RITA Intelligent Transportation Systems
Joint Program Office

Cooperative Intersection Collision Avoidance System Limited to Stop Sign and Traffic Signal Violations (CICAS-V)

Task 8 Final Report

Prototype Build and Testing

(Appendix F)

June 24, 2008

U.S. Department of Transportation
Research and Innovative Technology Administration

Crash Avoidance Metrics Partnership (CAMP) Produced
In conjunction with Virginia Tech Transportation Institute for
ITS Joint Program Office
Research and Innovative Technology Administration
U.S. Department of Transportation

CAMP Members:

Mercedes-Benz
General Motors (GM)
Toyota
Honda
Ford

Notice

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Cooperative Intersection Collision Avoidance System Limited to Stop Sign and Traffic Signal Violations (CICAS-V), System Design Specification – Infrastructure		5. Report Date June 24, 2008 6. Performing Organization Code:
7. Author(s) Michael A. Maile, David Masselink, Priyantha Mudalige, Chuck Pall		8. Performing Organization Report No.
9. Performing Organization Name and Address Crash Avoidance Metrics Partnership on behalf of the Vehicle Safety Communications 2 Consortium 39255 Country Club Drive Suite B-40 Farmington Hills, MI 48331		10. Work Unit No. (TRAIS) 11. Contract or Grant No. DTFH61-01-X-00014
12. Sponsoring Agency Name and Address United States Department of Transportation, Federal Highway Administration 1200 New Jersey Ave, S.E. Washington, DC 20590		13. Type of Report and Period Covered Final Task Report May 1, 2006 – May 31, 2008 14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This report describes the work performed in Task 8 of the CICAS-V project. Work in this task focused on the development and testing of the initial prototypes of the technology building blocks that make up the CICAS-V application. The task built upon the work that was conducted in the definition of the Concept of Operations (ConOps) and the initial requirements that were developed earlier in the project. An intersection system and a vehicle system were developed and tested along with the message sets needed to transmit information from the intersection to the vehicle. The message sets included a signal phase and timing message, a message to transmit an intersection map and local positioning correction message. In the course of the task, the intersection system was installed and tested in signalized and stop sign controlled intersections in California and in Michigan. The vehicle system was installed in five OEM vehicles that were used for developing the application. Results indicated that the CICAS-V ConOps was technically feasible and could be implemented such that a CICAS-V application would function reliably.

The CICAS V project is a four-year project to develop a cooperative intersection collision avoidance system to assist drivers in avoiding crashes in the intersection by warning the driver of an impending violation of a traffic signal or a stop sign. The Vehicle Safety Communications 2 Consortium (VSC2) is executing the project. Members of VSC2 are Ford Motor Company, General Motors Corporation, Honda R & D Americas, Inc., Mercedes-Benz Research and Development North America, Inc., and Toyota Motor Engineering & Manufacturing North America, Inc.

17. Key Words DSRC, intersection crashes, traffic safety, traffic signal violation, stop sign violation, signal phase and timing, intersection map, positioning corrections, map matching, driver warning algorithm, message sets, intersection selection, local safety system, system architecture	18. Distribution Statement No restrictions. This document is available through the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 73	22. Price

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorized**

Photos Credits

Photos and Illustration's courtesy of CAMP

Executive Summary

Task 8 developed and tested the initial prototypes of the technology building blocks that make up the CICAS-V application. The task was comprised of the purchasing and testing of the necessary equipment installed in the vehicles (as well as the vehicles themselves) and in the intersection. The task built upon the work that was conducted in the definition of the Concept of Operations (ConOps) and the initial requirements that were developed in the beginning of the project. The task developed the intersection system and the vehicle system as follows:

- The intersection system
- Transmitting signal phase and timing from the controller to the vehicle;
- Transmitting a small, highly accurate map (Geometric Intersection Description - GID) from the intersection to the vehicle;
- Developing and transmitting local positioning correction information from the intersection to the vehicle;
- The vehicle system
- Receiving the information from the intersection;
- Matching the vehicle to the correct approach and lane, using a map matching algorithm, the GID and the local positioning correction information;
- Warning the driver through a warning algorithm.

In the course of the task, the intersection system was installed and tested in signalized and stop sign controlled intersections in California and in Michigan as well as the Original Equipment Manufacturer (OEM) vehicles that are used for developing the application. It was found that the ConOps was technically feasible and could be implemented such that a CICAS-V application would function reliably. The CICAS-V project also gained experience in setting up and testing intersections as well as in working with local and state DOTs to make installations of the CICAS-V equipment in intersections a reality.

The project used laboratory-type equipment such as laptop PCs as computing platforms and the DSRC radios that were specified in the VSC1 project for communication. This preliminary equipment will be replaced in Task 10 by embedded computing platforms and advanced DSRC radios that embody the recent advances in the IEEE 802.11p and 1609.x standards.

The development activities and the test results provided the necessary input for the specifications of the software modules and interfaces that make up the pilot FOT prototype that will be used to test the system with naïve drivers in a naturalistic driving study.

The CICAS-V project found that transmission ranges of 300 m could easily be achieved with the available radios with a packet error rate (PER) of smaller than 70% at 300 m. PER is defined as (1 – (Number of received packets / Number of sent packets))x100.

Positioning improvement were achieved using local positioning corrections with accuracy of 0.5 m.

Intersection maps (GID) were developed with absolute accuracy of 10 cm and a size of less than 500 bytes.

Map matching algorithms were developed that enabled correct lane matching in real time using the GIDs and local positioning correction.

Initial warning algorithm logic was developed to provide the basis for further developments in other tasks.

The main outputs of this task were the specifications for the development of the final CICAS-V prototype system in Task 10 and the final Phase I CICAS-V prototype at the end of the project.

Table of Contents

List of Figures

List of Tables

LIST OF ACRONYMS

AGID	Area Geometric Intersection Description
AMA	Approach Matching Algorithm
ARP	Antenna Reference Point
Caltrans	California Department of Transportation
CAMP	Crash Avoidance Metrics Partnership
CAN	Controller Area Network
CICAS	Cooperative Intersection Collision Avoidance System
CICAS-V	Cooperative Intersection Collision Avoidance System for Violations
ConOps	Concept of Operations
CRC	Cyclic Redundancy Code
DGNSS	Differential Global Navigation Satellite System
DGPS	Differential Global Positioning System
DOT	Department of Transportation
DSRC	Dedicated Short Range Communication
DTC	Distance to Centerline
DVI	Driver-Vehicle Interface
ECEF	Earth-Centered, Earth-Fixed
EDMAP	Enhanced Digital Map
ESS	Electronic Stability System
FHWA	Federal Highway Administration
FOT	Field Operational Test
GHz	Gigahertz
GID	Geometric Intersection Description
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPSC	Global Positioning System Correction
GST	GPS Pseudorange Measurement Noise Statistics
HDOP	Horizontal Dilution of Precision
ID	Identification or Identifier
IEEE	Institute of Electrical and Electronics Engineers
IIM	Intersection Identification Module
IP	Internet Protocol
IRP	Intersection Reference Point
ITS	Intelligent Transportation Systems
IVI	Intelligent Vehicle Initiative
LMA	Lane Matching Algorithm
MCNU	Multiband Configurable Network Unit

MHz	Megahertz
NEMA	National Electrical Manufacturers Association
NHTSA	National Highway Traffic Safety Administration
NMEA	National Marine Electronics Association
NTCIP	National Transportation Communications for ITS Protocol
OBE	On-Board Equipment
OEM	Original Equipment Manufacturer
PATH	Partners for Advanced Transit and Highways
PBA	Panic Brake Assist
PED	Pedestrian
PDOP	Position Dilution of Precision
PHY	Physical
PSC	Provider Service Content
PSID	Provider Service Identifier
RCMD	Regional Coastal Monitoring Data
RF	Radio Frequency
RT	Reaction Time
RTCM	Radio Technical Committee for Maritime Applications
RTK	Real-Time Kinematic
RSE	Roadside Equipment
SAE	Society of Automotive Engineers
SPaT	Signal Phase and Timing
TMT	Technical Management Team
TOM	Transportation Object Message
TOMC	Transportation Object Message Compiler
TSVWG	Traffic Signal Violation Warning Given
UDP	User Datagram Protocol
UDRE	User Differential Range Error
USDOT	United States Department of Transportation
UTC	Coordinated Universal Time
UTM	Universal Transverse Mercator
VII	Vehicle-Infrastructure Integration
VSC	Vehicle Safety Communications
VSC2	Vehicle Safety Communications 2
VTTI	Virginia Tech Transportation Institute
WAAS	Wide Area Augmentation System
WAVE	Wireless Access in Vehicular Environment
WGS 84	World Geodetic System 1984
WiFi	Wireless Fidelity
WRM	WAVE Radio Module
WSA	WAVE Service Announcement

WSM	WAVE Short Message
WSU	Wireless Safety Unit
XML	Extensible Markup Language

1 Introduction

This document presents the Task 8 Final Report for the Cooperative Intersection Collision Avoidance System Limited to Stop Sign and Traffic Signal Violations (CICAS-V) project. The period covered by the report is from May 1, 2006 through November 30, 2007.

1.1 Project Description

The CICAS-V project is a four-year project to develop a cooperative intersection collision avoidance system that prevents crashes in the intersection by warning the driver of an impending violation of a traffic signal or a stop sign. Cooperative means that the system involves both infrastructure and in-vehicle elements working together. The Vehicle Safety Communications 2 Consortium (VSC2) is executing the project under Federal Highway Administration (FHWA) Cooperative Agreement No. DTFH61-01-X-00014, Work Order W-05-001. Members of VSC2 include Ford Motor Company, General Motors Corporation, Honda R & D Americas, Inc., Mercedes-Benz Research & Development North America, Inc. and Toyota Motor Engineering & Manufacturing North America, Inc. Funding for this project is provided from the Joint Program Office of the United States Department of Transportation (U.S. DOT). The project is also supported by Virginia Tech University (Virginia Tech), who plays a major role in the research to define and evaluate the CICAS-V warning system. The work at Virginia Tech is being conducted through its research group at the Virginia Tech Transportation Institute (VTTI).

The project was initiated in May 2006 and the technical work was completed at the end of August 2008. At this time the USDOT decided not to conduct the planned FOT.

1.2 Purpose for Implementing the System

The purpose of implementing CICAS-V is to reduce crashes due to violation of traffic control devices (both traffic signals and stop signs).

When deployed, this system is intended to:

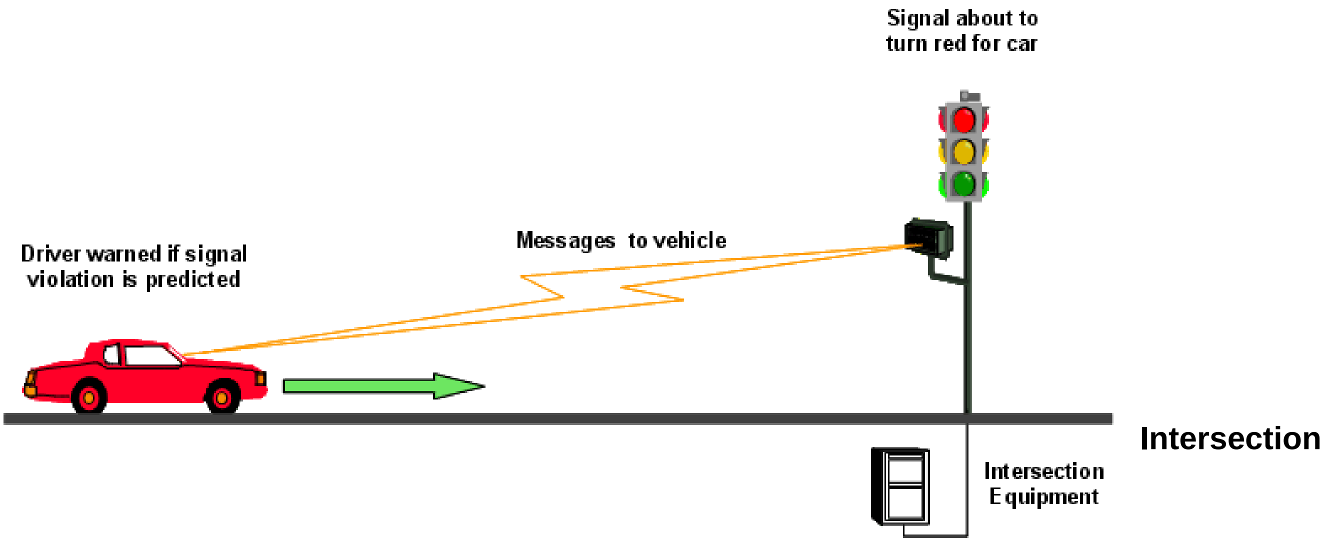
- Reduce fatalities at controlled intersections
- Reduce the number of injuries at controlled intersections
- Reduce the severity of injuries at controlled intersections
- Reduce property damage associated with collisions at controlled intersections
- Create an enabling environment that additional technologies can leverage to further extend safety benefits

Each year about 5,000 fatal crashes occur in intersections with traffic signals or stop signs (National Highway Traffic Safety Administration, 2005[0]). About 44% occur at traffic signals and 56% at stop signs. About 400,000 injury crashes occur at those intersections each year. About 600,000 property damage crashes also occur at those intersections annually.

An initial analysis of relevant National Highway Traffic Safety Administration (NHTSA) crash databases shows that violation crashes have a variety of causal factors. The CICAS-V system is intended to address the causal factors that include driver distraction (a frequent factor cited by Campbell, Smith and Najm, 2004, p. 65[0]), obstructed/limited visibility due to weather or intersection geometry or other vehicles, the presence of a new control device not previously known to the driver, and driver judgment errors. Driver warnings, such as those planned for CICAS-V, may prevent many violation-related crashes by alerting the distracted driver, thus increasing the likelihood that the driver will stop the vehicle and avoid the crash.

1.3 Goals and Objectives

CICAS-V is intended to provide a cooperative vehicle and infrastructure system that assists drivers in avoiding crashes at intersections by warning the vehicle driver that a violation, at an intersection controlled by a stop sign or by traffic signal, is predicted to occur. The basic concept of CICAS-V is illustrated at a high level in Figure 1 for a signalized intersection. In the figure, a CICAS-V equipped vehicle approaching a CICAS-V equipped intersection receives messages about the intersection geometry and status of the traffic signal. The driver is issued a warning if the equipment in the vehicle determines that, given current operating conditions, the driver is predicted to violate the signal in a manner which is likely to result in the vehicle entering the intersection. While the system may not prevent all crashes through such warnings, it is expected that, with an effective warning, the number of traffic control device violations will decrease, and result in a decrease in the number and severity of crashes at controlled intersections.



Specific goals of CICAS-V include the establishment of:

- A warning system that will be effective at reducing the number of fatal crashes, the severity of injuries and property damage at CICAS-V intersections
- A warning system that is acceptable to users
- A vehicle-infrastructure cooperative system that helps vehicle drivers avoid crashes due to violations of a traffic signal or stop sign
- A system that is deployable throughout the United States

2 Task Results

2.1 Task 8 Description

The objective of Task 8 was to develop the technology basis for the Phase I prototype and to test the individual elements that will comprise the overall system before the complete prototype is developed. The main technical work in Task 8 consisted of the development of the intersection map (called a geometric intersection description (GID)), positioning correction, a map matching algorithm, message sets and transmission formats, interface between the RSE and the traffic signal controller at the intersection and the initial driver warning algorithm. Also included in the work conducted in this task was the selection of intersections and vehicles to support the research and development efforts in Phase I. This included the mapping and classification of the intersections selected for use in the project.

The work led to the initial specifications for the CICAS-V system that were subsequently used to develop the systems software in Task 10 of the project. The work to actually develop the FOT-ready prototype system was not a part of the work conducted in Task 8 and will be reported separately at the end of Phase I.

2.1.1.1 Task 8 Scope of Work

Work in Task 8 was organized into eight subtasks, which focused on the development of core elements of the CICAS-V system or the preparations for testing the system elements. These are described below.

2.1.1.2 Task 8.1 Vehicle Selection and Purchase

This task involved identification and procurement of one engineering development vehicle by each OEM. The vehicles that were purchased will become the reference design for each OEM that will be given to VTTI in Phase II in preparation for the FOT.

These vehicles contained a computing platform, DSRC radio (using the existing VSC/WAVE radio), DGPS system, and other hardware software components that were integrated with the vehicle bus to obtain vehicle-sensor data. The computing platform performed the data processing, warning algorithms, DVI, and other software functions required by the application. The equipped vehicles were used for the evaluation of the elements of the concepts of operation (e.g., map broadcast, message set, and timing) and the functional evaluation of the in-vehicle and the intersection application. Development of these system elements is described below. As much as possible, the OEMs used the same components for all the vehicles. The task also included the functional testing of the vehicle build at the subsystem and system level.

2.1.1.3 Task 8.2 Intersection Selection for CICAS-V Development

Working cooperatively with the USDOT, the CICAS-V Team selected three intersections each in California, Michigan and Virginia. The intersections selected have varying road geometry complexity and varying degrees of GPS availability. At least one intersection in each state has the following intersection characteristics: one signalized intersection with

WhichRoad accuracy requirements, one stop-sign intersection with WhichRoad requirements and one signalized intersection with WhichLane accuracy requirements. WhichRoad and WhichLane are metrics that refer to the accuracy of the map matched position of the vehicle, developed in the CAMP Enhanced Digital Mapping (EDMAP) Project (Enhanced Digital Mapping Project Final Report, 2004). WhichRoad describes the accuracy necessary to match the vehicle to the road that it is currently driving on. WhichLane accuracy means that the vehicle is capable to match itself to the lane it is currently driving in. The intersections with WhichLane requirements have adequate GPS coverage so that reliable lane matching can be performed. The CICAS-V Team then engaged the local DOT agency with responsibility for each intersection and arranged to have the needed CICAS-V infrastructure components (i.e., RSEs, antennas, GPS equipment, etc.) installed at each intersection.

2.1.1.4 Task 8.3: Development of Traffic Signal Load Switch Sensor

Work in this task developed a traffic signal load switch sensor connected to the signal controller cabinet to detect the current signal indication, observe the yellow time and provide the countdown from yellow onset to red in sub-second increments. This device, referred to as a signal “sniffer” bypasses the traffic signal controller and provides a non-intrusive method for ascertaining the information needed for the CICAS-V system. Such a method was needed to obtain the signal phase and timing information from older traffic signal controllers, which are either not able to output this information or not able to provide it with suitable frequency.

2.1.1.5 Phase I, Task 8.4: RSE System and Antenna Installation at Intersections

Work conducted in this task focused on the integration of the prototype RSE system and antenna at the selected intersections. The RSE system and antenna installation included the wiring, antenna mounts, power supply, and interfaces to the traffic control devices. It should be noted that the RSE used in Task 8 was not the VII RSE that is planned for use in the Phase II FOT. Because the FOT RSE was under development and not available for use in the CICAS-V project, a substitute RSE was used with sufficient functionality in support the functional testing conducted in Task 8. The RSE will be upgraded to the FOT-ready RSE in Task 10 later in the project.

2.1.1.6 Task 8.5: Intersection Infrastructure Functional Testing

In Task 8.5, the CICAS-V Team developed a test plan and conducted functional testing of the intersection infrastructure in advance of the CICAS-V application tests. The goal was to verify and optimize the basic functionality of the infrastructure-vehicle communication interface. This testing was necessary to support the adequate completion of the RSE system and antenna installation process.

2.1.1.7 Task 8.6: Intersection Map Development and Verification

In this task, the CICAS-V Team developed the intersection map format and defined the process for transmitting maps from a local intersection RSE through the DSRC radio link to the vehicle. The CICAS-V intersections were digitally mapped in accordance with the map specifications and formats developed earlier which include intersection geometry, location of the stop lines, and other necessary attributes. This task also included the validation of the digital maps at each CICAS-V intersection.

2.1.1.8 Task 8.7: Application Development

In Task 8.7, the CICAS-V Team developed the CICAS-V application using the results from the development of the initial concepts of operation and system requirements, which include control interfaces, system design, application software development, and the DVI for engineering evaluations.

The performance of the applications was evaluated in real-world environments and engineering evaluations were conducted. The OEMs demonstrated application functionality interoperability of OEM-developed applications with communications infrastructure and the compliance with the objective test procedures developed previously. This task also contains the documentation of problems encountered and the mitigation strategies that were employed.

The system was built on a common architecture and shared application core that was common to all OEMs. This enabled faster development and facilitates easier maintenance of the application. Each OEM was responsible for the interface definition to their specific vehicle environment and system bus. The OEMs were also responsible for the interface between the application and the vehicle-specific DVI.

Work in Task 8.7 was also subdivided into three subtasks. A summary of each of these is provided, below.

2.1.1.8.1 Subtask 8.7.1: Basic Algorithms Design

This subtask focused on the design of the algorithms for the basic warning algorithm countermeasures and included algorithms for map matching, vehicle positioning, threat assessment, warning generation, DVI (Task 3.3), warning timing (Task 3.2), and so forth.

2.1.1.8.2 Subtask 8.7.2: Software Development

Work in this subtask centered around the software development effort to implement the CICAS-V application on the OEMs engineering vehicle platforms. The task also included development of the tools for data collection and analysis for the CICAS-V application evaluation.

2.1.1.8.3 Subtask 8.7.3: CICAS-V Functional Testing

This subtask will feature the functional testing of the integrated CICAS-V application at the identified intersections. Feedback from this initial round of tests was used as an input to Task 8.6.

3 Task Results

3.1 Task 8.1 – Vehicle Selection

The OEMs selected one vehicle each that was used for system testing and system development as well as being the vehicle used for the development of the FOT prototype in a joint decision between USDOT and the OEMs. Those vehicles will serve as the templates to build the vehicles in the FOT fleet. Several requirements figured into the vehicle selection – stability of the platform for the next four years and the possibility to install brake pulse and pre-charged panic brake assist features. This required the presence of Electronic Stability Systems (ESS) and Panic Brake Assist (PBA) on the vehicles as factory installed equipment. Having these systems would enable brake pulse or panic brake functionality via software change sinca ll the hardware would already be available. The vehicles selected and purchased were:

- Mercedes-Benz ML350
- Volvo S80
- Cadillac STS
- Acura RL
- Toyota Prius

In addition to the five OEM vehicles, Virginia Tech Transportation Institute (VTTI) purchased two Cadillac STS vehicles for the DVI development and the user tests on the VTTI Smart Road.

The vehicles were instrumented with initial equipment (some of which was later used in the Task 10 prototype build), such as:

- Laptop computer with CAN interface
- NovAtel OEM-V GPS receiver
- M/A Com DSRC antenna
- DENSO WRM DSRC radio

The NovAtel OEM-V GPS receiver was chosen because it accepts RTCM 104 v3 corrections, which were used to encode the differential corrections that were sent from the intersection to the vehicle. The DENSO Wave Radio Modules were used as a DSRC radio on the vehicle since this was the only functional radio available at the time of the development. The equipment was installed in an experimental configuration; this means that the components were not fully integrated or hidden. The radio and the GPS receiver were connected to the laptop computer through Ethernet and serial port, respectively. The vehicle data were obtained by connecting the CAN bus on the vehicle to the laptop through a CAN interface.

3.2 Task 8.2 – Intersection Selection for CICAS-V Development

The CICAS-V team (OEMs and VTTI) selected intersections in California (3), Michigan (3) and Virginia (4) for system development, testing and Pilot FOT (Virginia).

To aid in the selection of the intersections, criteria were developed that included the relevant parameters for the intersections

.The criteria that identified the best intersections for the development of the prototype were the positioning accuracy needed, GPS coverage, geometric features of the intersection and DSRC coverage. Other considerations emerged during the selection process such as controller type and the constraints of local DOTs. Reviewing the rest indicated no impact on the development of the prototype.

The required positioning accuracy can be divided into two classes:

- WhichRoad – the combined positioning error plus map error has to be smaller than 5 m to enable the vehicle to position itself on the correct approach road
- WhichLane – the combined positioning error plus map error has to be smaller than 1.5 m to enable the vehicle to position itself on the correct lane

The maximal allowed map error was specified as 30 cm, the maximum positioning error for lane level positioning was defined as 1 m with 95% probability in the error ellipse.

The intersections had to include in each location (CA, MI and VA):

- At least one stop sign controlled intersection with WhichRoad accuracy (stop sign)
- At least one signalized intersection with WhichLane accuracy (complex signalized)
- One signalized intersection with WhichRoad accuracy (simple signalized)

To select the intersections, a combination of DSRC coverage tests and GPS tests were conducted at the intersections and attention was paid to satellite obstructing buildings and terrain features. Candidate intersections were discussed with the supporting local DOT (Caltrans in California, Road Commission for Oakland County in Michigan and Virginia DOT in Blacksburg, VA) to obtain their permission. The final intersections were submitted to USDOT for a joint decision. The selected intersections were in California:

- 5th Avenue and El Camino Real in Redwood City (simple signalized and complex signalized)
- Hillview Rd. and Hanover Rd. in Palo Alto (simple signalized)
- Raimundo Rd. and Peter Coutts Rd. in Palo Alto (stop sign)

Due to several issues that were encountered after the initial selection had been completed, the Hillview Rd. and Hanover Rd. intersection in Palo Alto had to be dropped and another intersection at Dunbarton/Oakwood and El Camino Real in Redwood City was selected (complex signalized on El Camino, simple signalized on Dunbarton/Oakwood). This intersection has the advantage that it is in communication range of the 5th and El Camino Real intersection, which allows the study of issues that arise with several intersections within communication range of each other broadcasting the signal phase and timing (SPaT) and GID and the vehicle having to identify the correct messages for the intersection it approaches.

The intersections in Michigan are located at:

- W. 10 Mile Rd. and Orchard Lake Rd. (simple signalized)
- W. 12 Mile Rd. and Farmington Rd. (complex signalized)
- W. 11 Mile Rd. and Drake Rd. (stop sign)

The intersections in Blacksburg, VA were selected as the intersections for the naturalistic intersection approach data collection for Task 3.2 and they will be used for the Pilot FOT. Even though this is not technically a Task 8 activity, the Virginia intersections are listed here for completeness. The intersections are:

- Franklin St. and Elm St. (complex signalized)
- Franklin St. And Depot St. (complex signalized)
- Business Rte. 460 and Highway 114 (complex signalized)

The intersections in California and Michigan were used for the engineering development whereas the intersections in Virginia supported the Human Factors research and final systems testing.

3.3 Task 8.3 – Development of a Signal Sniffer

A signal sniffer, also known as load switch sensor, is a device to determine the current phase for a traffic signal without accessing the traffic signal controller. The reason for such a device is that older traffic signal controllers, such as the 170 series controllers, cannot export traffic signal phase through a standard serial or Ethernet interface at all or the information is not available with the update rate that CICAS-V requires. In those cases, a signal sniffer could be employed that (for example, inductively) measures which of the power lines that go from the traffic signal controller cabinet to the signal heads carry an electric current. Sensing this for all the signal heads and indications will give a complete picture of which phases are currently displayed for all the directions. The sniffer will not have the information about the time left in phase for any of the signal indications but this can be supplied as a priori knowledge for the amber phase. With this information, a state machine can determine the current phase and the time to phase change from amber to red, which is the minimum requirement for a CICAS-V system. The flow of events is shown in Figure 2.

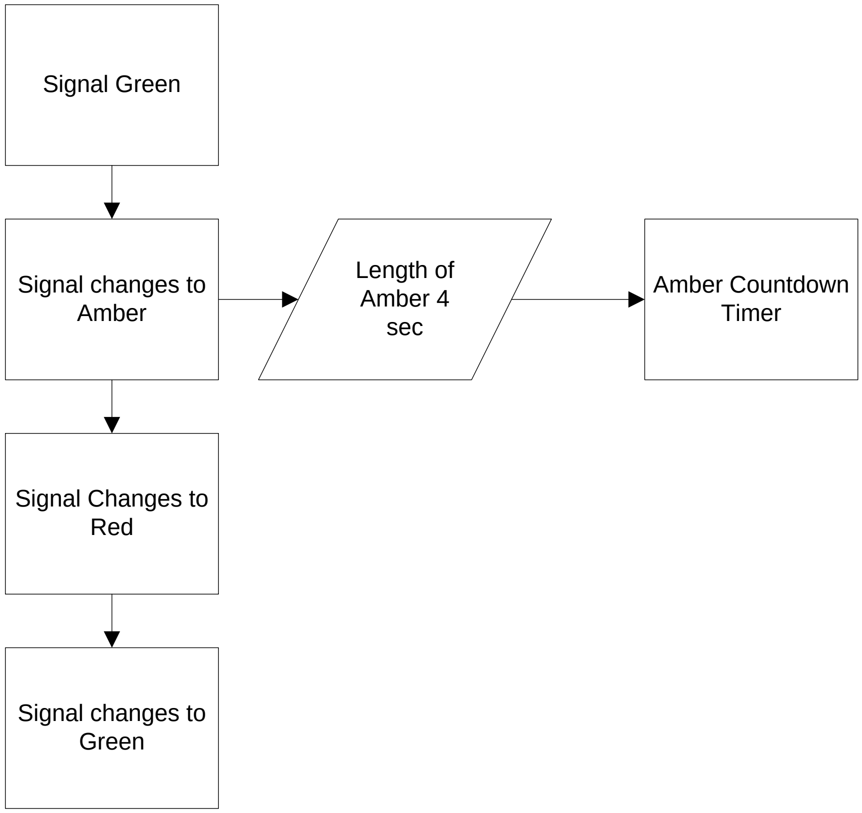


Figure 2 - Event Diagram

There are currently two different signal sniffers available. The University of California, Berkeley, PATH program has developed a signal sniffer that is used in the intersection of Page Mill Rd. and El Camino Real to detect signal phase and timing. Also, Virginia Tech Transportation Institute has a signal sniffer that they use for the intersection at the Smart Road. Both sniffers were tested and were found to have satisfactory performance, but due to the availability of signal phase and timing directly from the traffic signal controller in the intersections in California and in Michigan, the signal sniffer was not used in the intersection build or for any of the functional testing. VTTI used their signal sniffer to obtain phase information during the tests on the Smart Road.

During the CICAS-V Concept of Operations workshop in Austin, Texas, on November 1, 2006, the representatives from the traffic signal controller industry, NEMA and local and state DOTs argued strongly against the signal sniffer due to its adding complexity to the intersection installation. Also, the information that the signal sniffer collects is already available in the conflict monitor and at least one intersection equipment manufacturer is planning on unveiling a conflict monitor to the public that can export the phase information through a standard computer interface. If intersection operators are amenable to using conflict monitors for the purpose of collecting signal phase and timing information for CICAS-V installations, then intersections with legacy controllers can become CICAS-V equipped without the use of a signal sniffer. For those reasons no further development of the signal sniffer beyond what is currently available was conducted.

3.4 Task 8.4 – Installation of RSE and Antenna at the Intersections

The setup of equipment at the intersection included the installation of the following components:

- Road-side equipment (RSE)
- GPS base station
- DSRC antenna
- GPS antenna
- Connection between signal controller and the RSE
- Housing for components
- New signal controller

Due to variations in intersection configuration between Michigan and California the installations had significant differences. The intersection installation in California will be presented first followed by a discussion of the installation in Michigan. The installation of the intersection equipment in California was supported by Caltrans Sacramento, Caltrans District Four in Oakland and the University of California, Berkeley PATH. The intersection installation in Michigan was supported by the Road Commission for Oakland County. Outfitting a signalized intersection with CICAS-V equipment is more complicated than a stop controlled intersection and therefore this process is described in some detail. Stop controlled intersections only require that the vehicle has the GID of the intersection, which can be picked up from the RSEs in signalized intersections in the local area. .

3.4.1 Intersection Installation in California

The two signalized intersections in California selected for the CICAS-V project are under the authority of Caltrans. The configuration of the two intersections is similar. The intersection controller is housed in a cabinet at the side of one of the approach roads. The power to the signal heads run in special conduits from the controller cabinet underneath the road to the antenna masts. In the case of 5th and El Camino Real intersection, the intersection controller cabinet is located at the southeastern corner of the intersection as seen in Figure 3. At this corner there is no convenient intersection mast, only a short B-Pole that was not suitable for the mounting of the antennas. Another pole located at the corner could not be used due to the presence of high-voltage power lines (75 kV) that would have negatively impacted the antennas. The only possibility was to place the antennas on the mast arm shown in Figure 3 and run the signal line through the power conduit to the mast and from there to the RSE. Since California wiring codes do not permit an electrical signal cable in a power conduit, the signal had to be transmitted through a fiber-optic cable (multi-mode).

The Econolite 170 controller was replaced with a newer Econolite 2070 controller that can export signal phase and timing through a serial port on the controller. This will be described in more detail later. Since the signal had to be transmitted through the fiber-optic cable, several converters were needed to convert the serial information to Ethernet, from Ethernet to fiber-optic (using the LBH100A-H-ST-12 from Black Box) and on the other side back to serial. Figure 4 shows the block diagram of the components and the connections between the components.

Figure 3 - Intersection Configuration at 5th Ave. and El Camino Real in Redwood City, CA

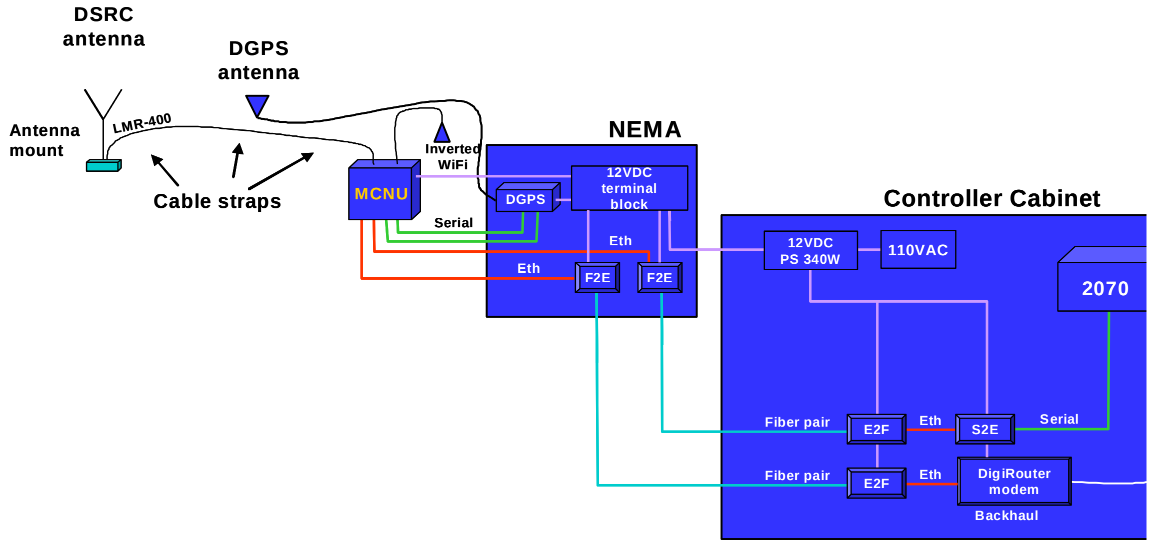


Figure 4 - Block Diagram of Intersection Installation

The RSE, GPS base station and the antennas were installed on the intersection mast and mast arm at the northeast corner of the intersection. The GPS base station (NovAtel OEM IV), the converters from fiber-optic to Ethernet and the power supply for the GPS system were housed in a NEMA cabinet. The RSE itself was a Multiband Configurable Network Unit (MCNU) from TechnoCom Corporation in San Diego, CA. This unit contained a 400 MHz PC104 processor and a DSRC radio that was compatible with the WSM radios that were originally developed in the VSC Project and manufactured by DENSO. Both the MCNU and the NEMA cabinet were mounted on a backboard that was then fastened to the mast. Power for the MCNU and the other equipment was provided from the intersection. Figure 5 shows a picture of the setup at the intersection.

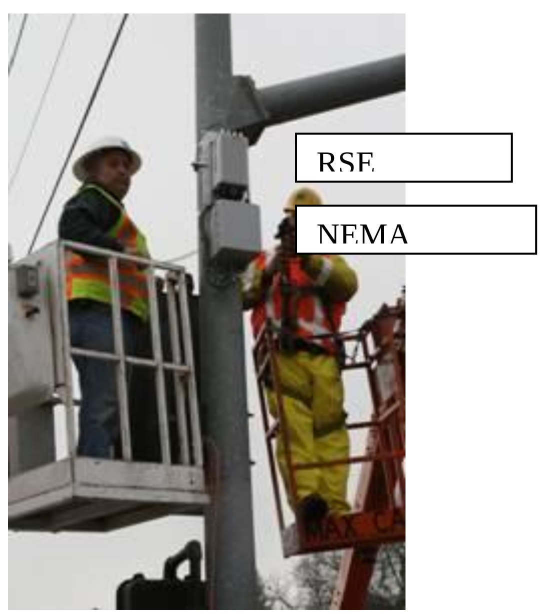


Figure 5 - Installation of RSE on the Mast

The DSRC and GPS antennas were mounted on the mast arm as can be seen in Figure 6.

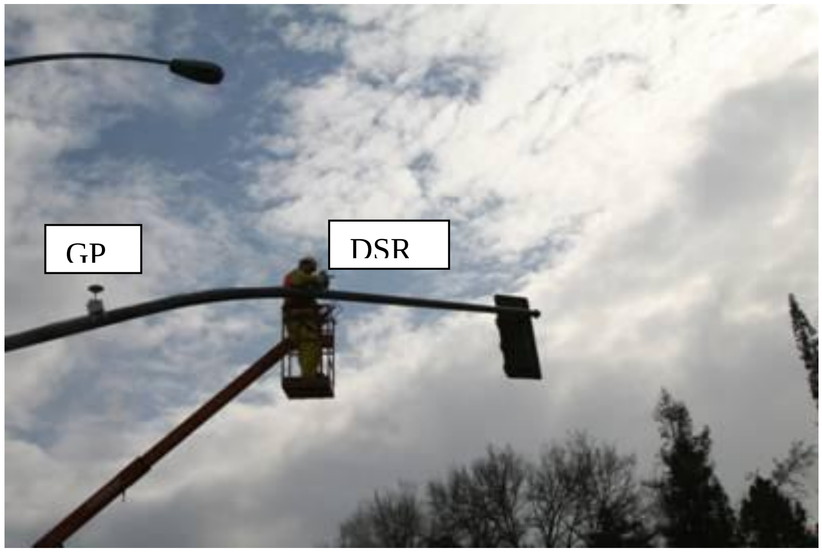


Figure 6 - Antenna Installation

In addition to the DSRC and GPS antennas, the RSE is also connected to a WiFi antenna that allows local software updates and system maintenance.

Despite the apparent complexity of the intersection installation, the CICAS-V system in the intersection has proved to be remarkably stable and has worked without interruption (aside from weather related power outages) since February 2007.

The intersection controller that is used in the CICAS-V intersections in California is an Econolite 2070 controller and is running the AB3418 protocol. The California legislature passed legislation (Assembly Bill 3418) requiring all signal controllers purchased in the state after January 1, 1996 to be compliant with a standardized protocol. The legislation delegated the creation of the technical standard to Caltrans. As a result, Caltrans worked with the developers of the National Transportation Communications for ITS Protocol (NTCIP) to develop a protocol that would: (1) be ready by the mandated deadline and (2) be as consistent as possible with the future NTCIP standard. The resulting protocol from this effort is referred to as the AB3418 (See References, NTCIP Guide Version 2.0[0].). The AB3418 protocol is a query-response protocol. This means that the RSE has to constantly query the status of the controller to obtain the necessary information. Also, the CICAS-relevant information that can be obtained through this protocol is only whether a light is green or not green. Information about when the light is amber or red is not available directly from the controller but has to be inferred. As with the signal sniffer, the knowledge of the length of the amber phase is critical to predict the state of the signal and the countdown to red. In fact, the AB3418 protocol requires a more sophisticated state machine for phase prediction than the sniffer since it is not known when the light turns red. Figure 7 shows the diagram for the phase prediction for one signal. The process to infer the amber and red states is shown in Figure 7 to the right of the box labeled “light not green”.

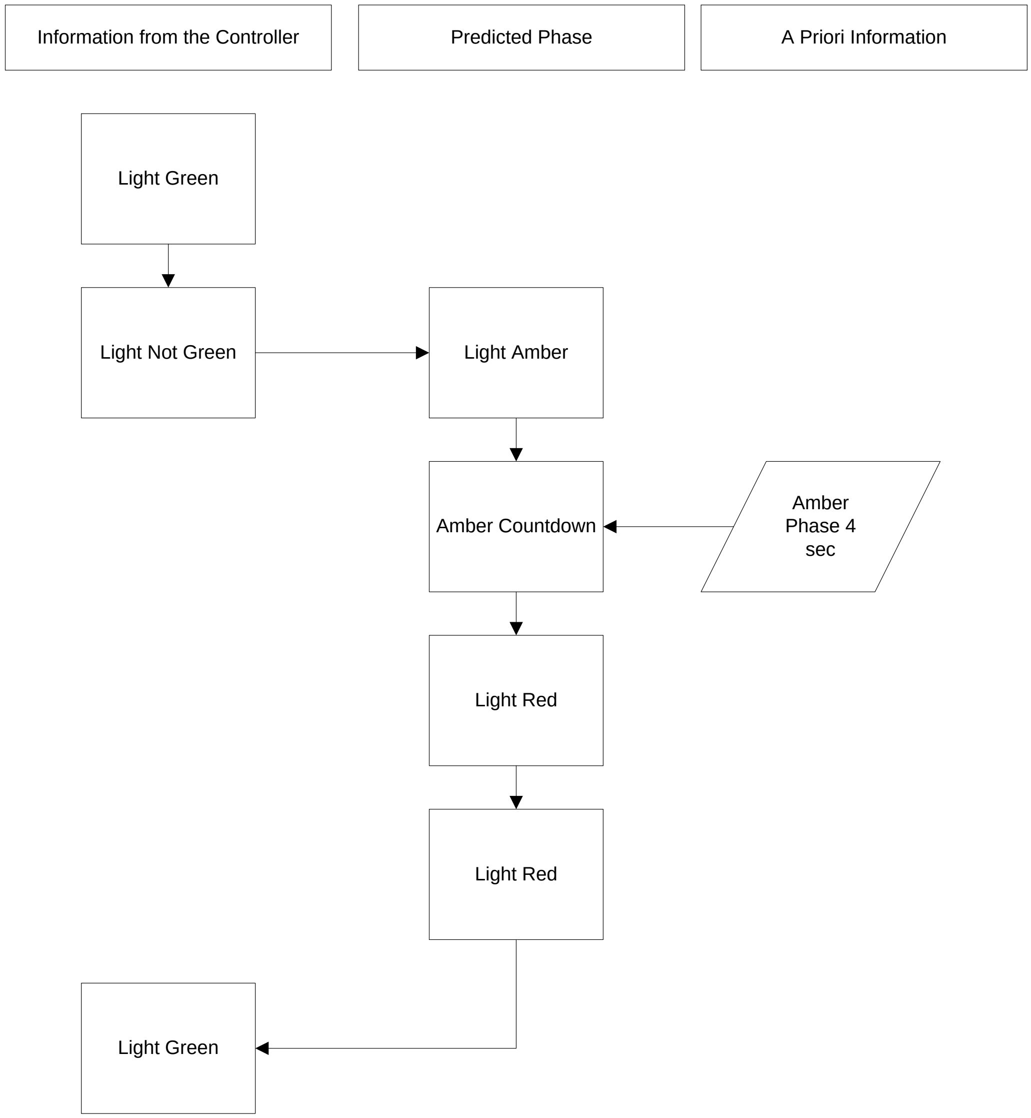


Figure 7 - Phase Prediction Using the Information from the Controller using the AB 3418 Protocol

The information that can be queried from the Econolite 2070 controller through the serial port has a time granularity of 150 ms. For the CICAS-V intersection in California, the state machine software was supplied by UC Berkeley PATH.

3.4.2 Intersection Installation in Michigan

The intersection configuration in Michigan differs from the one in California in that the intersection controller cabinet is smaller and it is mounted on a pole at the corner of the intersections. The Road Commission for Oakland County permitted the CICAS-V project to install a separate cabinet on the opposite side of the pole with the controller cabinet that was used to house the MCNU RSE, the GPS base station and power supplies. Since the cabinets were located at ground level, it made exchanging the equipment, debugging of software and maintenance much easier than in California where components are mounted on a pole. The antennas were mounted on the top of the pole on a cantilever arm. The disadvantage of this installation is that the antenna cable is relatively long and incurs signal loss due to the length of the cable (approx. 6 dB). The range of the DSRC communication was still large enough so that the functioning of the CICAS-V system in the intersection was not impacted. Some of the signal loss was offset by using a higher gain DSRC antenna (12 dB). The first intersection to come online in Michigan was the intersection at Orchard Lake and 10 Mile Roads. Figure 8 shows an aerial view of the intersection with the location of the controller cabinet. The intersection has left and right turn lanes that have the same signal phasing as the through lanes in the same approach, which makes this intersection effectively a WhichRoad intersection. .

Figure 8 - Intersection of Orchard Lake Rd. and West 10 Mile Rd.

Figure 9 through Figure 11, as shown below, identify the CICAS-V cabinet, the signal controller cabinet and the installation of the antennas on the same pole.

Figure 9 - Equipment in the CICAS-V Cabinet

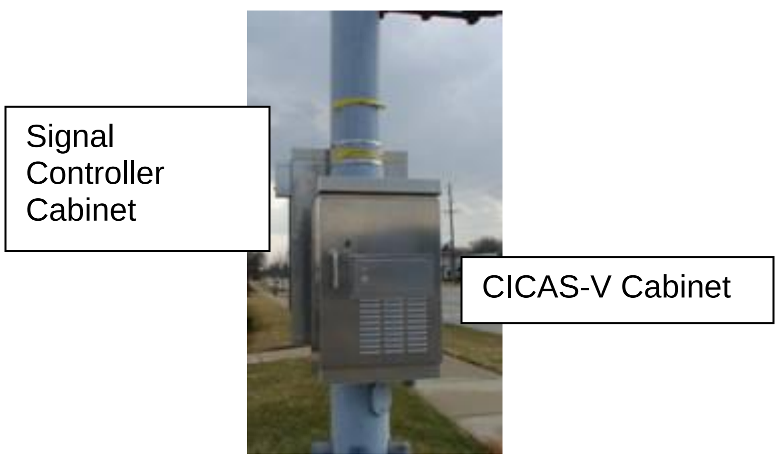


Figure 10 - CICAS-V and Intersection Controller Cabinets

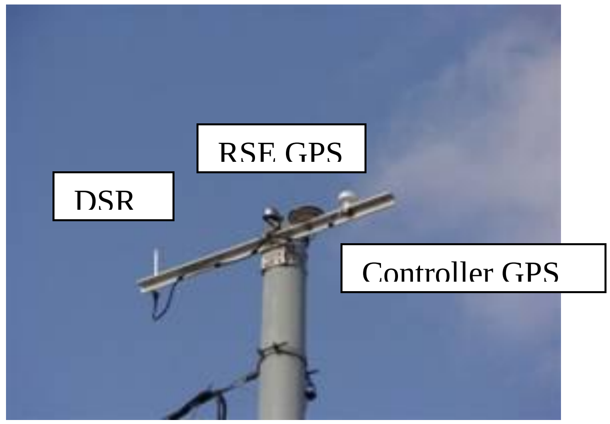


Figure 11 - Antenna Installation

The block diagram that shows the components and connections is shown in Figure 12.

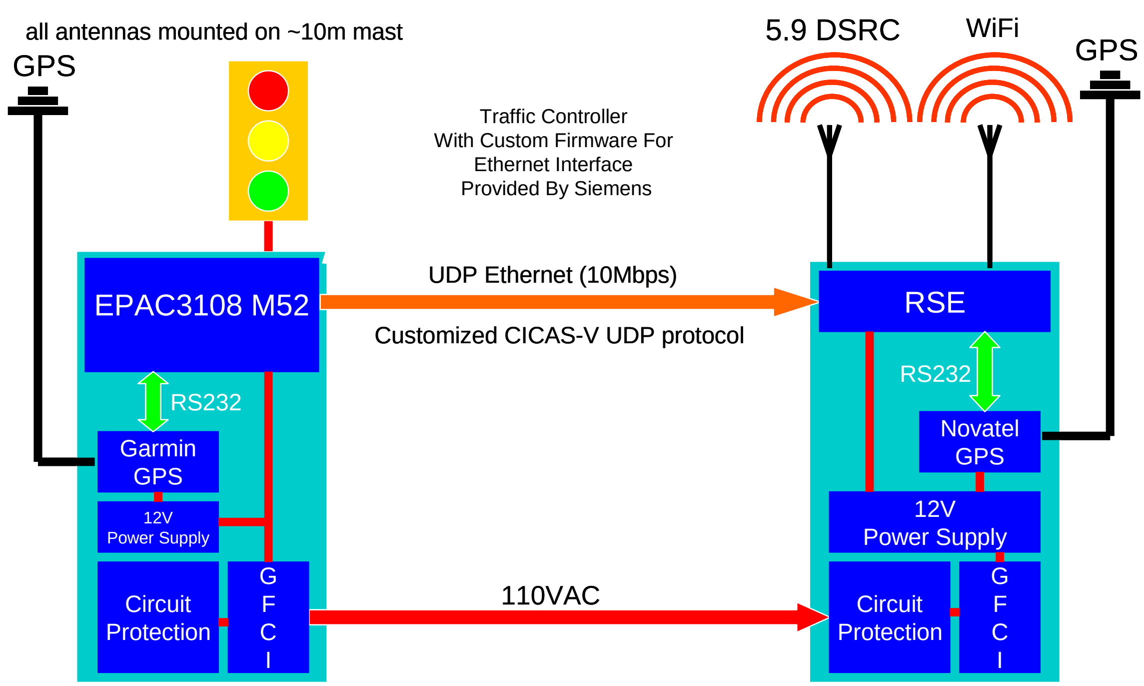


Figure 12 - Block Diagram of the Intersection Installation in Michigan

The controller that is used in the CICAS-V intersections in Michigan is the Siemens Eagle EPAC 3108 M52 controller. This controller includes custom firmware for an Ethernet interface and a customized protocol that unicasts the information over the Ethernet link to the IP address of the RSE as a UDP packet as soon as a state changes.

The latencies that were measured are between 30 and 60 ms, which are excellent for CICAS-V purposes.

3.5 Task 8.5: Intersection Functional Testing

The intersections that were outfitted in Task 8.4 were tested for CICAS-V functionality. The tests included mapping the intersection DSRC coverage and determining whether the intersection equipment sent out all the required messages. To test the message reception, an RSE monitor application was developed that used the DENSO WRM module to receive the messages and to display them on a computer in a test vehicle. The applications showed the current phases for all the approaches, as well as the intersection map (GID) and the reception of the positioning correction message (GPSC). Figure 13 shows a screen shot of the RSE monitor for the intersection at 5th Ave. and El Camino Real.

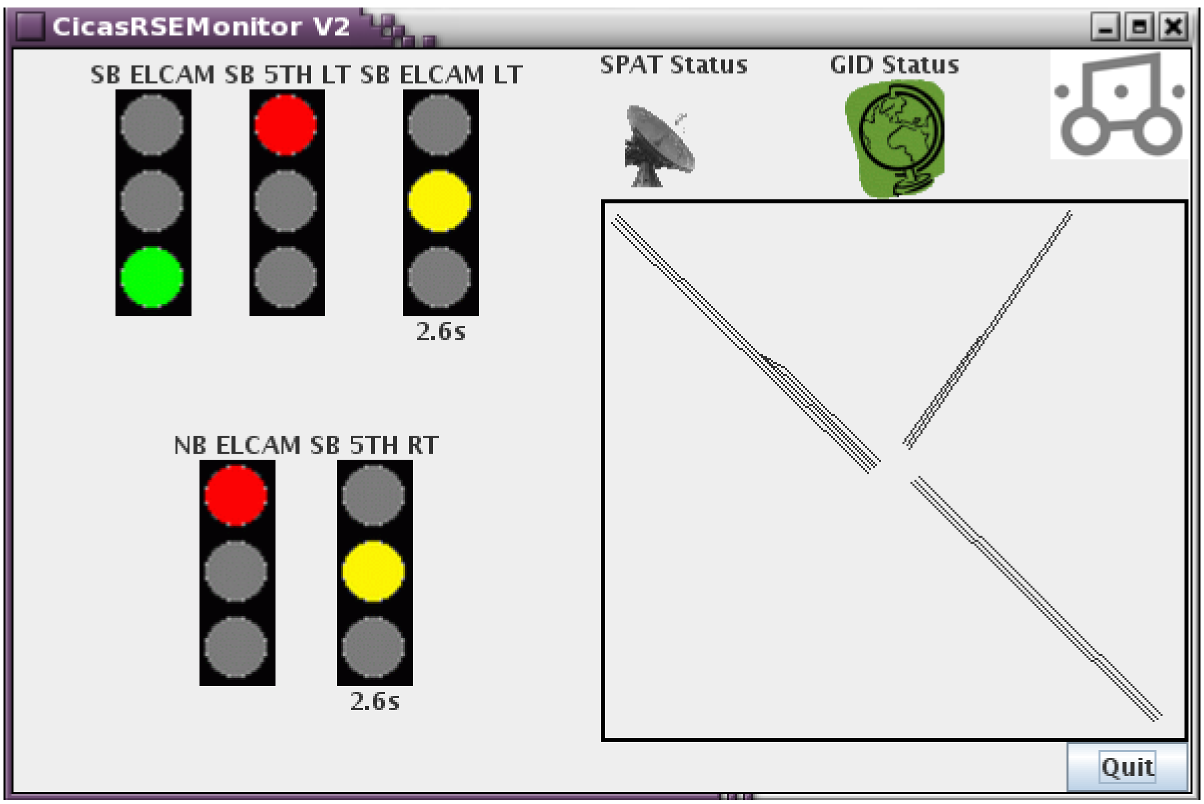


Figure 13 - Display of Received Messages on the RSE Monitor

The picture shows the phases for all the movements together with the countdown timer for the amber phases (5th Avenue Southbound 2.6 sec, Southbound El Camino Real Left Turn, 2.6 sec), the SPaT status (currently receiving messages), the GID status (intersection map received) and a display of the received map.

The DSRC communication range was measured by parking the test car on the side of the road and determining the packet error rate. Figure 14 shows the DSRC communication range for the intersection at 5th Ave. and El Camino Real.

Figure 14 - DSRC Communication Range for 5th Ave. and El Camino Real

The red dots signify a packet error rate larger than 70%, the blue dot show packet error rates smaller tan 70%. At a packet error rate of 70%, the likelihood of a packet reception of larger than 99% requires the sending of five packets.

The measurements showed that the communication range extended more than 400 m along El Camino Real in both directions and to about 200 m along 5th Ave. The limited communication range on 5th Ave. was due to an underpass at about 200 m.

The intersections in Michigan were similarly tested and the communication range on Orchard Lake Rd. and 10 Mile Rd. had a communication range exceeding 400 m in all directions and it was ascertained that all the intersection messages were received. From this, it was determined that the intersection installation was successful and that the antenna placements were sufficient to provide the needed coverage.

3.6 Task 8.6: Intersection Map Development and Verification

The CICAS-V system requires an intersection map of sufficient accuracy to enable matching the vehicle to the road or even to the lane in intersections with dedicated turn lanes. As mentioned above, the CICAS-V concept of operations defines two levels of accuracy – WhichRoad and WhichLane[1] – combining both positioning and map accuracy. The map has to have the following properties:

- Sufficiently accurate road/lane geometry for all lanes/approach roads;
- Intersection identification, including whether the intersection is stop sign controlled or signalized;
- Stop bar locations for all lanes;
- An intersection reference point;
- Lane widths for all the lanes;
- Correspondence between lane and traffic signal applying to the lane.

There are no commercially available maps that have these properties and the ConOps assumed that it could not be required that vehicles would have such a map stored on-board, even if it existed. Therefore the ConOps required that such a map be transmitted from an RSE at the intersection to the vehicle through DSRC. In order to increase the probability of reception of the map, it should fit within one DSRC Wave Short Message (WSM) packet. The maximum size of one packet is 1.4 K as specified in the IEEE 802.11p.[0] About 400 bytes of this packet size were assumed to be security overhead and not available for the actual message content. Those constraints led to the design of a map that was called a *Geometric Intersection Description* (GID) and that will be used for the remainder of the project. The specifications were also entered in the Society of Automotive Engineers (SAE) J2735[0] standards process to become an automotive standard in the future.

To minimize the size of the GID the following design choices were made:

All geometry points are Cartesian offsets from an intersection reference point that is given in (Latitude, Longitude, and Altitude) coordinates in the WGS 84 system. This means that all the points that are used to describe the geometry are described as distance in decimeters from the intersection reference point (x [decimeters], y [decimeters], z [decimeters]);

All roads/lanes are described as an ordered set of geometry points together with the lane width at each point;

[1] These terms originated in the CAMP EDMAP Project.

The lane geometry is described by specifying the centerline of the lane;

The stop bar location for each lane is the first geometry point for the lane;

Lane geometries are represented out to a distance of 300 m from the intersection reference point for approaching lanes;

Outgoing lanes are optional but can be included, if necessary.

Figure 15 shows the elements of a GID.

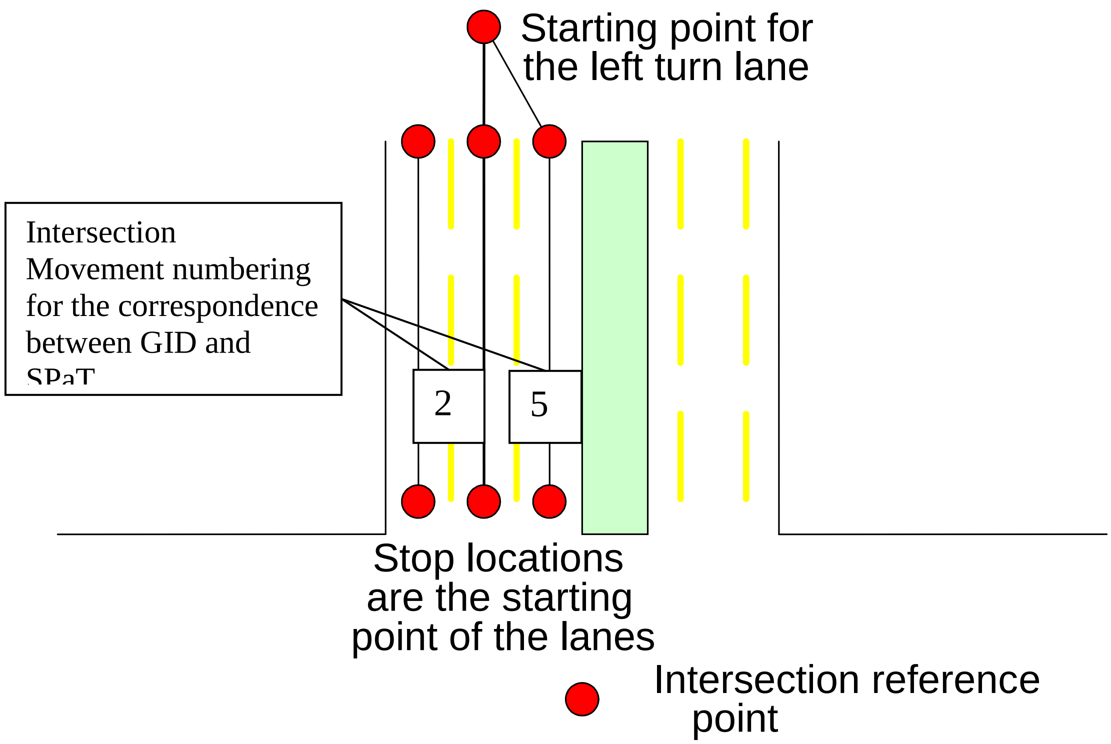


Figure 15 - GID Elements

As mentioned above, the basic element of the GID is a point or "node." There are two kinds of nodes in the GID:

- The Intersection Reference Point (IRP), given as Latitude, Longitude, Altitude
- The Nodes that describe the lanes, given as offsets in Cartesian coordinates from the IRP.

The set of nodes that describe a lane are collected in the "Node List."

There are currently two kinds of lanes:

- Reference Lane

- Computed Lane

A reference lane is a lane that is fully specified by a list of points. A computed lane is a lane that can be derived from a reference lane by a simple parallel shift of the reference lane. This is just a way to reduce the size of the GID message in cases where several parallel lanes can be grouped into one approach. An approach is defined as all lanes of traffic governed by a single, independent signal phase cycle, moving towards an intersection from one direction. This corresponds to the term “Movement” used by Traffic Engineers. Figure 16 shows the seven approaches to the intersection of 5th Ave. and El Camino Real in California. Approach 6 consists of three lanes, where the rightmost lane is wider than the other ones due to parking possibilities. Approach 2 contains three lanes and approach 5 contains two lanes. For approach 2 the GID has to specify the leftmost through lane as a reference lane and the other two lanes in the approach can be represented as computed lanes. The same is true for approach 6 where again the leftmost lane has to be specified through a node list and the other two lanes can be specified by the offset from the reference lane. It should be noted that the computed lane is not a mandatory feature of the GID but a device to minimize the size. In the GID, all lanes can be specified through node lists (as reference lanes) if the size of the GID permits.

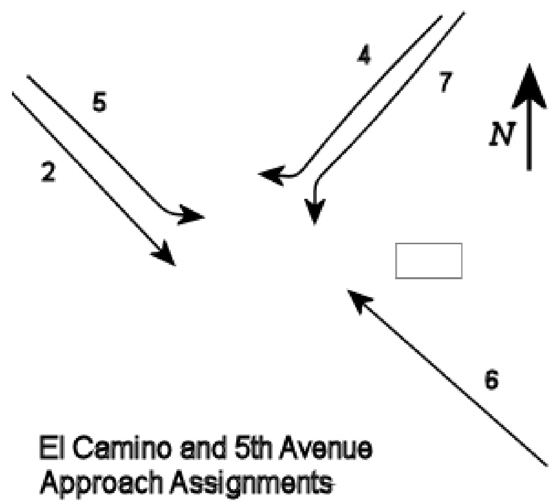


Figure 16 - Approaches for the Intersection at 5th Ave. and El Camino Real in California

Since there are no commercial maps available that describe the geometry to the required accuracy level, and some of the required GID attributes such as stop location are similarly missing, the CICAS-V project had to generate the GIDs. After looking at several alternatives, aerial surveying was selected as the method to map the intersections. The company chosen to map the intersections was HJW GeoSpatial, Inc. (HJW) in Oakland, California. The CICAS project developed specifications for the GID that were transmitted to HJW and HJW then took a high-resolution aerial photograph of the intersections. The resulting image had to be ortho-rectified. Also for this purpose a number of points on the picture were mapped by a surveyor on site. The company took the lane markings on the image to determine the location of the centerline for each lane and delivered the geometry of the lanes as a set of points, as specified. Those points were subsequently converted into the GID message, using a compiler that was specifically developed for CICAS-V.

It should be pointed out that the geometry data can also be collected by using probe vehicle traces, possibly combined with a vision system that recognizes the lane markings, to position the vehicle with high precision within the lane. This technique was used successfully by NAVTEQ in the EDMAP Project to make the high precision maps that were used in the project (Enhanced Digital Mapping Project Final Report, 2004[0]).

In mapping the intersection, work has been done to specify the "North" direction accurately as the Geographic North in the WGS 84 coordinate system. Using the "North" direction in the State Plane Coordinate System or the UTM Coordinate System, which are both used widely to specify geography, will lead to a rotation of the GID with respect to the Ground Truth by an angle that is location dependent. For example, in the UTM system the local 6 degree zone is centered on the central meridian which coincides with the geographic north. The farther the location of the intersection is away from the central meridian, the larger the angle between UTM north and geographic north. For the mapping of the intersection, this can amount to several meters of discrepancy between the position on the GID and the GPS position that the vehicle receives from the positioning system.

The GID is a very compact map of the intersection. The sizes of the GIDs for the intersections in the CICAS-V project are:

- 5th Ave. and El Camino Real: 352 bytes
- Orchard Lake and 10 Mile Roads: 483 bytes
- Farmington and 11 Mile Roads: 503 bytes
- Drake and 11 Mile Roads: 331 bytes

The accuracy of the GIDs was determined by using the CICAS-V system map matching functionality to determine whether map matching was working for all lanes in all approaches. If map matching was working according to performance specifications, it was determined that the GID was of sufficient accuracy for the purpose of CICAS-V.

3.7 Task 8.7: CICAS-V Prototype Development

Task 8.7 developed the initial algorithms and message sets to test the technological building blocks of the CICAS-V system. The initial specifications for the FOT system development were also developed in this task and then given to DENSO. The development activities included:

- Message sets
- Map matching
- Positioning correction
- Warning algorithm

The individual elements were developed and tested but not formally combined into a complete prototype. During the fourth quarterly briefing a prototype that combined the map matching and the warning algorithm was demonstrated.

3.7.1 CICAS-V Message Sets

The cooperative nature of the CICAS-V system requires the definition of the messages that are being sent from the intersection to the vehicle. The CICAS-V project defined the following messages as necessary for the system to function:

- Wave Service Announcement (WSA)
- Signal Phase and Timing (SPaT)
- GID Message (GID)
- GPS Correction Message

There are two messages that are to some degree optional but that were implemented. These are:

- Area GID (AGID)
- Traffic Signal Violation Warning Given (TSVWG)

3.7.1.1 Message Framework

In order to provide a common framework for all the messages, the CICAS-V project created the Transportation Object Message (TOM) that is based on XML but streamlines the message for byte efficiency. Here we only provide an overview over the basic concept.

XML is a meta-language ideally suited to dynamic data markup. It is very popular in the software engineering community. Descended from SGML, it is very expressive, flexible and powerful. Its disadvantage is the low byte-efficiency for RF transmissions. TOM's inner workings are patterned after the work conducted in the W3C XML Binary Characterizations Working Group. It was created to be like XML but highly-streamlined for byte efficiency for transmitting complex application data over DSRC. The TOM was designed as:

- Object-Oriented, Binary XML
- Easy to understand
- Minimal overhead

XML is well suited for describing data of arbitrary complexity. TOM has similar capability but is limited by the maximum size of an object.

A key feature of TOM is *minimal overhead*. There are no 'counts' of enclosed elements, no 'lengths' of arbitrary data - apart from Object Size (below). This serves three purposes: (1) to prevent bugs from incorrectly setting counts or lengths; (2) to eliminate the need for foreign parsers to know something about the payload to get by it; and (3) to improve message reception probability by keeping DSRC message size to a minimum.

Considerable attention was paid to byte-efficiency due to the nature of the transmission medium. A small, tight message places a smaller load on the RF channel and will more likely be received.

Message frameworks provide a basic set of services for message transmission. A TOM frame begins each message with a Message Header and ends it with a Message Footer. The framework provides message differentiation and a basic measure of integrity.

There may only be one frame per message. Ideally, that frame never exceeds 1,024 bytes to fit into a WSM packet, assuming 200-400 bytes for the security overhead and WSM frame overhead. It must begin with a TOM Message Header and end with a TOM Message Footer.

Everything between header and footer is considered message content. The content is a set of object tags.

The TOM framework allows all the messages that are received to be treated in the same way in their decoding, which creates efficiency in application development and robustness of the code. Also, it allows consistent authoring of content across all the different messages. In order to develop the various messages, a compiler was developed that allows the authoring of the message in XML and then converts the XML message into a TOM message that is far more byte efficient.

The Transportation Object Message Compiler (TOMC) is a software tool that converts arbitrary XML data into a message consisting of hierarchical binary XML objects. It does so by employing a TOM Schema file which describes how to code objects. The schema itself is expressed in XML.

The compiler implements several basic operations. One packs bits specified by XML tags into bitmasks. Another translates latitudes and longitudes from text into 4-byte integers while maintaining a high degree of precision. Still another governs how many contiguous objects and elements of a given type are allowed to pack together. The software does not, however, dictate much beyond that. The schema, an ordinary XML file loaded by the compiler at runtime, directs message composition.

3.7.1.2 Wave Service Announcement Message (WSA)

The WSA follows the IEEE 1609 standard by specifying the service that the RSE is offering and on which channel this service is being offered. The message is designed to work within the framework of the Vehicle Infrastructure Integration as one of the services that is defined within the overall WSA message. The WSA message within CICAS also uses the TOM framework described above when used in a stand-alone mode without the VII.

The WSA message specifies the Provider Service ID (PSID) which identifies the service to the vehicle, the channel number for the channel on which the service is offered and the Provider Service Content (PSC) which can be a string of characters up to 32 bytes long. The PSC can be freely defined by the service provider and the vehicle application will have to be able to parse the string.

The basic use of the WSA for CICAS-V is defined in the CICAS-V Concept of Operations[0]:

Conditions: The CICAS-V enabled vehicle is approaching a CICAS-V enabled traffic signal at a simple intersection with no dedicated turn lanes, where all vehicles on the same approach have the same traffic signal indication.

As the vehicle approaches a CICAS-V intersection and comes in range of the system's communications, the vehicle receives a CICAS-V service announcement on the control channel indicating the availability of the intersection's GID, area geospatial information, the status of the intersection, and differential GPS corrections. The vehicle decides if it needs either or both of the GID or the area geospatial information broadcast. If necessary, the vehicle switches to the service channel and receives the intersection's GID and/or the area geospatial information. The vehicle stores the new GID and/or geospatial information in its data store, replacing any older information.

The PSC contains the GID version number of the intersection that the vehicle approaches or the Area GID version number together with other content that will be specified later, such as the status of the CICAS-V system in the intersection.

In the current form, only the GID and the Area GID are planned on being broadcast on the service channel. The system is flexible, though, in that all messages can be sent over the control channel (or a service channel).

3.7.1.3 Signal Phase and Timing Message (SPaT)

The SPaT is central to the cooperative aspect of the CICAS-V system in that it conveys the information about the status of the traffic signal infrastructure to the vehicle.

The SPaT, or Signal Phase and Timing message layer, is designed to provide traffic signal phase and timing information organized in such a way that a vehicle can reliably determine (1) whether it has right-of-way or must stop; (2) what movements are allowed from a given lane; and (3) an extensible scheme capable of incorporating future content with minimal negative impact. The SPaT follows the TOM framework described earlier. For purposes of CICAS-V, the vehicle only needs to know whether it needs to stop at the stop bar or whether it has a green light. In this sense the SPaT provides a simplification of the states of the intersection but the information can be included to enable various gap acceptance applications within the intersection that need somewhat more information than the CICAS-V application.

The SPaT is designed to work with the GID so that the vehicle, correctly matched to its lane, can filter from the SPaT message the signal indication that applies to its current trajectory. It should be understood that the SPaT sends out the information for all the traffic signals and movements through the intersections and the vehicle will pick out the applicable one.

In the following, the structure and the various elements of the SPaT message are discussed.

3.7.1.3.1 SPaT Object IDs

Consult Table 1 for SPaT Object ID values within the SPaT section.

Table 1 - SPaT Object IDs

Intersection	2
Approach	3
Pedestrian	4
Preempt	5
Location	6
Label	7
Sensor	8
Current time	9

3.7.1.3.2 SPaT Layout

Typically, the general layout of a SPaT is organized as follows. Indentation indicates relative nesting levels. Only one object repeats: SPaT Approach Object. An intersection with eight approaches has eight SPaT Approach Objects in it. The relationship is one approach object for one approach. SPaT Approach Objects in this scenario do not need matching Close SPaT Approach Objects because they do not encapsulate other objects (i.e., they are standalone). Standalone objects do not require closure.

The layout of required objects in a SPaT Layer is as shown below. Indentation implies encapsulation.

TOM Header
Layer Object: SPaT
| SPaT Intersection Object
| | SPaT Approach Objects
| Close SPaT Intersection Object
Close Layer Object
TOM Footer

Optional SPaT objects may appear elsewhere in the message layer at the points indicated below.

TOM Header
Layer Object: SPaT
| *Optional SPaT Object*
| SPaT Intersection Object
| | *Optional SPaT Object*
| | SPaT Approach Object
| | | *Optional SPaT Objec*[2]*t*
| | Close SPaT Approach Object
| Close SPaT Intersection Object
Close Layer Object
TOM Footer

Standalone and encapsulating Approach Objects may be mixed as needed.

3.7.1.3.3 SPaT Layer Object

The SPaT Layer (Figure 17) encapsulates one or more SPaT Intersections. The SPaT Layer Object must be closed with the Close Layer object immediately following all of its interior objects.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
Layer Type	unsigned 16-bit integer
Layer ID	unsigned 8-bit integer
Content Version	unsigned 8-bit integer
Format Version	unsigned 8-bit integer

Figure 17 - SPaT Layer Object

The Layer Type for SPaT is 2.

[2] Note the addition of the Close SPaT Approach Object in this scenario. This is now necessary because the SPaT Approach Object is no longer standalone. It encapsulates something.

Content Version is used to indicate a change in SPaT content. For example, the change of a Countdown Timer is a content change. The Content Version for that SPaT layer must differ from the prior broadcast SPaT message layer for the intersection. Receiving applications that notice a change must parse the contents of the SPaT Layer. If Content Version hasn't changed, the application is free to disregard the layer, using cached data.

Assuming SPaT Content Version changes on the order of ten times per second, wrapping around will occur about once every 25 seconds. Layer ID is used to tell layers in the same message frame apart. If there is only a single SPaT layer in a message then the Layer ID is set to 0.

The Format Versions table (see Table 2) specifies which value to use for the SPaT Layer Object's Format Version field.

Table 2 - Format Version

TOM Framework Version	1
GID Format Version	2
SPAT Format Version	2
GPSC Format Version	1
TSVWG Format Version	1
RCMD Format Version	1

3.7.1.3.4 SPaT Intersection Object

This is a required object. The SPaT Intersection Object (Figure 18) uniquely identifies the intersection it corresponds to. It encapsulates all SPaT Approach Objects at the intersection, and possibly other optional objects that may apply to the entire intersection. Encapsulated SPaT Approach Objects may occur in any order. The SPaT Intersection Object must be closed with the Close Intersection object. It is possible to have multiple intersections per SPaT layer but this is not recommended.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
Intersection ID	unsigned 32-bit integer

Figure 18 - SPaT Intersection Object

3.7.1.3.5 SPaT Approach Object

The SPaT Approach Object (Figure 19) provides signal phase and timing for an individual approach. Its parent is the SPaT Intersection Object. This is a required object.

> *The role of this object is to convey all relevant signal phase and timing information needed by drivers in controlled lanes. There must be one Approach Object for every unique signal phase cycle from a given direction. An Approach ID is provided to allow vehicles to match geometry data in the GID to signal phase data. This ID must match its counterpart in the GID Approach Object.*

The SPaT Approach Object may encapsulate other objects that apply to a given approach, except other Approach objects. When encapsulation occurs there must be a Close SPaT Approach object. Otherwise, the Close SPaT Approach object should be omitted for byte efficiency as the object is intended to be standalone under normal circumstances.

<table>
<tr><td colspan="2">Object ID</td><td>unsigned 8-bit integer</td></tr>
<tr><td colspan="2">Object Size</td><td>unsigned 8-bit integer</td></tr>
<tr><td colspan="2">Approach ID</td><td>unsigned 8-bit integer</td></tr>
<tr><td colspan="2">Signal Phase Indications</td><td>32-bit bitmask</td></tr>
<tr><td>Countdown Timer Confidence</td><td>Yellow Duration Confidence</td><td>2 x unsigned 4-bit integer (two nibbles)</td></tr>
<tr><td colspan="2">Time until next signal phase change
(in hundredths of a second)
aka Countdown Timer</td><td>unsigned 16-bit integer</td></tr>
<tr><td colspan="2">Yellow Duration</td><td>unsigned 8-bit integer</td></tr>
</table>

Figure 19 - SPaT Approach Object

3.7.1.3.6 Signal Phase Indications

The Signal Phase Indications field (Table 3) is a bitmask to facilitate combinations of various indications or signal lights. One would ‘OR’ the bits together before storing the value in the field (in network order).

Table 3 - Signal Phase Indication Bit Values

All 0	Dark
0x00000001	Green Ball
0x00000002	Yellow Ball
0x00000004	Red Ball
0x00000010	Green Left Arrow
0x00000020	Yellow Left Arrow
0x00000040	Red Left Arrow
0x00000100	Green Right Arrow
0x00000200	Yellow Right Arrow
0x00000400	Red Right Arrow
0x00001000	Soft Green Left Arrow
0x00002000	Soft Yellow Left Arrow
0x00004000	Soft Red Left Arrow
0x00010000	Soft Green Right Arrow
0x00020000	Soft Yellow Right Arrow
0x00040000	Soft Red Right Arrow
0x00100000	Straight Green Arrow
0x00200000	Straight Yellow Arrow
0x00400000	Straight Red Arrow
0x01000000	Flashing Ball
0x02000000	Flashing Left Arrow
0x04000000	Flashing Right Arrow
0x08000000	Flashing Soft Left Arrow
0x10000000	Flashing Soft Right Arrow
0x20000000	Flashing Straight Arrow

The Flashing bits are modifiers. To indicate a Flashing Red Ball, for example, the Red Ball and Flashing Ball bits must be on. See Table 4. Soft arrows are oblique or angled arrow indications.

Table 4 - Signal Indication Bit Combination Guide

	Green	**Yellow**	**Red**	**Flashing**
Ball	1h	2h	4h	1000000h
Left Arrow	10h	20h	40h	2000000h
Right Arrow	100h	200h	400h	4000000h
Soft Left Arrow	1000h	2000h	4000h	8000000h
Soft Right Arrow	10000h	20000h	40000h	10000000h
Straight Arrow	100000h	200000h	400000h	20000000h

Sample combinations are:

- Green Ball = 1h
- Yellow Ball = 2h
- Red Ball = 4h
- Flashing Ball (1000000h) + Red Ball (4h) = Flashing Red Ball (1000004h)
- Red Ball (4h) + Green Right Arrow (100h) = 104h
- Green Ball (1h) + Green Left Arrow (10h) + Soft Green Left Arrow (1000h) = 1011h
- All Red bits ON (444444h) = Red Mask (used to determine if any signal indications in the approach are Red)
- All Flashing bits ON = Flashing Mask (used to determine if any signal indications in the approach are Flashing)

3.7.1.3.7 SPaT Confidences

Countdown Timer Confidence (high) and Yellow Duration Confidence (low) are 4-bit fields within the same byte. These fields are associated with like-named fields described below. Each is set independently to one of the values in Table 5. Any other value is undefined and should be treated as “disregard.”

A receiving application must check a field’s Confidence before consulting the associated field.

Table 5 - Confidence

Value	Confidence
0	Disregard associated field
1	Associated field is inexact (at least)
2	Associated field is inexact (at most)
3	Associated field is exact

3.7.1.3.8 SPaT Timers

The Countdown Timer indicates either exactly how much time is left until the signal phase changes, at least that much time, or at most that much time. If, however, Countdown Timer Confidence is zero, the meaning of the Countdown Timer value is undefined. Countdown Timer specifies time remaining in hundredths of a second.

Yellow Duration specifies how long a yellow light lasts on that approach, in tenths of a second. This is typically a static value in the traffic signal. Each phase or overlap may have a different value from its counterparts.

SPaT Pedestrian Object

The SPaT Pedestrian Object is presented in Figure 20. This is an optional, transient object. It will only be active when PED is active. It is encapsulated in the Approach it applies to.

Object ID			unsigned 8-bit integer
Object Size			unsigned 8-bit integer
Ped Countdown Timer Confidence	P	Ped Signal State	unsigned 8-bit integer
Time until the next signal state change (in hundredths of a second) aka Ped Countdown Timer			unsigned 16-bit integer

Figure 20- SPaT Pedestrian Object

3.7.1.3.9 SPaT Preempt Object

Figure 21 illustrates the SPaT Preempt Object. This is an optional, transient object. It will only be provided when signal preemption is in effect and encapsulated where appropriate. It is expected this will be applied per-approach. It is encapsulated in the Approach it applies to.

Object ID		unsigned 8-bit integer
Object Size		unsigned 8-bit integer
PC	Pre-emption State	high bit and unsigned 7-bit integer

Figure 21 - SPaT Preempt Object

If PC bit is on, a preemption call is in effect. The normal signal program for lanes on that approach has been overridden.

Table 6 identifies values for preemption states.

Table 6 - Preemption States

No preemption	0
Preempt Call – Delay interval	1
Advancing to clear phase	2
Timing clearance interval	3
Advancing to dwell phase	4
Servicing dwell phase	5

SPaT Location Object

The SPaT Location Object is presented in Figure 22. This is an optional object. It is a standalone object. No close tag is necessary.

The purpose of the SPaT Location Object is to convey a GPS location to recipients. This is typically a per-intersection concept.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
Latitude (LSBit = 10^{-7} decimal degrees)	signed 32-bit integer
Longitude (LSBit = 10^{-7} decimal degrees)	signed 32-bit integer
Altitude (LSBit = 1 dm)	*partially signed* 16-bit integer

Figure 22 - SPaT Location Object

3.7.1.3.10 SPaT Sensor Object

The SPaT Sensor Object (Figure 23) is an optional object, which means that it is only incorporated in the SPaT message if information about the presence of other vehicles or trains, etc., is available. It is also expected to be transient; it only appears when applicable. If no bits in the bitmask are on, the entire object becomes redundant and will be ignored.

Object ID								unsigned 8-bit integer
Object Size								unsigned 8-bit integer
		B	P	L	T	S	V	8-bit bit mask

Figure 23 - SPaT Sensor Object

V: If on, one or more vehicles detected on approach.
S: If on, there are vehicles stopped in intersection.
T: If on, high-priority train alarm.
L: If on, transit vehicle (e.g., light rail) is approaching.
P: If on, public safety vehicle is approaching.
B: If on, bus priority call in effect.

The high two bits are currently unused. V is bit 0 or the least significant bit; S is bit 1, and so on.

3.7.1.3.11 SPaT Current Time Object

The SPaT Current Time (Figure 24) is an optional object. It is encapsulated by a SPaT Intersection to convey the current RSE system time at that intersection.

This is a standalone, optional object. No close tag required.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
Year	unsigned 16-bit integer
Month	unsigned 8-bit integer
Day	unsigned 8-bit integer
Hour	unsigned 8-bit integer
Minute	unsigned 8-bit integer
Milliseconds	unsigned 16-bit integer

Figure 24 - SPaT Current Time Object

Time values reflect UTC.

The intent is that recipients may add this time to any in-scope future time offset (e.g., Countdown Timer) and, assuming both producer and consumer sync their system clocks to the same reference, know when a signal light will change.

3.7.1.3.12 SPaT Sample

A sample SPaT message captured with Ethereal is shown in Figure 25.

```
⊞ Frame 1 (131 bytes on wire, 131 bytes captured)
⊞ Ethernet II, Src: Intel_ad:0a:79 (00:02:b3:ad:0a:79), Dst: Broadcast (ff
⊞ Internet Protocol, Src: 192.168.1.4 (192.168.1.4), Dst: 192.168.1.255 (1
⊞ User Datagram Protocol, Src Port: 32773 (32773), Dst Port: 6061 (6061)
   Data (89 bytes)

0000  ff ff ff ff ff ff 00 02  b3 ad 0a 79 08 00 45 00   ........ ...y..E.
0010  00 75 00 00 40 00 40 11  b6 24 c0 a8 01 04 c0 a8   .u..@.@. .$......
0020  01 ff 80 05 17 ad 00 61  b6 78 f1 01 00 59 1c 1d   .......a .x...Y..
0030  01 07 00 02 01 00 01 02  06 00 00 00 01 03 0b 01   ........ ........
0040  00 00 00 02 30 00 78 00  00 03 03 0b 02 00 00 00   ....0.x. ........
0050  02 30 00 14 00 00 03 03  0b 03 00 00 00 02 30 00   .0...... ......0.
0060  14 00 00 03 03 0b 04 00  00 00 02 30 00 78 00 00   ........ ...0.x..
0070  03 03 0b 05 00 00 00 04  00 00 00 00 00 03 00 02   ........ ........
0080  00 01 f1                                           ...
```

Figure 25 - SPaT Sample

The highlighted section in Figure 25 represents an entire sample SPaT message. The first seven bytes on row 0030 comprise the SPaT Layer Object: **01 07 00 02 01 00 01**. The first byte is the SPaT Layer Object ID, 01. Next is the Object Size, 7; the Layer object occupies 7 bytes. The next pair of bytes, 00 02, is the SPaT Layer Type, 2. Layer ID, 01, Content Version, 00, and Format Version, 01, follow.

The layer is closed near the end of the message frame with these two bytes: **00 01**, which indicate Close Layer. The TOM Footer byte, **f1**, completes the message frame.

Here is another way of looking at a complete (and different) SPaT message. Each object (and the TOM components) is laid out on its own line. Note the nesting.

```
TOM HEADER:              f1 01 00 59 59 6f
LAYER:                   01 07 00 02 01 00 01
  INTERSECTION:          02 06 00 00 00 01
    APPROACH:            03 0b 01 00 00 00 01 33 93 a8 2b
    APPROACH:            03 0b 02 00 00 00 01 33 93 a8 2b
    APPROACH:            03 0b 03 00 00 00 04 03 ff ff 2b
    APPROACH:            03 0b 04 00 00 00 04 03 ff ff 2b
    APPROACH:            03 0b 05 00 00 00 01 33 93 a8 2b
    APPROACH:            03 0b 06 00 00 00 01 33 93 a8 2b
    APPROACH:            03 0b 07 00 00 00 04 03 ff ff 2b
    APPROACH:            03 0b 08 00 00 00 04 03 ff ff 2b
  CLOSE INTERSECTION:    00 02
CLOSE LAYER:             00 01
TOM FOOTER:              f1
```

Approach IDs in this SPaT message range from 01 to 08.

As mentioned above, the GID message is comprised of a set of objects that are hierarchically organized. The advantage of this format is that new objects can be introduced without disturbing the overall message structures and older applications that cannot parse the new object can just skip it. An example would be a "pedestrians present in crosswalk X" object that later applications could use for pedestrian protection and older versions that do not have this functionality could skip.

3.7.1.4 GPSC Message Layer

The GPSC message layer format is largely described below but the inner workings of GPS Correction, the RTCM data format, etc., are described in the Task 10 final report. .

3.8 GPSC Object IDs

Table 7 lists Object ID values to use within the GPSC message layer.

Table 7 - GPSC Object IDs

RTCM_3.0_L1_CORRECTION	2

3.9 GPSC Layer Object

The GPSC Layer (Figure 26) encapsulates one or more GPSC correction objects.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
Layer Type	unsigned 16-bit integer
Layer ID	unsigned 8-bit integer
Content Version	unsigned 8-bit integer
Format Version	unsigned 8-bit integer

Figure 26 - GPSC Layer Object

The Layer Type for GPSC is 3.

It is not unusual for RTCM correction data to change on a second to second basis. The fact that the correction data has changed is communicated through Content Version. This field is incremented on change from previously published content. The new value doesn't matter to receiving applications so much as the fact that it differs from the previous value seen. This allows the unsigned field to wrap around naturally.

Layer ID is used to tell layers in the same message frame apart. If there is only a single GPSC layer in a message then set Layer ID to 0.

The Format Versions (previously presented in Table 2) specifies which value to use for the GPSC Layer Object's Format Version field.

3.10 GPSC RTCM 3.0 L1 Correction Object

The GPSC RTCM 3.0 L1 Correction Object (Figure 27) object conveys GPS time and RTCM version 3.0 L1 correction data.

Object ID	unsigned 8-bit integer
Object Size	unsigned 8-bit integer
GPS Status	16-bit bitmask
GPS Week Number	unsigned 16-bit integer
GPS Milliseconds in Week	unsigned 32-bit integer
Length of both RTCM 1005 and RTCM 1001 data	unsigned 8-bit integer
RTCM 1005 Message Array	25 bytes of data
RTCM 1001 Message Array	0 to 101 bytes of data

Figure 27 - GPSC RTCM 3.0 L1 Correction Object

All integer fields have their bytes in network order. Object ID is set to the RTCM_3.0_L1_CORRECTION Object ID value from Table 7, previously presented. Object Size varies with the size of the RTCM data. It must be set to the size in bytes of the entire object.

GPS Status is a 16-bit bit mask composed of bit values (Table 8) and stored in network order. One should logically 'OR' them together to obtain the field value.

Table 8 - GPSC GPS Status

0x0001	Unhealthy (UDRE=7)
0x0002	Unmonitored (UDRE=6)
0x0004	Horizontal StdDev > 1.5m
0x0008	PDOP > 5
0x0010	Satellites < 5
0x0020	Local GPS Corrections
0x0040	Network GPS Corrections
0x0080	Other GPS Corrections

GPS Week Number and GPS Milliseconds in Week provide the current GPS time.

The two RTCM message array fields provide the core data of the correction object. The sum of their sizes must be stored in the 'Length of both RTCM 1005 and RTCM 1001 data' field.

Consult the RTCM Standard 10403.1 for Differential GNSS Service – Version 3[4] for details of the message arrays.

3.10.1.1.1 GPSC Sample

This sample GPSC message frame (the selected data bytes in Figure 28) is 109 bytes long (message length is 0x006d or 109).

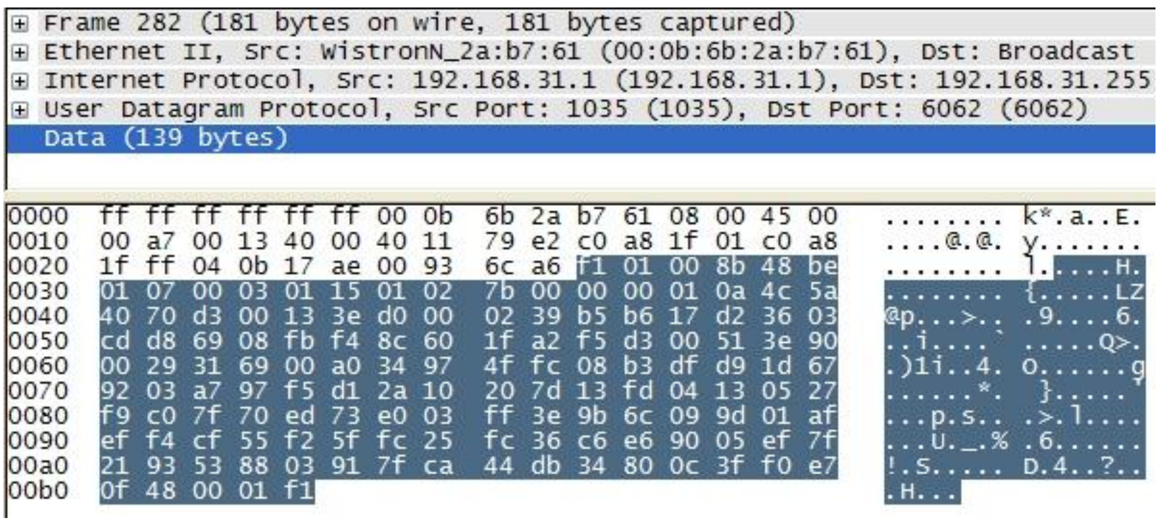


Figure 28 - GPSC Sample

The highlighted section in Figure 28 represents a complete sample GPSC message. Note the bytes on row 0030 in Figure 28: **01 07 00 03 01 15 01**. The first is the GPSC Layer Object ID, 01. Next is the Object Size, 7; the Layer object occupies 7 bytes. The next pair of bytes, 0003, is the GPSC Layer Type, 3. Layer ID, 01, Content Version, 15, and Format Version, 1, follow.

Beyond the GPSC Correction Object, the GPSC layer is terminated with the Close Layer object: **00 01**, which is then followed by the TOM Footer: **f1**.

3.10.2 Map Matching

Lane matching is the OBE process of determining the most likely vehicle travel lane. As different lanes of an intersection will have different traffic signal states, correct identification of travel-lane is required for appropriate driver warning.

3.10.2.1 Map Matching Design Overview

Map matching has two main sub-functions: the identification of the intersection that the vehicle is approaching and the matching of the vehicle to the correct approach and/or lane on the GID. Once the lane or approach has been determined, the information is sent to the Warning Algorithm for further processing.

3.10.2.1.1 Lane and Approach Matching Definitions

An approach is a combination of one or more lanes as defined in the GID that form the same movement through the intersection. The approach matching algorithm (AMA) is the OBE process of using the output of the lane matching algorithm (LMA) to determine the GID-defined approach of travel. The warning algorithm uses GID defined approach to determine if a warning is required. The AMA converts the lane matching output into a format that the Warning Algorithm can use. Further, the AMA may increase or decrease the reported certainty of the approach match by combining the confidences of lane matches if they share a common approach.

Figure 29 illustrates an example of the relationship between lanes and approaches at a typical intersection.

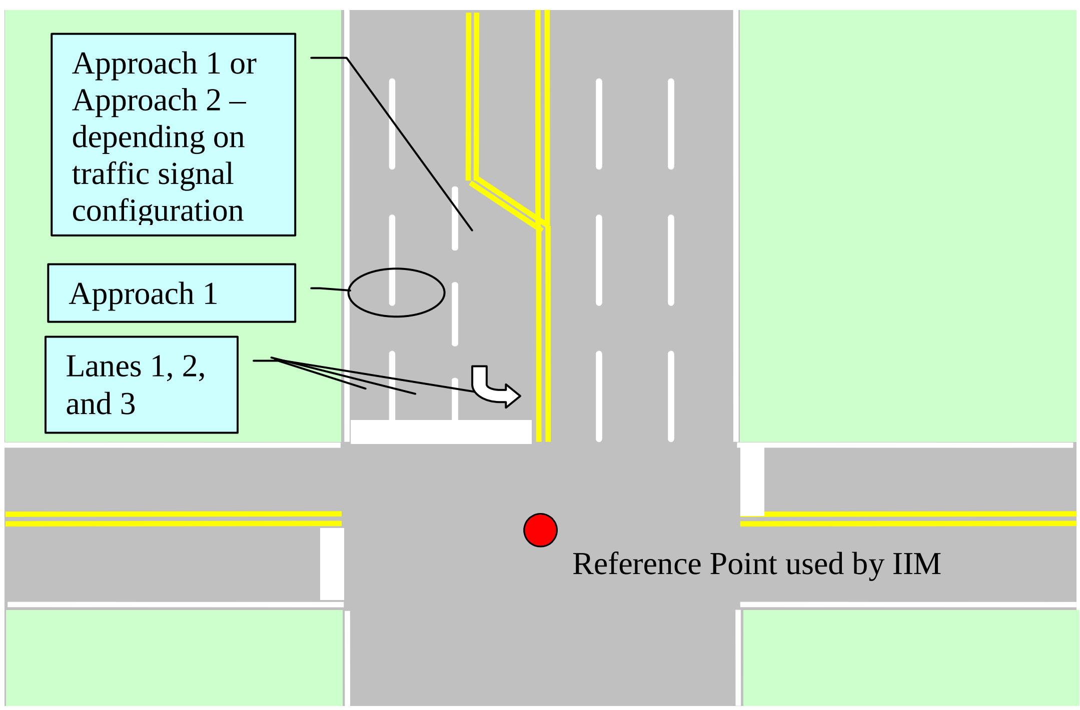


Figure 29 - Example of Lanes and Approaches at an Intersection

In Figure 29, lanes 1, 2, and 3 are defined from leftmost to rightmost from the driver's perspective. The LMA processes lanes 1, 2, and 3 independent regardless of them sharing or not sharing an approach.

If lane 3 has no right-turn specific traffic signal, lanes 2 and 3 share an approach if their traffic signals always operate simultaneously. In this case, the AMA will combine the output of the LMA for lanes 2 and 3 into a single approach match. Lane 1 could also be in approach 1 if the intersection does not have a left turn signal.

However, if this intersection was equipped with a left-turn specific traffic signal, then lane 1 could be in a different approach.

3.10.2.1.2 Intersection Identification

Intersection identification is the process of determining if the vehicle is approaching an intersection that may be capable of issuing a warning. Once the Intersection Identification Module (IIM) determines that the vehicle is approaching an intersection, other logic modules such as the Lane Matching Algorithm, the Warning Algorithm, and the Arbiter Algorithm will begin operation.

The basic logic flow of the Intersection Identification module can be seen in Figure 30 below.

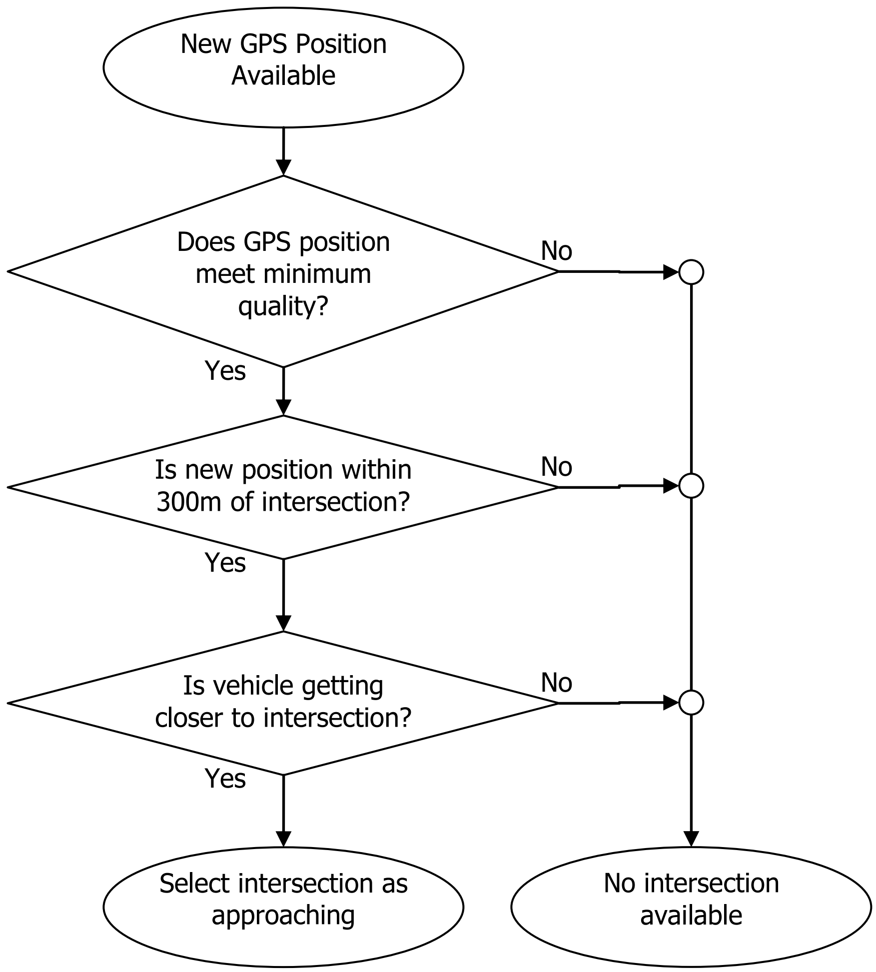


Figure 30 - Basic Intersection Identification Logic Flow

Determining if a new GPS position available: The Intersection Identification Module (IIM) operates at the frequency of the GPS position system. There is no benefit to operating the IIM more frequently than GPS because important inputs to the IIM will not have changed – greatly reducing the likelihood of any change in the outcome of algorithm.

Determining if the position reported by the GPS meets minimum quality requirements: For the IIM to operate, the position returned by the GPS must meet minimum requirements. The latitude and longitude returned by the GPS must be valid numbers for a position in North America, and the GPS calculated Horizontal Dilution of Precision (HDOP) must be below an adjustable maximum for an adjustable length of time.

Is the new position within 300m of intersection? The IIM will calculate the distance from the vehicle's new position to the intersection's Reference Point. The intersection's Reference Point represents the center point of the intersection and will either have been stored from a previous download or will be downloaded from the intersection as the vehicle approaches the CICAS-V RSE. Each distance will be calculated by an independent calculation of the great circle distance between the two points using the Haversine formula given below:

$$\Delta\hat{\delta} = 2\arcsin\left(\sqrt{\sin^2\left(\frac{\phi_f - \phi_s}{2}\right) + \cos\phi_s \cos\phi_f \sin^2\left(\frac{\Delta\lambda}{2}\right)}\right)$$

Where $\phi_{s,}$, λ_s, ϕ_f, λ_f are the latitude and longitude of vehicle and the GID being tested.

Is the vehicle getting closer to the intersection? The IIM will only identify intersections that the vehicle is approaching. Once the vehicle passes through the intersection – the intersection will no longer be identified.

Select intersection as approaching: Once an intersection is identified, other logic modules such as the Lane Matching Algorithm, the Warning Algorithm, and the Arbitrator Algorithm will begin operation.

3.10.2.1.3 Lane Matching

The Lane Matching Algorithm (LMA) is a mathematical algorithm used to determine the most likely lane in which the vehicle is driving. The output of the lane matching algorithm is the current calculated lane of travel and a percentage of confidence that the calculation is correct.

Calculation of current travel-lane begins with the calculation of the distance from the current vehicle position to nearest point on each lane segment provided by the GID. In Figure 31, an example of calculating the distance is shown. The black arrows represent lane segments as defined by the GID, the yellow arrows represent the measured distance from the vehicle to the nearest point on those lane segments.

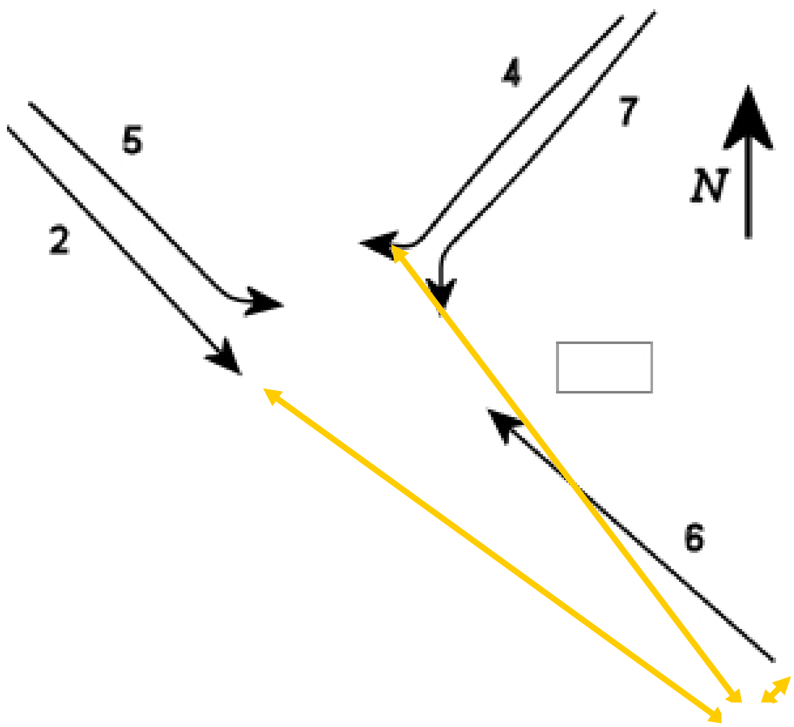


Figure 31 - Calculating Distance from Vehicle to Each GID Lane Segment

In practice, the vehicle in Figure 31 would calculate the distance to every segment available in the GID, and not just the three shown in yellow. Once the nearest GID lane segment has been identified, the GID provided lane width is used to calculate the distance to the edge of the lane, as shown below.

$$\text{Distance from Edge of Lane} = \frac{(\text{Lane Width for Segment})}{2} - (\text{Distance from Segement})$$

Figure 32 is a graphical representation of this process.

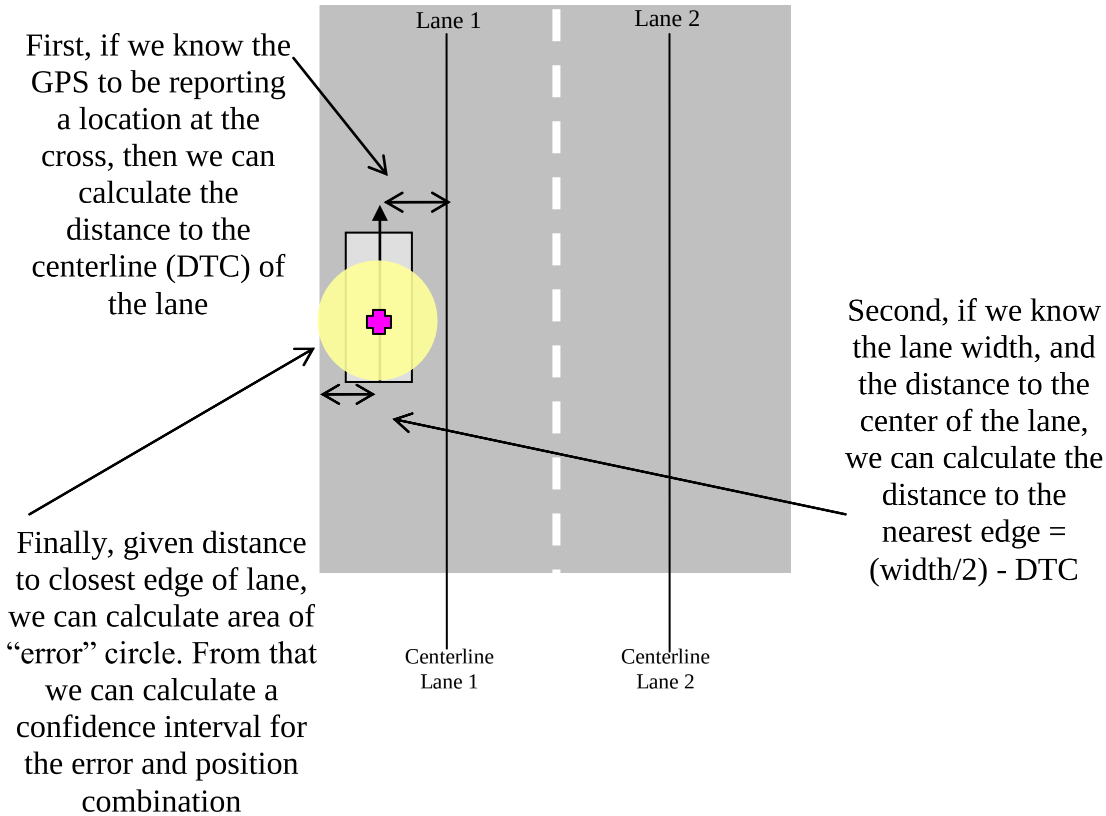


Figure 32 - Determining Lane Confidence

Once the distance from the vehicle to the nearest lane edge has been determined, the LMA will determine a confidence of that interval.

The LMA will use GST-error to determine a match confidence. GST error is a standard output available on most commercial GPS receivers. Two of the data bytes provided in GST describe an ellipse that is expected to represent a 1-sigma likelihood of containing the true position of the current position. However, the ellipse description is given relative to true north, making it subject to possible error dependent on the accuracy of vehicle heading. For this reason the LMA converts the error ellipse is converted to an error circle using:

$$\text{Error Circle Radius} = \sqrt{(\text{latitude error})^2 + (\text{longitude error})^2}\ .$$

Figure 33 is a graphical representation of this process.

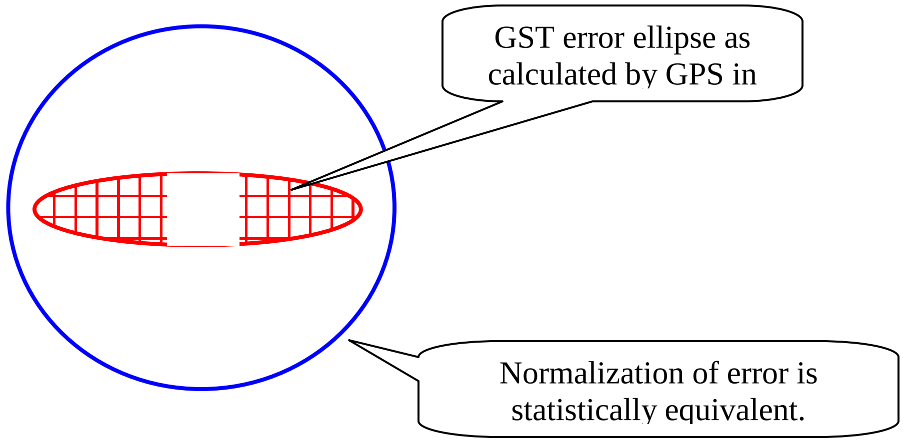


Figure 33 - GPS Provided Error Ellipse

Once the conversion to an error circle and the calculation of distance to the nearest lane edge is complete, a confidence value can be calculated. The ratio of the distance to nearest lane edge (D.N.E) to the length of radius of the error circle is the confidence sigma number of the closest lane.

$$\textit{if}\ \text{D.N.E} \geq \text{Error of Error Circle then, Sigma of Confidence} = \frac{\text{Distance from Nearest Lane Edge}}{\text{Radius of Error Circle}}$$

A geographical representation of this is provided by Figure 32.

3.10.2.1.4 Approach Matching

Once the LMA has calculated a confidence for any possible lanes, the Approach Matching Algorithm will use the confidences for lanes further away from the vehicle to increase or decrease approach likelihood for each GID provided approach. The final approach match and approach confidence will be output to the warning algorithm and be matched with the signal phase and timing provided by the SPaT message from the intersection RSE.

3.10.2.2 Map Matching Testing

3.10.2.2.1 Prototype Logic Development

The Intersection Identification (IIM) and the Lane Matching Algorithm (LMA) were initially tested during Task 8 by the development of a Matlab script that reproduced the core logic described above. The prototype IIM and LMA logic was developed using Matlab scripting language running under Matlab 2006b interfacing to the serial output of a 5 Hz CSI Minimax DGPS on an IBM T42 laptop computer. For the prototype work, U.S. Coast Guard DGPS correction signals intended for the boating community were substituted for the RTK solutions being developed for CICAS-V project. A screen capture from the Lane Matching prototype is presented in Figure 34.

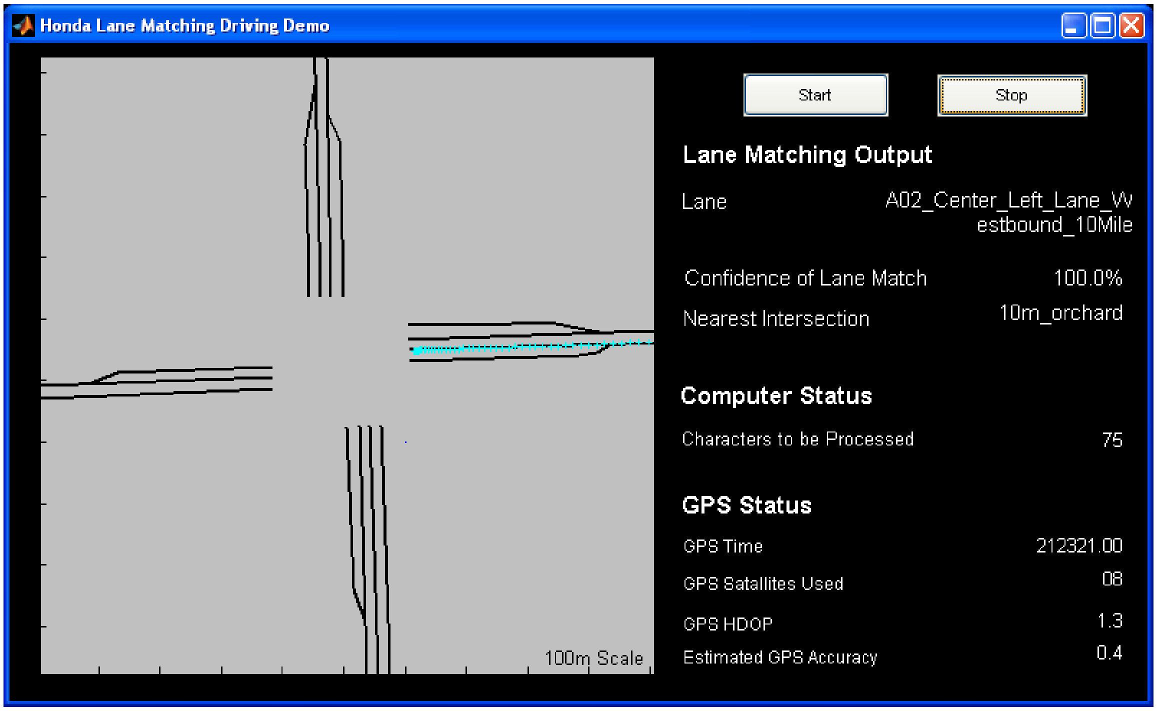


Figure 34 - Screen Capture from Task 8.7 Lane Matching Prototype

The core logic developed during this prototype effort was used as the basis for furthering the research work carried out as part of Task 10. The Matlab scripting language source code developed as part of Task 8 was used as possible model reference implementation for the development of an integrated prototype as part of Task 10.

3.10.2.2.2 Testing

Over a period of months, thousands of real time drives were conducted using the lane matching prototype logic in Farmington Hills, Michigan and Palo Alto, California to verify that the lane matching logic could meet a variety of lane matching conditions. These included fast and slow lane changes, differing GPS conditions, early and late lane shifts, and off-map conditions.

Results from the testing were processed and used to identify possible improvements in the map matching algorithm, such as grouping of lanes, using the lane center as the reference for wider right lanes and using variable lane width. These improvements were integrated into ongoing Task 10 research.

3.10.3 CICAS-V GPS Corrections Generation and Vehicle Positioning System

The overarching goal of the CICAS-V GPS correction generation and vehicle positioning subsystems is to design and prototype a vehicle positioning system. The purpose is to achieve real-time sub-meter vehicle positioning near CICAS-V intersections for CICAS-V equipped vehicles at relatively low cost while using commercial off-shelf hardware.

The prototype design is dependent on the availability of Road Side Equipment (RSE) at CICAS-V signalized intersections that have a local Global Positioning System (GPS) base station receiver. The GPS base station receiver is configured to compute correction factors for the GPS Satellite signals that are needed to make the position result from GPS position estimation algorithms match the base station's known (surveyed) fixed location. This locality of scope contrasts with other popular correction techniques, such as the Wide Area Augmentation System (WAAS), which has ground reference stations spaced approximately 500 miles apart, and, therefore, computes corrections on a regional basis. The field test results conducted to date at real intersections indicate significantly higher real-time vehicle positioning accuracy when compared to the position accuracies obtained through WAAS and DGPS corrections based vehicle positioning systems. For example, at the CICAS-V traffic intersection located in Farmington Hills, Michigan, absolute real-time vehicle positioning errors in the order of less than 0.5m is consistently achieved using the CICAS-V test vehicles.

3.10.3.1 Design Overview

Figure 35 shows the local DGPS correction generation and broadcast subsystem installed at a traffic point of interest, such as a controlled intersection. The GPS receiver shall be configured in base-station mode, where it will compute corrections to GPS satellite signals for other moving (vehicle-mounted) GPS receivers in its vicinity. The correction information shall be encoded in Radio Technical Commission for Maritime Services (RTCM)-standardized format, such as the RTCM Recommended Standards for Differential GNSS (Global Navigation Satellite Systems) Service as defined by the Special Committee (SC) 104 on Differential Global Navigation Satellite Systems (DGNSS). For brevity, the corrections data message format shall be referred to as either RTCMv3.0 or RTCMv2.3, depending on which SC-104 release of the "Recommended Standards for Differential GNSS" being used.

The RTCMv3.0 message format used in the CICAS-V system design consists of single frequency (L1) GPS information. The TMT recommended using single channel (L1) NovAtel OEMV ("OEM-Five") receivers for CICAS-V vehicles, and single channel (L1) NovAtel OEM4 or OEMV receivers for the base station, assuming that accurate reference survey information is available to use in the intersection base station GPS receiver..

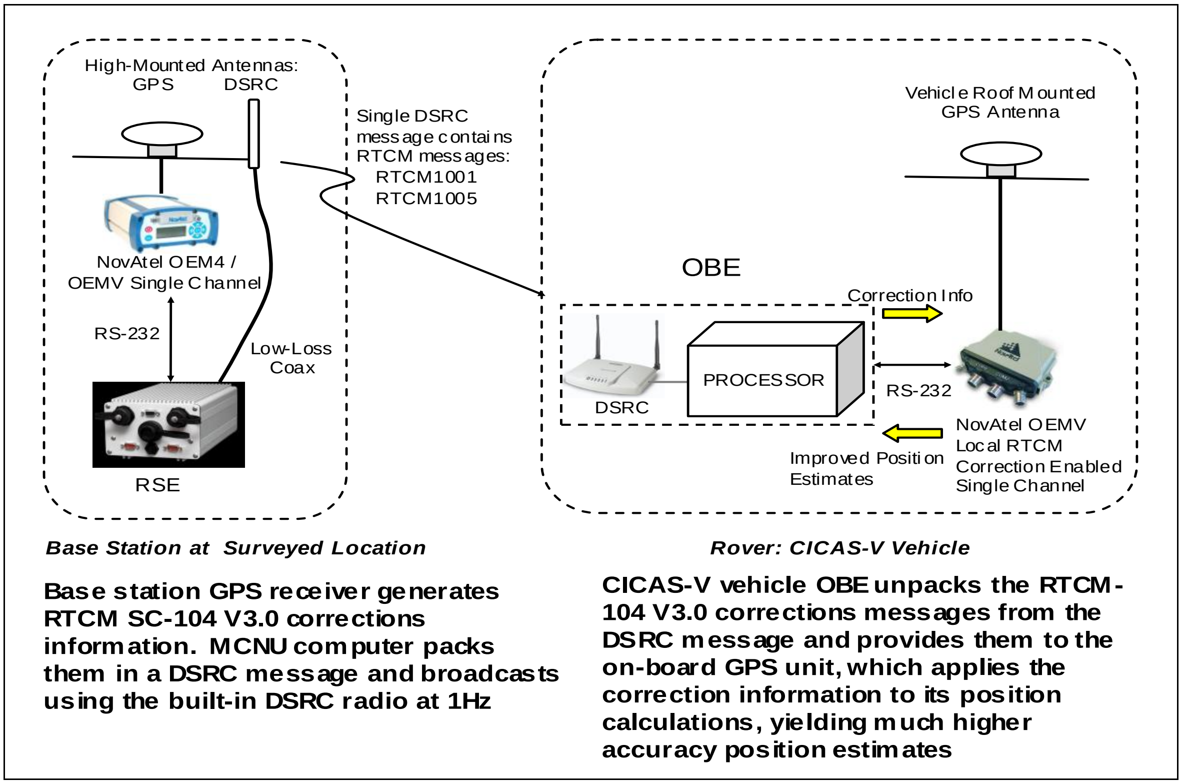


Figure 35 - Positioning Correction Equipment

3.10.3.2 Local GPS Correction Generation in the RSE

The prototype software program for broadcasting the local GPS corrections messages in the Road Side Equipment (RSE) is called **Gypsy**. The current version of this prototype is discussed below, along with other design recommendations for its future functionality.

As a single executable program, **Gypsy** receives RTCM data, verifies the integrity of the messages and streams the messages over a DSRC wireless link to the vehicle OBE system in pairs of two as follows: base-station information (RTCM message 1005), plus one set of per-satellite correction information (RTCM message 1001). The Gypsy software design makes the simplifying assumption that the attached base station GPS receiver is pre-configured to calculate GPS corrections and transmit the corrections data in the RTCM format through an RS-232 serial link. Such GPS receiver configuration can easily be done through vendor-provided software utilities or interactive commands at a terminal emulation program such as Microsoft HyperTerminal. With this assumption, Gypsy may simply "listen" for the RTCM messages on the serial port.

For broadcast over DSRC radio, the Gypsy program uses the built-in 802.11p radio module. The wireless software stack of the DENSO WSU supports the current DSRC upper layer protocols, namely 802.11p (draft standards) and 1609.3 and 1609.4 protocol standards. The DSRC radio was configured for the following transmission characteristics for GPS corrections:

- Bitrate: 6Mbps
- Periodicity = 1Hz
- Tx Power: 17 dBm (Maximum allowed by the current WSU hardware)
- Channel: 178
- Bandwidth: 10MHz
- Tx & Rx sharing Antenna 1

The sequence diagram below in Figure 36 shows the GPS correction generation procedure in the RSE.

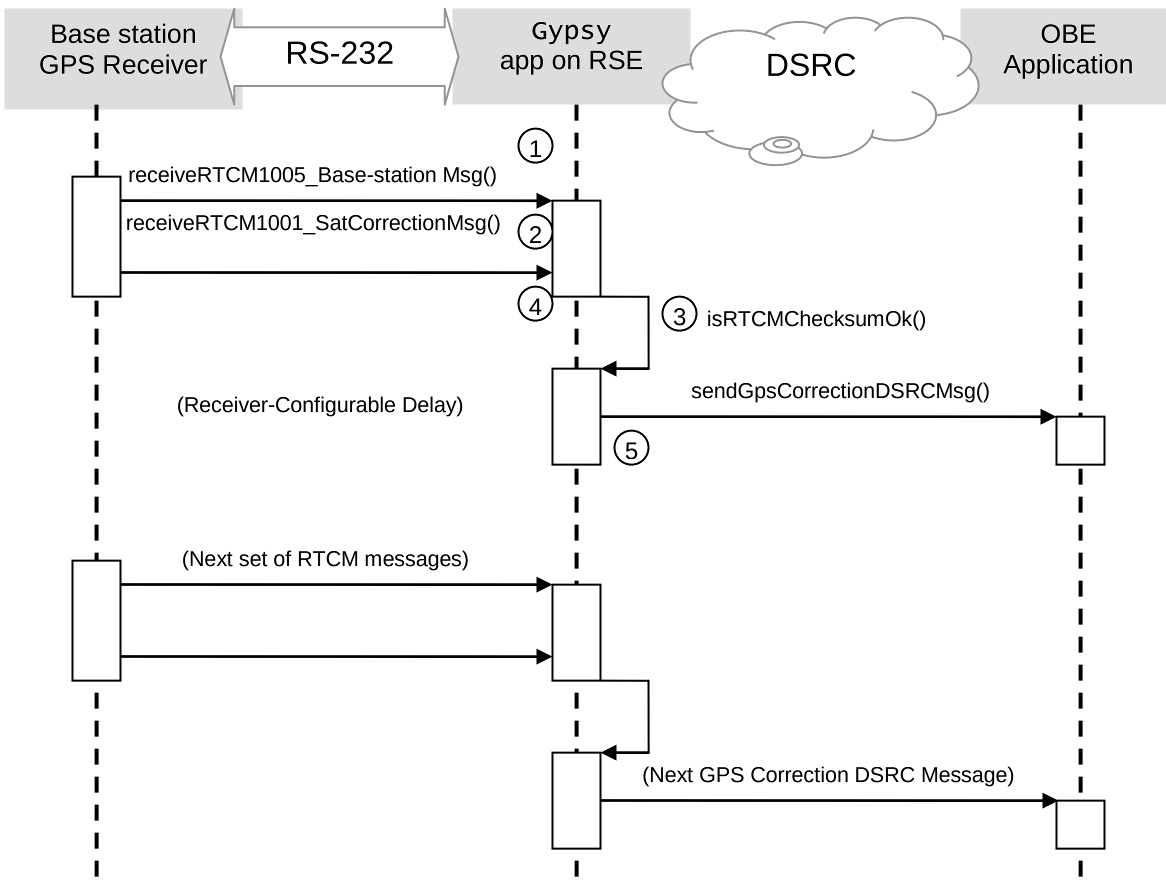


Figure 36 - GPS Correction Sequence Diagram

The RSE waits for the beginning of a RTCM Transport Link Layer frame for the base-station information message (type 1005 in RTCM v3.0).

The RSE receives the number of bytes indicated in the Transport Link Layer header.

If an error occurs, the RSE software will re-parse the recently-received bytes to look for the next possible RTCM Transport Link Layer message frame start (this reduces the number of “thrown away” received bytes that could contain the frame start sequence, shortening synchronization times).

If the RTCM Transport Link Layer checksum passes, the software uses a similar method to acquire and verify an RTCM per-satellite corrections data set message (type 1001 in RTCM v3.0). If the checksum does not pass, the RSE software will wait for the next RTCM Link Layer frame.

When both the base-station information RTCM message and the per-satellite corrections RTCM message are received correctly, a DSRC message is sent with the complete RTCM Transport Link Layer message frames for both messages.

3.10.3.3 CICAS-V Vehicle Positioning in the OBE

The sequence diagram for the vehicle positioning system is shown in Figure 37.

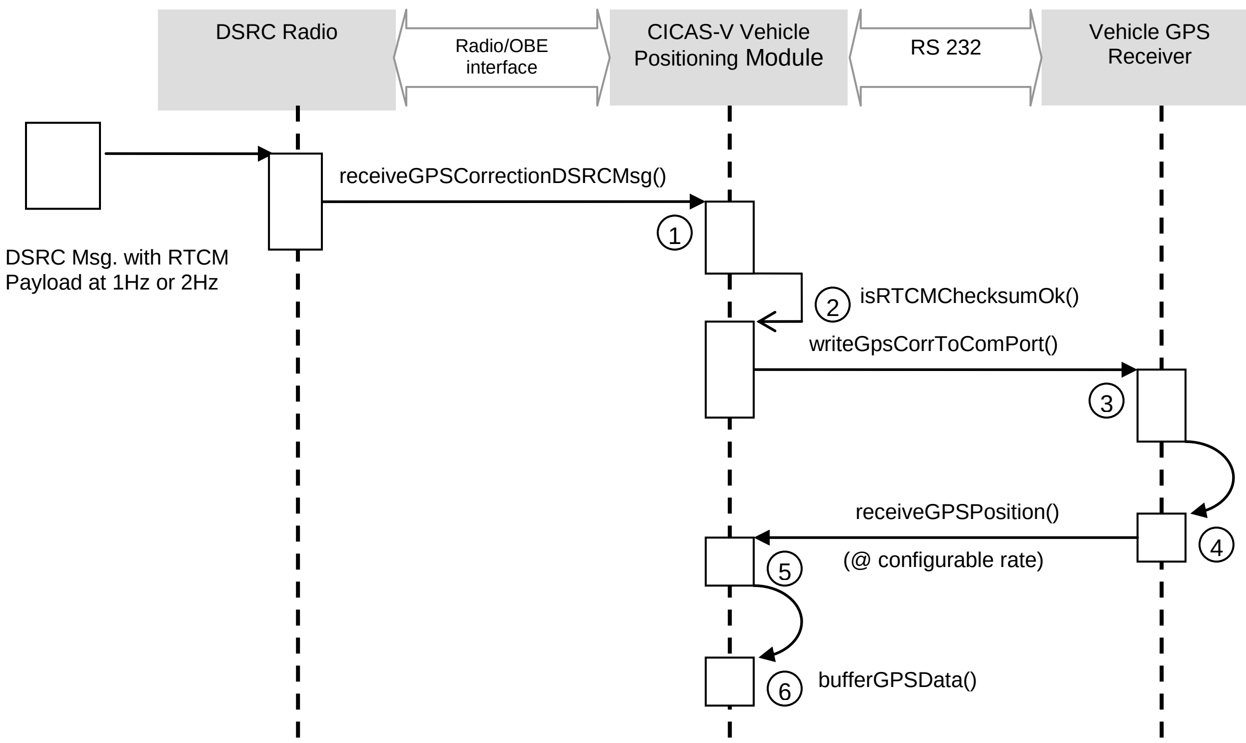


Figure 37 - Sequence Diagram for the Positioning Correction

The DSRC radio receives the broadcast message containing the RTCM payload and the base station GPS status and passes the message payload to the CICAS-V application.

CICAS-V application checks the checksum of the RTCM message 1005 and 1001 sequentially. If there is an error, the CICAS-V application raises an internal error condition and drops the DSRC packet.

If there are no error conditions, CICAS-V application buffers the RTCM payload internally, and writes RTCM payload data into the vehicle GPS receiver's COM port in the same incoming order, i.e., RTCM1005 followed by RTCM1001.

Vehicle GPS receiver receives and uses local corrections and generates position solution at a configurable rate (e.g., 10Hz) and makes the solution available at its COM port.

CICAS-V vehicle positioning module receives the GPS position solution from the vehicle GPS receiver in the NMEA standards through a serial communications link, which may be the same physical link used to provide the RTCM data to the GPS receiver in step 3.

CICAS-V vehicle positioning module decodes the NMEA string and buffers GPS information such as; latitude, longitude, heading, velocity, GPS time, number of satellites, position precision, etc. into internal CICAS-V application memory.

The RTCM1001 corrections provide per-satellite GPS pseudoranges and carrier phase measurements so that the on-board (moving) GPS receiver can compute its position estimate with much higher accuracy and reliability. The RTCM 1001 form of L1-only correction information provides a good accuracy improvement with rather modest communications requirements and impact on GPS receiver workload. For example, only 101 bytes are required per RTCM 1001 binary message that includes range corrections for 12 satellites, and is often smaller according to the number of visible GPS satellites in the current constellation. See the following document section for further detail on correction types and formats.

3.10.3.4 GPS Correction Message Formats

The amount of correction data that has to be broadcasted in the DSRC link is dependent on the RTCM version used and on the number of visible satellites. For example, the RTCM v2.3 format (RTCM, 2001) requires about 4800 bits per second (bps) to broadcast dual-frequency code and carrier-phase observations or observation corrections of 12 satellites. Similar information content can be transmitted using 1800 bps in the newer RTCM v3.0 format (i.e., for 12 visible satellites, v2.3 requires 372 bytes to transmit data, whereas RTCM v3.0 requires only 8+7.25*12 bytes). The RTCM v3.0 is primarily designed to support Real-Time Kinematic (RTK) operations that normally require broadcasting relatively large amounts of information, and generally implies highly sophisticated forms of correction analysis and error removal. However, the L1-only subset of the RTCM v3.0 format provides good performance improvements for modest system resource requirements, and works well even with moderately-priced receivers.

A minimum of two RTCM SC 104 standard messages are required from the RSE to support local differential L1 solution correction for onboard GPS receivers:

RTCM1001 Message: Contains the satellite observations, in particular the single frequency (L1-only) GPS pseudorange and carrier phase measurements, as derived by the base station GPS receiver by comparing the position estimate determined from current satellite pseudorange observations with the surveyed fixed location of the base station antenna. The base station "works backwards" to compute corrections to the satellite pseudoranges that would yield a much more accurate position estimate. Other roving GPS receivers in the surrounding area will generally face the same set of inaccuracies in the GPS satellite pseudorange observations, so when they apply these pseudorange correction factors to their own observations, they too will be able to significantly reduce the errors and obtain a more accurate position estimate. The size of the RTCM1001 message is a variable, depending on the number of satellites (**Figure 38**). The maximum size of this binary message is 101 bytes for 12 satellites (see reference [0]), that is including the 3 bytes of header and 3 bytes of footer introduced by the transport link layer.

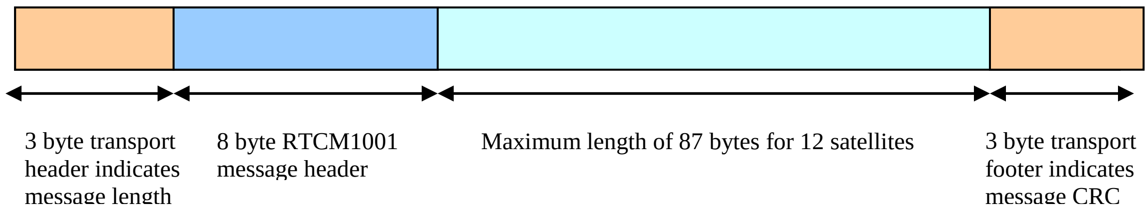


Figure 38 - RTCM1001 Message

RTCM1005 Message: Contains the base station coordinates, in particular it provides the earth-centered, earth-fixed coordinates (ECEF) of the antenna reference point (ARP) for the intersection base station. The size of this binary message is fixed at 25 bytes, including the 3 bytes of header and 3 bytes of footer introduced by the transport link layer. See Figure 39. It is not mandatory for the CICAS-V application to decode the data elements of this message in the OBE. However, it is required that the CICAS-V application check any data bit-wise errors (data integrity) in the RTCM1005 correction message using CRC values included in the message structure. The last three bytes of the RTCM1005 message payload contains its 24-bit CRC value.

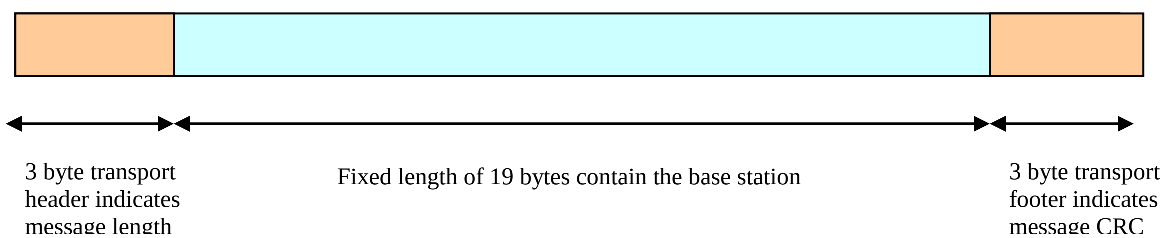


Figure 39 - RTCM1005 Message

3.10.3.5 DSRC Message Format for RTCM Payload

Table 9 illustrates the proposed byte order for the DSRC message format that contains the RTCM corrections and base station GPS information.

Table 9 - DSRC Message Format for the RTCM Corrections

DSRC GPS LOCAL CORRECTION MESSAGE FOR CICAS-V				
	Version: 1.0.6		*NOTE: All multibyte integer fields are in network order (ie. big endian).*	
	Date Modified: 1/23/2008			
	Field	**Description**	**Values**	
Byte \ Bit	8 (msb) · 7 · 6 · 5 · 4 · 3 · 2 · 1 (lsb)			
1	Message Type	Identifies this as a Transportation Object Message	0xF1	HEADER
2	Message Version	Increments when the TOM Framework (including Object Tag) format changes	0..255	HEADER
3, 4	Message Length	Total length of message in bytes including header and trailer	unsigned 16-bit int	HEADER
5, 6	CRC-16	CRC-16 computed from Message Type thru Message Termination Flag (inclusive) with this field set to 0	unsigned 16-bit int	HEADER
7	Object ID	Indicates type of object	**LAYER**	
8	Object Size	Total size of object in bytes	0..255	
9, 10	Layer Type	Indicates type of layer	unsigned 16-bit int (GID=1,SPAT=2,**GPSC=3**,...)	
11	Layer ID	Layer identifier indicates which one; it is possible to have more than one of each type in a frame	0..255	
12	Content Version	Increments when layer content changes	0..255	
13	Object ID	Indicates type of object	**RTCM_3.0_L1_CORRECTION**	
14	Object Size	Total size of object in bytes	0..255	
15, 16	GPS Status	GPS Base Station Status (bitmask)	0x0001=Unhealthy, 0x0002=Unmonitored, 0x0004=(reserved), 0x0008=PDOP > 5, 0x0010=Satellites < 5, 0x0020=Local GPS Corrections, 0x0040=Network GPS Corrections, 0x0080=Other GPS Corrections, …	
17, 18	GPS Week Number	Official GPS week for the GPS epoch associated with the included GPS Correction(s)	unsigned 16-bit integer (0..65,535)	
19, 20, 21, 22	GPS Milliseconds in Week	Number of Milliseconds since GPS midnight (Sunday/Monday) for the GPS epoch associated with the GPS Correction messages	0..604,799,999	
23	Length of both RTCM 1005 and RTCM 1001 data	Length in bytes of the 1005 and 1001 RTCM message "blobs." Length includes 25 bytes (fixed) for RTCM 1005 data and 14 bytes plus 7.25 * (number of satellites) for RTCM 1005 data [RTCM 3.0 protocol].	25..126	
24 to 48	RTCM 1005 Message Array	Copy of full RTCM 1005 "transport link layer" base station message data, always 25 bytes long	(binary data bytes)	
49 to <= 149	RTCM 1001 Message Array	Copy of full L1-only RTK observable RTCM 1001 "transport link layer" message data (variable length)	(binary data bytes)	
<= 150	Object ID		**CLOSE**	0
<= 151	Which Object to Close		**• *LAYER***	1
<= 152	Message Termination Flag	Indicates end of message, same value as Message Type	0xF1	FOOTER

3.10.3.6 Preliminary Field Test Results

The data logs from the base station and vehicle dual-frequency GPS receivers (NovAtel-OEM4) were used to generate the 2D ground truth (reference) solution in the post-mission processing to compare the 2D real time vehicle positioning accuracies at CICAS-V intersections. The GPS error, with respect to post-processed base line reference, is in the order of less than 0.5m when there is a reliable local GPS correction messaging system available. The GPS differential correction age increases when the vehicle looses DSRC correction messages. Currently, the OEMV has a DGPS correction age timeout parameter set to 60 seconds, during which time the OEMV receiver still maintains sub-meter level vehicle positioning accuracy. If the vehicle doesn't receive the DSRC correction messages consecutively during the timeout period, the vehicle GPS receiver changes its position solution from the DGPS solution to a single-point GPS solution, and the vehicle positioning accuracy degrades significantly. The value of 60 seconds is not defined by any explicit constraints, but represents a reasonable trade-off between uninterrupted performance despite temporary signal loss (due to obstructions or signal noise) and low-latency response to those few expected situations in which the vehicle would obtain the best possible correction from an alternative source.

When local RSE corrections are available, CICAS-V vehicles use local corrections to generate high precision L1 RTK Float GPS solutions. When local corrections are not available, CICAS-V vehicles use satellite based WAAS corrections to generate differential GPS solutions. The OEMV receiver can be pre programmed to automatically change the receiver positioning mode to WAAS from L1 RTK Float when there are no RSE generated local corrections.

Figure 40 shows the CICAS-V GPS corrections (GPSC) DSRC Message Coverage (Orchard Lake and West 10 Mile Intersection).

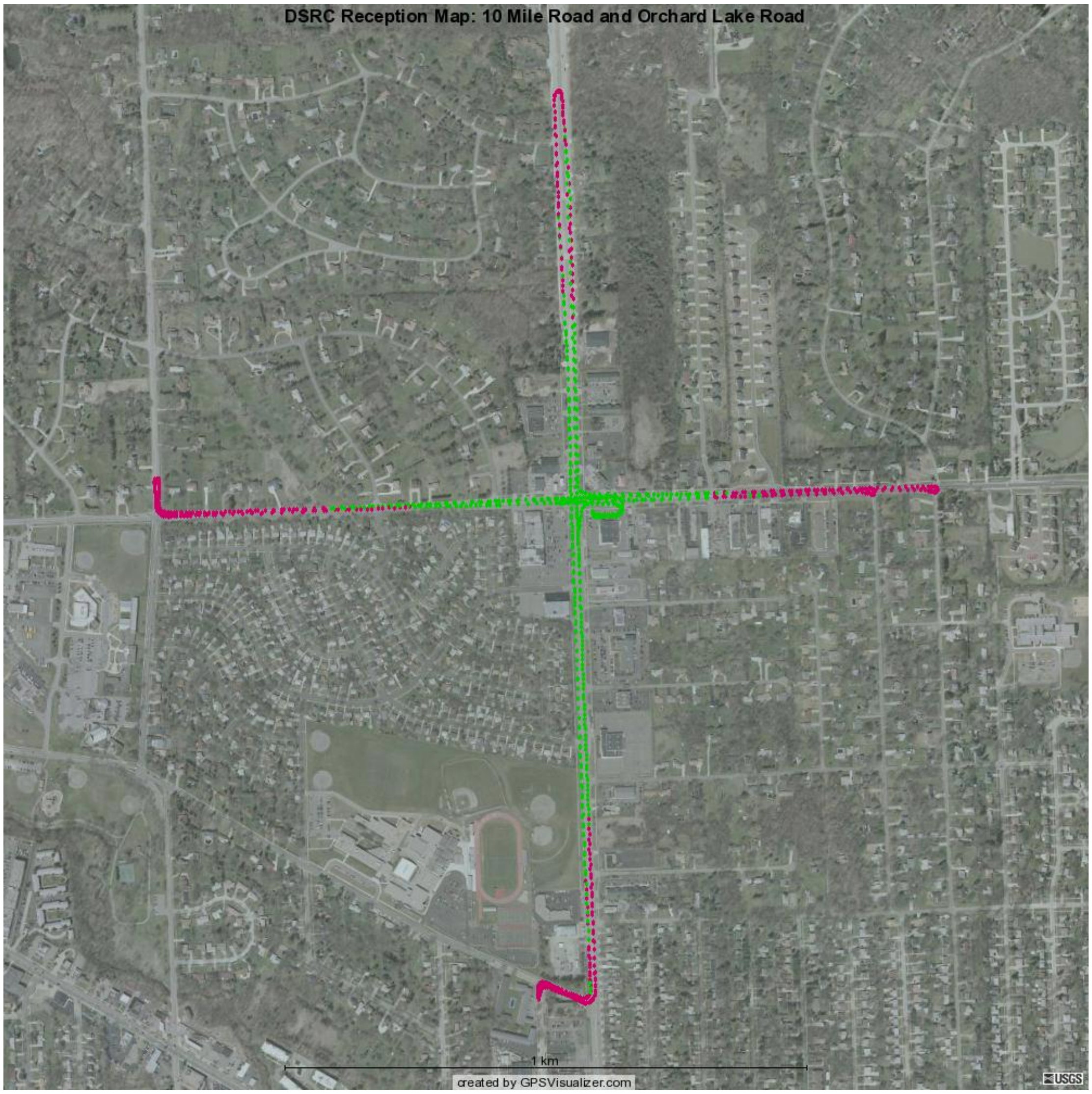


Figure 40 - CICAS-V GPS Positioning Corrections Message Coverage (Image Courtesy of U.S. Geological Survey)

Figure 41 shows the availability of High Accuracy GPS Positioning at the Orchard Lake and 10-Mile Intersection, Farmington Hills, Michigan. Dots of both colors represent the location of a vehicle as it drove around the intersection. The data is shown at 1 Hz output rate. The Green dots represent locations where the vehicle GPS is operating in the high accuracy mode using the GPS data sent from the intersection Magenta dots represent the locations where the vehicle GPS is operating in a standalone mode without CICAS support. This can be standalone GPS or WAAS-enabled GPS. The results show at 1Hz GPSC message transmission frequency, the receiver switches to high Accuracy positioning mode (L1 RTK) from WAAS or Single Point modes within four seconds upon receiving the first epoch of RTCM corrections. The data shows that the real-time vehicle positioning accuracy with respect to the post processed solution typically converges to ~0.6m on average as soon as the OEMV receiver switches to High Accuracy positioning mode. After 4-6 seconds, the error in the High Accuracy mode converges to ~0.3m on average.

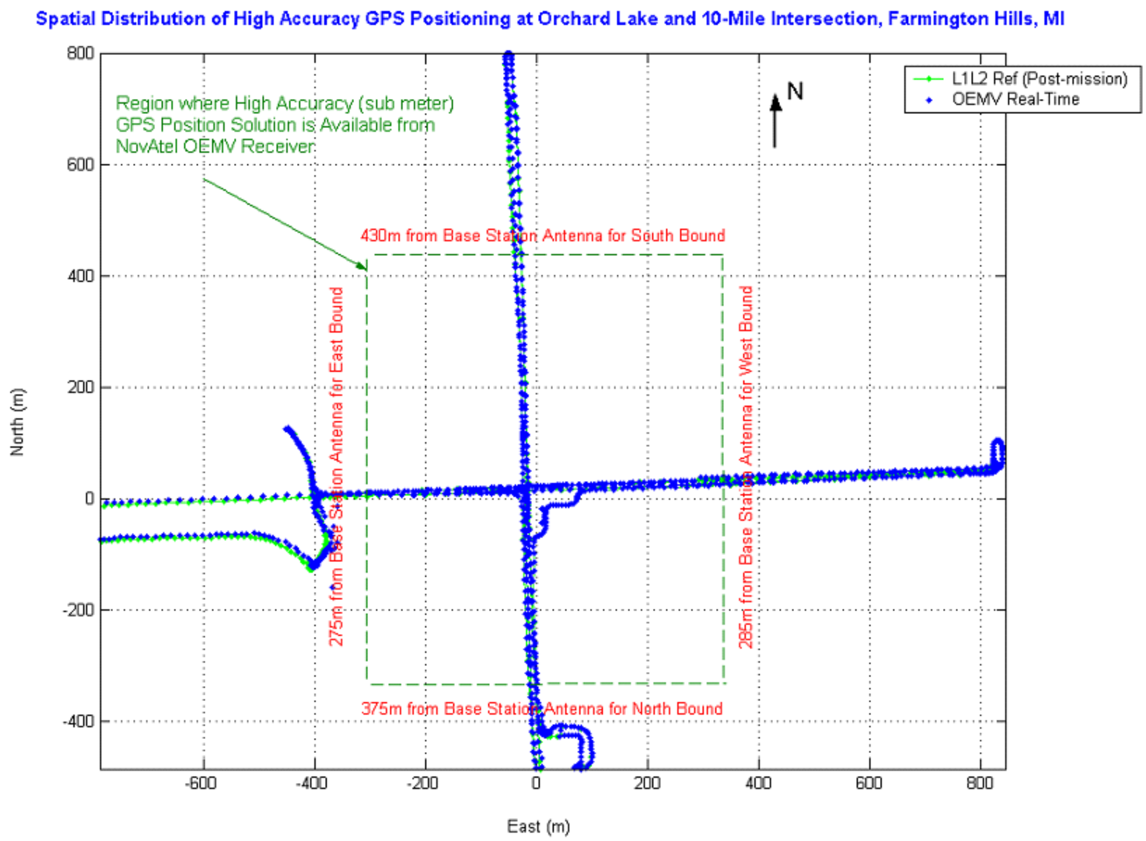


Figure 41 - Availability of High-Accuracy GPS Positioning at the Orchard Lake and West 10 Mile Road Intersection

The data plot in Figure 42 shows the transition distance and error difference from WAAS or Single Point Solution to L1 RTK Solution for the West Bound Approach at the Orchard Lake and 10 Mile intersection in Farmington Hills, MI.

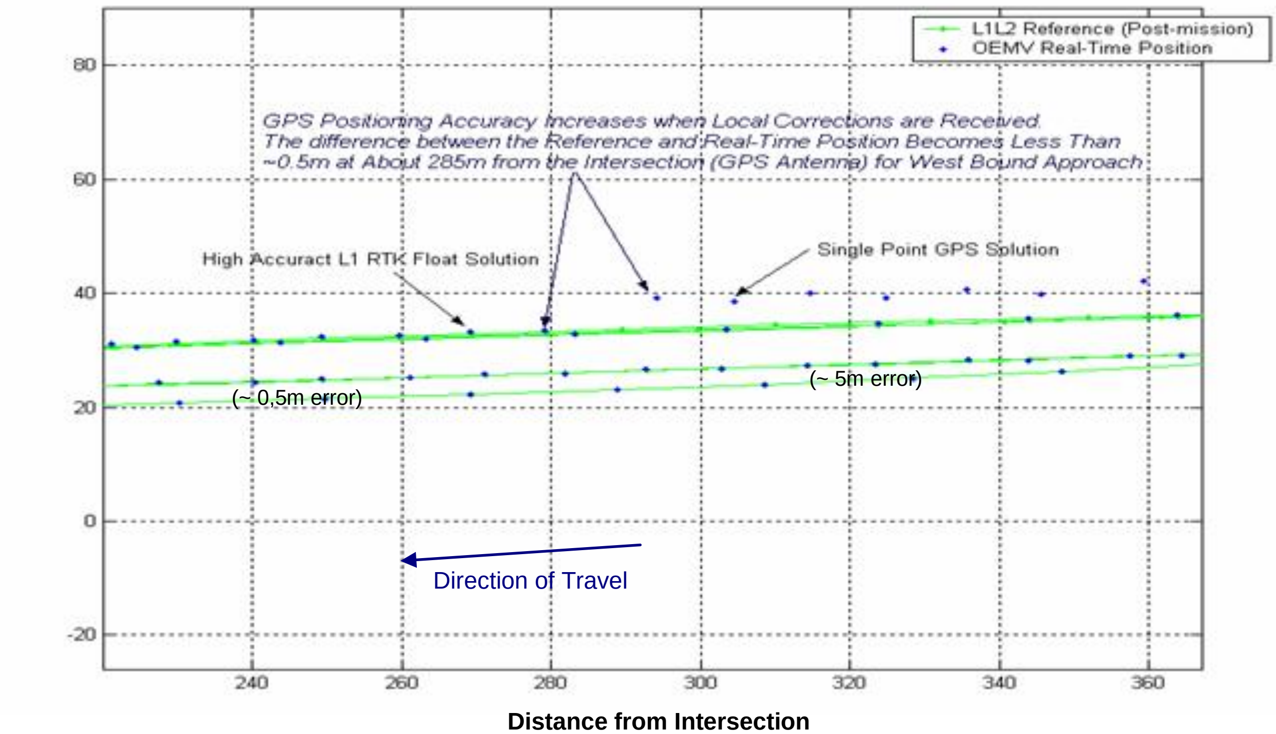


Figure 42 - Transition Distance and Error Difference from WAAS or Single Point Solution to L1 RTK Solution

Figure 43 shows the 2D Error for Real Time OEMV Data vs. Post Processed Reference Data at the Orchard Lake and West 10-Mile intersection in Farmington Hills, Michigan. The average Real-Time Vehicle Positioning Error in the High Accuracy L1 RTK mode is less than 0.3m. The data plot also shows OEMV GPS receiver's solution convergence times when the positioning mode transit from WAAS/Single Point to High Accuracy L1-RTK. The data shows that the average receiver switching time from WAAS/ Single Point to L1 RTK is 3 - 4 seconds at 1Hz. It is expected that the switching delay could be reduced up to ~2 seconds by increasing the GPSC Tx rate to 2Hz or more.

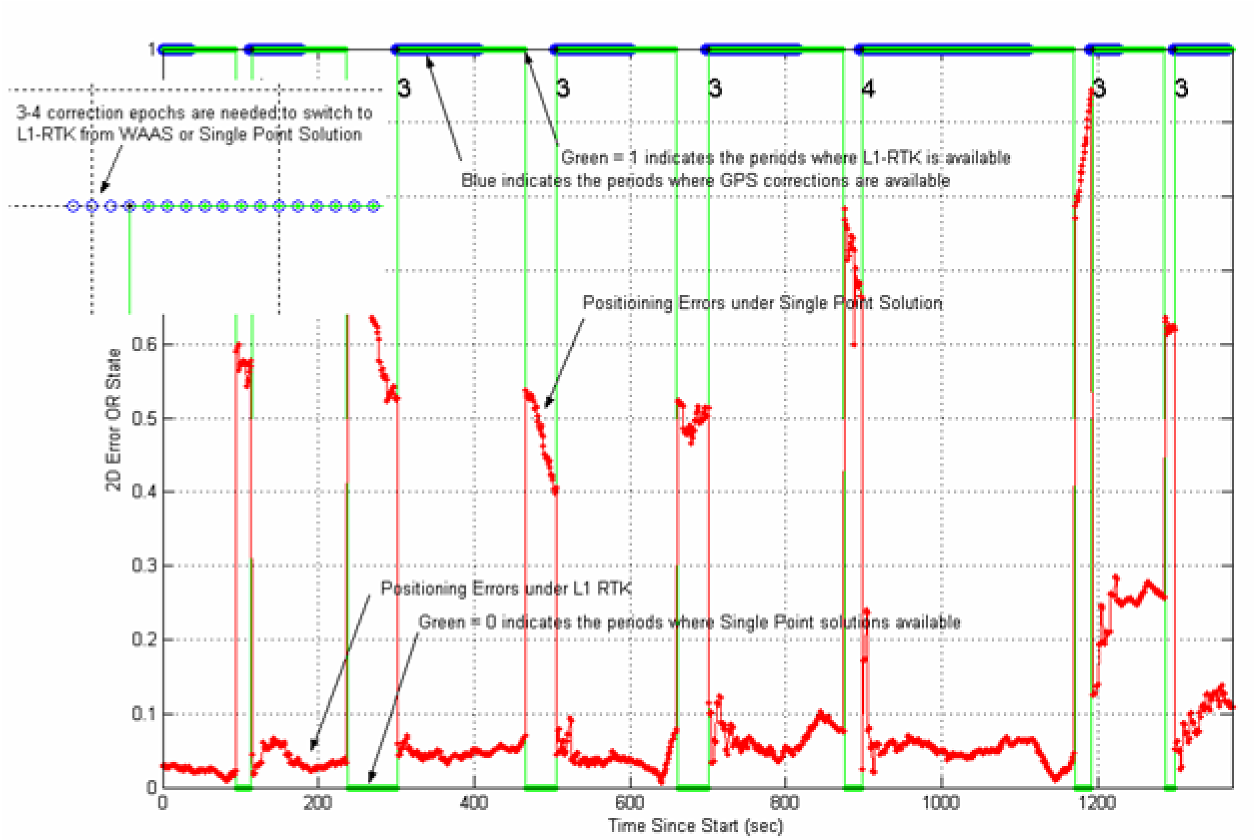


Figure 43 - 2D Error for Real Time OEMV Data vs. Post Processed Reference Data

As can be seen from the measurements, the initial positioning error of approximately 5 m will reduce to around 0.5 m after the reception of 4 correction messages and stays stable throughout the intersection approach. If a dual frequency receiver such as the NovAtel L1/L2 receiver is used in the vehicle and in the intersection, this error can be further reduced to a few centimeters. This line of inquiry was not pursued aside from initial tests since a positioning error of 0.5 m or less was attained with the single channel receiver, which is sufficient for the correct functioning of the CICAS-V application.

3.10.4 CICAS-V Warning Algorithm

The CICAS-V program has developed an algorithm framework for warning the driver of a CICAS-V equipped vehicle approaching a CICAS-V equipped intersection of an impending violation of a stop light or a stop sign. The algorithm framework was developed with a high degree of flexibility in mind to be able to adapt the algorithm to the results of Task 3.2 (naturalistic data collection at intersections). The goal of those activities was to develop an algorithm that, in conjunction with the CICAS-V system, is effective in compelling drivers to stop before entering the intersection crash box. The warning algorithm is designed to determine if a vehicle will violate a control device, using the stop bar as a reference. The algorithm will determine if a violation is likely to occur and provide sufficient time after a warning is issued for the driver to stop the vehicle before the crash box to reduce the number of crashes.

This section presents the details of the algorithm used by CICAS-V to decide under which conditions the driver should be warned for red light violations. This algorithm also applies to stop sign violations. Figure 44 illustrates the placement of the algorithm within the CICAS-V in-vehicle system.

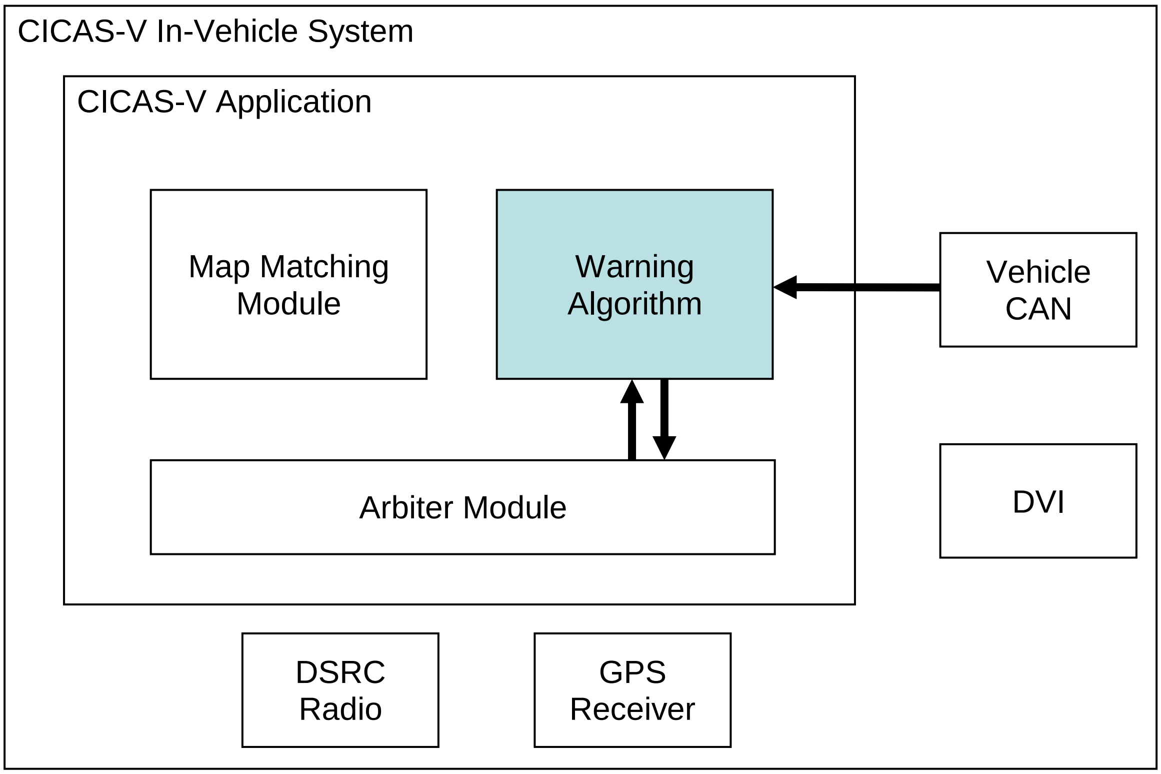


Figure 44 - Warning Algorithm's Placement within the CICAS-V In-vehicle System (Note: only inputs/outputs directly related to the warning algorithm are shown)

3.10.4.1 Interfaces

This section describes the warning algorithm interface in detail. Table 10 explicitly lists the input parameters necessary for the periodic execution of the CICAS-V warning algorithm as well as its outputs at the completion of each cycle. As shown Table 10, the warning algorithm takes inputs from Vehicle CAN and Arbiter modules and outputs it result to the Arbiter module. The units of input and output parameters are specified along with the direction of dataflow in the warning algorithm.

Table 10 - Warning Algorithm Interface Control Table

Input / Output	Data	Units (If Applicable)	Where Data is Coming from/Going to
Input	Brake Switch	N/A	Vehicle CAN
Input	Driver Intended Brake Value	N/A	Vehicle CAN
Input	Vehicle Speed	Meters per second (m/s)	Vehicle CAN
Input	Distance to Stop Line	Meters (m)	Arbiter (from Map Matching Algorithm)
Input	Phase	N/A	Arbiter (from DSRC - Signal Phase and Timing Message)
Input	Time to Phase Change	Seconds (s)	Arbiter (from DSRC - Signal Phase and Timing Message)
Input	Yellow Duration	Seconds (s)	Arbiter (from DSRC - Signal Phase and Timing Message)
Output	Warning State	N/A	Arbiter

Scope

The warning algorithm should be able to identify when the driver may be about to violate a red light. When the on-board warning algorithm can assert that the vehicle will not stop by the stop bar based on vehicle speed, distance to stop bar and current deceleration, the algorithm will pass along the warning status to the Arbiter module.

The Warning Status is a Boolean value. Each time the algorithm is run, either a warning is given or no warning is given. The driver intended brake value is a parameter that indicates how hard the driver intends to brake. It is included for completeness but was not used in the algorithm.

3.10.4.2 Overall Algorithm Structure

Only the overall structure of the algorithm is presented in Figure 45. The specifics on how different algorithm variables are to be determined are presented in subsequent sections.

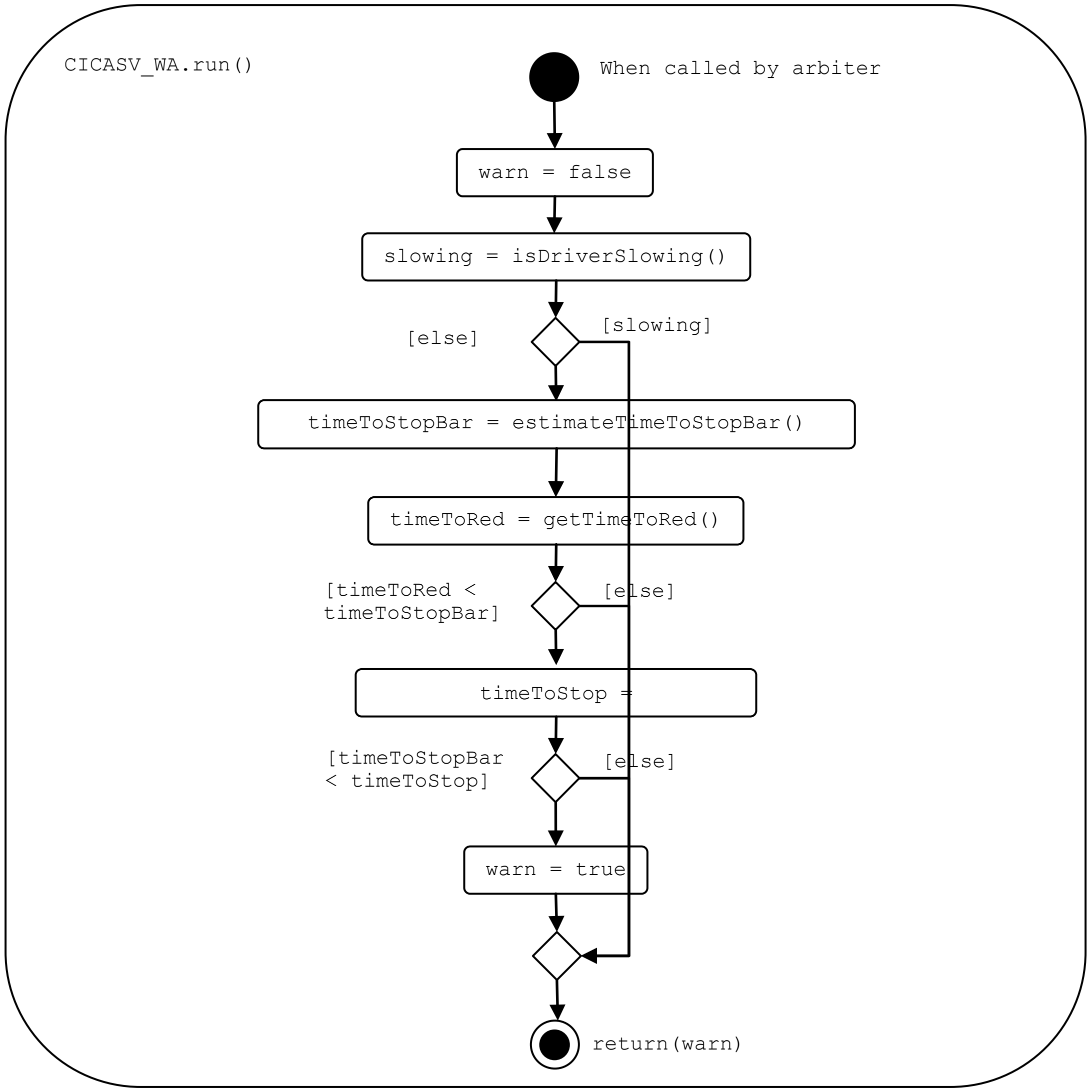


Figure 45 - CICAS-V Warning Algorithm Main Flow of Events

The following flow of events is executed whenever requested by the arbiter module.

If the driver is applying the brakes then he/she is not warned. In this case, from a human factor prospective, the driver is already considered aware of the situation since he/she is maneuvering the vehicle to slow down its speed and there is no need to warn him/her.

Otherwise, the "time to stop bar" is calculated as the estimated time necessary for the vehicle (with unchanged vehicle dynamics – currently only speed is taken into account) to reach the intersection stop line.

The "time to red" is determined as the amount of time until the intersection traffic light will switch to red. The "time to red" is always zero for a stop sign controlled approach.

If the "time to stop bar" is shorter than the "time to red," the driver is not warned. In this case, the vehicle is estimated to arrive at the intersection stop bar before the traffic light will switch to red. Therefore, the driver will not be warned.

Otherwise, the "time to stop" is determined as the time that it will take for the vehicle to come to a complete stop when performing a "hard" braking maneuver. This "time to stop" time is a summation of driver brake reaction time (RT), vehicle stopping time, and other system delay times (e.g., interface delay). Results from the Human Factors Task 3 will be used to determine the assumed driver brake RT and stopping time values.

If the "time to stop" is shorter than the "time to stop bar," the driver is not warned. In this case, the driver is estimated to have enough time left to stop the vehicle (performing a "hard" braking maneuver) before he/she is at risk of an intersection collision due to a violation.

Otherwise, the driver is warned. In this case, a high percentage of drivers (as determined by the Human Factor Study group's research) will have enough time to bring the vehicle to a full stop such that they avoid any potential intersection collision due to a violation. Therefore, the driver must start "hard" braking immediately.

The warning status bit is set high.

Return the warning status Boolean value.

Driver Slowing Check

The logic shown in Figure 46 checks if the driver is slowing down the vehicle indicating, from a human factors perspective, that he/she is aware of the situation and should not be warned.

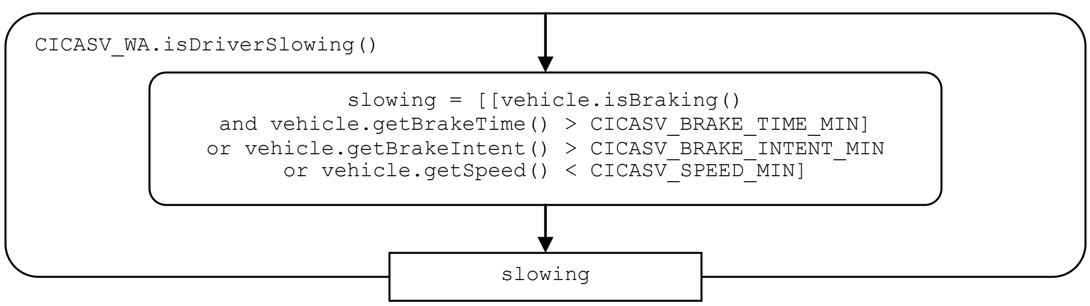


Figure 46 - Checks if Driver Is Slowing the Vehicle

This logic also checks if the vehicle speed is below a minimum threshold. This check allows a CICAS-V vehicle to very slowly cruise through a red light without receiving a warning, which 100 car study show happens frequently with Right Turn on Red (Sudweeks et al, Task 3.1 final report). This low speed will indicate the driver's attentiveness. The minimum speed limit will also allow the "turning right at red light" scenario to be carried out without a warning given.

In order to judge that the vehicle is slowing, the logic checks whether the brakes are applied and if they have been for a minimum period of time. If the driver is braking lightly and has been for some time, this should be recognized as slowing. This section of logic also checks a value that indicates that the driver has a strong intent to stop the vehicle. This value is constructed with data such as brake pressure, brake torque, brake pedal position, brake switch, and extended brake switch as available by each OEM. The driver is classified as attentive if this value is above some minimum threshold. If the driver is braking hard, this should be immediately recognized as slowing.

NOTE: The minimum speed threshold will need to be modified during testing and may vary for traffic light and stop sign intersections. It should be configurable.

NOTE: Brake time is not a value that comes directly from the Vehicle I/F. The brake switch must be monitored to determine this value. Brake time is defined as zero whenever the brake is not pressed and the cumulative brake press time (since the last transition) whenever the brake is pressed. The timer may be implemented as most convenient.

3.10.4.3 Estimate Time to Stop Bar

The logic shown in Figure 47 estimates the time that it will take the vehicle to reach the intersection stop line with unchanged dynamics (e.g., speed, etc).

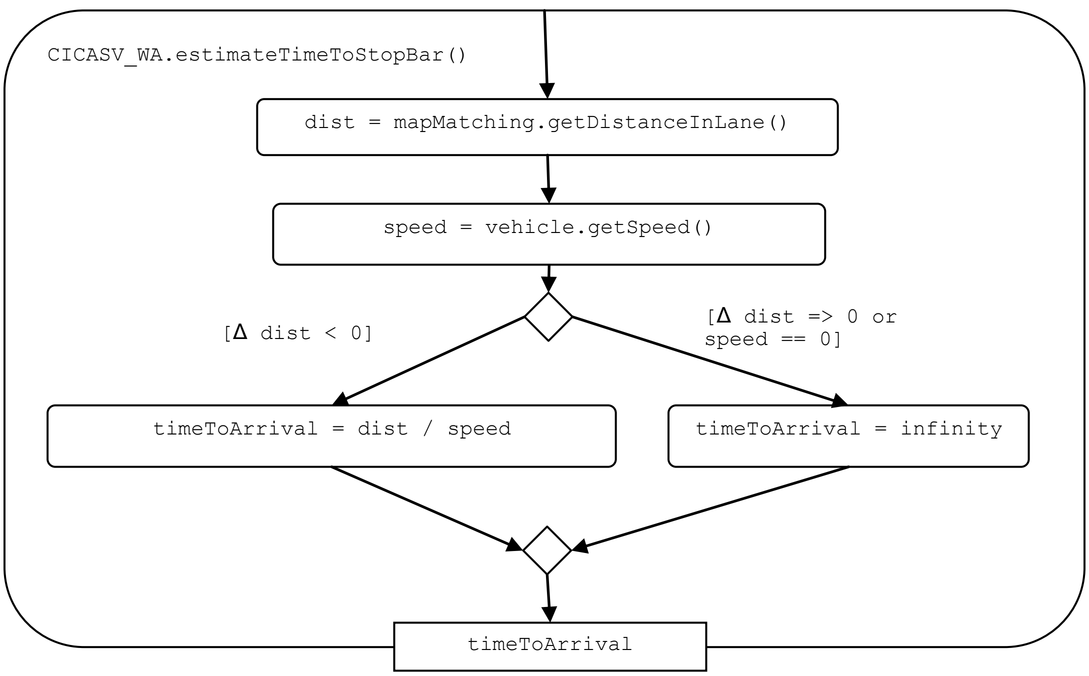


Figure 47 - Estimates the Time Before the Vehicle Arrives at the Intersection Stop Line

The formula used here assumes the vehicle will follow its current lane at a constant speed. Distance to stop line will be measured through the use of in-lane distance from true lane geometry broadcasted in the intersection map. This distance value should be provided by the map matching algorithm which can calculate an in-lane distance from the GID map. More complex formulas, using acceleration and/or steering angle, could be used to the increase accuracy of the “time to arrival” calculation. This should be considered further only if testing indicates the need for a more accurate "time to stop bar" prediction.

NOTE: As indicated in Figure 47, above, there will need to be a mechanism for determining if the vehicle's distance to stop line is increasing or decreasing. If it is decreasing, the left branch in Figure 47 is followed. If it is increasing, not changing or the vehicle's speed is 0, the time to arrival will be infinity since the vehicle is not approaching an equipped intersection. This could be as simple as maintaining a record of the distance on the last loop iteration and subtracting the current from the previous distance, and could be implemented however convenient.

NOTE: Acceleration is not used as it is a hard value to measure because of the effect of gravity on accelerometers (when the vehicle is traveling up or down a grade) used in most production vehicles.

NOTE: The distance to stop bar value used above doesn't come directly from the Map Matching module; it is passed along through the Arbiter Module. The Map Matching module, however, is mentioned as it is the original source of the value.

3.10.5 Calculate Time to Red

The logic presented in

Figure 48 calculates the time it will take for the traffic light to become red.

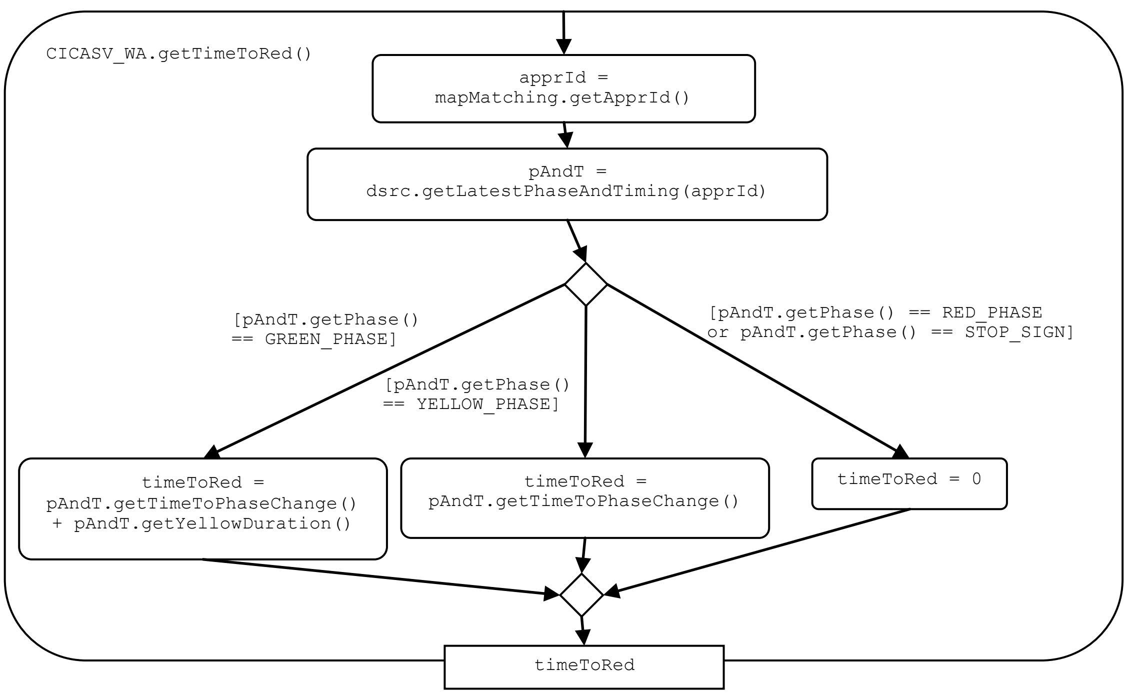


Figure 48 - Calculates Time it Will Take for the Intersection Traffic Light to Become Red (Also Handles Stop Signs)

The information received from the latest (and not yet expired) "Signal Phase and Timing" DSRC message is used. Phase, time to phase change, and yellow duration are extracted from the DSRC message for the vehicle's approach (as decided by the map matching module). The time to phase change is included in the message and not calculated by the vehicle software.

NOTE: For CICAS-V Pilot FOT, the only phases which will be considered are: GREEN, YELLOW, and RED. Any other flashing or arrow phases will not be supported.

NOTE: The approach ID used above doesn't come directly from the Map Matching module; it is passed along through the Arbiter Module. The Map Matching module, however, is mentioned as it is the original source.

NOTE: The Signal Phase and Timing data doesn't come directly from the DSRC radio as indicated above. Instead, it is passed through the Arbiter module. The DSRC Radio, however, is mentioned as it is the original source of the data.

3.10.5.1 Calculate Time to Stop Vehicle

This logic (Figure 49) calculates the time to stop the vehicle performing a "hard" braking maneuver.

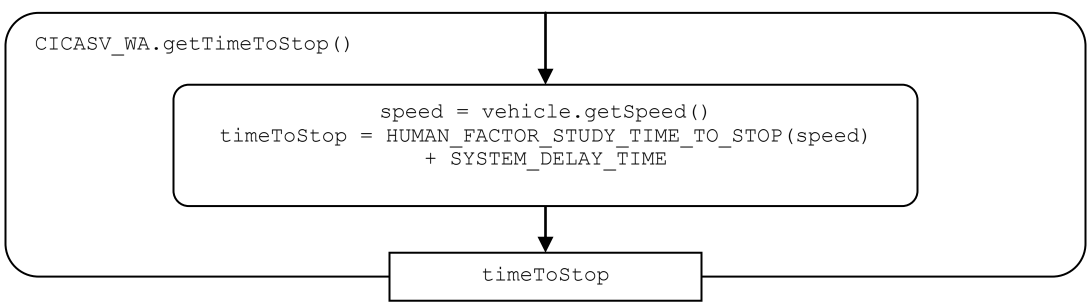


Figure 49 - Calculate Vehicle's Time to Stop When Performing a "Hard" Braking Maneuver

This time is extracted directly from the CICAS-V Task 3 Human Factors Research efforts. This stopping time takes into consideration the driver's braking RT, vehicle stopping time, and other system delay times. This time may also be a function of the vehicle's current velocity. While this time is in principle different from driver to driver and from vehicle to vehicle, the CICAS-V Human Factor Study is trying to identify a warning/timing approach which will be effective for a wide range of drivers across vehicle models. The stopping time values given by the Human Factor group should be well within the capabilities of the driver and vehicle under most road and environmental conditions, i.e. dry, clear, day.

Vehicle stopping time and driver reaction time (derived from Task 3 Human Factor group efforts) are added to form the primary "time to stop" component of the timing algorithm. Additionally, hardware and software system delay times must also be added to construct a complete "time to stop" value.

3.10.5.2 Time to Stop Tradeoff

In general, the "time to stop" should have the goals of:

- Providing a high percentage of drivers enough time to stop so that they can avoid a signal violation which results in an intersection collision.
- Minimizing the alerts perceived by the driver as "too early".

3.10.5.3 Current Warning Algorithm Limitations

The following material and Table 11 describe the assumptions that were made in the initial implementation of the warning algorithm. It should be noted that the warning algorithm developed in Task 8 was needed only to support evaluation of the system design during the early stages of the CICAS-V development. The research conducted in Task 3 led to modifications of the initial algorithm that are reported in the Task 3.4 report. (Neale et al, Task 3.4 final report). The information presented in this section and in the table below are provided to document the work that was done in Task 8.

- The algorithm has to strive toward minimizing false positive and nuisance alerts while maximizing correct warnings.
- False Positive alert: warning given though objectively no reason for alert
- Nuisance alert: when driver perceives there is no reason for the alert even though the alert was issued at an appropriate time during an intersection approach (too early, too late and unwanted)

Table 11 - Assumptions Made in Initial Warning Algorithm Implementation

No.	Assumption	Rationale
1	The warning algorithm will be enabled once all relevant CICAS message data have been received and the driver is within *tbd* meters from the stop bar. At this time the intersection ahead icon will signalize to the driver that the intersection is CICAS-V equipped. The algorithm will be deactivated if the credibility (or validity) of the algorithm inputs is below the allowable level.	Avoidance of alerting the driver to the presence of a CICAS-V equipped intersection unless the functioning of the warning can be guaranteed. (The intersection ahead icon will typically be activated at a pre-determined distance from the intersection and when we can place the vehicle on the GID.)
2	The warning algorithm is designed to	We cannot guarantee that the

No.	Assumption	Rationale
	determine if a vehicle will violate a control device, using the stop bar as a reference. The algorithm will determine if a violation is likely to occur and provide sufficient time after a warning is issued for the driver to stop the vehicle before the crash box to avoid possible collisions.	vehicle comes to a stop before the stop bar because we cannot exactly determine the position (based on GPS error) of the vehicle relative to the stop bar. Given these assumptions – the distance between the stop bar and the crash box allows late alerts to be issued– reducing nuisance alerts - while maintaining the system objective crash reduction (by getting vehicles to stop short of the crash box).
3	The ability to alert will be disabled when braking is detected. (Currently we have not defined a threshold). The alert will be enabled again if the driver steps off the brake and the vehicle speed is not below the speed threshold. Consequently, the system does currently not evaluate whether braking is sufficient to bring a vehicle to a stop. I suggest that the braking level (to suppress alerts) needs to 'near' the threshold to stop short of crash box.	We assume that this shows that the driver is paying attention to the traffic situation. This will also minimize nuisance alerts. Ease of implementation, especially defining a homogenous braking threshold across various vehicle platforms
4	If a driver brakes, he/she is assumed to be attentive to the forward scene and therefore no warning is needed.	The main focus of the warning is to bring the driver's attention to the actual driving situation (impending violation) so that the driver can make the proper decision to stop before entering the intersection This will also minimize nuisance alerts.
5	The system will not alert on rolling stops at stop signs or right turns on red where the driver slows down below a speed threshold before crossing the stop bar. This speed threshold will result from work in Task 3.2	Minimization of nuisance alerts. CICAS is not primarily aimed at changing driver's normal intersection approach behavior

No.	Assumption	Rationale
		(for example: rolling over the stop bar to check for cross traffic)
6	No multi-stage warning is required.	The warning is a late warning and there is no time for a multistage warning The intersection ahead icon can provide some insight towards a future warning.
7	The alert will be disabled after a vehicle has passed the stop bar,	Avoidance of warning in the intersection box where providing a warning could result in the driver stopping in adjacent traffic lanes increasing the opportunity for a crash
8	After a driver has been warned, there will be a *tbd* period of time which the warning is suppressed.	Drivers should not be warned multiple times at the same approach. The *tbd* period of time will likely be on the order of 30 seconds.

There are several scenarios where a violation occurs but a late alert or no alert is issued. These are presented in Table 12. All scenarios assume that a CICAS-V equipped vehicle approaches a CICAS-V equipped intersection.

Table 12 - Scenarios Where a Violation Occurs But a Late Alert or No Alert Is Issued

No	Scenario	Assumption
1	A vehicle approaches a controlled (stop sign or traffic signal) intersection. The driver brakes lightly because of another vehicle, e.g., and runs the intersection control due to the fact that the warning will be delayed or disabled because the driver has the foot on the brake. (The task 3.2 data analysis will give us more insight into how frequent this scenario would occur.)	4,8
2.	The vehicle approaches a signalized CICAS-V equipped intersection. The vehicle comes to a stop before the stop bar but then starts up again (while maintaining a speed less than the threshold up to the stop bar) while the traffic signal is still red and goes straight through the intersection. This situation could arise if the driver were behind a large truck and could not see the signal. If the truck decides to run the red light, the driver might assume it is ok to go, as well.	4,5,8
3	The vehicle approaches a signalized intersection on the rightmost lane. The vehicle brakes to slow down but does not come to a full stop. The vehicle continues past the stop bar and makes a right turn.	4,5,8
4	The vehicle approaches a stop sign controlled intersection below the speed threshold without braking and continues through the intersection without stopping.	5
5	The vehicle is traveling in a lane with a green indication and then changes lanes near the intersection to a lane with a red light before violating. The warning could be delayed or would not occur.	11
6	The driver has received a warning recently (most likely due to violating the last intersection approached) and will not be warned or will be warned too late at the upcoming intersection (if the warning suppression time has not yet passed).	12

4 References

National Highway Traffic Safety Administration. (2005). Traffic Safety Facts 2005. (Report No. DOT HS 810 631). Washington, DC: National Highway Traffic Safety Administration.

Campbell, B., Smith, J., & Najm, W. (2004). *Analysis of Fatal Crashes Due to Signal and Stop Sign Violations*. (Report No. DOT HS 809 779). Washington, DC: National Highway Traffic Safety Administration.

NTCIP Guide Version 2.0, http://www.ntcip.org/library/documents/html/chapter4.html.

IEEE P802.11p/D30 - Draft Standard for Information Technology, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, July 2007.

Draft SAE J2735 Dedicated Short Range Communications (DSRC) Message Set Dictionary - Revision 22.

Enhanced Digital Mapping Project Final Report (2004). IVI Light Vehicle Enabling Research Program (Cooperative Agreement No. DTFH61-01-X-00014). Washington, DC, U.S. Department of Transportation.

Crash Avoidance Metrics Partnership (2007). Cooperative Intersection Collision Avoidance System – Violation (CICAS-V) Concept of Operations Interim Release v1.03. June 29, 2007.

RTCM Standard 10403.1 for Differential GNSS (Global Navigation Satellite Systems) Services - Version 3 Developed by Radio Technical Commission For Maritime Services (RTCM) Special Committee No. 104, October 27, 2006.

U.S. Department of Transportation
ITS Joint Program Office-HOIT
1200 New Jersey Avenue, SE
Washington, DC 20590

Toll-Free “Help Line” 866-367-7487
www.its.dot.gov

FHWA-JPO-10-068

U.S. Department of Transportation

Research and Innovative Technology Administration