Minnesota Department of Transportation

RESEARCH SERVICES

Office of Policy Analysis, Research & Innovation

Research Implementation of the SMART SIGNAL System on Trunk Highway (TH) 13

Henry X. Liu, Pricipal Investigator
Department of Civil Engineering
University of Minnesota

February 2013

Research Project
Final Report 2013-06

Your Destination...Our Priority

To request this document in an alternative format, please contact the Affirmative Action Office at 651-366-4723 or 1-800-657-3774 (Greater Minnesota); 711 or 1-800-627-3529 (Minnesota Relay). You may also send an e-mail to ADArequest.dot@state.mn.us.

(Please request at least one week in advance).

Technical Report Documentation Page

1. Report No. MN/RC 2013-06	2.	3. Recipients Accession No.
4. Title and Subtitle Research Implementation of the SMART SIGNAL System on Trunk Highway (TH) 13		5. Report Date February 2013
		6.
7. Author(s) Henry X. Liu, Jianfeng Zheng, Heng Hu, and Jie Sun		8. Performing Organization Report No.
9. Performing Organization Name and Address Department of Civil Engineering University of Minnesota 500 Pillsbury Drive SE Minneapolis, MN 55455		10. Project/Task/Work Unit No. CTS Project #2010054
		11. Contract (C) or Grant (G) No. (C) 89261 (G) 178
12. Sponsoring Organization Name and Address Minnesota Department of Transportation Research Services 395 John Ireland Boulevard, MS 330 St. Paul, MN 55155		13. Type of Report and Period Covered Final Report
		14. Sponsoring Agency Code
15. Supplementary Notes http://www.lrrb.org/pdf/201306.pdf		

16. Abstract (Limit: 250 words)

In our previous research, the SMART-SIGNAL (Systematic Monitoring of Arterial Road Traffic and Signals) system that can collect event-based traffic data and generate comprehensive performance measures has been successfully developed by the University of Minnesota. In this research, a new set of interfaces are developed for SMART-SIGNAL system including new prototypes of data collection unit (DCU) and refined web-based user interface. To collect high resolution event-based traffic data including both vehicle detector actuation event and signal phase change event, two types of DCUs are designed, the TS-1 DCU and TS-2 DCU for corresponding traffic signal cabinet. TS-1 DCU connects with TS-1 cabinet using pin to pin interface, and the TS-2 DCU interfaces directly with SDLC bus within TS-2 cabinet. The DCUs uses high performance microcontroller modules, and are compact and easy to install. Both DCUs are designed to be vender independent add-on module for traffic cabinet, and can be used as flexible solution to enhance data collection by agencies. The refined web-based user interface features various performance measures to public users, such as Level of Service (LOS), queue length, travel time and intersection delays. The new set of interfaces have been deployed with the SMART-SIGNAL system at 13 intersections along Trunk Highway (TH) 13 in Burnsville, MN.

17. Document Analysis/Descriptors Traffic signal control systems, SMART SIGNAL system, Data collection, Traffic data, Performance measurement		18. Availability Statement No restrictions. Document available from: National Technical Information Services, Alexandria, Virginia 22312	
19. Security Class (this report) Unclassified	20. Security Class (this page) Unclassified	21. No. of Pages 60	22. Price

Research Implementation of the SMART SIGNAL System on Trunk Highway (TH) 13

Final Report

Prepared by:

Henry X. Liu
Jianfeng Zheng
Heng Hu
Jie Sun

Department of Civil Engineering
University of Minnesota

February 2013

Published by:

Minnesota Department of Transportation
Research Services
395 John Ireland Boulevard, MS 330
St. Paul, Minnesota 55155

Acknowledgements

This work was supported by Minnesota Department of Transportation. The authors would like to thank Steve Misgen, Timothy Bangsund and Curt Krohn for their assistance of the field deployment of TS-2 DCUs on TH13. Ron Christopherson also offered constructive suggestion for the design of TS-2 DCUs. The authors thank his help.

Dr. Henry Liu and the University of Minnesota have equity and royalty interests in SMART Signal Technologies, Inc., a Minnesota-based private company which may commercially benefit from the results of this research. These relationships have been reviewed and managed by the University of Minnesota in accordance with its Conflicts of Interests policies.

Table of Contents

List of Figures

List of Tables

Executive Summary

Efficient traffic signal operation is vital for smooth traffic flows in signalized network, and well-maintained signal operation is greatly beneficial for road users. With increasing congestion in traffic network, performance monitoring for arterial traffic control and management has become an area with emerging focus in the United States. Although many existing signal-control systems can generate data for performance assessment, most do not make it convenient for the agencies. And the majority of agencies do not monitor or archive traffic performance data for traffic operation improvement. In an effort to make up for such deficiency of data collection and performance measurement, a system for high-resolution traffic signal data collection, named as SMART-SIGNAL (Systematic Monitoring of Arterial Road Traffic and Signals) was developed by the University of Minnesota. The SMART-SIGNAL system can simultaneously collect event-based high resolution traffic data from signalized intersections and generate comprehensive performance measures in real time at both intersection level and arterial level.

This project takes the SMART-SIGNAL system one step further connecting basic research and engineering practices. To be compatible with the TS-2 Controller Cabinets, the data collection system for SMART-SIGNAL is re-designed so that it is able to connect to the Bus Interface Units (BIUs) directly. Utilizing high performance microcontrollers, the re-designed data collection unit (DCU) is designed as an independent add-on module that can serve as a flexible and cost-effective solution to enhance data collection capability for existing traffic signal systems. Two types of DCUs are developed for interfacing with TS-1 cabinet and TS-2 cabinet. Both types of DCUs are designed as a compact embedded device that greatly eases the installation within the cabinet. The ultimate goal for development of DCUs is to achieve plug-and-play capability at the field intersections and to reduce the effort of customized installation as much as possible.

The system interface is also developed as a Web portal for convenient access. Via the Web portal, users can easily access performance measurement data including historical travel time, real-time travel time, historical queue length, and real-time queue length. The installation and configuration of a server side software package is described in detail so that the system can be easily maintained by the agency.

Compared with the previous version of the SMART-SIGNAL system, the refined one described in this project is much easier to install and access. The manufacturing cost is also much lower. Thus transportation agencies could use the design described in this project to monitor the performance of their signalized arterials on a large scale. For demonstration purposes, the manufacturing process and implementation on 13 intersections along TH13 are described in detail, and can be referred to for guidance by the agencies.

Chapter 1: Introduction

1.1 Project Motivation

Efficient traffic signal operation is vital for smooth traffic flows in signalized network and well-maintained signal operation is greatly beneficial for road users. With increasing congestion in traffic network, performance monitoring for arterial traffic control and management has become an area with emerging focus in the United States. Although many existing signal-control systems are capable of generating data to support performance assessment, most do not make it "easy" for the managing agencies to prioritize improvements and plan for future needs (Liu et al., 2009 a). The overall D+ grade indicated in **Figure 1.1** by 2012 National Traffic Signal Operation Self-Assessment Survey shows that the "improvement and investment in traffic signal operations remains critical" (National Transportation Operations Coalition, 2012). And the majority of agencies do not monitor or archive traffic performance data for traffic operation improvement.

Figure 1.1: 2012 National Traffic Signal Report Card

[Source: National Transportation Operations Coalition, 2012]

In an effort to fill in the gap for data collection from traffic signal systems, a system to collect high-resolution traffic signal data, named as SMART-SIGNAL (Systematic Monitoring of Arterial Road Traffic and Signals) system was developed by University of Minnesota (Liu et al., 2009a). The SMART-SIGNAL can simultaneously collect event-based high resolution traffic data from signalized intersections and generate performance measures in real time. At single intersection level, the maximum queue length on a cycle-to-cycle basis is monitored by the system, with other performance measures as intersection delay and Level of Service (LOS) that can be further calculated. At arterial level, travel time, speed, and average number of stops can be calculated and reported in real time.

With support from Minnesota LRRB, Minnesota DOT, and U of M ITS Institute, the research development of SMART-SIGNAL system was initially completed in Sept. 2008. The system has been implemented on 11 intersections along France Ave in Hennepin County and 6 intersections along TH 55. In the meantime, independent evaluation of SMART-SIGNAL system was performed by Alliant Engineering, Inc., and results were very encouraging. Innovative algorithms to identify and manage oversaturated signalized intersections have also been proposed by further research developments funded by National Cooperative Highway Research Program (NCHRP) with promising outcomes (Wu et al., 2010).

1.2 Project Objectives

This project is intended to take SMART-SIGNAL system one step further connecting basic research and engineering practices at the traffic management agencies. In order to be compatible with the NEMA TS-2 Controller Cabinet that is being adopted by Minnesota Department of Transportation (MnDOT), the data collection system will be re-designed so that it is able to connect to the Bus Interface Units (BIUs) directly. In addition, graphical interface will be refined to be more user-friendly so that the system can be installed and managed easily. The refined SMART-SIGNAL system will be tested on Trunk Highway (TH) 13 with 13 intersections between Yankee Doodle Rd and TH 101.

1.3 Project Overview

The work planned in this project can be described from three aspects, including:

a) Hardware development
b) Software development
c) On-site implementation and system maintenance

Hardware Development

The hardware development for this project will focus on enhancing compatibility of data collection unit (DCU) for SMART-SIGNAL system. Since MnDOT is upgrading the controller cabinets on TH 13 to TS-2 type, the DCUs of SMART-SIGNAL need to be re-designed so that they are able to connect to the BIUs directly. In doing so, the wires used in the previous design become unnecessary and this will greatly increase the system connectivity and adoptability. In addition to the connectivity to the BIUs, DCUs will be field-hardened and running on 24 volt power (or 110 volt with power adapters) that can be obtained steadily from controller cabinets. The ultimate goal for development of DCUs is to achieve plug-and-play capability at the field intersections and to reduce the effort of customized installation as much as possible.

Software Development

The major effort on the software development is to refine the graphical user interface. While some prototypical user interfaces for SMART-SIGNAL system has been reported previously

(Liu et al., 2009 a), but they are still in the primitive stage and not suitable for implementation on a large scale. In this project, two user interfaces will be refined including:

1. Graphical user interface for field DCU, which enable an easy setting-up and resetting.
2. Graphical user interface at central server, which allows remote access to local field computers and generates real-time display on arterial performance measures. Summary statistics on the archived performance measures can be also displayed and reported.

On-Site Implementation

For demonstration purposes, the refined SMART-SIGNAL system is tested on TH13 with 13 intersections between Yankee Doodle Rd and TH 101. System configuration at field intersections and regional traffic management center (RTMC) are also demonstrated.

1.4 Report Organization

The rest of this report is organized as following: The state-of-the- practice of data collection will be reviewed in Chapter 2, in which data collections regarding event-based data are highlighted. Chapter 3 described the development of re-designed DCU for SMART-SIGNAL system in detail along with manufacturing procedure that can viewed as a guidance for small quantities production. The corresponding server side software installation and configuration are then discussed in Chapter 4 together with a refined Web-based user interface for public access. In the end, concluding remarks are stated in Chapter 5.

Chapter 2: Review of the State-of-the-Practice

2.1 Data Collection for Signalized Intersections

While real-time performance measures for freeway are commonly available in the forms of travel speeds and travel time nowadays, they are still relatively new for signalized intersections (Day et al., 2010), partially due to a lack of data collection efforts for signalized intersections. Although gradual improvement occurs for traffic signal operation from D grade by 2007 National Traffic Signal Operation Self-Assessment Survey to D+ in 2012, the data collection still remains at grade F, showing significant needs and rooms for improvement (National Transportation Operations Coalition, 2007; National Transportation Operations Coalition, 2012).

Current traffic controller for traditional traffic signals systems normally features capability to generate very basic MOE reports. For example, volume, occupancy and average speed (if speed detectors are available) aggregated in 5-60 minutes can be obtained from Econolite ASC/3 family controller (Econolite ASC/3 Controller Specification, 2012). Evaluation of Eagle's® MOE by researchers showed capability to generating MOE report with Volume, stops, delays and utilization of Eagle® EPAC 300 controller (Balke et al., 2005). However MOE generated within controller is still away from standardization, and is unrecognized by most of engineers operating the traffic system (Balke & Herrick, 2004; Balke et al., 2005).

One of noteworthy work of data collection system for signalized network is the arterial Performance Measurement Systems (PeMS) developed by University of California, Berkeley in cooperation with the Los Angeles Department of Transportation (Petty et al., 2005), which serves as the counterpart of freeway PeMS. Arterial PeMS aggregates traffic volume and occupancy data in 30 second intervals, and can generate performance measures such as link travel time and intersection delays. Other important works includes the development of ACS Lite as a light version of adaptive control software by Federal Highway Administration (Luyanda et al., 2003).

Recently, increasing efforts can be found using high resolution event-based data for performance measure of a traffic signal system, and achieves encouraging results. The "event" here is referred to vehicle-detector actuation and de-actuation and traffic signal phase changes, all in high resolution time intervals. Attractiveness of such event-based data over aggregated data are: (a) Performance indexes based on large (hour and day) or medium time interval (phases and cycle) can be easily calculated and (b) Investigation on microscopic level is feasible and could help to make more accurate and detailed traffic evaluation, particularly during congested traffic condition.

The Texas Transportation Institute (TTI) initiated the first effort to retrieve such traffic event data from existing traffic detection systems. The proposed prototype system, the Traffic Signal Performance Measurement System (TSPMS) can automatically collect traffic event data and generate performance measurement for individual intersections. Using event-based traffic data, Smaglik et al., (2007) proposed methodologies for actuated controller performance measure. Applying same type of data, Day et al., (2009) and Day et al., (2010 b) revisited Webster's seminal work (Webster, 1958) and illustrated procedure to identify opportunity to shorten cycle length and to identify correctable phase failures. In another work by Day et al., (2010 a), Purdue Coordination Diagram (PCD) is developed to visualize event-based data. Based on the PCD, arterial coordination evaluation is performed applying Combination Method (CM) to determine optimal offset adjustments. Based on the same type of high-resolution data, the University of Minnesota researcher successfully developed algorithms that can accurately estimate arterial travel time and intersection queue length in real time (Liu & Ma, 2009; Liu et al., 2009b).

2.2 Event-Based Data Collection

Although detectors are commonly available in many signalized intersections, additional adjustments are needed for existing traffic signal system to access event-based data, since standard traffic signal controller normally aggregate data in 5-60 minutes. To address the issue of non-standardization interface of controller MOE output, Bullock and Catarella (1998) developed the controller interface device (CID) that can interface signal controller with PC regarding signal events and detector events in real time. With CID, the hardware-in-the-loop simulation (HILS) can be established incorporating traffic controllers from different vendors. CID for TS-2 Controller has been developed by Advanced Traffic Analysis Center (ATAC) of North Dakota State University (Smadi & Birst, 2006). However, since the CIDs are designed for interfacing controller with simulation package in PC, it can’t obtain vehicle detector information independently.

TTI’s TSPMS extended CID with I/O data acquisition card to obtain signal and detector event data from TS-1 cabinet. The data is further processed by an industrial computer within the cabinet. For TS-2 cabinet, it uses enhanced BIU by Naztec Inc. to interface with SDLC bus, and the industrial computer is replaced by a rugged laptop for easier user interaction (Balke et al., 2005; Sunkari et al., 2011). This architecture for data collection system for TS-1 Cabinets is adopted and extended by the first version of the SMART-SIGNAL system.

As mentioned by Smaglik et al., (2007) one version of Econolite Controller Software, Econolite ASC/3 Data Logger (2006), features a data logger capability that can retrieve event-based data. Event are recorded with 0.1 sec time-stamping, and stored as binary-formatted files. The raw data can be stored within controller for >24 hours, and uploaded via FTP.

National Institute for Advanced Transportation Technology (NIATT) in University of Idaho also developed a data logging device (DLD) for high resolution traffic event data adopting similar architecture of data collection system in TSPMS for TS-1 cabinets (Ahmed & Brain, 2008). Although SDLC bus-based interface is recognized in the report, the connection of DLD for TS-2 cabinet is still based on harness cable connecting connection matrix on the back panel. A snap shot of components for DLD by NIATT can be viewed in **Figure 2.1**.

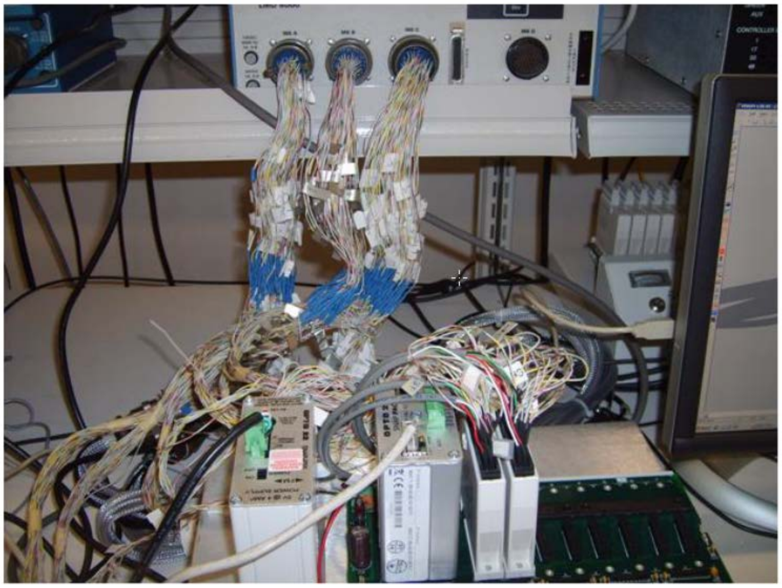


Figure 2.1: Data Logging Device Components Developed by NIATT

[Source: Ahmed & Brain, 2008]

Major limitations for available solutions can be summarized as following. 1) Solution with multiple components incurs unnecessary cost and space within the cabinet. It further increases the difficulty of installation in the field. 2) Vender-Based solution such as Econolite ASC/3 Data Logger, is still far from standardization and is not convenient for agencies with controllers from various venders. Although efforts can be seen recently from vender side to enhanced event-based data collection within controller, for a perceived future, the progress for standardization and a vender independent solution could be gradual and time consuming.

2.3 Summary

While it remains critical for better operation of a traffic signal system, the data collection effort of agencies involved in traffic signal system operation is still dispersed. With availability of detectors in signalized intersections, poor data storage and analysis lead to information wastage of improvement opportunities and result in unnecessary delay to drivers. Given the recent advances in computer and communication technologies, technical issue is no longer obstacle for data collection for traffic signal system. Among notable works in literatures, event-based data collection shows particular potentials of capability to address complexity pertaining to interruptive state of traffic signal systems.

This project intends to take SMART-SIGNAL system one step further with one of the main objectives to address major implementation issues for event-based data collection. With function as an independent add-on module, DCU is redesigned to serve as a vender independent and flexible solution to enhance data collection capability for existing traffic signal system. Further refinement of web user interface is also conducted for easier field setting-up and internet access.

Chapter 3: Design, Manufacturing and Implementation of the DCU

3.1 SMART-SIGNAL System Architecture

The SMART-SIGNAL system is a cohesive event-based data collection, storage, and analysis system. The System consists of three major components: data collection system for traffic event, performance measure system and user interface. The system architecture is shown in **Figure 3.1**. The first box indicates the data collection system implemented in the field. Two types of data, the signal event and detector event are collected by the system and stored in a log file and then transmitted to the database. Then, as indicated in the second box, various performance measures are generated, including both intersection level and arterial level, such as queue length, delay, travel time and etc. The third box shows the Web-based user interface for traffic performance information.

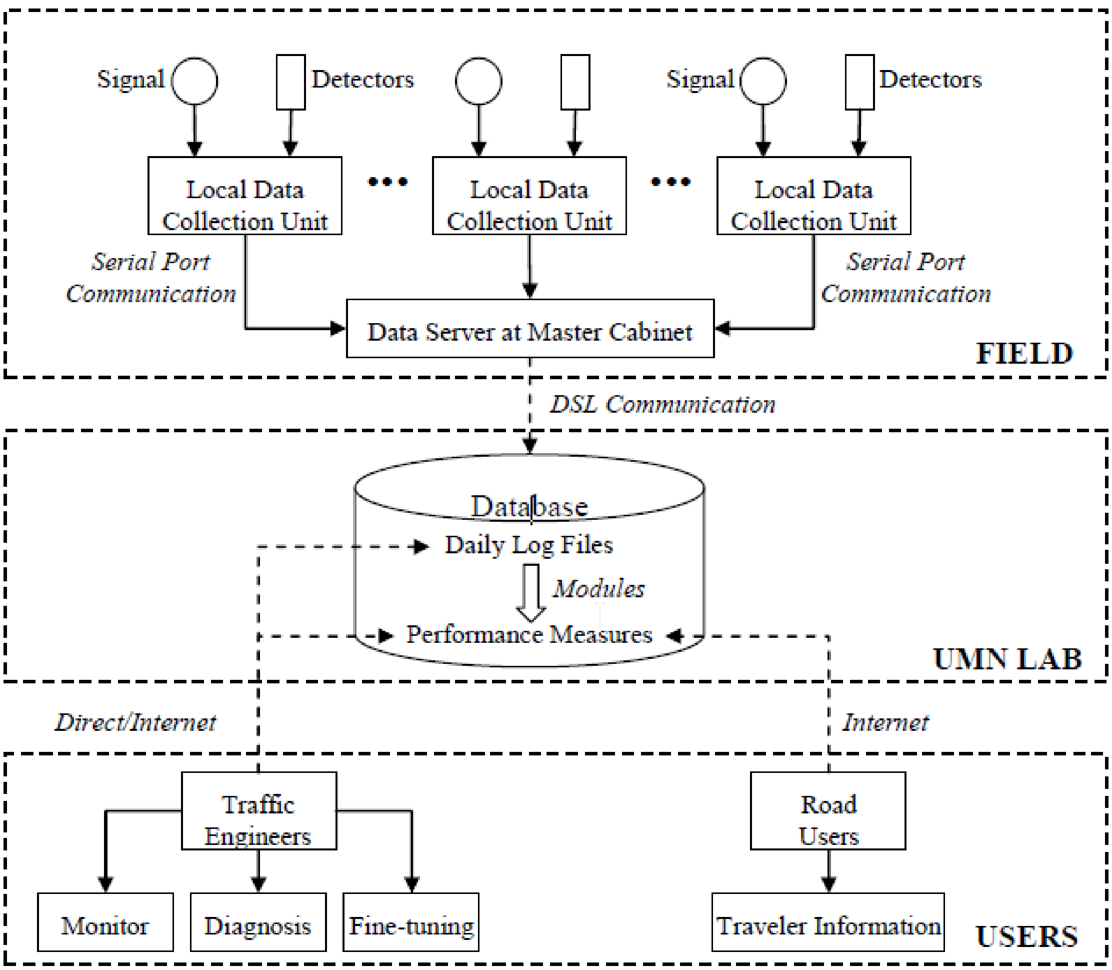


Figure 3.1: SMART-SIGNAL System Architecture

[Source: Liu et al., 2009 a]

Figure 3.2. illustrates data collection components for SMART-SIGNAL system. Two categories of elements are needed: the existing traffic signal elements and the data collection elements. Existing traffic signal elements include the vehicle detection units, the traffic signal controller and available cabinet interface for additional components. Data collection elements include the DCUs in each intersections and a data server in remote center.

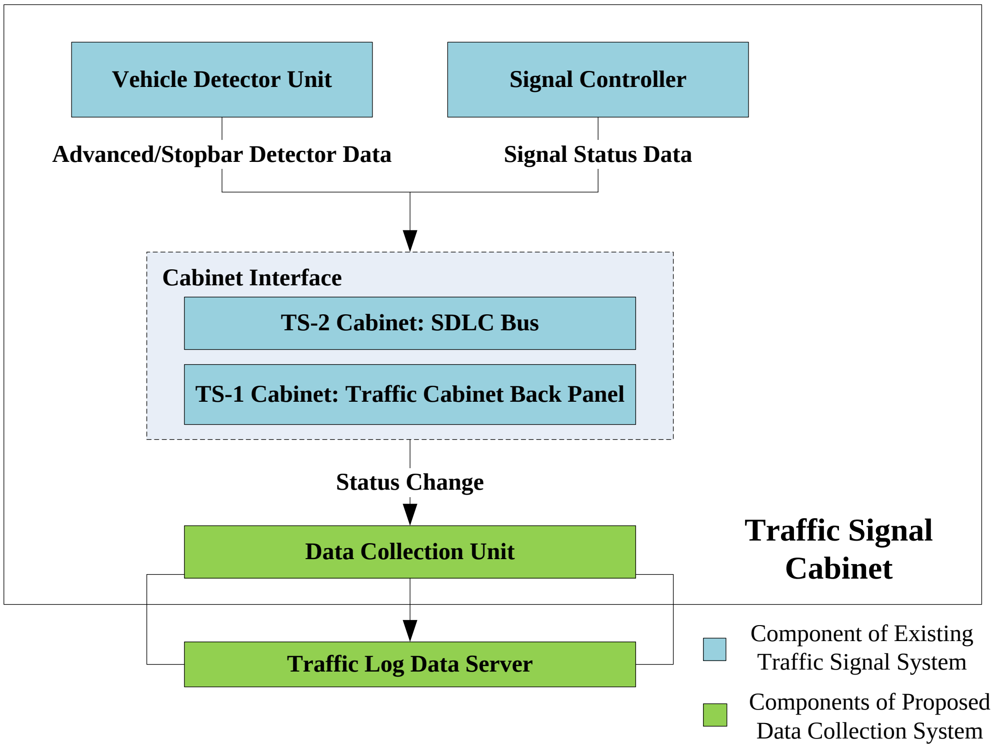


Figure 3.2: Flow Chart of the Traffic Data Collection Components [Adapted from Balke et al., 2005]

3.2 Design of DCU

In SMART-SIGNAL system, the DCU serves as a basic unit to obtain high-resolution data in real time. Previous development of DCU for TS-1 cabinets are based on prototype system of TSPMS system (Balke et al., 2005), using industrial computer with PCI-6528 digital input / output (I/O) data acquisition card. A view of data collection component for previous deployments can be found in **Figure 3.3**. With popularization of TS-2 Controller Cabinets over TS-1 Controller Cabinets, compatibility issues occur for DCU with interface based on contact closure instead of SDLC bus. Other implementation issues also exist since multiple components occupy cumbersome spaces and complicate wiring within cabinets. The re-designed DCU are aiming at addressing these issues. According to two different interfaces in TS-1 and TS-2 cabinets, we designed two types of DCUs: TS-1 DCU for TS-1 Cabinets interfaces with contact closure on the back panel, and TS-2 DCU for TS-2 Cabinet interfaces with SDLC bus with frame format defined by NEMA TS-2 standard (National Electrical Manufacturers Association, 2003). While the brief information of TS-1 DCU will be provided, the main focus of this report is on TS-2 DCU for TS-2 Controller Cabinets.

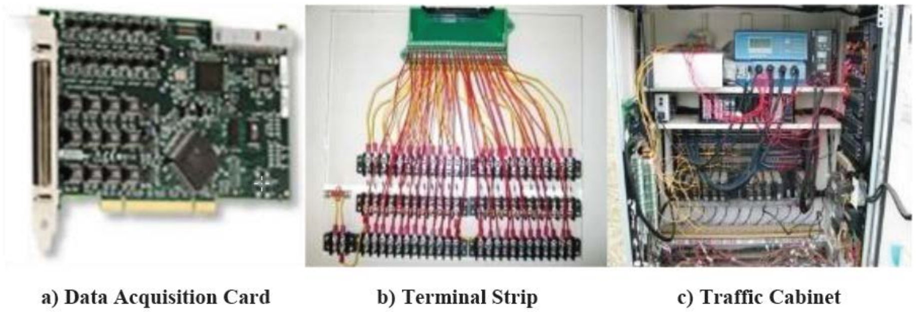


Figure 3.3: Traffic Data Collection Components for TS-1 Cabinet

[Source: Liu et al, 2009 a]

3.2.1 Design Overview

Both types of the DCUs are designed as an industrial single board computer. For interfacing with TS-1 Cabinet, 80 I/O channels are provided for TS-1 DCU to connect with contact closures. SDLC interface is provided by TS-2 DCU. Both types of DCUs provide interfaces to media card for local storage and standard RJ45 port for data transmission using Ethernet. Additional ports are also reserved for field debugging and future expansion. For reliability concerns, interfaces with TS-1/TS-2 cabinet are designed to be unidirectional with inputs only, and is not able to output to contact closures in a TS-1 cabinet and SDLC BUS in a TS-2 cabinet. The general architecture for two types of DCU can be found in **Figure 3.4** with views of sample devices.

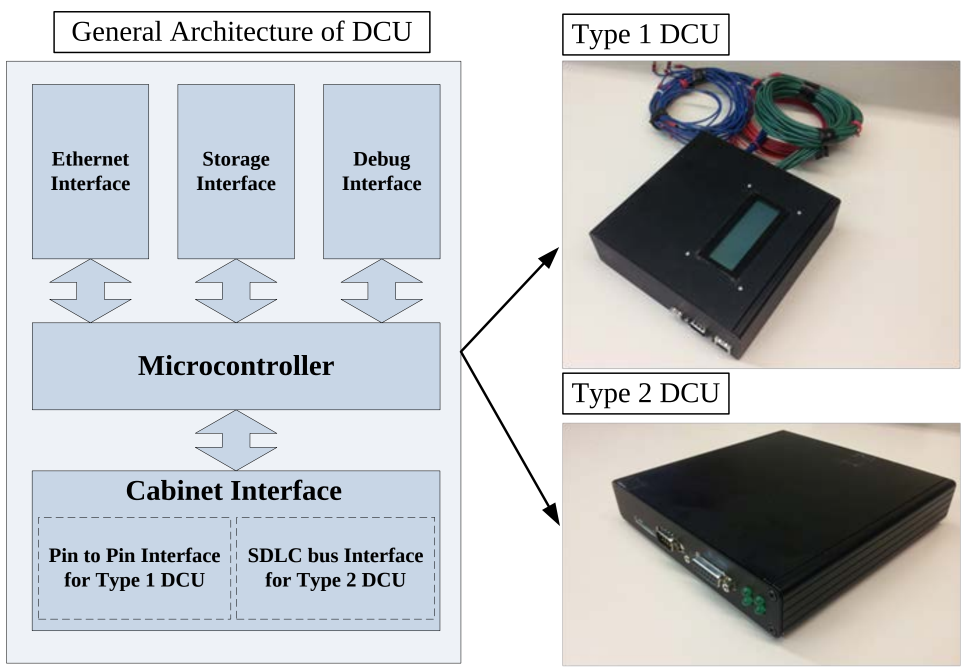


Figure 3.4: General Architecture and Snapshot of Two Types of DCUs

3.2.2 Design of TS-1 DCU

TS-1 DCU is designed to integrate terminal box with single board computer. LPC 1769 microcontroller produced by NXP Semiconductor is selected as the data processing core for TS-1 DCU. LPC 1769 is ARM Cortex-M3-based microcontrollers operating at CPU frequencies of up to 120 MHz. Rich peripheral interfaces including a USB controller and Ethernet interface make it suitable for various secure applications (UM10360-LPC 17XX User manual, 2010). The port layout for TS-1 DCU can be found at **Figure 3.5.**

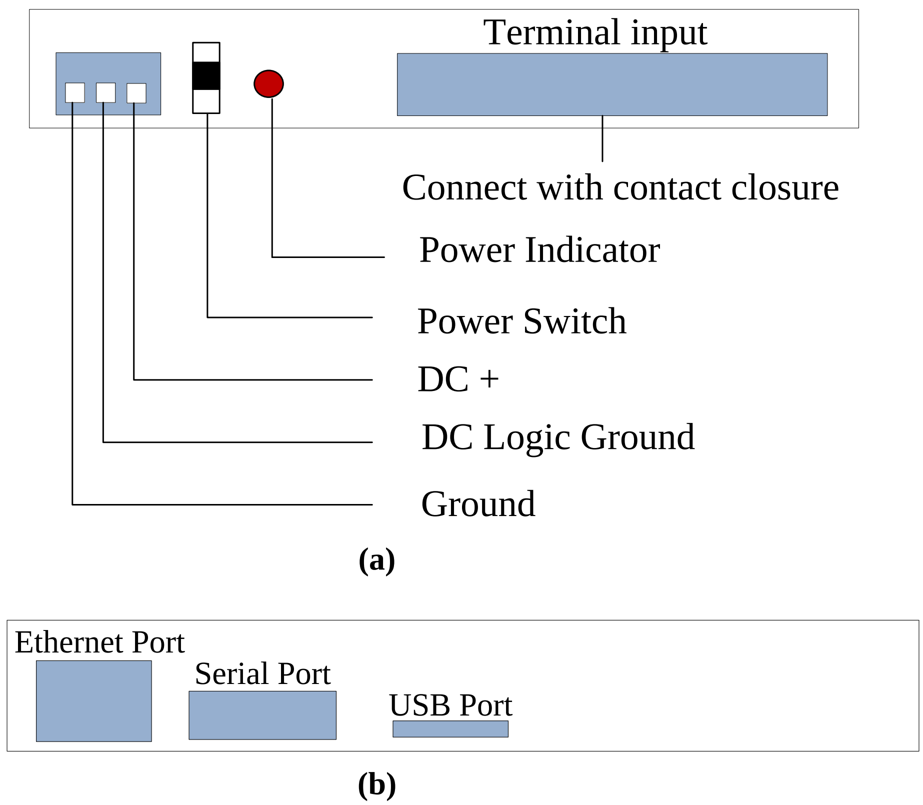


Figure 3.5: Port Layout of Front Panel (a) and Back Panel (b) for TS-1 DCU

3.2.3 Design of TS-2 DCU

The TS-2 DCU consists of two microcontroller unit (MCU) modules (Module A & B) with different peripheral ports with TS-1 DCU. Module A's main function is to process frame, archive data locally to mass storage media (SD Memory Card), and send data to remote server. Module B is placed to capture data frame from SDLC BUS in a TS-2 Cabinet. ConnectCore™ 9M 2443 of Digi International Inc. (Figure 3.6) is used as Module A for user customized tasks. Its main components include: 533 MHz Microcontroller, 128 MB Flash with 64 MB SDRAM Memory (Digi International Inc.). RabbitCore RCM4100 module is used as Module B for SDLC processing. Both modules are programmable for user extension. Embedded Linux is used as Module A’s operating system and can provide field engineer with a stand Linux console for configuring and troubleshooting the DCU from serial port.

Figure 3.6: Digi ConnectCore™ 9M 2443 Core Module

[Source: Digi International Inc.]

Several peripheral ports including D-sub 15 pin SDLC port (Port 1), D-sub 9 pin EIA RS232 port (Port3), RJ 45 Ethernet port and a SD Media Card slot are provided in TS-2 DCU. When operating in a TS-2 cabinet, the module B captures frames from port 1 and communicates with Module B via universal asynchronous receiver/transmitter (UART) port. Module B stores the data into SD Card, and sends data via Ethernet to central server. The block diagram for DCU is shown at **Figure 3.7**. The details of each peripheral port are described in following.

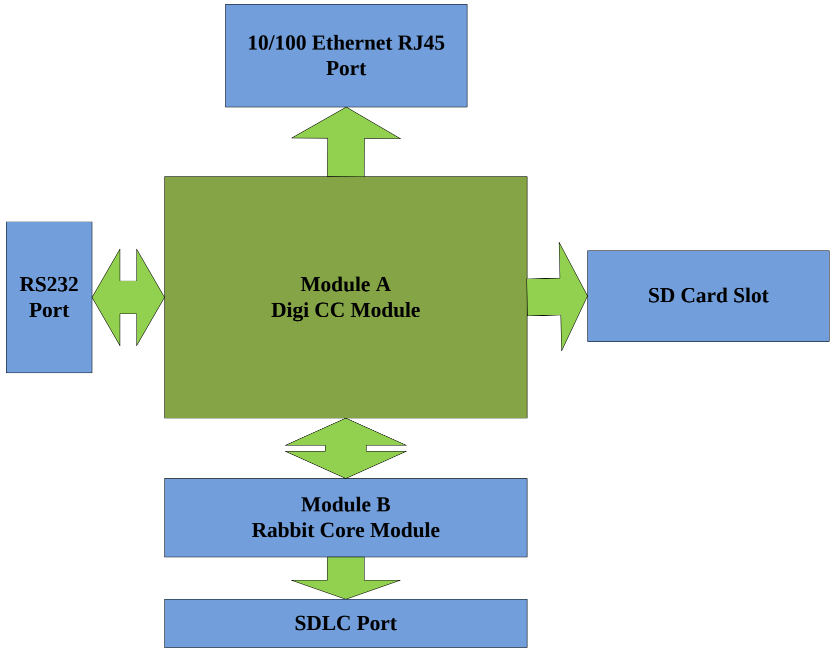
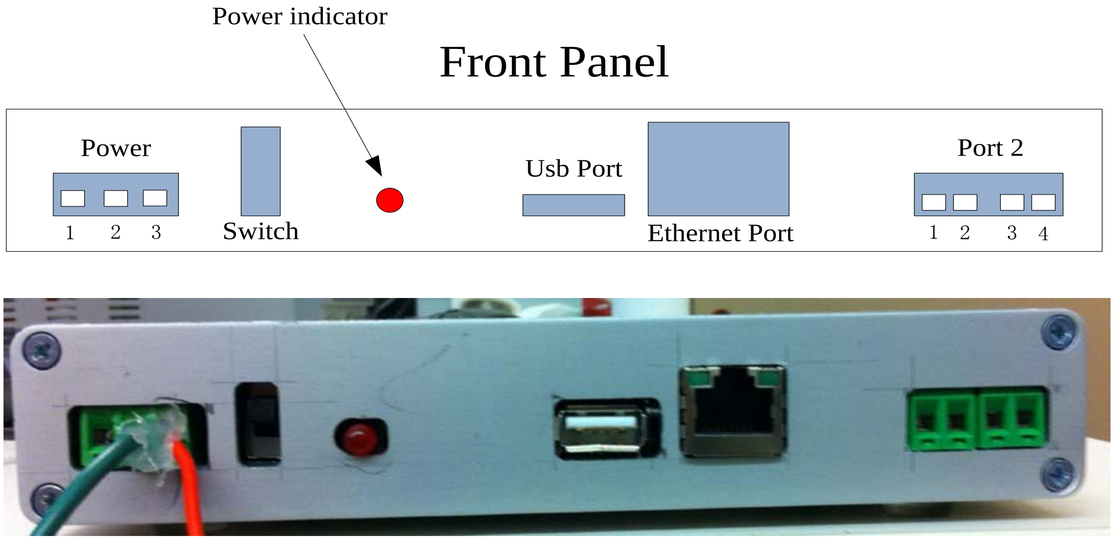


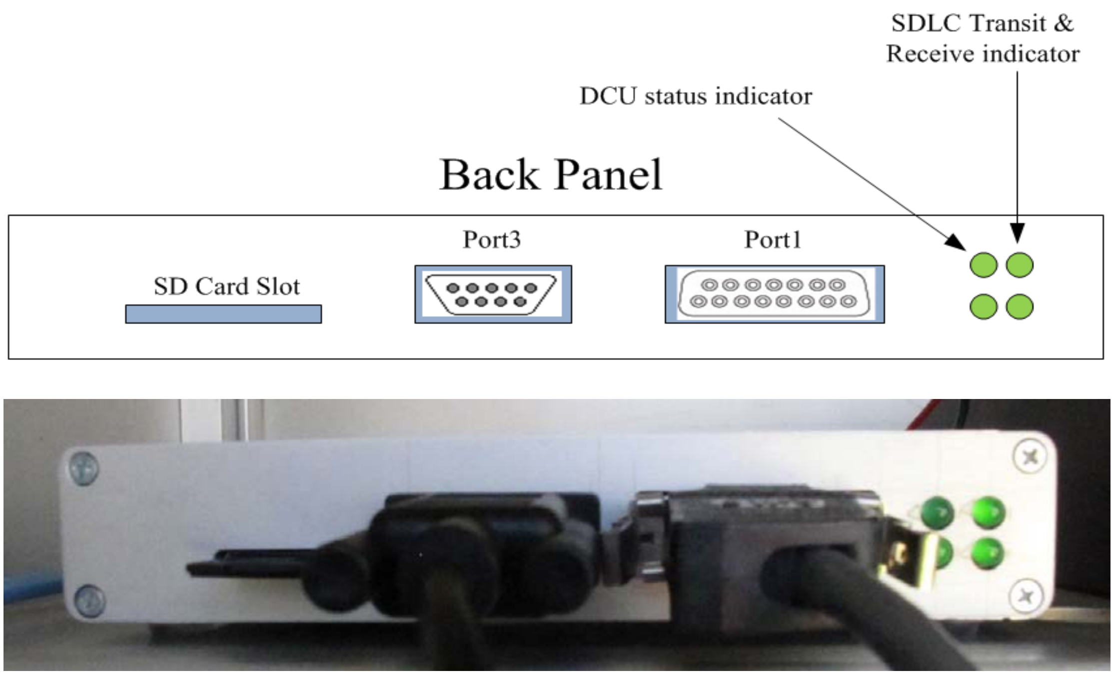
Figure 3.7: Block Diagram of TS-2 DCU

Peripheral Ports in TS-2 DCU

Port layout of TS-2 DCU can be found in **Figure 3.8.**



(a)



(b)

Figure 3.8: Ports Layout for TS-2 DCU

Power Port for DCU

The design input power range for DCU is DC 12-30 V. The pin assignment for power port can be found at **Figure 3.9**.

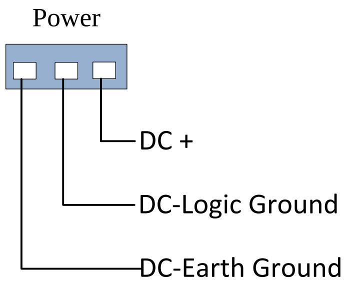
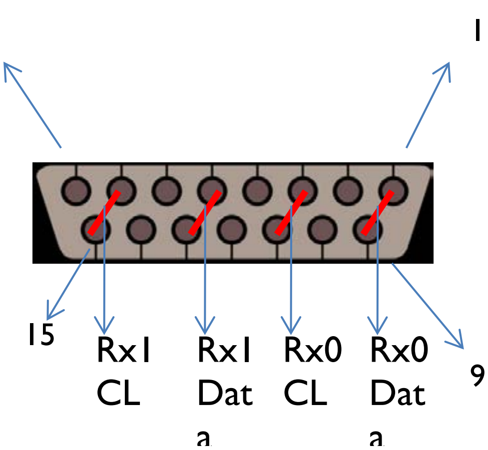


Figure 3.9: Layout of the Power Port

a) Port 1—D-Sub 15 pin SDLC Port

Port1 use a Ti SN65LBC175 EIA-485 Receiver along with SMBJ5 Transient Voltage Diodes for EIA-485 connection with surge protection. Two SDLC inputs are provided with 2 data and clock inputs operates at EIA-485 differential mode. The layout and pin assignment of Port 1 are provided in **Figure 3.10.**

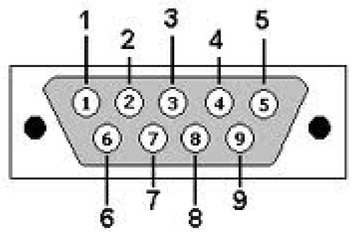


Pin ID	**1**	**2**	**3**	**4**	**5**
Function	Rx0 Data+	N/A	Rx0 Clock+	N/A	Rx1 Data+
Pin ID	6	7	8	9	10
Function	N/A	Rx1 Clock +	N/A	Rx0 Data -	N/A
Pin ID	11	12	13	14	15
Function	Rx0 Clock -	N/A	Rx1 Data -	N/A	Rx1 Clock-

Figure 3.10: Layout and Pin Assignment of Port 1

b) Port 2—D-sub 9 pin EIA-232 port

Port 2 is a D-sub 9 pin EIA-232 port, designed for a field engineer to configure the DCU. It is the terminal port for Linux system running on module A. It can be used to directly connect to serial port of a personal computer at a Baudrate of 38400 Bits/sec. Minimum 3 pin connection, pin 2, pin 3, pin 5 are required for to establish a connection using a null modern cable. The PIN assignment and layout for Port 2 is provided in **Figure 3.11**.



Pin	**1**	**2**	**3**	**4**	**5**
Function	N/A	Rx Data	Tx Data	N/A	Logic Ground
Pin	6	7	8	9	
Function	N/A	RTS	CTS	N/A	

Figure 3.11: Layout of Port 2

c) Led Indicators

In total, 5 indicators are provided in DCU. The red indicator on front panel is the power indicator. Indicators on the back panel includes: 2 DCU status indicators & 2 SDLC receiving activity indicators. The 2 DCU status indicators flashes with predefined frequency after the DCU is powered up. If status indicator stops flashing, it indicates a malfunction of Module B of DCU. The flashing of receiving activity indicators shows ongoing SDLC data transmissions.

3.2.4 Data Format and Sample

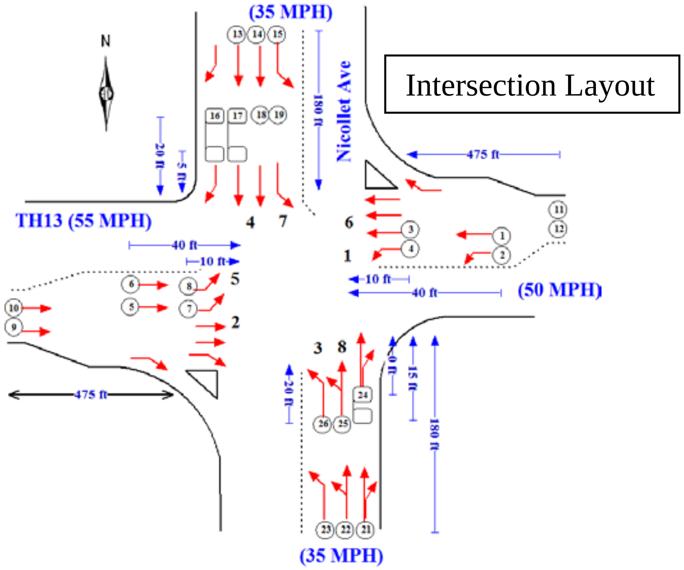
Signal Event Data

The signal status can be obtained from frames transmitted from controller to Signal Load Switch BIUs. DCU records signal event information including phase switching time and current signal status with a 0.1 sec resolution, which is also the resolution defined at NEMA TS-2 Standard for controller unit.

Vehicle Detection Data

The vehicle detection data can be obtained from frames transmitted from detector BIUs to the controller. As defined in NEMA TS-2 Standard, the information from Detector Rack BIUs to controller units includes vehicle status and timer status which has a resolution of 1 milliseconds recording detector actuation information. Thus, the highest resolution within one lane can be 1 millisecond. The synchronization with other lane, however, remains at a resolution of 0.1 seconds.

A sample data collected in Intersection Nicollet Ave & TH13 can be found at **Figure 3.12** with intersection layout.



Detector occupation event

Start Time of Detector "ON"	Detector ID	Occupied Duration(Sec)
05:00:16.400	13	0.321
05:00:21.157	10	0.403
05:00:20.989	16	1.369
05:00:22.580	9	0.403

Signal Event

Start Time	Signal Status	Phase	Duration(Sec)
05:00:16.800	G	3	9.800
05:00:26.600	Y	3	4.000
05:00:33.100	G	4	10.300
05:00:43.400	Y	4	4.000

Figure 3.12: Intersection Layout and Data Sample

3.3 DCU Manufacture Process

There are 18 sets of the TS-2 DCUs, with 13 of them instrumented on TH13 and 5 sets of spares manufactured. In the following, the manufacturing process will be described, starting from the preparation of the design files, and prototype verification, to the small amount production.

3.3.1 Preparation of the Design Files

Before getting into any hardware manufacturing process, we need to have the all design files ready. Those are:

1. Schematic drawing,
2. Layout of Printed Circuit Board (PCB),
3. Bill of Materials (BOM), and
4. Enclosure drawing to drill ports of panels to fit the assembled PCB.

The schematic file and PCB layout are needed for PCB fabrication, and BOM is the reference for the selected components to be placed on PCB. A detailed BOM, in consistent with the schematic and layout file, lists all necessary information for each component. The information consists of ID (which should be clearly printed on the PCB for each component), footprint, and part number defined by component manufacturer or distributers i.e., the merchandise ID for purchasing the components. The general procedure of hardware manufacturing can be viewed at **Figure 3.13.**

Figure 3.13: General Procedure of Manufacturing

3.3.2 Prototype Verification

With the design files, it's typically desired to do a prototype assembly at first to verify the design. Such practice is, essentially, to assemble 1-2 units manually in lab before any formal production, since most factories (or assembly companies) require a minimum amount of production (>5) and cost per unit varies significantly with the order amount. A purchase of 3-5 sets of components and PCB fabrication are recommended for the verification. After hand-soldering the components to PCB, if the indicator led lights on normally and reference voltage is stable when power is up, then we can mount our module into the PCB and completes assembling circuit board without enclosures. The next step is to download firmware into the device. A software and hardware environment testing (>1 hour) is also needed to complete the process. The procedure of prototype verification is shown in **Figure 3.14**.

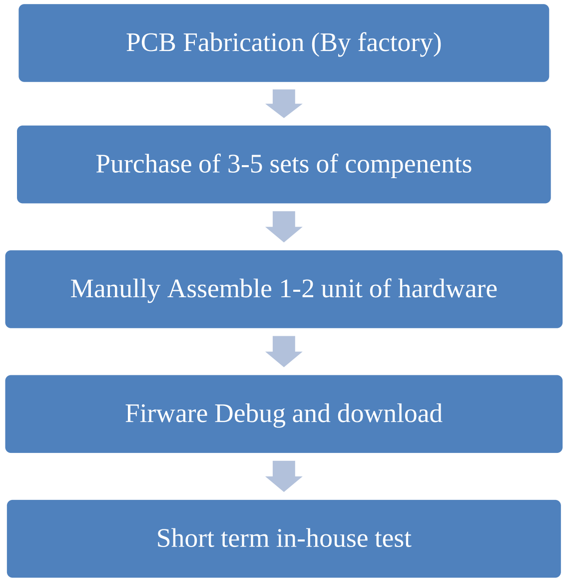


Figure 3.14: Procedures of Prototype Verification

3.3.3 Small Amount Production

After a successful verification, it's time to go through the small amount production, or so called "pre-production". A turn-key service by a PCB assembly company is recommended. With the turn-key service, the procedure is straightforward — package the design files (PCB layout), and send them to a sale representative from an assembly company to get a quote and place the order. An assembled PCB with all components placed should be available shortly. The rest procedure is needed to get devices ready to be installed in the fields: mounting the modules, downloading firmware into device, drilling enclosure panel port (done by machine shop), and mounting PCB into enclosure. The manufacture process is illustrated in **Figure 3.15**.

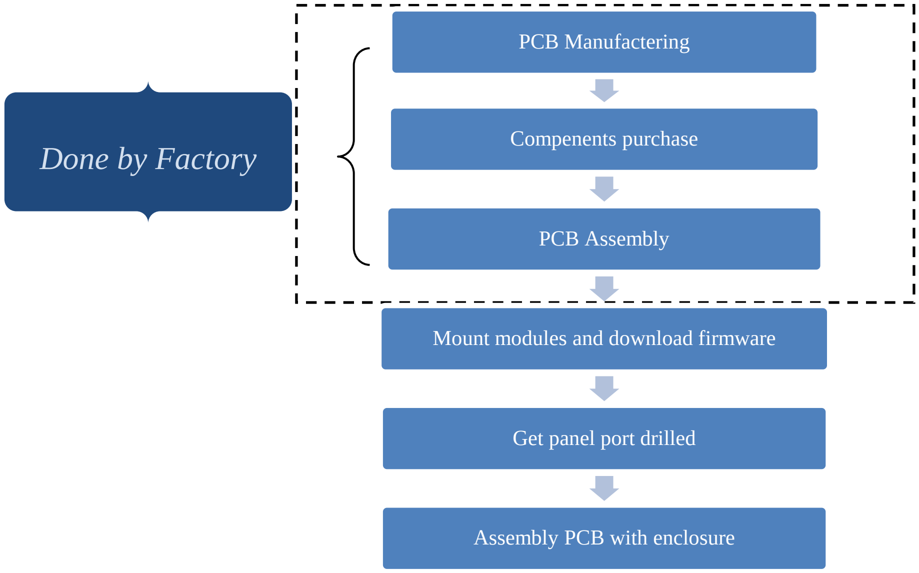


Figure 3.15: Manufacture Process for Small Amount of TS-2 DCUs

3.4 Field Implementation of DCUs

Thirteen sets of TS-2 DCUs were installed on TH13 since December, 2011. In the following, we will first make a brief description of the implementation sites, followed by a discussion on the implementation procedure of DCUs, as well as the procedure of checking the working status of DCU from both the field & the central server.

3.4.1 Description of Implementation Sites

There are 13 intersections on TH 13 implemented with the SMART-SIGNAL DCUs. The planned installation at intersection Highway 101 is cancelled due to construction nearby. These implementation sites are using fiber optics for communication between intersections and Regional Traffic Management Center (RTMC). A unique IP address is provided to the SMART-SIGNAL system DCUs at each intersection. Information of implementation site can be found at **Table 3.1.** and a map with implementation sites can be found at **Appendix A.** To make notation brief, only intersection IDs are used for referring an instrumented site in the rest of this report.

Table 3.1: Information of Implementation Sites

Intersection ID	Cross Street	Comment
0401	Highway 101	Not Installed Due to construction
0402	Lynn Ave.	
0403	Washburn Ave S	
0404	Co Rd 5	
0405	Nicollet Ave	
0406	Portland Ave	
0407	Parkwood Dr.	
0408	River Hills Dr W	
0409	Cliff Rd E	
0410	River Hills Dr	
0411	Diffley Rd	
0412	Silver Bell Rd	
0413	Blackhawk Rd	
0414	Yankee Doodle Rd	

3.4.2 Installation Procedure

Because the TS-2 DCU is designed to be a plug-and-play device, the implementation the field is quite simple. The implementation steps are:

a) Plug the power adapter into 110 voltages AC socket, then
b) Connect the SDLC port in DCU with SDLC socket in the cabinet, then
c) Connect DCU to the Ethernet switch inside cabinet using fiber cable, then
d) Turn on the power switch.

A graphical representation and a snapshot of DCU installed in field (intersection Lynn Ave & TH 13) can be found at **Figure 3.16, Figure 3.17** respectively.

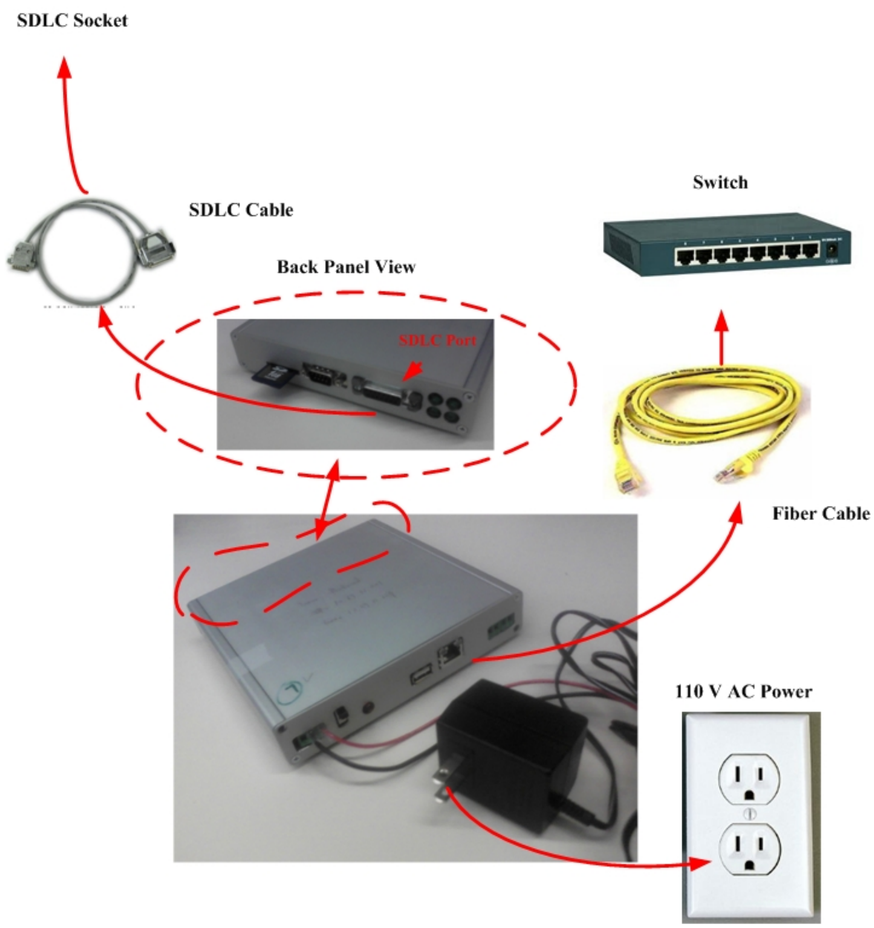


Figure 3.16: Plug and Play Installation of TS-2 DCU

Figure 3.17: Snapshot of DCU Installed in Intersection Lynn Ave & TH 13

3.4.3. Checking for the Working Status of DCU after Implementation

After implementation, we need to check the working status of DCU, i.e., whether the data collection and transmission are functioning properly. There are two ways to do so, either from the field or from the central server.

Checking at Instrumented Sites

To check the working status of DCU in the field, we need to connect the serial port of DCU (Port 2) to a serial port of a PC or laptop to see the console output. The procedure to set up HyperTerminal for serial connection with DCU can be found in **Appendix B.** As the DCU will be seeking for the server to establish TCP/IP connection when booting up, if no host is available, error information will be printed on the console output, as shown in **Figure 3.18**. As soon as connection is established, the console will print every event collected by the DCU to the console display, as shown in **Figure 3.19**.

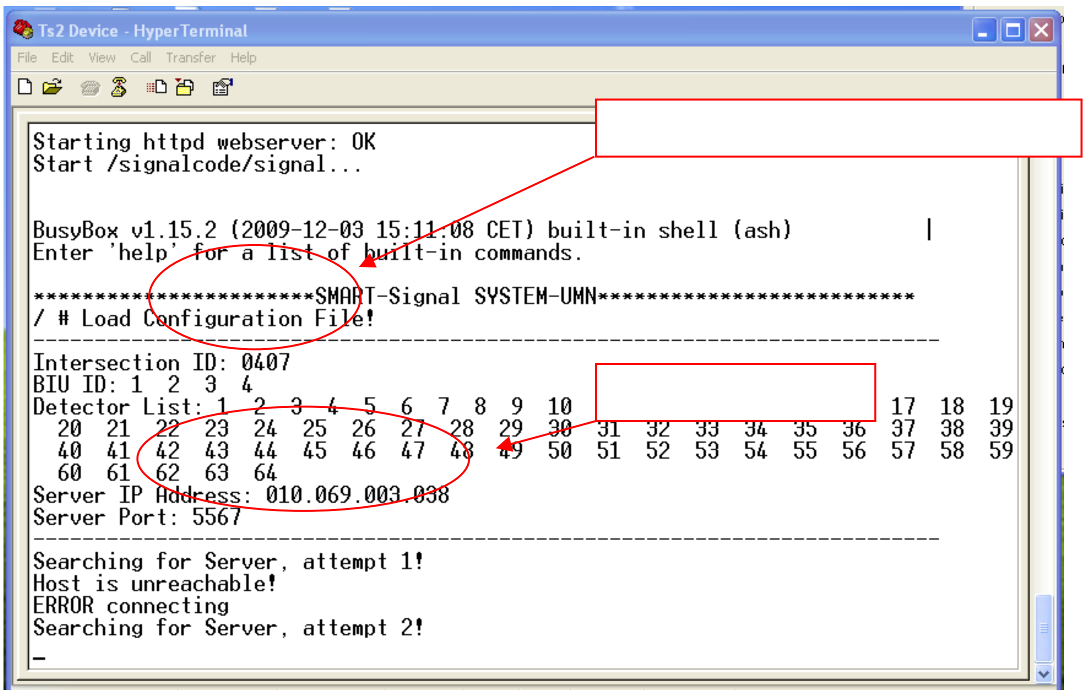


Figure 3.18: Illustration of Console Output with Error Information

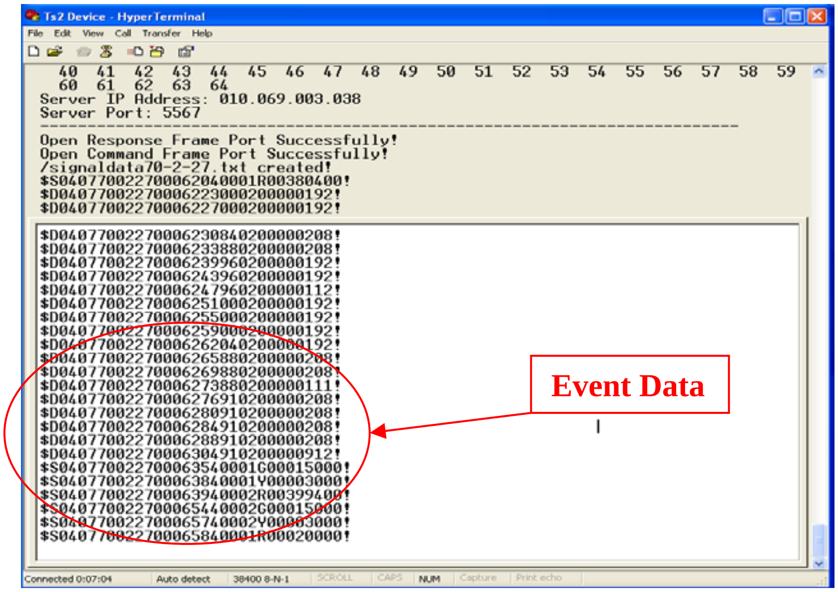


Figure 3.19: Illustration of Console Output with Event Data Displayed

Check from the Server

As soon as the Ethernet connection is established between the DCU and the server, and if the SMART-SIGNAL server software package is installed in the server, we should expect to see **Figure 3.20** from the SMART-SIGNAL server software GUI that indicates successful connection and data transmission status.

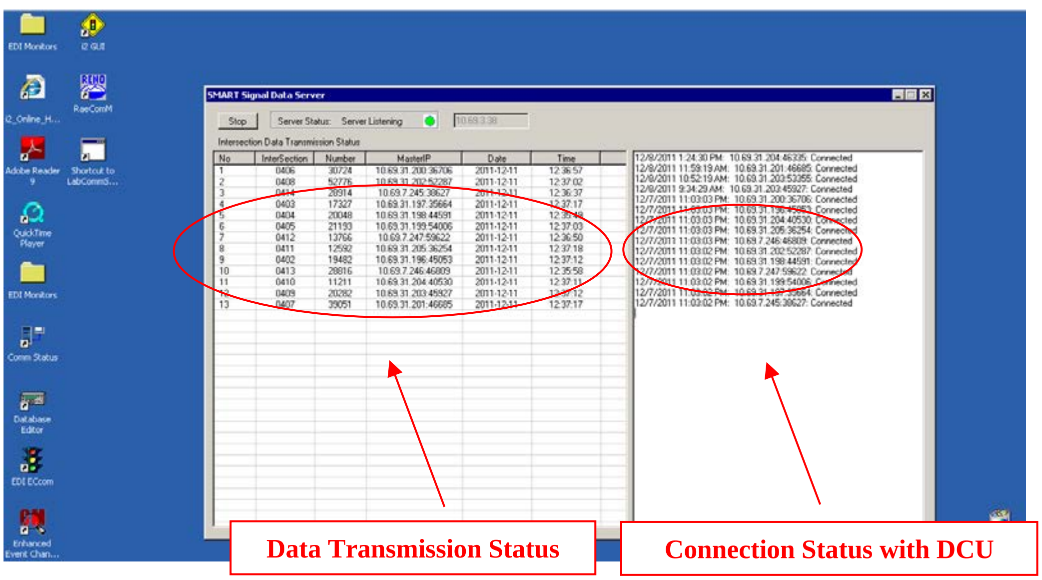


Figure 3.20: GUI of the SMART-SIGNAL Server Software Package

In addition,, we can log into the DCU using FTP or Telnet from the central server, to get access to the Linux console to make configuration changes or to transfer files. How to get access to DCU via FTP & Telnet can be found in **Appendix C.**

Chapter 4: Web-Based User Interface, Software Installation and Configuration

4.1 SMART-SIGNAL Web-Based User Interface

In order to set up and maintain the SMART-SIGNAL system more easily, the web-based user interface is refined. The web-based user interface can be accessed either inside or outside the MnDOT network. And users can conveniently monitor the real-time intersection LOS, retrieve the historical and real-time queue information, and estimate arterial travel time and intersection delay.

4.1.1 Overview of Web-Based User Interface

Through the SMART-SIGNAL web-based user interface, various performance measures of the implemented sites can be obtained. If users are inside the MnDOT network, the website can be accessed through "**http://10.69.*.***" or "**http://10.69.*.*/smartsignal**"; However, if the users are outside the MnDOT network, i.e., in a public Internet, the website can be accessed through "**http://dotapp4.dot.state.mn.us/smartsignal**". **Figure 4.1** described the structure for SMART-SIGNAL website. Performance measurement of TH 13 can be obtained from subpages under category "Instrumented Sites", and webpage of project summary and resources lists related information like publications and news for SMART-SIGNAL system.

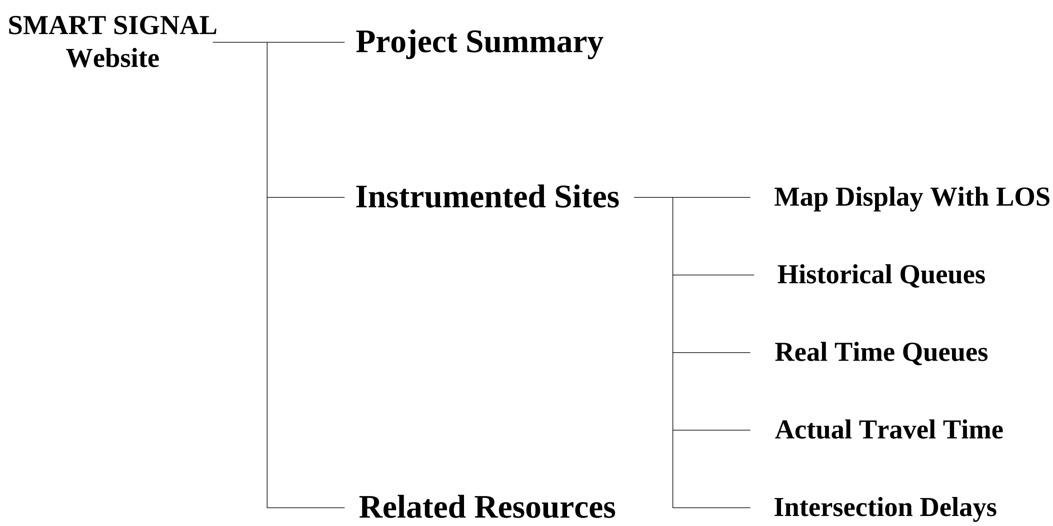


Figure 4.1: SMART-SIGNAL Website Structure

4.1.2 Webpages for Performance Measurements

Figure 4.2 shows the main page of the website. The major components, the performance measurements can be found on the left column under **"Implementation Site"**→"**TH 13**", with four major sub-tabs, i.e., **Historical Queue**, **Real-time Queue**, **Actual Travel Time** and **Intersection Delays**. The methodologies to calculate corresponding performance measurements can be found elsewhere (Liu et al., 2009), and will not be stated in this report.

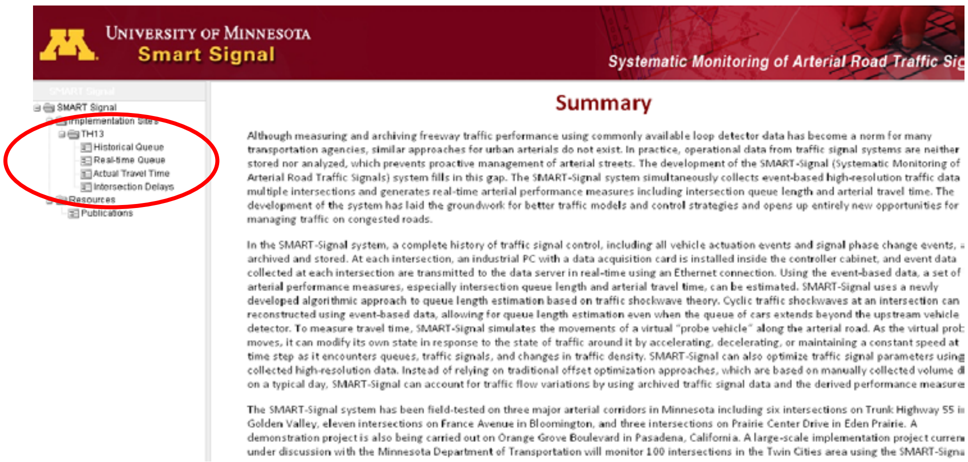


Figure 4.2: Main page of the SMART-SIGNAL Web Application

(1) Level of Service (LOS)

If users click on the "**TH 13**" tab, an area map of the implemented site will be loaded, see **Figure 4.3.** The different colors of the dots indicate the different levels of service at the corresponding intersections. The correspondence can be found in **Table 4.1**.

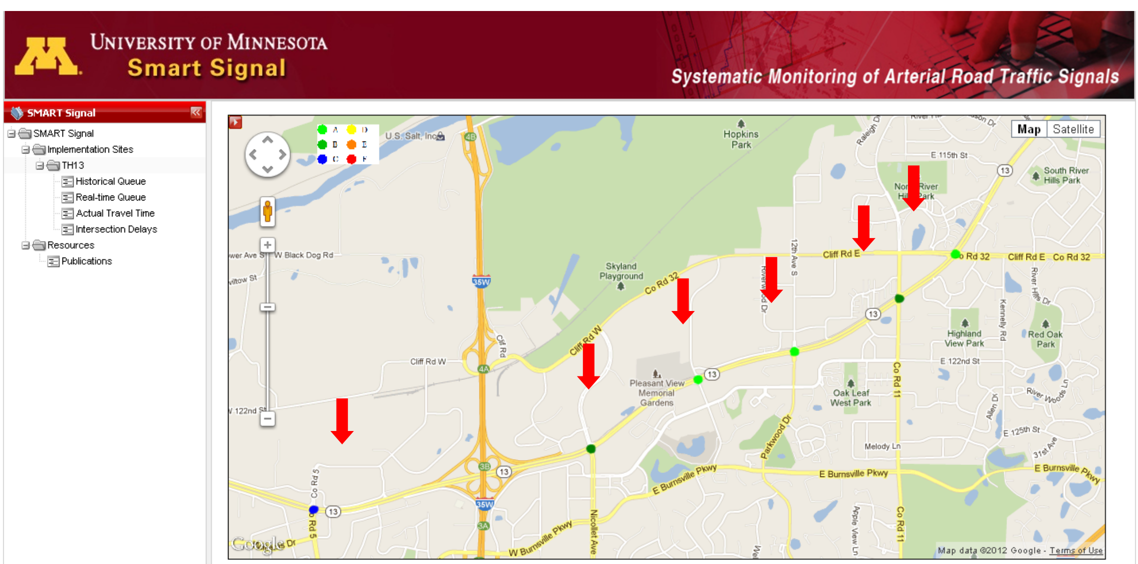


Figure 4.3: Map Display with LOS of TH13

Table 4.1: LOS Color Map

Color	LOS
Light Green	A
Dark Green	B
Blue	C
Yellow	D
Orange	E
Red	F

(2) Historical Queue

Once users click on the "**Historical Queue**", a webpage like **Figure 4.4** will be displayed.

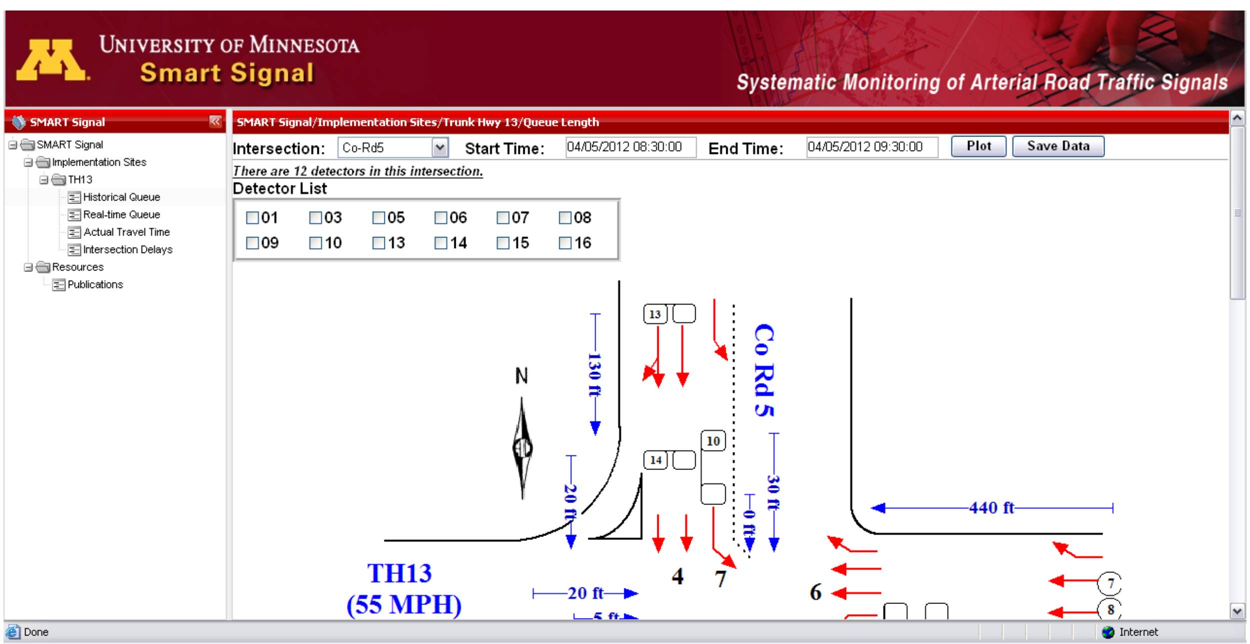


Figure 4.4: Webpage for Historical Queue Retrieval

One can first choose an intersection, a time interval and one or multiple detectors, and click the button "**Plot**" or "**Export**". When "**Plot**" is clicked, the queue length profiles for selected detectors during the specified time interval will be displayed (see **Figure 4.5**), where the horizontal axis represents time and vertical axis represents queue length in "ft". When "**Export**" is clicked, the queue length information will be exported into ".csv" file. As shown in **Figure 4.6**, users can choose to save the file at their preferred location.

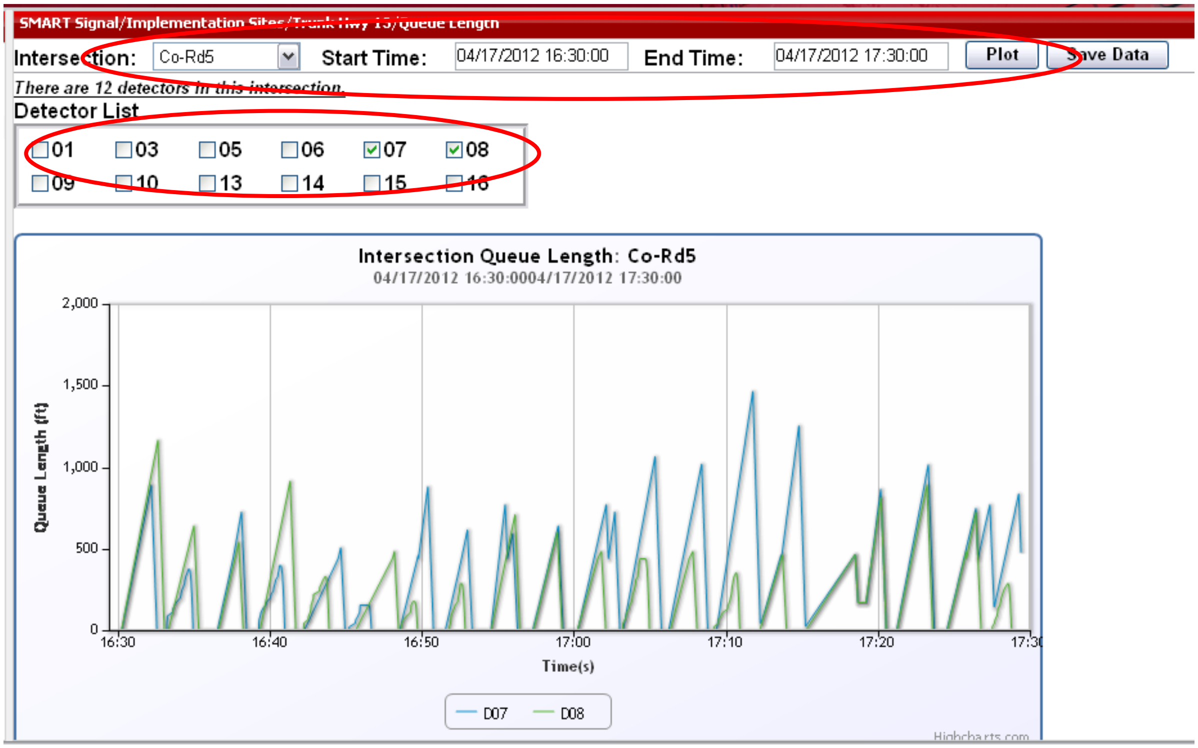


Figure 4.5: Illustration of Historical Queue Retrieval

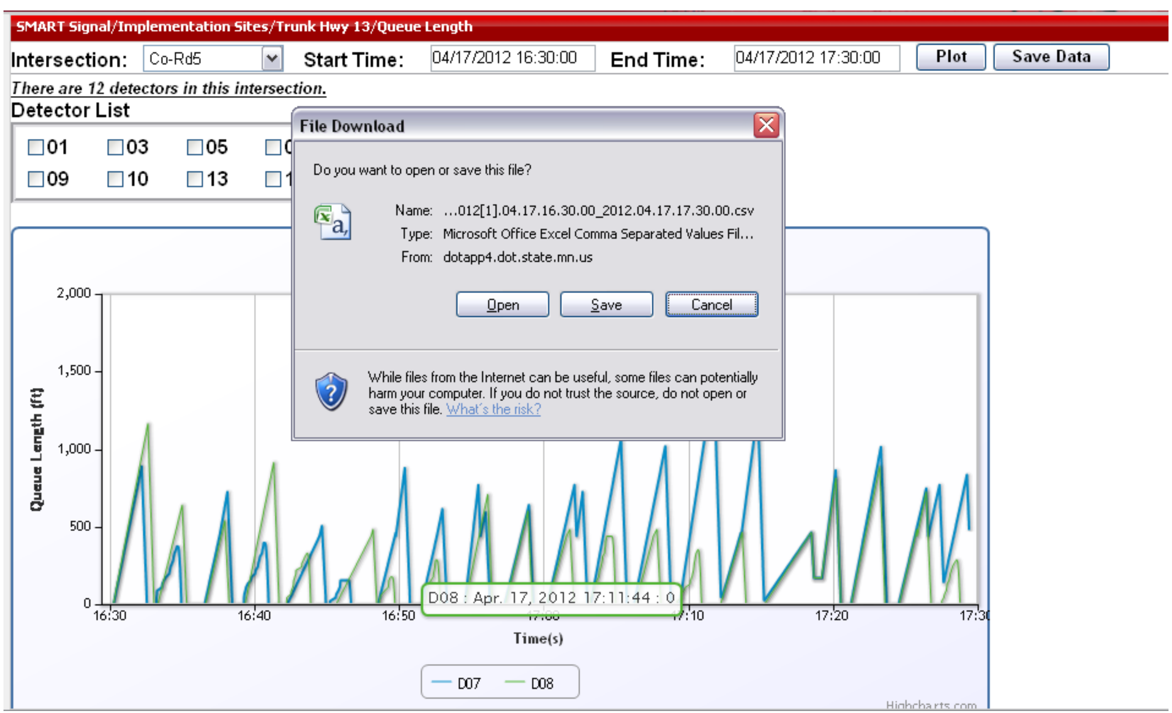


Figure 4.6: Pop-Up Window for Export Function

(3) Real-Time Queue

Once users click on the "**Real-time Queue**", a webpage like **Figure 4.7** will be displayed.

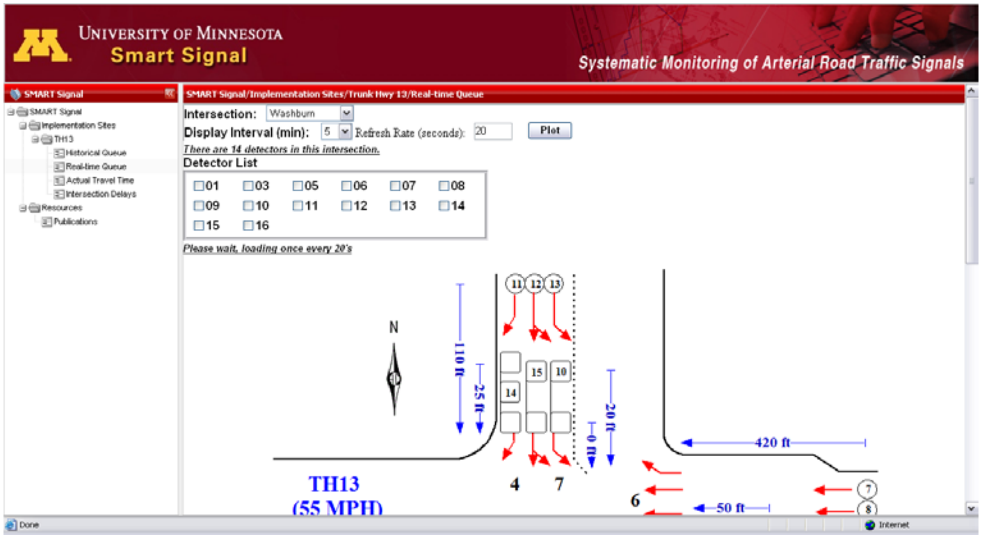


Figure 4.7: Webpage for Real-Time Queue Display

One should first select the "**Display Interval**" and "**Refresh Rate**". Once you click "**Plot**", the real-time queue length profile will be displayed (**Figure 4.8**). **Display Interval** specifies the time window to display the real-time queue length information. For example, if the user chooses the interval to be "15 min", the figure will show the queue length profile from 15 minutes prior to the current time instant. Further, the figure will be automatically updated as time goes on and **Refresh Rate** determines the update frequency.

Figure 4.8: Illustration of Real-Time Queue Display

(4) Travel Time

Once users click on the "**Travel Time**", a webpage like **Figure 4.9** will be displayed.

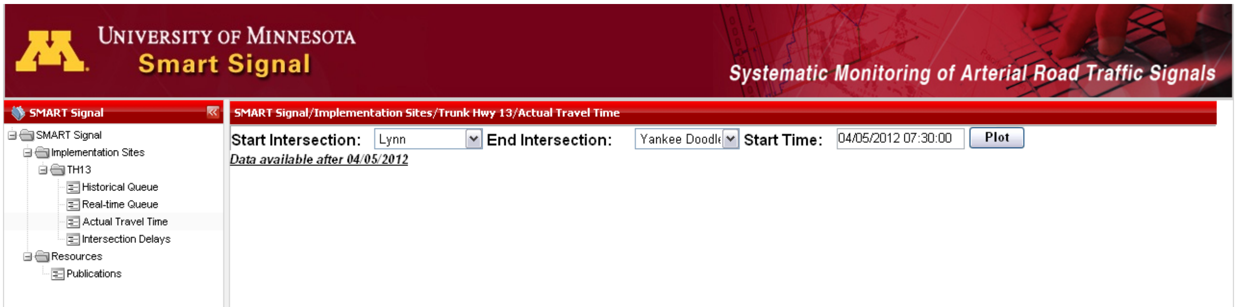


Figure 4.9: Webpage for Travel Time Calculation

In order to calculate the travel time, the user needs to specify the "**Start Intersection**", "**End Intersection**" and "**Start Time**", see **Figure 4.10**. After that, once the user clicks the "**Plot**" button, the total travel time of one vehicle departing at the specified start time from the stop bar of the start intersection to the stop bar of the end intersection will be calculated. The trajectory of the vehicle will also be displayed. If the user clicks the "**Export**" button, the time stamps when the vehicle passes each intersection will be exported into a ".csv" file.

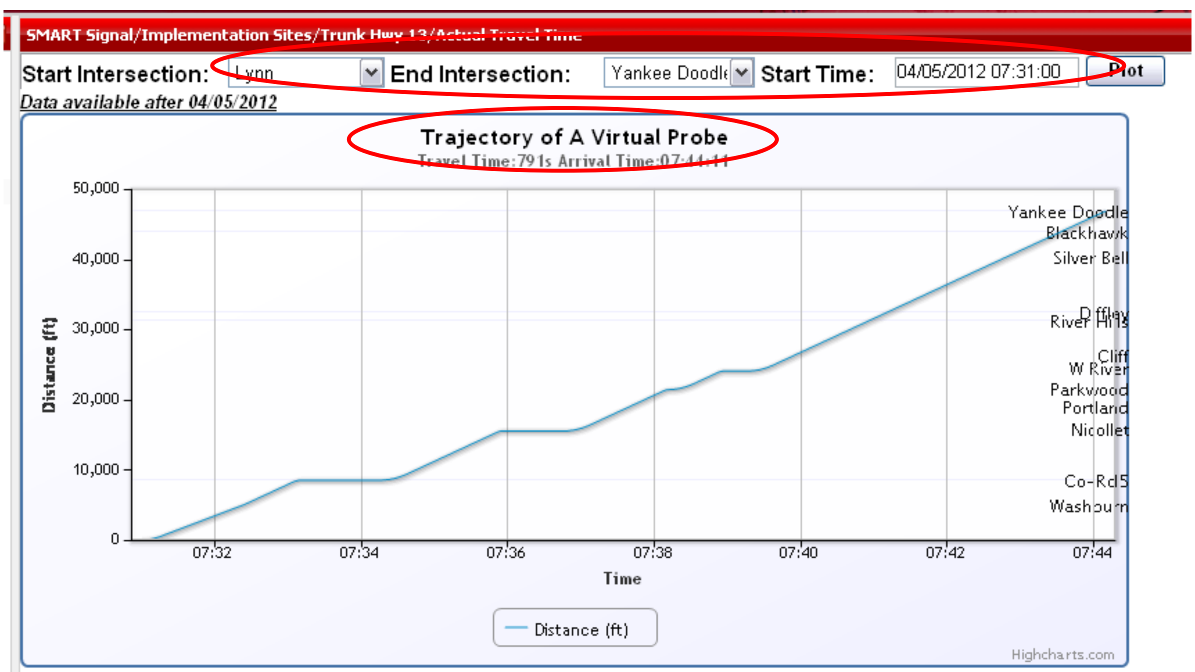


Figure 4.10: Illustration for Travel Time Calculation

(5) Intersection Delay

Once users click on the "**<u>Intersection Delay</u>**", a webpage like **Figure 4.11** will be displayed.

Figure 4.11: Webpage for Intersection Delay Calculation

Users need to first specify the "**Direction**" (i.e. Eastbound or Westbound) and "**Start Time**", see **Figure 4.12**. Click the "**<u>Plot</u>**" button, the delay for the selected direction at the specified time will be displayed for each intersection.

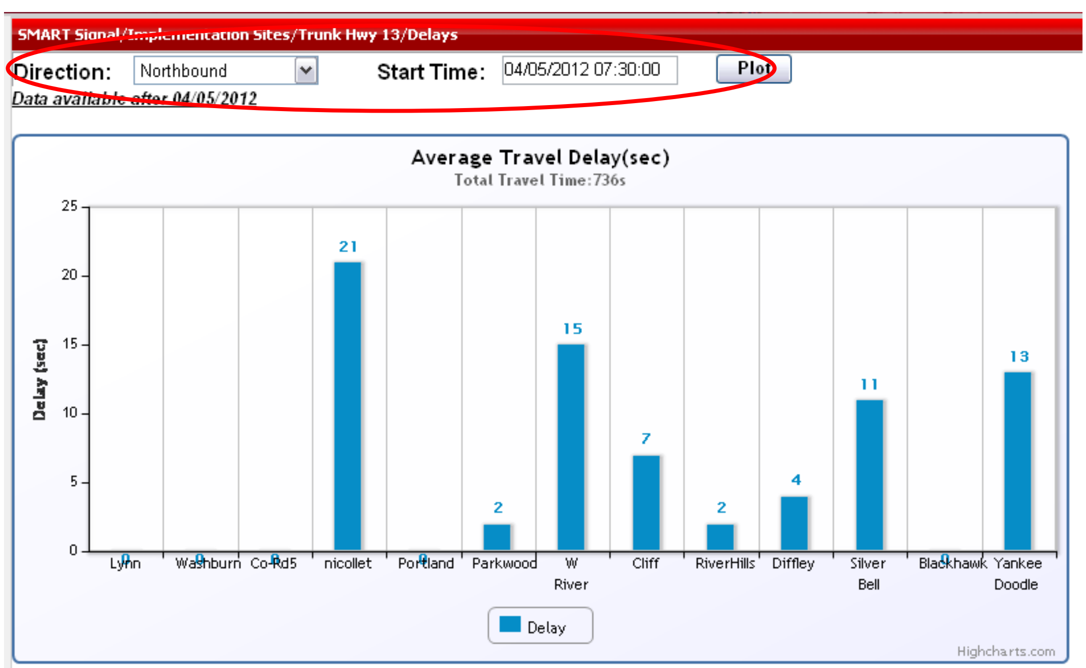


Figure 4.12: Illustration of Intersection Delay

4.2 Server Software Installation and Configuration

This section illustrates the procedure to install the required software for the SMART-SIGNAL system on the server computer. The installed software includes the **Microsoft SQL Server 2005,** the **Microsoft Internet Information Services (IIS) 6.0,** the **SMART-SIGNAL Data Server** software, the **SMART-SIGNAL Queue Calculation** software and the **SMART-SIGNAL Web Application.** In the following, the installation and configuration details will be presented.

4.2.1 Software Installation

The firmware in the SMART-SIGNAL field device (DCU) is pre-embedded before the implementation. In this section, the installation procedure of the required software on the server computer will be illustrated.

1. Install the **Microsoft SQL Server 2005**, configure the SQL Server according to the installation manual. Write down the **User Name** and **Password** to the database.
2. Generate a new database "SmartSignalDB".
3. Copy the folders "**Data Server**" and "**Queue Calculation**" to "**C:**\".
4. Open the folder "**Data Server**", there is one file named "**DatabaseInfo.txt**". Open the file and modify the database login information, "**IP Address**" of the SQL database server, "**Database Name**", "**User Name**" and "**Password**", save and close it. Do the same things to folder "**Queue Calculation**".
5. Copy the two shortcuts in "**Data Server**" and "**Queue Calculation**" folders to "**Startup**" folder, locating at "**Start**"-> "**All Programs**" -> "**Startup".**
6. Go to "**Start**"-> "**Control Panel**" -> "**Add or Remove Programs**" and select the "**Add/Remove Windows Components**". Follow the wizard to install the **Microsoft Internet Information Services (IIS) 6.0**, be sure to enable the **"ASP.Net"** service in the installation process.
7. Copy the "**SMART-SIGNAL Web Application**" to the folder of "**C:\Inetpub\wwwroot\".**
8. Open the "**IIS Manager**", right-click on "**Websites**". Go to "**New**" -> "**Website…**" and follow the wizard to generate a new website, pointing the home directory to the one generated in (7).

4.2.2 Software Interface Illustration

(1) Microsoft SQL Server

The Microsoft SQL Server 2005 was installed on the server computer to store all the data. It interconnects different parts of the SMART-SIGNAL system. There are mainly three types of tables in the database (see **Figure 4.13**), namely detector tables, signal tables and queue tables. The detector tables archive all detector actuations, the signal tables archive all signal status changes and the queue tables archive all queue information of every detector.

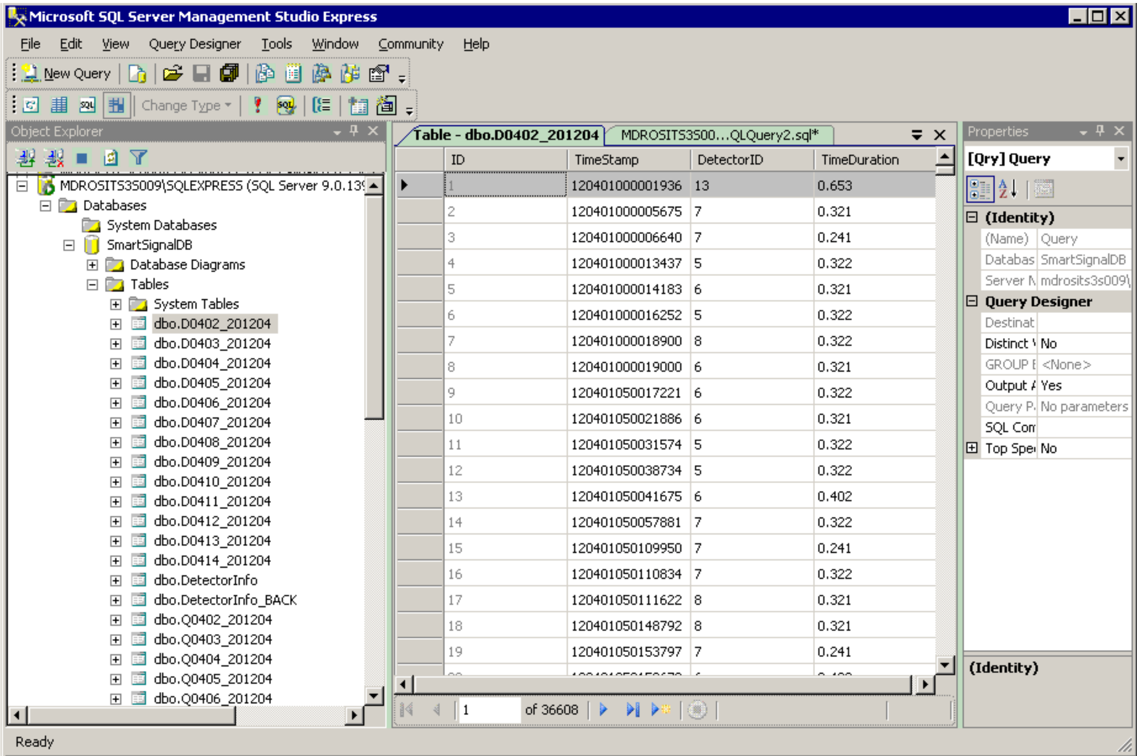


Figure 4.13: Microsoft SQL Server Management Studio Interface

(2) SMART-SIGNAL Data Server

The SMART-SIGNAL data server software is running on the server computer in the traffic management center. It is used to receive the data packages sent out by field computers and store them into the database; meanwhile, it can also be used as a system monitor to check the latest status of each intersection. The data server software will be automatically started once the system is on. Sometimes, it might be minimized to the lower right-hand side tray area, see **Figure 4.14**, the user can double click the icon to open the software interface.

Figure 4.14: The SMART-SIGNAL Data Server Software in the Tray Area

The interface is shown in **Figure 4.15**. Area 1 indicates the IP address of the server computer, area 2 lists out the data transmission status for each of the intersections (i.e. intersection ID number, total number of received events from the intersection in the current day, IP address of specific intersection and the date and time when the last data package was received) and area 3 indicates the time when each intersection is connected or disconnected with the server.

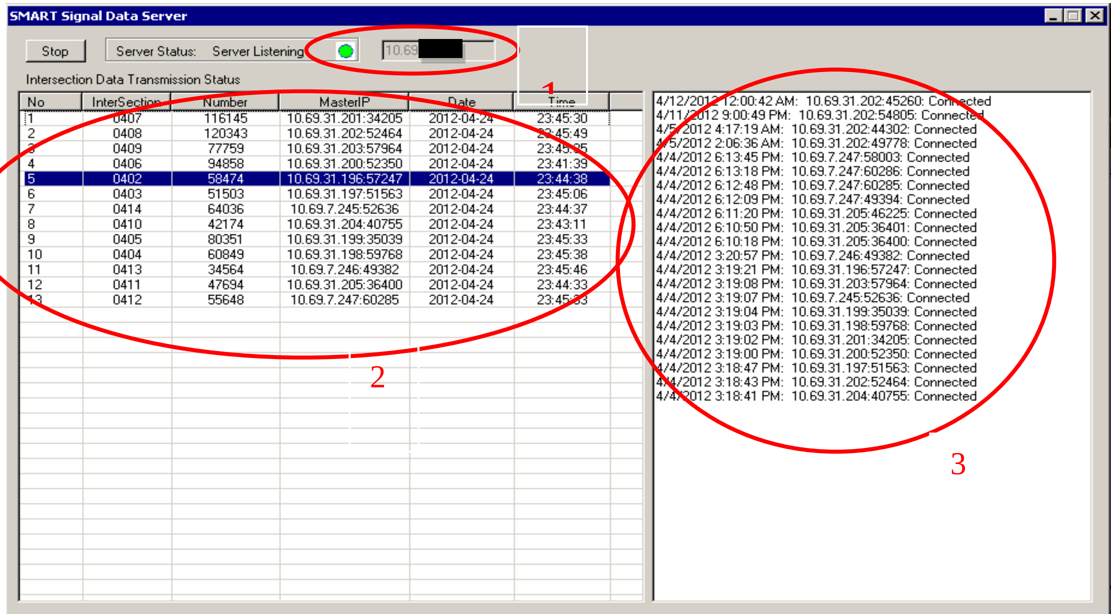


Figure 4.15: The SMART-SIGNAL Data Server Interface

(3) SMART-SIGNAL Queue Calculation Software

The queue calculation software (see **Figure 4.16**) generates the real-time queue information for each detector of each intersection and stores it into the database. The software will be automatically started once the system is on. Users can see the software running once they log into the system. The queue calculation software stays running all the time since queue length information is needed for the web interface. From the console window, users can see the most up-to-date information of queue calculation for each intersection.

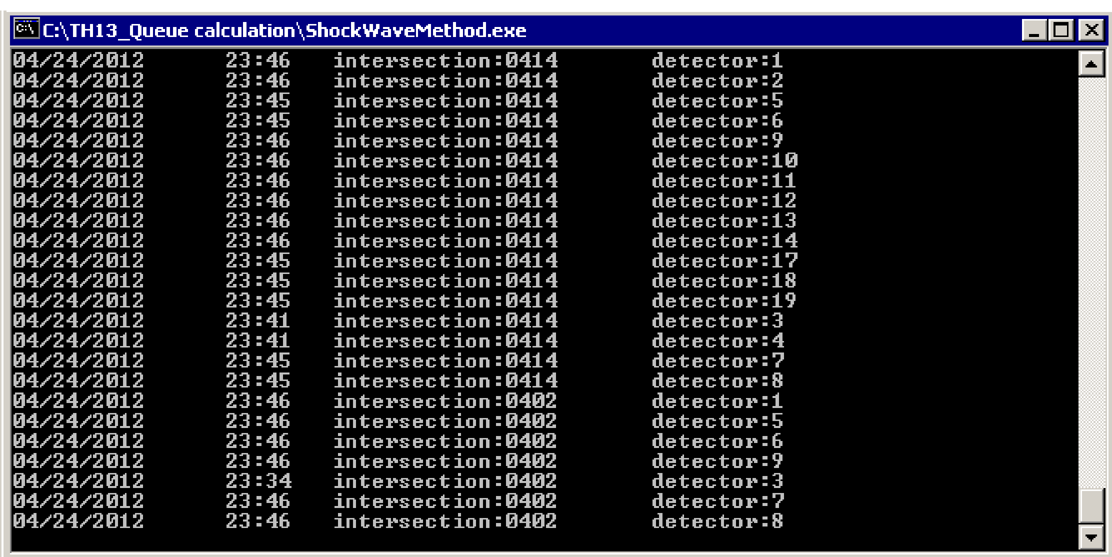


Figure 4.16: SMART-SIGNAL Queue Calculation Software

4.2.3 Internet Information Services (IIS) Configuration

The Internet Information Services (IIS) needs to be properly configured to allow both internal and external access to the SMART-SIGNAL website. As shown in **Figure 4.17**, the name of the website is “SmartSignal” and the IP address to the website is “**10.69.3.38**”. The TCP port is set to “**80**” and the SSL port is set to “**443**”, which provide “**Http**” and “**Https**” services respectively.

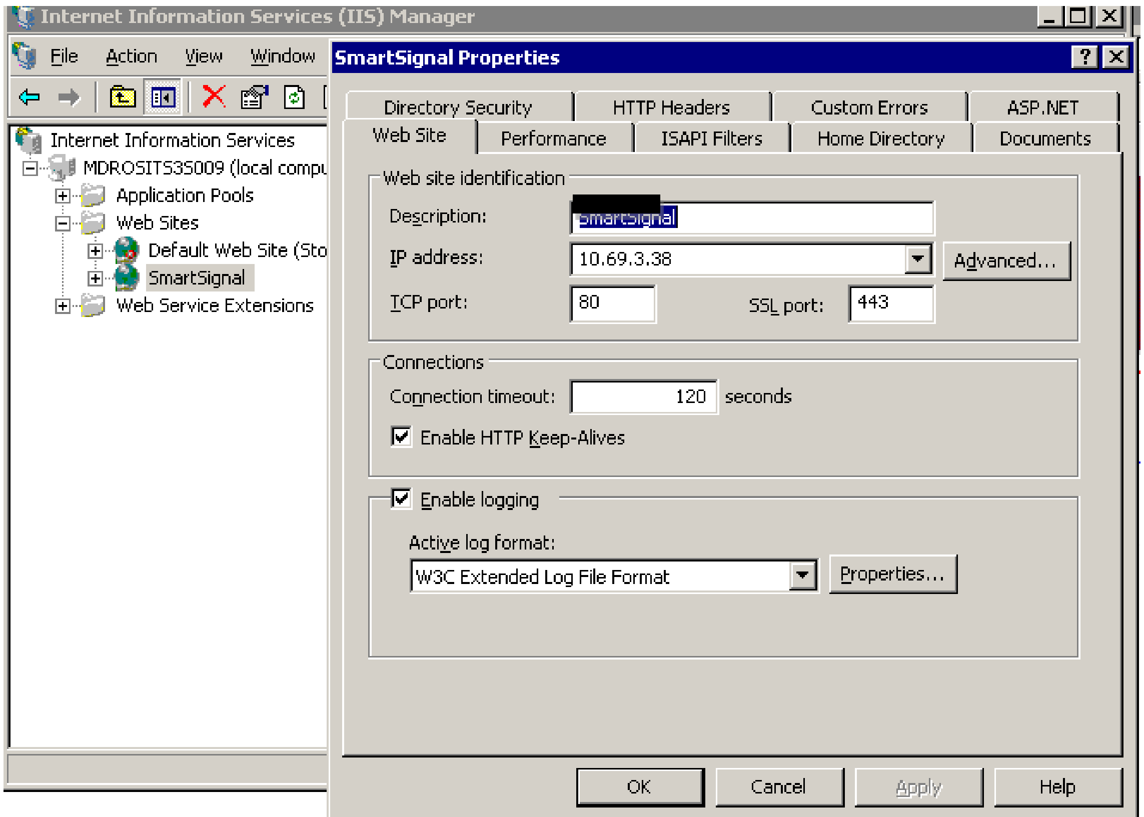


Figure 4.17: The Internet Information Services (IIS) Settings

Since the SMART-SIGNAL website is hosted behind the MnDOT firewall, in order to make the web publicly accessible (i.e., from a public Internet outside the MnDOT network) and at the same time make sure the safety of the website and network, following procedures need to be applied:

1. First of all, in the IIS a virtual directory named “smartsignal” is created and all the web application files are copied into this folder. This allows the website to be accessed by using “ http://10.69.*.*/smartsignal”, as showing in **Figure 4.18**.

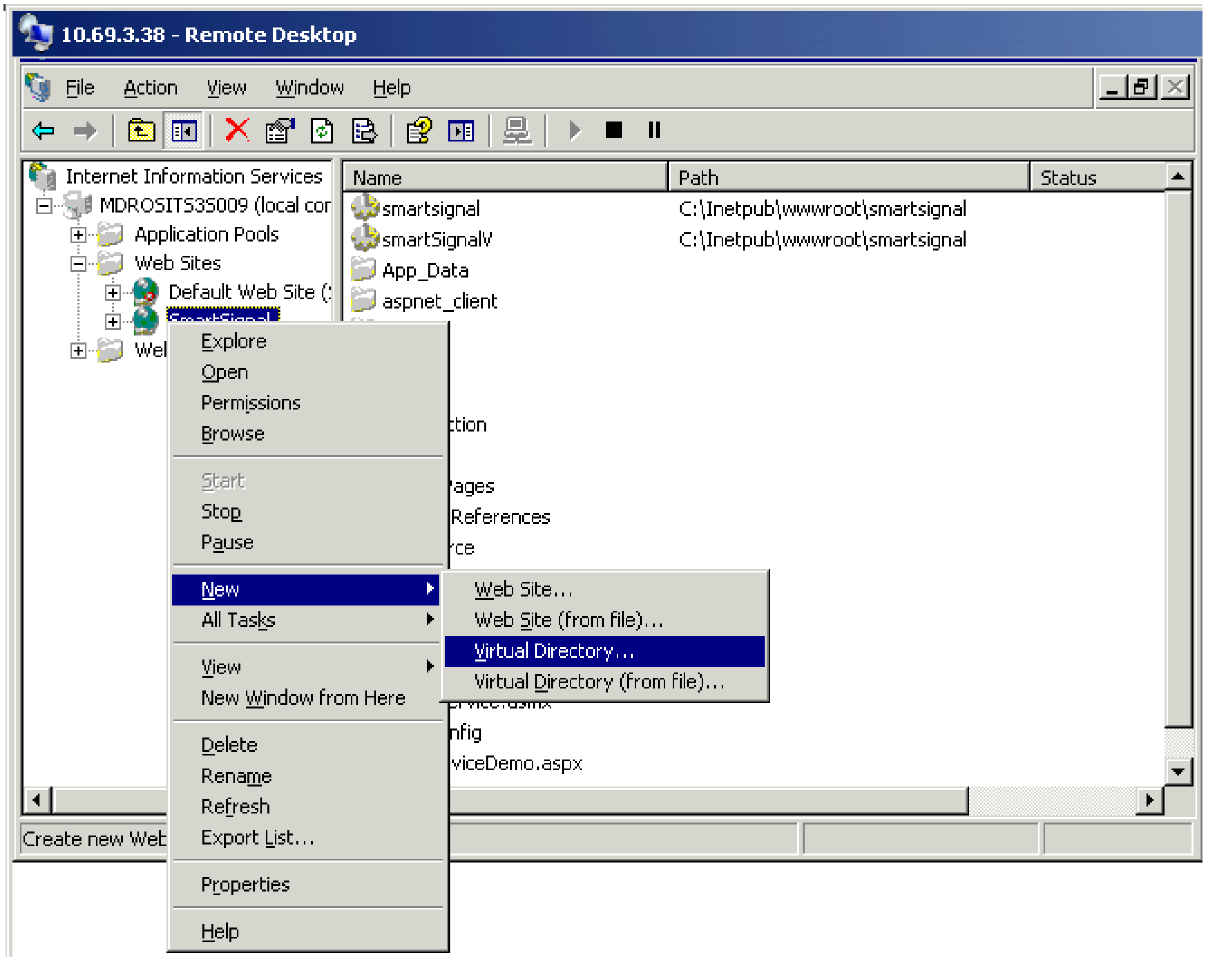


Figure 4.18: Illustration of Generating Virtual Directory in IIS

2. Then, a reverse proxy approach is utilized to map the SMART-SIGNAL website (http://10.69.*.*/smartsignal) to http://dotapp4.dot.state.mn.us/smartsignal. As shown in **Figure 4.19** , a reverse proxy takes requests from the Internet and forwards them to servers in an internal network; on the other hand, it gets the response from the web server and passes it to the Internet. This will protect the internal network and web server from outside attack, since users can only make requests directly to the proxy and have no direct access to the internal server. By doing so, the SMART-SIGNAL website can be accessed internally by using http://10.69.*.* or "http://10.69.*.*/smartsignal", and externally by using "http://dotapp4.dot.state.mn.us/smartsignal".

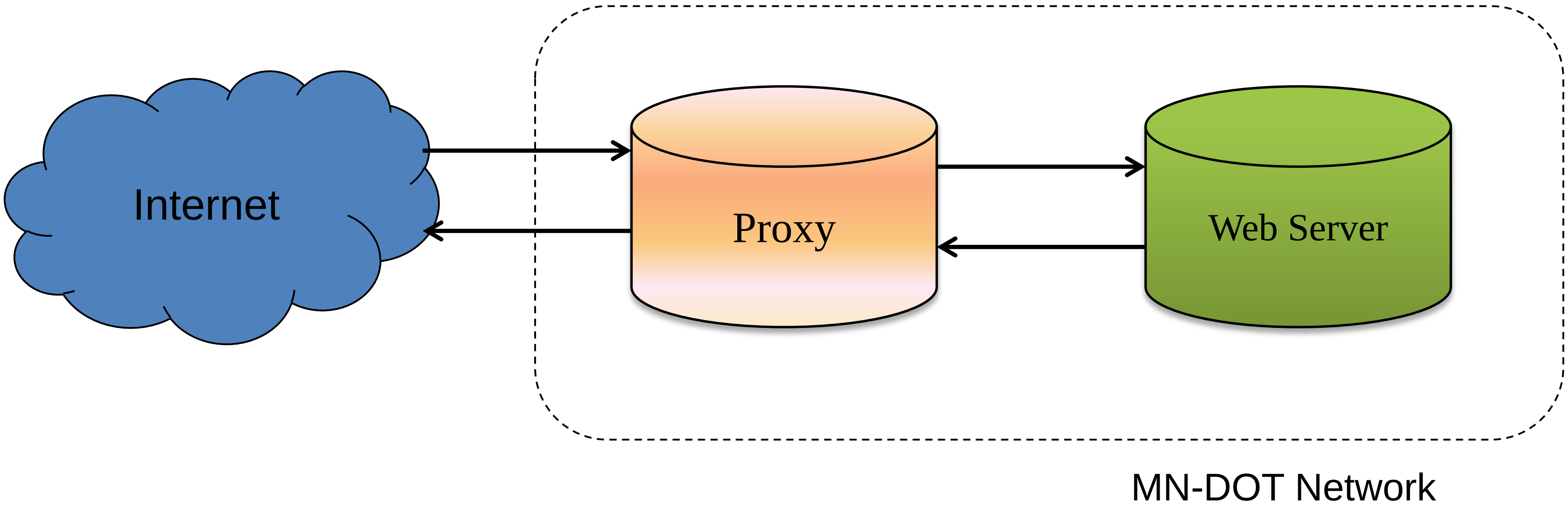


Figure 4.19: Illustration of the Reverse Proxy Approach

Chapter 5: Concluding Remarks

Efficient traffic signal operation is vital for smooth traffic flows in signalized network, and well-maintained signal operation is greatly beneficial for road users. Data collection and performance monitoring are critical for operation improvements of a traffic signal system, and yet most agencies involved in operating traffic signal systems do not archive and analyze traffic performance data. With the increasing congestion on traffic network nowadays, it becomes even more urgent. These facts lead to a data-poor situation for signalized traffic networks and means missed the opportunity to improve operation. This is the motivation of the development of the SMART-SIGNAL system. The previous development and deployment of SMART-SIGNAL system have successfully demonstrated its great potentials with accurately estimated queue length and travel time at both intersection level and arterial level. This project goes one step further to address the implementation issues for SMART-SIGNAL system with a re-designed DCU and system interface.

Utilizing high performance microcontrollers, the DCU is designed as an independent add-on module that can serve as a flexible and cost-effective solution to enhance data collection capability for existing traffic signal systems. Two types of DCUs are developed for interfacing with TS-1 cabinet and TS-2 cabinet. TS-1 DCU connects with contact closures on the back panel in TS-1 cabinet. TS-2 DCU interfaces with SDLC bus in TS-2 cabinet to obtain traffic event data. Both types of DCUs are designed as compact embedded devices that greatly ease the installation difficulty within the cabinet. The ultimate goal for development of DCUs is to achieve plug-and-play capability at the field intersections and to reduce the effort of customized installation as much as possible.

The manufacturing process and implementation at 13 intersections along TH13 are explained in detail, and can be referred to be guidance for agencies to customize and produce small quantities of TS-2 DCUs.

A refined website with public access is highlighted as the user friendly system interface for SMART-SIGNAL system. Via this website, users can access performance measurement data, including historical travel time, real-time travel time, historical queue length and real-time queue length conveniently. The installation and configuration of a server side software package is described in detail so that the system can be easily maintained by the agency.

Compared with the previous version of the SMART-SIGNAL system, the refined one described in this project is much easier to install and access. The manufacturing cost is also much lower. Thus transportation agencies could use the design described in this project to monitor the performance of their signalized arterials on a large scale. For demonstration purposes, the manufacturing process and implementation on 13 intersections along TH13 are described in detail, and can be referred to for guidance by the agencies.

References

Ahmed, A. & J. Brian (2008). *An Intersection Traffic Data Collection Device Utilizing Logging Capabilities of Traffic Controllers and Current Traffic Sensors*. NIATT. Moscow, ID.

Balke, K., H. Charara, & R. Parker. (2005). *Development of a Traffic Signal Performance Measurement System (TSPMS).* Texas Transportation Institute. College Station, TX.

Balke, K. & C. Herrick. (2004). *Potential Measures of Assessing Signal Timing Performance Using Existing Technologies*. Texas Transportation Institute. College Station, TX.

Birst, S. & A. Smadi. (2006). "Use of Hardware-in-the-Loop Traffic Simulation in a Virtual Environment." *Applications of Advanced Technology in Transportation: Proc. 9th International Conference*, ASCE, Chicago, IL, 635–640.

Bullock, D. & A. Catarella (1998). "A Real-Time Simulation Environment for Evaluating Traffic Signal Systems." Paper presented at the 77th Annual Transportation. Research Board Meeting, Washington D.C.

Day, C.M., D.M. Bullock, & J.R. Sturdevant (2009). "Cycle-Length Performance Measures: Revisiting and Extending Fundamentals." *Transportation Research Record*, 2128: 48-57.

Day, C.M., R. Haseman, H. Premachandra, T.M. Brennan, Jr., J.S. Wasson, J.R. Sturdevant, & D.M. Bullock (2010 a). "Evaluation of Arterial Signal Coordination." *Transportation Research Record*, 2192: 37–49.

Day, C.M., J.R. Sturdevant, & D.M. Bullock (2010 b). "Outcome-Oriented Performance Measures for Management of Signalized Arterial Capacity." *Transportation Research Record*, 2192: 24–36.

Digi International Inc. ConnectCore® 9M 2443. Available at: http://www.digi.com/products/wireless-wired-embedded-solutions/solutions-on-module/connectcore/connectcore-9m2443#overview. Accessed on Aug 3, 2012.

Econolite Group Inc. (2012). ASC3 controller specification. Available online at: http://www.econolite.com/assets/pdf/controller_asc3_specification.pdf. Accessed on Aug 3, 2012.

Liu, H. & W. Ma (2009). "A virtual vehicle probe model for time-dependent travel time estimation on signalized arterials." *Transportation Research Part C*, 17 (1): 11-26.

Liu, H., W. Ma, X. Wu, & H. Hu (2009 a). *Development of a Real-Time Arterial Performance Monitoring System Using Traffic Data Available from Existing Signal Systems*. MNDOT. St Paul, MN.

Liu, H., X. Wu, W. Ma, & H. Hu (2009 b). "Real-time queue length estimation for congested signalized intersections." *Transportation Research Part C*, 17 (4): 412–427.

Luyanda, F., D. Gettman, L. Head, S. Shelby, D. Bullock, & P. Mirchandani (2003). "ACS-Lite Algorithmic Architecture: Applying Adaptive Control System Technology to Closed-Loop Traffic Signal Control Systems." *Transportation Research Record*, 1856: 175–184.

National Electrical Manufacturers Association. (2003). *NEMA Standards Publication TS 2*. National Electrical Manufacturers Association. Rosslyn, VA.

National Transportation Operations Coalition. (2007). *2007 National Traffic Signal Report Card-Technical Report*. National Transportation Operations Coalition. Washington, D.C.

National Transportation Operations Coalition. (2012). *2012 National Traffic Signal Report Card-Technical Report*. National Transportation Operations Coalition. Washington, D.C.

NXP Semiconductors. (2010). UM10360-LPC 17XX User manual. Available online at: http://www.nxp.com/documents/user_manual/UM10360.pdf. Accessed on Aug 3, 2012

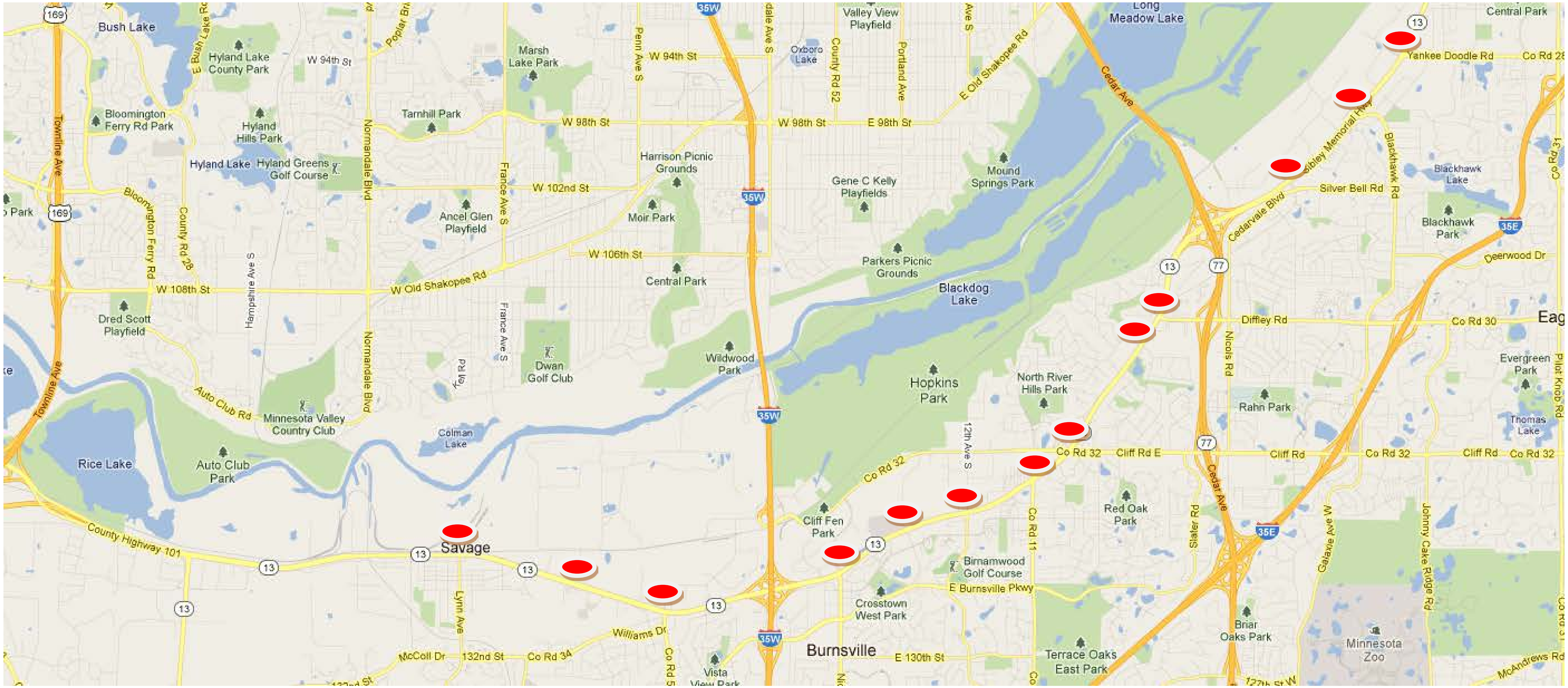
Petty, K., J. Kwon, & A. Skabardonis (2005). "A-PeMS: An Arterial Performance Measurement System." 2006 Annual Meeting Workshop. Washington, D.C.

Smaglik, E.J., A. Sharma, & D.M. Bullock. (2007). "Event-Based Data Collection for Generating Actuated Controller Performance Measures." *Transportation Research Record*, 2035: 97–106.

Sunkari, S.R., H.A. Charara, & P. Songchitruksa (2011). *A Portable Toolbox to Monitor and Evaluate Signal Operations*. Texas Transportation Institute. College Station, TX.

Wu, X., H. Liu, & D. Gettman (2010). "Identification of Oversaturated Intersections Using High-Resolution Traffic Signal Data." *Transportation Research Part C*, 18 (4): 626–638.

Appendix A: Locations of Implementation Sites on TH 13



: Implementation site

[Map Source: Google Maps - ©2012 Google]

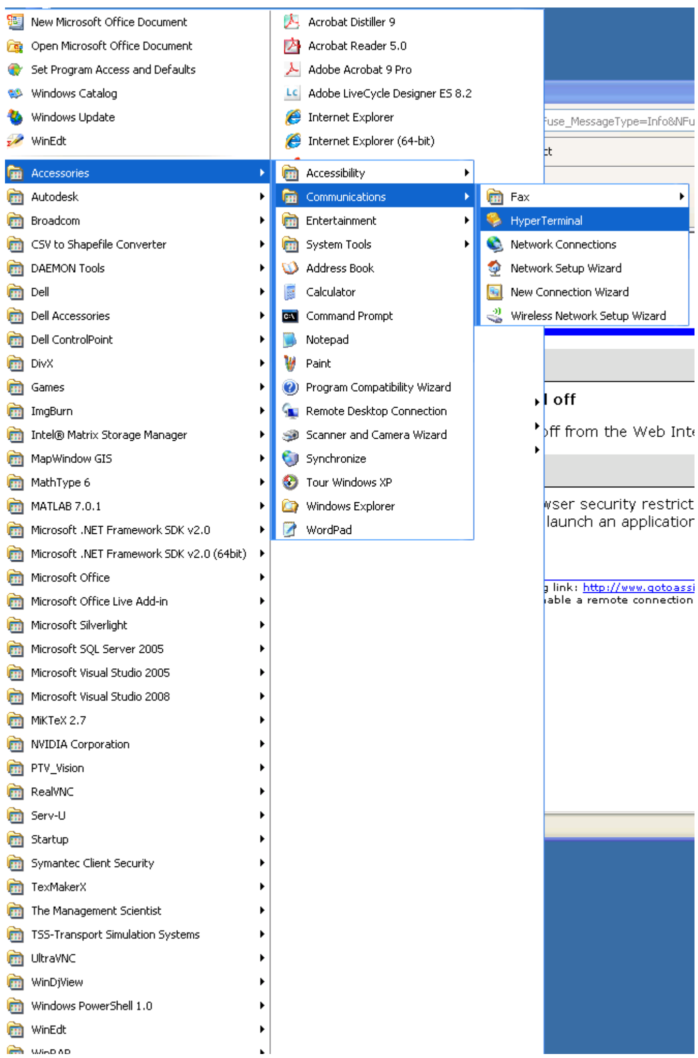
Appendix B: Procedure of Configuring HyperTerminal for DCU Connection

We need to configure the Serial Port setting from a PC or Laptop (running Windows OS) as the following to connect to COM of DCU. Figures below show the step by step instruction of how to set the Hyper Terminal.

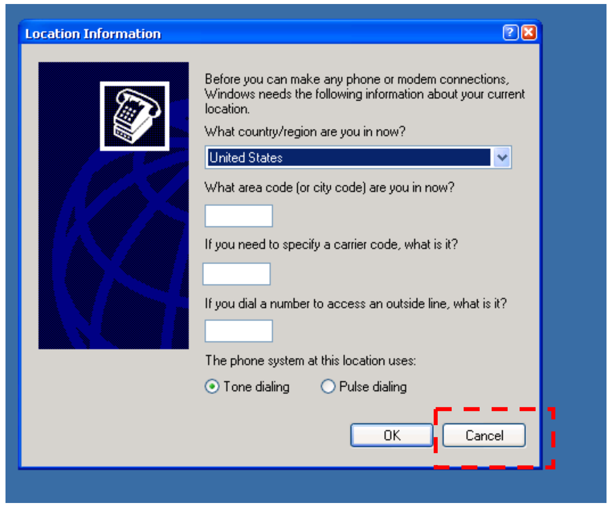
The port settings are:

- Bit per sec: 38400
- Data bit: 8
- Stop bit: 1
- Flow control: None.
- (Follow the instruction circled by:)

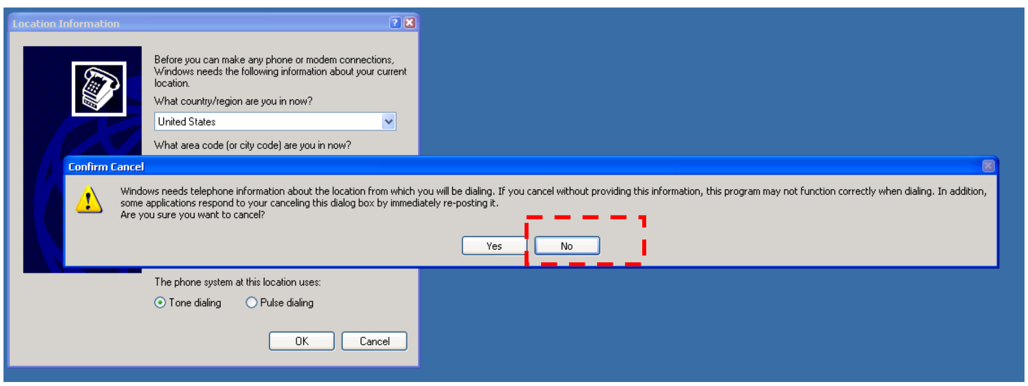
Step 1



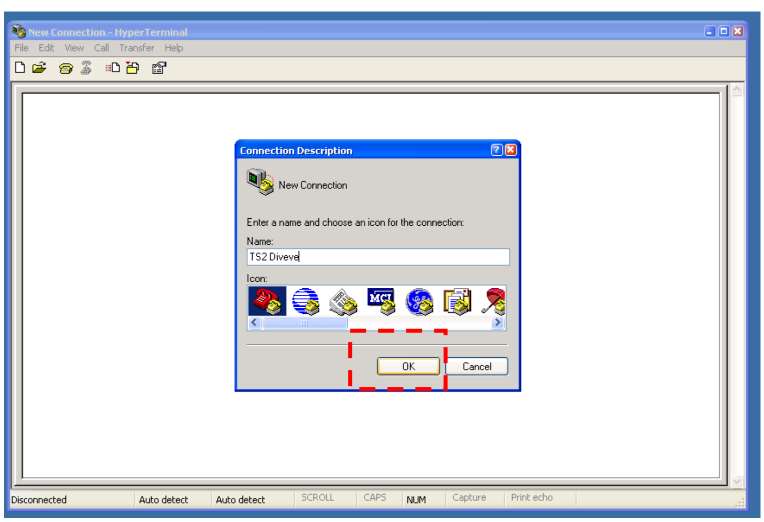
Step 2



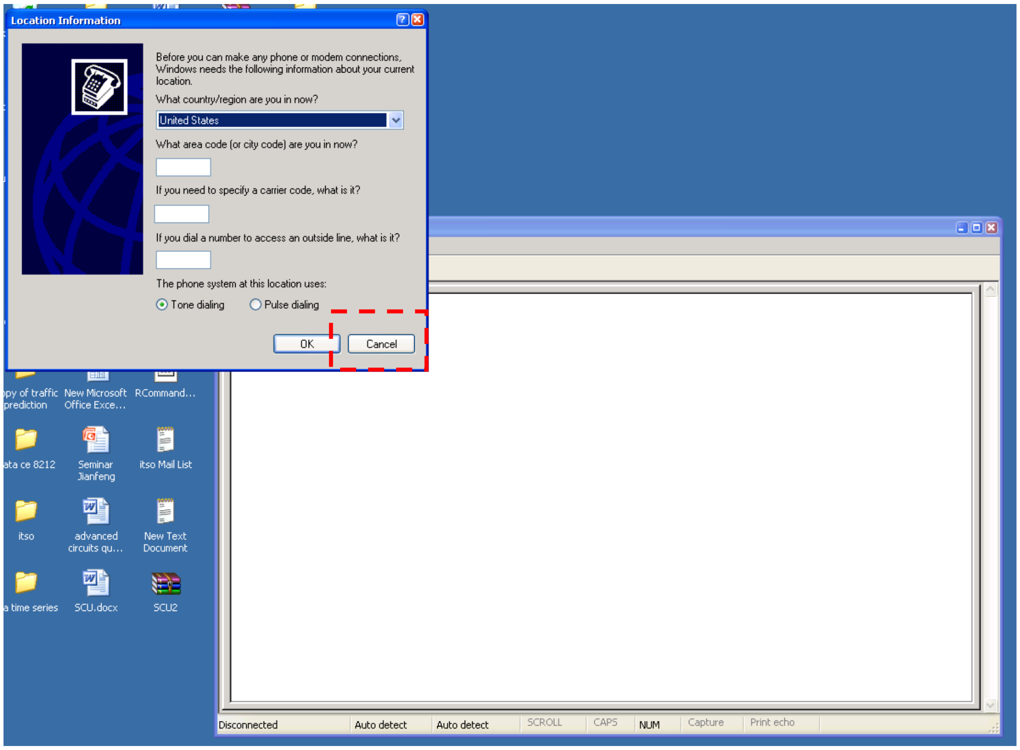
Step 3



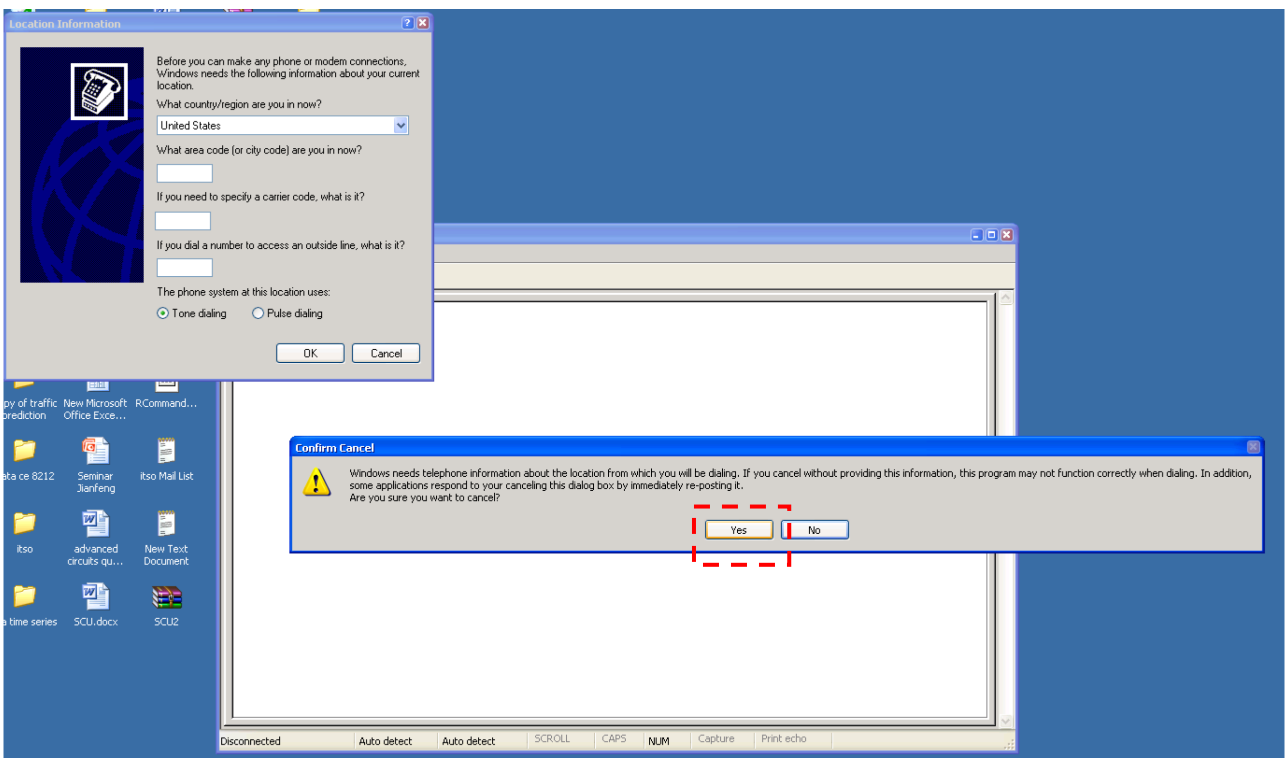
Step 4



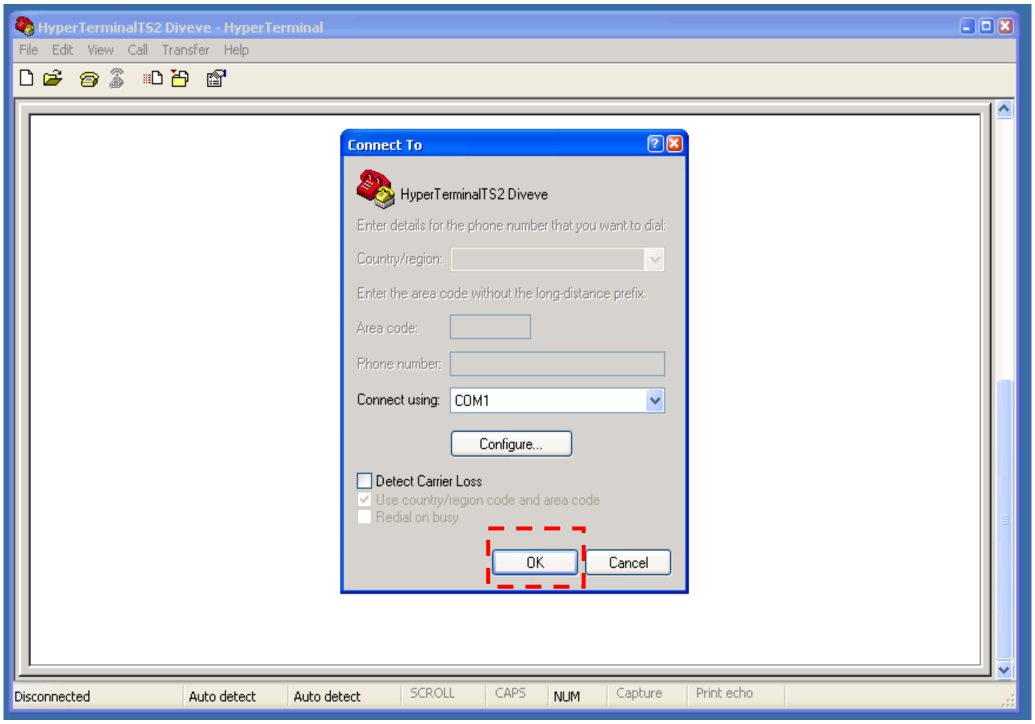
Step 5


Location Information
Before you can make any phone or modem connections, Windows needs the following information about your current location.
What country/region are you in now?
United States
What area code (or city code) are you in now?
If you need to specify a carrier code, what is it?
If you dial a number to access an outside line, what is it?
The phone system at this location uses:
Tone dialing
Pulse dialing
OK
Cancel
Disconnected
Auto detect
Auto detect
SCROLL
CAPS
NUM
Capture
Print echo

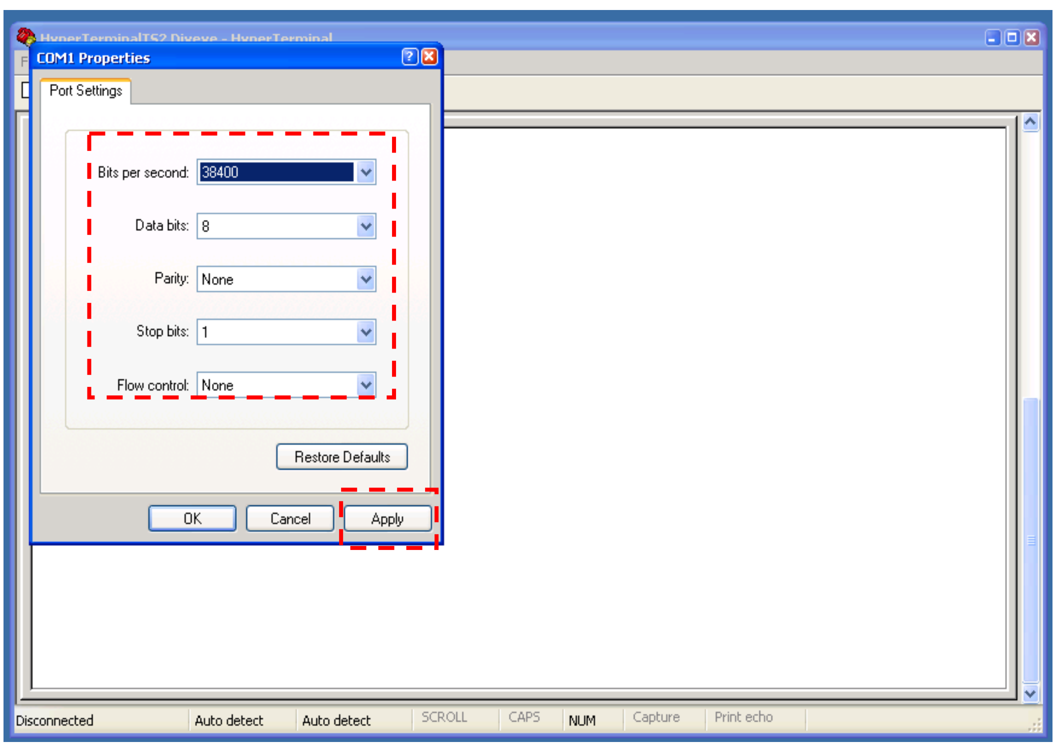
Step 6


Location Information
Before you can make any phone or modem connections, Windows needs the following information about your current location.
What country/region are you in now?
United States
What area code (or city code) are you in now?
If you need to specify a carrier code, what is it?
If you dial a number to access an outside line, what is it?
The phone system at this location uses:
Tone dialing
Pulse dialing
OK
Cancel
Confirm Cancel
Windows needs telephone information about the location from which you will be dialing. If you cancel without providing this information, this program may not function correctly when dialing. In addition, some applications respond to your canceling this dialog box by immediately re-posting it.
Are you sure you want to cancel?
Yes
No
Disconnected
Auto detect
Auto detect
SCROLL
CAPS
NUM
Capture
Print echo

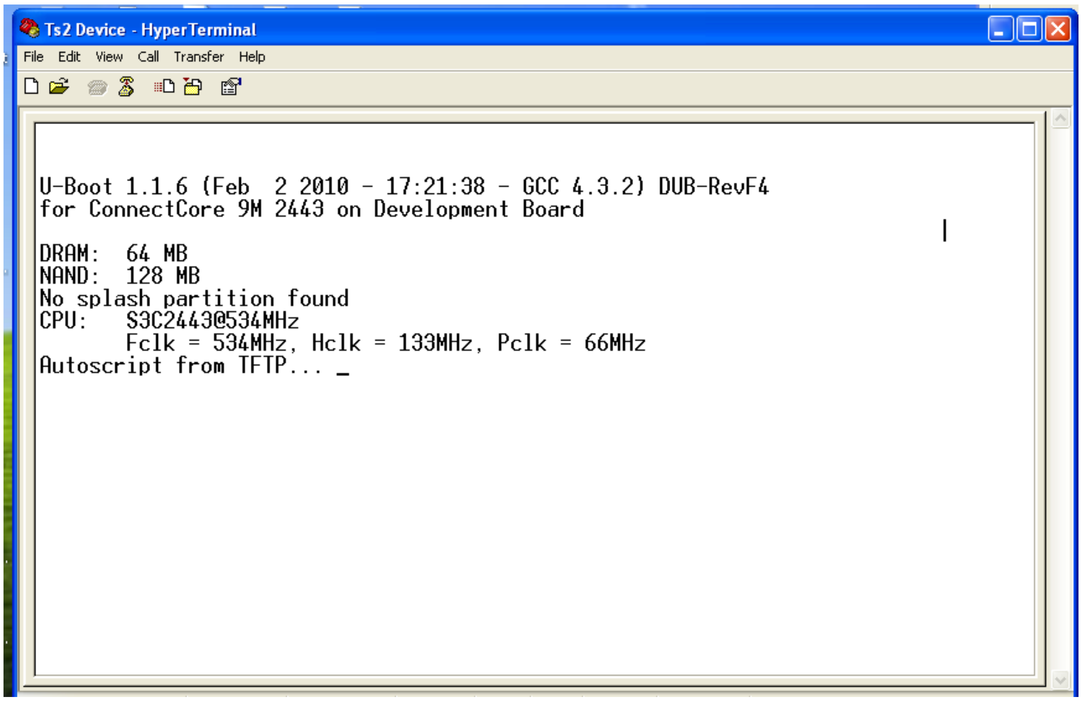
Step 7


HyperTerminalTS2 Diveve - HyperTerminal
File Edit View Call Transfer Help
Connect To
HyperTerminalTS2 Diveve
Enter details for the phone number that you want to dial:
Country/region:
Enter the area code without the long-distance prefix.
Area code:
Phone number:
Connect using:
COM1
Configure...
Detect Carrier Loss
Use country/region code and area code
Redial on busy
OK
Cancel
Disconnected
Auto detect
Auto detect
SCROLL
CAPS
NUM
Capture
Print echo

Step 8


HyperTerminalTS2 Diveve - HyperTerminal
COM1 Properties
Port Settings
Bits per second:
38400
Data bits:
8
Parity:
None
Stop bits:
1
Flow control:
None
Restore Defaults
OK
Cancel
Apply
Disconnected
Auto detect
Auto detect
SCROLL
CAPS
NUM
Capture
Print echo

Step 9 (When DCU booting up, the console will output as follows)

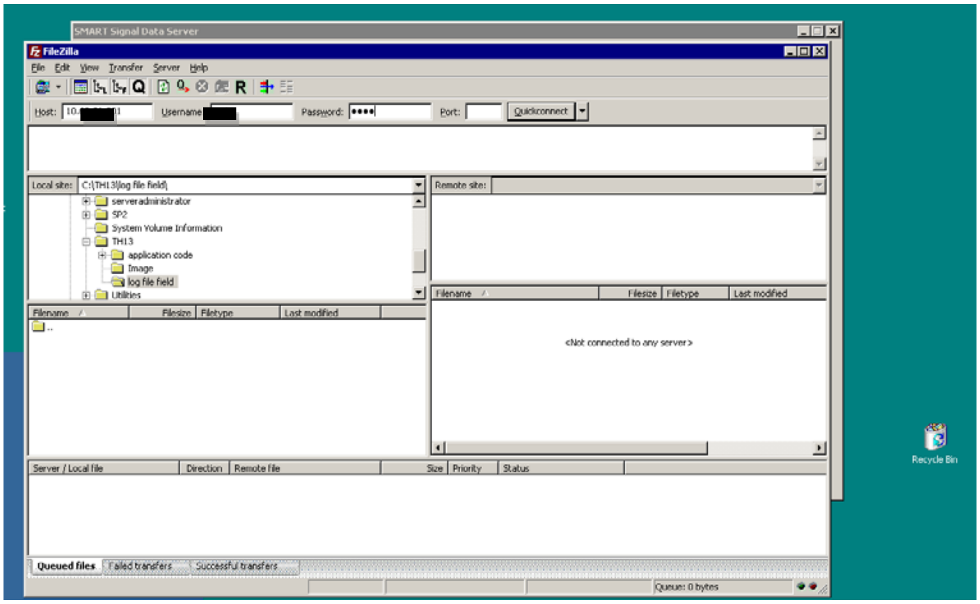


Appendix C: Illustration of Accessing DCU from Server using FTP and Telnet

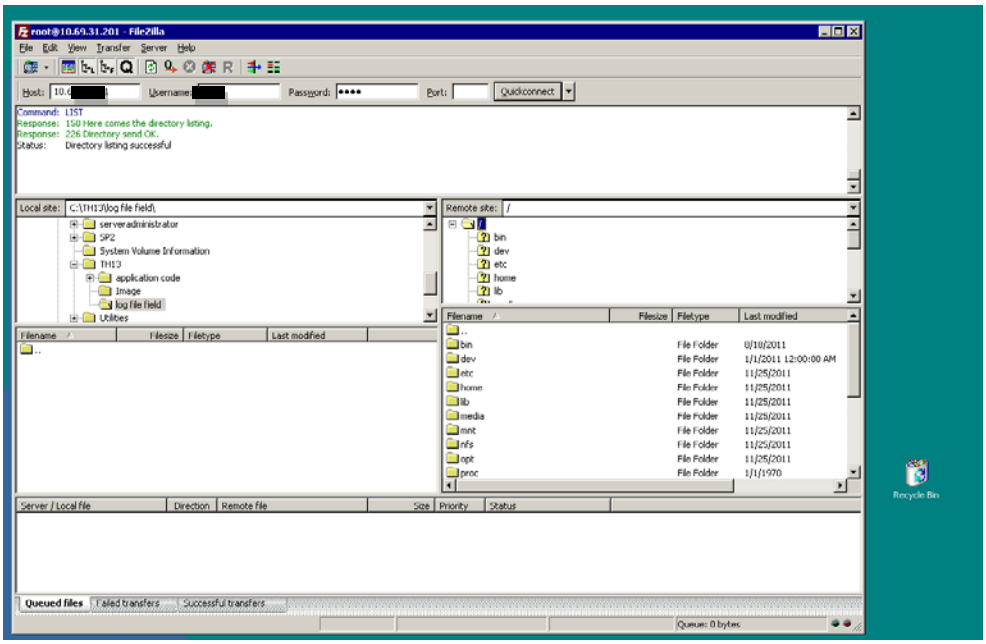
A. FTP Connection

A FTP server is running in the DCU, making it possible to transfer file with server. Figures below state how to connect with DCU using FileZilla.

Step 1



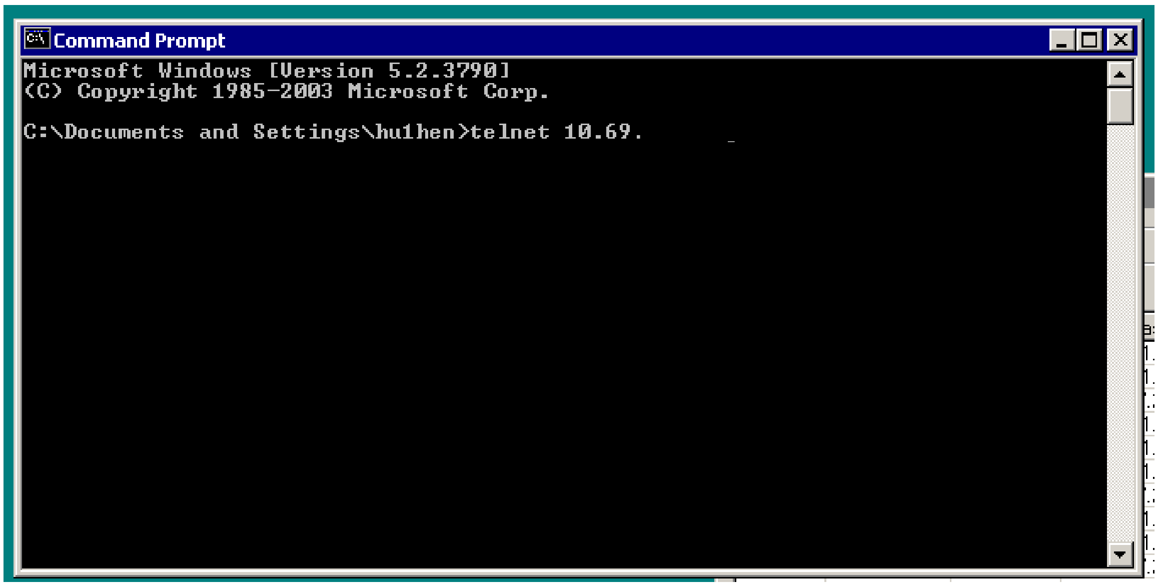
Step 2



B. Telnet connection with DCU

The Linux OS in the DCU can be logged into via telnet, and thus configured remotely. No username and password are required. Figures below show that how to connect DCU using telnet command provided by Window Server.

Step 1



Step 2