(10)];
    title(title_str)
    xlabel('Time [sec]')
    ylabel('[g], [rev], [rad/s], [mph/20]')

    f_stop = find(sp == 0); % determine if subject stopped
    if length(f_stop) > 0
        stop = 1;
    else
        stop = 0;
    end
    turn_dis = 1;
    rec_dis = 1;
    app_dis = 1;
    app_time = t(aei) - t(asi);     % time to complete approach phase
    turn_time = t(tei) - t(tsi);    % time to complete turning phase
    rec_time = t(rei) - t(rsi);     % time to complete recovery phase
    com = 0;
```

```
for i = 1:length(al)
    com(i) = sqrt(af(i)^2+al(i)^2);    % combined acceleration for turn
end

% calculate statistics for each measurement of each phase
% approach
if k == 7 | k == 9
    factor = 1;
    app_yaw_min = min(ar(asi:aei)); % minumum
    app_yaw_max = max(ar(asi:aei)); % maximum
    app_lat_min = min(al(asi:aei));
    app_lat_max = max(al(asi:aei));
    app_str_min = min(sw(asi:aei));
    app_str_max = max(sw(asi:aei));
else
    factor = -1;
    app_yaw_min = -max(ar(asi:aei));
    app_yaw_max = -min(ar(asi:aei));
    app_lat_min = -max(al(asi:aei));
    app_lat_max = -min(al(asi:aei));
    app_str_min = -max(sw(asi:aei));
    app_str_max = -min(sw(asi:aei));
end
app_fwd_min = min(af(asi:aei));
app_fwd_max = max(af(asi:aei));
app_spd_min = min(sp(asi:aei));
app_spd_max = max(sp(asi:aei));
app_com_min = min(com(asi:aei));
app_com_max = max(com(asi:aei));

app_yaw_avg = mean(factor.*ar(asi:aei)); % mean
app_yaw_rms = norm(factor.*ar(asi:aei))/sqrt(length(ar(asi:aei))); % rms
app_yaw_int = sum(factor.*ar(asi:aei))*0.1*length(ar(asi:aei)); % integral
app_yaw_var = var(factor.*ar(asi:aei),1); % variance
app_yaw_med = median(factor.*ar(asi:aei)); % median
app_yaw_ran = app_yaw_max - app_yaw_min;   % range
app_yaw_skw = skewness(factor.*ar(asi:aei),0); % skewness
app_yaw_kur = kurtosis(factor.*ar(asi:aei),0); % kurtosis
app_yaw_s_t = sum(factor.*ar(asi:aei))/app_time; % sum/time
app_yaw_s_d = sum(factor.*ar(asi:aei))/app_dis;  % sum/distance

app_lat_avg = mean(factor.*al(asi:aei));
app_lat_rms = norm(factor.*al(asi:aei))/sqrt(length(al(asi:aei)));
app_lat_int = sum(factor.*al(asi:aei))*0.1*length(al(asi:aei));
app_lat_var = var(factor.*al(asi:aei),1);
app_lat_med = median(factor.*al(asi:aei));
app_lat_ran = app_lat_max - app_lat_min;
app_lat_skw = skewness(factor.*al(asi:aei),0);
app_lat_kur = kurtosis(factor.*al(asi:aei),0);
app_lat_s_t = sum(factor.*al(asi:aei))/app_time;
app_lat_s_d = sum(factor.*al(asi:aei))/app_dis;

app_fwd_avg = mean(af(asi:aei));
app_fwd_rms = norm(af(asi:aei))/sqrt(length(af(asi:aei)));
app_fwd_int = sum(af(asi:aei))*0.1*length(af(asi:aei));
app_fwd_var = var(af(asi:aei),1);
app_fwd_med = median(af(asi:aei));
app_fwd_ran = app_fwd_max - app_fwd_min;
app_fwd_skw = skewness(af(asi:aei),0);
app_fwd_kur = kurtosis(af(asi:aei),0);
app_fwd_s_t = sum(af(asi:aei))/app_time;
app_fwd_s_d = sum(af(asi:aei))/app_dis;

app_str_avg = mean(factor.*sw(asi:aei));
app_str_rms = norm(factor.*sw(asi:aei))/sqrt(length(sw(asi:aei)));
app_str_int = sum(factor.*sw(asi:aei))*0.1*length(sw(asi:aei));
app_str_var = var(factor.*sw(asi:aei),1);
app_str_med = median(factor.*sw(asi:aei));
app_str_ran = app_str_max - app_str_min;
app_str_skw = skewness(factor.*sw(asi:aei),0);
app_str_kur = kurtosis(factor.*sw(asi:aei),0);
app_str_s_t = sum(factor.*sw(asi:aei))/app_time;
app_str_s_d = sum(factor.*sw(asi:aei))/app_dis;

app_spd_avg = mean(sp(asi:aei));
app_spd_rms = norm(sp(asi:aei))/sqrt(length(sp(asi:aei)));
app_spd_int = sum(sp(asi:aei))*0.1*length(sp(asi:aei));
app_spd_var = var(sp(asi:aei),1);
app_spd_med = median(sp(asi:aei));
app_spd_ran = app_spd_max - app_spd_min;
app_spd_skw = skewness(sp(asi:aei),0);
```

```
app_spd_kur = kurtosis(sp(asi:aei),0);
app_spd_s_t = sum(sp(asi:aei))/app_time;
app_spd_s_d = sum(sp(asi:aei))/app_dis;

app_com_avg = mean(com(asi:aei));
app_com_rms = norm(com(asi:aei))/sqrt(length(com(asi:aei)));
app_com_int = sum(com(asi:aei))*0.1*length(com(asi:aei));
app_com_var = var(com(asi:aei),1);
app_com_med = median(com(asi:aei));
app_com_ran = app_com_max - app_com_min;
app_com_skw = skewness(com(asi:aei),0);
app_com_kur = kurtosis(com(asi:aei),0);
app_com_s_t = sum(com(asi:aei))/app_time;
app_com_s_d = sum(com(asi:aei))/app_dis;

% turn
if k == 7 | k == 9
    factor = 1;
    turn_yaw_min = min(ar(tsi:tei));
    turn_yaw_max = max(ar(tsi:tei));
    turn_lat_min = min(al(tsi:tei));
    turn_lat_max = max(al(tsi:tei));
    turn_str_min = min(sw(tsi:tei));
    turn_str_max = max(sw(tsi:tei));
else
    factor = -1;
    turn_yaw_min = -max(ar(tsi:tei));
    turn_yaw_max = -min(ar(tsi:tei));
    turn_lat_min = -max(al(tsi:tei));
    turn_lat_max = -min(al(tsi:tei));
    turn_str_min = -max(sw(tsi:tei));
    turn_str_max = -min(sw(tsi:tei));
end
turn_fwd_min = min(af(tsi:tei));
turn_fwd_max = max(af(tsi:tei));
turn_spd_min = min(sp(tsi:tei));
turn_spd_max = max(sp(tsi:tei));
turn_com_min = min(com(tsi:tei));
turn_com_max = max(com(tsi:tei));

turn_yaw_avg = mean(factor.*ar(tsi:tei));
turn_yaw_rms = norm(factor.*ar(tsi:tei))/sqrt(length(ar(tsi:tei)));
turn_yaw_int = sum(factor.*ar(tsi:tei))*0.1*length(ar(tsi:tei));
turn_yaw_var = var(factor.*ar(tsi:tei),1);
turn_yaw_med = median(factor.*ar(tsi:tei));
turn_yaw_ran = turn_yaw_max - turn_yaw_min;
turn_yaw_skw = skewness(factor.*ar(tsi:tei),0);
turn_yaw_kur = kurtosis(factor.*ar(tsi:tei),0);
turn_yaw_s_t = sum(factor.*ar(tsi:tei))/turn_time;
turn_yaw_s_d = sum(factor.*ar(tsi:tei))/turn_dis;

turn_lat_avg = mean(factor.*al(tsi:tei));
turn_lat_rms = norm(factor.*al(tsi:tei))/sqrt(length(al(tsi:tei)));
turn_lat_int = sum(factor.*al(tsi:tei))*0.1*length(al(tsi:tei));
turn_lat_var = var(factor.*al(tsi:tei),1);
turn_lat_med = median(factor.*al(tsi:tei));
turn_lat_ran = turn_lat_max - turn_lat_min;
turn_lat_skw = skewness(factor.*al(tsi:tei),0);
turn_lat_kur = kurtosis(factor.*al(tsi:tei),0);
turn_lat_s_t = sum(factor.*al(tsi:tei))/turn_time;
turn_lat_s_d = sum(factor.*al(tsi:tei))/turn_dis;

turn_fwd_avg = mean(af(tsi:tei));
turn_fwd_rms = norm(af(tsi:tei))/sqrt(length(af(tsi:tei)));
turn_fwd_int = sum(af(tsi:tei))*0.1*length(af(tsi:tei));
turn_fwd_var = var(af(tsi:tei),1);
turn_fwd_med = median(af(tsi:tei));
turn_fwd_ran = turn_fwd_max - turn_fwd_min;
turn_fwd_skw = skewness(af(tsi:tei),0);
turn_fwd_kur = kurtosis(af(tsi:tei),0);
turn_fwd_s_t = sum(af(tsi:tei))/turn_time;
turn_fwd_s_d = sum(af(tsi:tei))/turn_dis;

turn_str_avg = mean(factor.*sw(tsi:tei));
turn_str_rms = norm(factor.*sw(tsi:tei))/sqrt(length(sw(tsi:tei)));
turn_str_int = sum(factor.*sw(tsi:tei))*0.1*length(sw(tsi:tei));
turn_str_var = var(factor.*sw(tsi:tei),1);
turn_str_med = median(factor.*sw(tsi:tei));
turn_str_ran = turn_str_max - turn_str_min;
turn_str_skw = skewness(factor.*sw(tsi:tei),0);
turn_str_kur = kurtosis(factor.*sw(tsi:tei),0);
```

```
turn_str_s_t = sum(factor.*sw(tsi:tei))/turn_time;
turn_str_s_d = sum(factor.*sw(tsi:tei))/turn_dis;

turn_spd_avg = mean(sp(tsi:tei));
turn_spd_rms = norm(sp(tsi:tei))/sqrt(length(sp(tsi:tei)));
turn_spd_int = sum(sp(tsi:tei))*0.1*length(sp(tsi:tei));
turn_spd_var = var(sp(tsi:tei),1);
turn_spd_med = median(sp(tsi:tei));
turn_spd_ran = turn_spd_max - turn_spd_min;
turn_spd_skw = skewness(sp(tsi:tei),0);
turn_spd_kur = kurtosis(sp(tsi:tei),0);
turn_spd_s_t = sum(sp(tsi:tei))/turn_time;
turn_spd_s_d = sum(sp(tsi:tei))/turn_dis;

turn_com_avg = mean(com(tsi:tei));
turn_com_rms = norm(com(tsi:tei))/sqrt(length(com(tsi:tei)));
turn_com_int = sum(com(tsi:tei))*0.1*length(com(tsi:tei));
turn_com_var = var(com(tsi:tei),1);
turn_com_med = median(com(tsi:tei));
turn_com_ran = turn_com_max - turn_com_min;
turn_com_skw = skewness(com(tsi:tei),0);
turn_com_kur = kurtosis(com(tsi:tei),0);
turn_com_s_t = sum(com(tsi:tei))/turn_time;
turn_com_s_d = sum(com(tsi:tei))/turn_dis;

% recovery
if k == 7 | k == 9
    factor = 1;
    rec_yaw_min = min(ar(rsi:rei));
    rec_yaw_max = max(ar(rsi:rei));
    rec_lat_min = min(al(rsi:rei));
    rec_lat_max = max(al(rsi:rei));
    rec_str_min = min(sw(rsi:rei));
    rec_str_max = max(sw(rsi:rei));
else
    factor = -1;
    rec_yaw_min = -max(ar(rsi:rei));
    rec_yaw_max = -min(ar(rsi:rei));
    rec_lat_min = -max(al(rsi:rei));
    rec_lat_max = -min(al(rsi:rei));
    rec_str_min = -max(sw(rsi:rei));
    rec_str_max = -min(sw(rsi:rei));
end
rec_fwd_min = min(af(rsi:rei));
rec_fwd_max = max(af(rsi:rei));
rec_spd_min = min(sp(rsi:rei));
rec_spd_max = max(sp(rsi:rei));
rec_com_min = min(com(rsi:rei));
rec_com_max = max(com(rsi:rei));

rec_yaw_avg = mean(factor.*ar(rsi:rei));
rec_yaw_rms = norm(factor.*ar(rsi:rei))/sqrt(length(ar(rsi:rei)));
rec_yaw_int = sum(factor.*ar(rsi:rei))*0.1*length(ar(rsi:rei));
rec_yaw_var = var(factor.*ar(rsi:rei),1);
rec_yaw_med = median(factor.*ar(rsi:rei));
rec_yaw_ran = rec_yaw_max - rec_yaw_min;
rec_yaw_skw = skewness(factor.*ar(rsi:rei),0);
rec_yaw_kur = kurtosis(factor.*ar(rsi:rei),0);
rec_yaw_s_t = sum(factor.*ar(rsi:rei))/rec_time;
rec_yaw_s_d = sum(factor.*ar(rsi:rei))/rec_dis;

rec_lat_avg = mean(factor.*al(rsi:rei));
rec_lat_rms = norm(factor.*al(rsi:rei))/sqrt(length(al(rsi:rei)));
rec_lat_int = sum(factor.*al(rsi:rei))*0.1*length(al(rsi:rei));
rec_lat_var = var(factor.*al(rsi:rei),1);
rec_lat_med = median(factor.*al(rsi:rei));
rec_lat_ran = rec_lat_max - rec_lat_min;
rec_lat_skw = skewness(factor.*al(rsi:rei),0);
rec_lat_kur = kurtosis(factor.*al(rsi:rei),0);
rec_lat_s_t = sum(factor.*al(rsi:rei))/rec_time;
rec_lat_s_d = sum(factor.*al(rsi:rei))/rec_dis;

rec_fwd_avg = mean(af(rsi:rei));
rec_fwd_rms = norm(af(rsi:rei))/sqrt(length(af(rsi:rei)));
rec_fwd_int = sum(af(rsi:rei))*0.1*length(af(rsi:rei));
rec_fwd_var = var(af(rsi:rei),1);
rec_fwd_med = median(af(rsi:rei));
rec_fwd_ran = rec_fwd_max - rec_fwd_min;
rec_fwd_skw = skewness(af(rsi:rei),0);
rec_fwd_kur = kurtosis(af(rsi:rei),0);
rec_fwd_s_t = sum(af(rsi:rei))/rec_time;
```

```
    rec_fwd_s_d = sum(af(rsi:rei))/rec_dis;

    rec_str_avg = mean(factor.*sw(rsi:rei));
    rec_str_rms = norm(factor.*sw(rsi:rei))/sqrt(length(sw(rsi:rei)));
    rec_str_int = sum(factor.*sw(rsi:rei))*0.1*length(sw(rsi:rei));
    rec_str_var = var(factor.*sw(rsi:rei),1);
    rec_str_med = median(factor.*sw(rsi:rei));
    rec_str_ran = rec_str_max - rec_str_min;
    rec_str_skw = skewness(factor.*sw(rsi:rei),0);
    rec_str_kur = kurtosis(factor.*sw(rsi:rei),0);
    rec_str_s_t = sum(factor.*sw(rsi:rei))/rec_time;
    rec_str_s_d = sum(factor.*sw(rsi:rei))/rec_dis;

    rec_spd_avg = mean(sp(rsi:rei));
    rec_spd_rms = norm(sp(rsi:rei))/sqrt(length(sp(rsi:rei)));
    rec_spd_int = sum(sp(rsi:rei))*0.1*length(sp(rsi:rei));
    rec_spd_var = var(sp(rsi:rei),1);
    rec_spd_med = median(sp(rsi:rei));
    rec_spd_ran = rec_spd_max - rec_spd_min;
    rec_spd_skw = skewness(sp(rsi:rei),0);
    rec_spd_kur = kurtosis(sp(rsi:rei),0);
    rec_spd_s_t = sum(sp(rsi:rei))/rec_time;
    rec_spd_s_d = sum(sp(rsi:rei))/rec_dis;

    rec_com_avg = mean(com(rsi:rei));
    rec_com_rms = norm(com(rsi:rei))/sqrt(length(com(rsi:rei)));
    rec_com_int = sum(com(rsi:rei))*0.1*length(com(rsi:rei));
    rec_com_var = var(com(rsi:rei),1);
    rec_com_med = median(com(rsi:rei));
    rec_com_ran = rec_com_max - rec_com_min;
    rec_com_skw = skewness(com(rsi:rei),0);
    rec_com_kur = kurtosis(com(rsi:rei),0);
    rec_com_s_t = sum(com(rsi:rei))/rec_time;
    rec_com_s_d = sum(com(rsi:rei))/rec_dis;

    % compile statistics by phase (contains all maneuvers)
    man = str2num(man_labels(k));
    app_temp =
[sub,age_1,age_2,gen,man,imp,stop,app_time,app_dis,app_yaw_min,app_yaw_max,app_yaw_avg,ap
p_yaw_rms,app_yaw_int,app_yaw_var,app_yaw_med,app_yaw_ran,app_yaw_skw,app_yaw_kur,app_yaw
_s_t,app_yaw_s_d,app_lat_min,app_lat_max,app_lat_avg,app_lat_rms,app_lat_int,app_lat_var,
app_lat_med,app_lat_ran,app_lat_skw,app_lat_kur,app_lat_s_t,app_lat_s_d,app_fwd_min,app_f
wd_max,app_fwd_avg,app_fwd_rms,app_fwd_int,app_fwd_var,app_fwd_med,app_fwd_ran,app_fwd_sk
w,app_fwd_kur,app_fwd_s_t,app_fwd_s_d,app_str_min,app_str_max,app_str_avg,app_str_rms,app
_str_int,app_str_var,app_str_med,app_str_ran,app_str_skw,app_str_kur,app_str_s_t,app_str_
s_d,app_spd_min,app_spd_max,app_spd_avg,app_spd_rms,app_spd_int,app_spd_var,app_spd_med,a
pp_spd_ran,app_spd_skw,app_spd_kur,app_spd_s_t,app_spd_s_d,app_com_min,app_com_max,app_co
m_avg,app_com_rms,app_com_int,app_com_var,app_com_med,app_com_ran,app_com_skw,app_com_kur
,app_com_s_t,app_com_s_d];
    if length(app) > 1
        app = [app;app_temp];   % approach stats
    else
        app = [app_temp];
    end
    turn_temp =
[sub,age_1,age_2,gen,man,imp,stop,turn_time,turn_dis,turn_yaw_min,turn_yaw_max,turn_yaw_a
vg,turn_yaw_rms,turn_yaw_int,turn_yaw_var,turn_yaw_med,turn_yaw_ran,turn_yaw_skw,turn_yaw
_kur,turn_yaw_s_t,turn_yaw_s_d,turn_lat_min,turn_lat_max,turn_lat_avg,turn_lat_rms,turn_l
at_int,turn_lat_var,turn_lat_med,turn_lat_ran,turn_lat_skw,turn_lat_kur,turn_lat_s_t,turn
_lat_s_d,turn_fwd_min,turn_fwd_max,turn_fwd_avg,turn_fwd_rms,turn_fwd_int,turn_fwd_var,tu
rn_fwd_med,turn_fwd_ran,turn_fwd_skw,turn_fwd_kur,turn_fwd_s_t,turn_fwd_s_d,turn_str_min,
turn_str_max,turn_str_avg,turn_str_rms,turn_str_int,turn_str_var,turn_str_med,turn_str_ra
n,turn_str_skw,turn_str_kur,turn_str_s_t,turn_str_s_d,turn_spd_min,turn_spd_max,turn_spd_
avg,turn_spd_rms,turn_spd_int,turn_spd_var,turn_spd_med,turn_spd_ran,turn_spd_skw,turn_sp
d_kur,turn_spd_s_t,turn_spd_s_d,turn_com_min,turn_com_max,turn_com_avg,turn_com_rms,turn_
com_int,turn_com_var,turn_com_med,turn_com_ran,turn_com_skw,turn_com_kur,turn_com_s_t,tur
n_com_s_d];
    if length(turn) > 1
        turn = [turn;turn_temp];% turn stats
    else
        turn = [turn_temp];
    end
    rec_temp =
[sub,age_1,age_2,gen,man,imp,stop,rec_time,rec_dis,rec_yaw_min,rec_yaw_max,rec_yaw_avg,re
c_yaw_rms,rec_yaw_int,rec_yaw_var,rec_yaw_med,rec_yaw_ran,rec_yaw_skw,rec_yaw_kur,rec_yaw
_s_t,rec_yaw_s_d,rec_lat_min,rec_lat_max,rec_lat_avg,rec_lat_rms,rec_lat_int,rec_lat_var,
rec_lat_med,rec_lat_ran,rec_lat_skw,rec_lat_kur,rec_lat_s_t,rec_lat_s_d,rec_fwd_min,rec_f
wd_max,rec_fwd_avg,rec_fwd_rms,rec_fwd_int,rec_fwd_var,rec_fwd_med,rec_fwd_ran,rec_fwd_sk
w,rec_fwd_kur,rec_fwd_s_t,rec_fwd_s_d,rec_str_min,rec_str_max,rec_str_avg,rec_str_rms,rec
_str_int,rec_str_var,rec_str_med,rec_str_ran,rec_str_skw,rec_str_kur,rec_str_s_t,rec_str_
s_d,rec_spd_min,rec_spd_max,rec_spd_avg,rec_spd_rms,rec_spd_int,rec_spd_var,rec_spd_med,r
```

```
ec_spd_ran,rec_spd_skw,rec_spd_kur,rec_spd_s_t,rec_spd_s_d,rec_com_min,rec_com_max,rec_co
m_avg,rec_com_rms,rec_com_int,rec_com_var,rec_com_med,rec_com_ran,rec_com_skw,rec_com_kur
,rec_com_s_t,rec_com_s_d];
    if length(rec) > 1
        rec = [rec;rec_temp];   % recovery stats
    else
        rec = [rec_temp];
    end
end

% write statisitics files for each phase (contains all maneuvers)
subject = ['subject_'];
appstr = ['approach_'];
AppName = ['SIM_',appstr,subject,num2str(sub),'.txt'];
dlmwrite(AppName,turn)

turnstr = ['turn_'];
TurnName = ['SIM_',turnstr,subject,num2str(sub),'.txt'];
dlmwrite(TurnName,turn)

recstr = ['recovery_'];
RecName = ['SIM_',recstr,subject,num2str(sub),'.txt'];
dlmwrite(RecName,rec)

status = 'Finished'
```

sim_sort.m

```
% **********************************************************************
%
%           Simulator Sorting Program (organizes stats by maneuver)
%
%   University of Florida
%   College of Publich Health and Health Professions
%   Occupational Therapy Department
%   Created by: Ethan J. Davis (Mechanical & Aerospace Engineering)
%   Date: 09-15-05
%
%   ------------------------ INSTRUCTIONS ----------------------------
%   Place this M-File in the same folder with all of the Statistic Files
%   (i.e. "SIM_turn_subject_23.txt" )
%   Open Matlab, Open this M-File
%   Run this M-File (F5)
%  --------------------------------------------------------------------
%   *** NOTE: In order to process a given subject, the subject's number
%   must be added to the "subs" vector on line 25 (see below) ***
%  --------------------------------------------------------------------
%
% **********************************************************************

clear all
status = 'Please Wait...'
subs = [];  % <-- ADD SUBJECT NUMBERS HERE

sim_str = ['SIM_'];
app_str = ['approach_'];
turn_str = ['turn_'];
rec_str = ['recovery_'];
sub_str = ['subject_'];
man_1_app = 0;
man_2_app = 0;
man_3_app = 0;
man_4_app = 0;
man_5_app = 0;
man_1_turn = 0;
man_2_turn = 0;
man_3_turn = 0;
man_4_turn = 0;
man_5_turn = 0;
man_1_rec = 0;
man_2_rec = 0;
man_3_rec = 0;
man_4_rec = 0;
man_5_rec = 0;

% process 1 subject at a time
for cnt = 1:length(subs)
    app_file_name = [sim_str,app_str,sub_str,num2str(subs(cnt)),'.txt'];
    app_data = dlmread(app_file_name);  % read approach file
```

```
    turn_file_name = [sim_str,turn_str,sub_str,num2str(subs(cnt)),'.txt'];
    turn_data = dlmread(turn_file_name);% read turn file
    rec_file_name = [sim_str,rec_str,sub_str,num2str(subs(cnt)),'.txt'];
    rec_data = dlmread(rec_file_name);  % read recovery file
    % sort by maneuver
    man_1A_temp_app = app_data(1,:);
    man_1B_temp_app = app_data(2,:);
    man_1C_temp_app = app_data(3,:);
    man_2A_temp_app = app_data(4,:);
    man_2B_temp_app = app_data(5,:);
    man_3A_temp_app = app_data(6,:);
    man_3B_temp_app = app_data(7,:);
    man_4A_temp_app = app_data(8,:);
    man_4B_temp_app = app_data(9,:);
    man_5A_temp_app = app_data(10,:);
    man_5B_temp_app = app_data(11,:);

    man_1A_temp_turn = turn_data(1,:);
    man_1B_temp_turn = turn_data(2,:);
    man_1C_temp_turn = turn_data(3,:);
    man_2A_temp_turn = turn_data(4,:);
    man_2B_temp_turn = turn_data(5,:);
    man_3A_temp_turn = turn_data(6,:);
    man_3B_temp_turn = turn_data(7,:);
    man_4A_temp_turn = turn_data(8,:);
    man_4B_temp_turn = turn_data(9,:);
    man_5A_temp_turn = turn_data(10,:);
    man_5B_temp_turn = turn_data(11,:);

    man_1A_temp_rec = rec_data(1,:);
    man_1B_temp_rec = rec_data(2,:);
    man_1C_temp_rec = rec_data(3,:);
    man_2A_temp_rec = rec_data(4,:);
    man_2B_temp_rec = rec_data(5,:);
    man_3A_temp_rec = rec_data(6,:);
    man_3B_temp_rec = rec_data(7,:);
    man_4A_temp_rec = rec_data(8,:);
    man_4B_temp_rec = rec_data(9,:);
    man_5A_temp_rec = rec_data(10,:);
    man_5B_temp_rec = rec_data(11,:);
    % create statistic arrays
    if length(man_1_turn) > 1
        man_1_app = [man_1_app;man_1A_temp_app,man_1B_temp_app,man_1C_temp_app];
        man_2_app = [man_2_app;man_2A_temp_app,man_2B_temp_app];
        man_3_app = [man_3_app;man_3A_temp_app,man_3B_temp_app];
        man_4_app = [man_4_app;man_4A_temp_app,man_4B_temp_app];
        man_5_app = [man_5_app;man_5A_temp_turn,man_5B_temp_turn];
        man_1_turn = [man_1_turn;man_1A_temp_turn,man_1B_temp_turn,man_1C_temp_turn];
        man_2_turn = [man_2_turn;man_2A_temp_turn,man_2B_temp_turn];
        man_3_turn = [man_3_turn;man_3A_temp_turn,man_3B_temp_turn];
        man_4_turn = [man_4_turn;man_4A_temp_turn,man_4B_temp_turn];
        man_5_turn = [man_5_turn;man_5A_temp_turn,man_5B_temp_turn];
        man_1_rec = [man_1_rec;man_1A_temp_rec,man_1B_temp_rec,man_1C_temp_rec];
        man_2_rec = [man_2_rec;man_2A_temp_rec,man_2B_temp_rec];
        man_3_rec = [man_3_rec;man_3A_temp_rec,man_3B_temp_rec];
        man_4_rec = [man_4_rec;man_4A_temp_rec,man_4B_temp_rec];
        man_5_rec = [man_5_rec;man_5A_temp_rec,man_5B_temp_rec];
    else
        man_1_app = [man_1A_temp_app,man_1B_temp_app,man_1C_temp_app];
        man_2_app = [man_2A_temp_app,man_2B_temp_app];
        man_3_app = [man_3A_temp_app,man_3B_temp_app];
        man_4_app = [man_4A_temp_app,man_4B_temp_app];
        man_5_app = [man_5A_temp_app,man_5B_temp_app];
        man_1_turn = [man_1A_temp_turn,man_1B_temp_turn,man_1C_temp_turn];
        man_2_turn = [man_2A_temp_turn,man_2B_temp_turn];
        man_3_turn = [man_3A_temp_turn,man_3B_temp_turn];
        man_4_turn = [man_4A_temp_turn,man_4B_temp_turn];
        man_5_turn = [man_5A_temp_turn,man_5B_temp_turn];
        man_1_rec = [man_1A_temp_rec,man_1B_temp_rec,man_1C_temp_rec];
        man_2_rec = [man_2A_temp_rec,man_2B_temp_rec];
        man_3_rec = [man_3A_temp_rec,man_3B_temp_rec];
        man_4_rec = [man_4A_temp_rec,man_4B_temp_rec];
        man_5_rec = [man_5A_temp_rec,man_5B_temp_rec];
    end
end

man_str = ['Maneuver_'];
sta_str = ['_Stats'];
app_str = ['_Approach'];
turn_str = ['_Turn'];
rec_str = ['_Recovery'];
```

```
% create output files
man_1_fn_app = [sim_str,man_str,num2str(1),app_str,sta_str,'.txt'];
man_2_fn_app = [sim_str,man_str,num2str(2),app_str,sta_str,'.txt'];
man_3_fn_app = [sim_str,man_str,num2str(3),app_str,sta_str,'.txt'];
man_4_fn_app = [sim_str,man_str,num2str(4),app_str,sta_str,'.txt'];
man_5_fn_app = [sim_str,man_str,num2str(5),app_str,sta_str,'.txt'];
dlmwrite(man_1_fn_app,man_1_app)
dlmwrite(man_2_fn_app,man_2_app)
dlmwrite(man_3_fn_app,man_3_app)
dlmwrite(man_4_fn_app,man_4_app)
dlmwrite(man_5_fn_app,man_5_app)

man_1_fn_turn = [sim_str,man_str,num2str(1),turn_str,sta_str,'.txt'];
man_2_fn_turn = [sim_str,man_str,num2str(2),turn_str,sta_str,'.txt'];
man_3_fn_turn = [sim_str,man_str,num2str(3),turn_str,sta_str,'.txt'];
man_4_fn_turn = [sim_str,man_str,num2str(4),turn_str,sta_str,'.txt'];
man_5_fn_turn = [sim_str,man_str,num2str(5),turn_str,sta_str,'.txt'];
dlmwrite(man_1_fn_turn,man_1_turn)
dlmwrite(man_2_fn_turn,man_2_turn)
dlmwrite(man_3_fn_turn,man_3_turn)
dlmwrite(man_4_fn_turn,man_4_turn)
dlmwrite(man_5_fn_turn,man_5_turn)

man_1_fn_rec = [sim_str,man_str,num2str(1),rec_str,sta_str,'.txt'];
man_2_fn_rec = [sim_str,man_str,num2str(2),rec_str,sta_str,'.txt'];
man_3_fn_rec = [sim_str,man_str,num2str(3),rec_str,sta_str,'.txt'];
man_4_fn_rec = [sim_str,man_str,num2str(4),rec_str,sta_str,'.txt'];
man_5_fn_rec = [sim_str,man_str,num2str(5),rec_str,sta_str,'.txt'];
dlmwrite(man_1_fn_rec,man_1_rec)
dlmwrite(man_2_fn_rec,man_2_rec)
dlmwrite(man_3_fn_rec,man_3_rec)
dlmwrite(man_4_fn_rec,man_4_rec)
dlmwrite(man_5_fn_rec,man_5_rec)

status = 'Finished'
```

APPENDIX E
EXAMPLE OF SPSS OUTPUT FILE FOR MAXIMUM YAW DURING THE TURN PHASE OF MANEUVER 1

Within-Subjects Factors

Measure: MEASURE_1

improv	Dependent Variable
1	yawmaxa
2	yawmaxb

Between-Subjects Factors

		N
agegp2	0	39
	1	32

Descriptive Statistics

	agegp2	Mean	Std. Deviation	N
yawmaxa	0	.3582633	.07513164	39
	1	.3436297	.03921199	32
	Total	.3516679	.06163604	71
yawmaxb	0	.4356249	.06452554	39
	1	.4364078	.07731970	32
	Total	.4359777	.07005650	71

Multivariate Tests[c]

Effect		Value	F	Hypothesis df	Error df	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[a]
improv	Pillai's Trace	.566	90.058[b]	1.000	69.000	.000	.566	90.058	1.000
	Wilks' Lambda	.434	90.058[b]	1.000	69.000	.000	.566	90.058	1.000
	Hotelling's Trace	1.305	90.058[b]	1.000	69.000	.000	.566	90.058	1.000
	Roy's Largest Root	1.305	90.058[b]	1.000	69.000	.000	.566	90.058	1.000
improv * agegp2	Pillai's Trace	.011	.739[b]	1.000	69.000	.393	.011	.739	.136
	Wilks' Lambda	.989	.739[b]	1.000	69.000	.393	.011	.739	.136
	Hotelling's Trace	.011	.739[b]	1.000	69.000	.393	.011	.739	.136
	Roy's Largest Root	.011	.739[b]	1.000	69.000	.393	.011	.739	.136

a. Computed using alpha = .05

b. Exact statistic

c.
Design: Intercept+agegp2
Within Subjects Design: improv

Mauchly's Test of Sphericity[b]

Measure: MEASURE_1

Within Subjects Effect	Mauchly's W	Approx. Chi-Square	df	Sig.	Epsilon[a] Greenhouse-Geisser	Epsilon[a] Huynh-Feldt	Epsilon[a] Lower-bound
improv	1.000	.000	0	.	1.000	1.000	1.000

Tests the null hypothesis that the error covariance matrix of the orthonormalized transformed dependent variables is proportional to an identity matrix.

a. May be used to adjust the degrees of freedom for the averaged tests of significance. Corrected tests are displayed in the Tests of Within-Subjects Effects table.

b.

Design: Intercept+agegp2
Within Subjects Design: improv

Tests of Within-Subjects Effects

Measure: MEASURE_1

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[a]
improv	Sphericity Assumed	.254	1	.254	90.058	.000	.566	90.058	1.000
	Greenhouse-Geisser	.254	1.000	.254	90.058	.000	.566	90.058	1.000
	Huynh-Feldt	.254	1.000	.254	90.058	.000	.566	90.058	1.000
	Lower-bound	.254	1.000	.254	90.058	.000	.566	90.058	1.000
improv * agegp2	Sphericity Assumed	.002	1	.002	.739	.393	.011	.739	.136
	Greenhouse-Geisser	.002	1.000	.002	.739	.393	.011	.739	.136
	Huynh-Feldt	.002	1.000	.002	.739	.393	.011	.739	.136
	Lower-bound	.002	1.000	.002	.739	.393	.011	.739	.136
Error(improv)	Sphericity Assumed	.195	69	.003					
	Greenhouse-Geisser	.195	69.000	.003					
	Huynh-Feldt	.195	69.000	.003					
	Lower-bound	.195	69.000	.003					

a. Computed using alpha = .05

Tests of Within-Subjects Contrasts

Measure: MEASURE_1

Source	improv	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[a]
improv	Linear	.254	1	.254	90.058	.000	.566	90.058	1.000
improv * agegp2	Linear	.002	1	.002	.739	.393	.011	.739	.136
Error(improv)	Linear	.195	69	.003					

a. Computed using alpha = .05

Tests of Between-Subjects Effects

Measure: MEASURE_1
Transformed Variable: Average

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[a]
Intercept	21.772	1	21.772	3657.032	.000	.981	3657.032	1.000
agegp2	.002	1	.002	.283	.596	.004	.283	.082
Error	.411	69	.006					

a. Computed using alpha = .05

LIST OF REFERENCES

Boyce Thomas E., Geller Scott E. "An instrumented vehicle assessment of problem behavior and driving style: Do younger males really take more risks?" Accident Analysis and Prevention 34 (2002): 51-64.

Dingus, Thomas A., Hulse Melissa C. "Effects of age, system experience, and navigation technique on driving with an Advanced Traveler Information System" Human Factors: The Journal of the Human Factors and Ergonomics Society 39 (1997): 177-200.

DOT HS 809 328. U.S. Department of Transportation, National Highway Traffic Safety Administration. < http://www-nrd.nhtsa.dot.gov/pdf/nrd-30/ncsa/tsf2000/2000oldpop.pdf> (accessed June, 2004)

Knodler Michael A., Noyce David A., Kacir Kent C., Brehmer Christopher L. "Evaluation of traffic signal displays for protected permissive left-turn control using driving simulator technology." Journal of Transportation Engineering 131.4 (2005): 270-278.

LabVIEW. Version 6.1. 2001. National Instruments. Austin, Texas.

MATLAB. Version 6.1.0.450 Release 12.1. 1984-2001. The MathWorks, Inc. Natick, Massachusetts

McGehee Daniel V., Lee John D., Rizzo Matthew, Bateman Kirk. "Examination of older driver steering adaptation on a high performance driving simulator." International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design (2001) <http://ppc.uiowa.edu/driving-assessment/2001/Summaries/Downloads/download.html> (accessed February, 2006)

Rizzo M., Jermeland J., Severson J. "Instrumented vehicles and driving simulators." Gerontechnology 1.4 (2002): 291-296.

Road & Track 53.6 (2002): 114-115.

SPSS for Windows. Version 13.0. 1989-2004. SPSS, Inc. Chicago, Illinois.

Uc E. Y., Rizzo M., Anderson S. W., Shi Q., Dawson J. D. "Driver route-following and safety errors in early Alzheimer's disease." Neurology 63 (2004): 832-837.

Wierwille W. W., Gutmann F. "Comparison of primary and secondary task measures as a function of simulated vehicle dynamics and driving conditions." Human Factors: The Journal of the Human Factors and Ergonomics Society 20.2 (1978): 233-344.

Wood J.M., Worringham C. "Quantitative assessment of driving performance in Parkinson's disease" Journal of Neurology, Neurosurgery, and Psychiatry 76 (2005): 176-180.

BIOGRAPHICAL SKETCH

Ethan Davis graduated from Pine View High School for The Gifted located in Osprey, Florida, in June 1999. He attended Florida Southern College in Lakeland, Florida, where he played 2 seasons of NCAA Div. II soccer. Ethan transferred to the University of Florida in Gainesville, Florida where he graduated magna cum laude with a bachelor's degree in mechanical engineering in May, 2004. He will receive a Master of Engineering degree in May, 2006, from the University of Florida.