โครงงานทางวิทยาศาสตร์
เรื่อง

ระบบควบคุมและติดตามการจราจรอัตโนมัติ

An Automatic Traffic Control and Monitoring System

โดย

นาย กิตติพงศ์ เตชะพานิชกุล 4232017323

นาย จักกนาท วิวัฒนาวรสิน 4232044223

นาย ณัฐพล พงษ์อดุลยสุข 4232100723

โครงการนี้เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี

สาขาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์

จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2545

โครงงาน	ระบบควบคุมและติดตามการจราจรอัตโนมัติ
	An Automatic Traffic Control And Monitoring System
โดย	นาย กิตติพงศ์ เตชะพานิชกุล เลขประจำตัว 4232017323
	นาย จักกนาท วิวัฒนาวรสิน เลขประจำตัว 4232044223
	นาย ณัฐพล พงษ์อดุลยสุข เลขประจำตัว 4232100723
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. พีระพนธ์ โสพัศสถิตย์

ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับโครงงานนี้เป็นส่วนหนึ่งของการศึกษาปริญญาตรี ในรายวิชา 2301499 SENIOR PROJECT

(รองศาสตราจารย์ ดร. วนิดา เหมะกุล)
หัวหน้าภาควิชาคณิตศาสตร์

(ผู้ช่วยศาสตราจารย์ ดร. พีระพนธ์ โสพัศสถิตย์)
อาจารย์ที่ปรึกษา

(รองศาสตราจารย์ พินิจ เพิ่มพงศ์พันธ์)
กรรมการสอบ

(อาจารย์ ดร. ไพศาล นาคมมหาชลาสินธุ์)
กรรมการสอบ

บทคัดย่อ

ในปัจจุบันความต้องการในการใช้รถยนต์ส่วนบุคคลในประเทศไทยมีปริมาณเพิ่มมากขึ้น ส่งผลให้การจราจรในถนนสายต่าง ๆ ติดขัด จนกลายเป็นปัญหาสำคัญของประเทศไปในที่สุด ซึ่งสำนักการจราจรและขนส่งได้หาทางแก้ปัญหานี้โดยใช้ระบบควบคุมสัญญาณไฟจราจรเป็นพื้นที่ หรือ ATC (Area Traffic Control) ที่นำระบบ SCOOT มาใช้ ซึ่งระบบนี้เป็นระบบที่ใช้กันอย่างแพร่หลายในหลาย ๆ ประเทศ เนื่องจากสามารถทำงานได้โดยใช้คอมพิวเตอร์ แต่ระบบดังกล่าวเป็นของต่างประเทศและจำเป็นต้องติดตั้งอุปกรณ์ที่ใช้ในการเก็บข้อมูลในทุก ๆ ทางแยก ทำให้ประเทศไทยต้องเสียงบประมาณกับระบบนี้เป็นจำนวนมาก ทางคณะผู้จัดทำได้เล็งเห็นถึงปัญหานี้จึงได้คิดค้นระบบที่จะนำมาใช้แทนระบบ SCOOT

ในโครงงานนี้ทางคณะผู้จัดทำได้สร้างระบบที่สามารถจำลองสถานการณ์การจราจรบนท้องถนนโดยพัฒนาให้ผู้ใช้สามารถศึกษาถึงวิธีการเปิด-ปิดสัญญาณไฟจราจรในสถานการณ์ที่จำลองขึ้นและนำสถิติมาวิเคราะห์หาวิธีการควบคุมสัญญาณไฟที่เหมาะสมที่สุดของถนนแต่ละสาย อีกทั้งยังสามารถนำข้อมูลจากสถิติไปประยุกต์ใช้ในงานด้านอื่น ๆ ต่อได้ นอกจากนี้ยังช่วยลดภาระค่าใช้จ่ายในการใช้ระบบ SCOOT อีกด้วย

Abstract

Nowadays, the demand of using private car grows rapidly and affects the traffic in many roads. That becomes one of the most important problems of Thailand. Although Bangkok Traffic and Transportation Department solves the problem using computerized Area Traffic Control (SCOOT) that is used in many countries, the annual cost of operating ATC that Thai government has to pay is very expensive. In order to reduce the SCOOT operating cost, we invent a new system that can be used in place of SCOOT.

In this project, we model the situation of the road so that the user can study and analyze traffic control statistics in order to arrive at the most appropriate way for traffic light control. Furthermore, the user can use these statistics for other purposes. As a consequence, SCOOT operating cost will reduce considerably.

กิตติกรรมประกาศ

โครงงานนี้จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดีหากปราศจากความช่วยเหลืออย่างดียิ่งของอาจารย์ที่ปรึกษาโครงงาน ผศ.ดร.พีระพนธ์ โสพัศสถิตย์ ซึ่งท่านได้ให้การดูแล แนะนำ สั่งสอน ให้ข้อมูลเพิ่มเติมที่เป็นประโยชน์อย่างยิ่งในการพัฒนาและปรับปรุงโครงงาน ระบบควบคุมและติดตามการจราจรอัตโนมัติ นี้

ขอขอบคุณ สำนักการจราจรและขนส่ง กรุงเทพมหานคร ที่เอื้อเฟื้อข้อมูลปริมาณการจราจรบริเวณทางแยกที่อยู่ภายในขอบเขตศึกษา

ขอขอบคุณ ผศ.ดร.สรวิศ นฤปิติ อาจารย์คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่จุดประกายแนวทางและให้เอกสารที่เป็นประโยชน์แก่ทางทีมงานพัฒนา

ขอขอบคุณ อาจารย์ทุกท่านที่ให้ความรู้ ความเข้าใจในสรรพวิชา ที่สามารถนำมาประยุกต์ใช้ในการแก้ไขปัญหาต่าง ๆ ที่ประสบระหว่างการทำงานให้ผ่านพ้นไปได้ด้วยดี

ขอขอบคุณ คุณดรุณี ที่คอยให้คำแนะนำและช่วยเหลือทีมพัฒนาให้สามารถพัฒนาผลงานได้อย่างเต็มความสามารถ

ขอขอบคุณ น้ำใจและมิตรภาพจากเพื่อน ๆ ทุกคน ที่คอยอำนวยความสะดวกในด้านต่าง ๆ อยู่เสมอ

ท้ายสุดนี้ ผู้จัดทำขอกราบขอบพระคุณ บิดา มารดา ผู้ให้กำเนิด ให้การอบรมสั่งสอน ให้การสนับสนุนด้านการเงิน การศึกษา และคอยให้กำลังใจเสมอมาจนสำเร็จการศึกษา

สารบัญ

	หน้า
หน้าอนุมัติผลการเรียน	ก
บทคัดย่อ	ข
Abstract	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญรูปภาพ	ช
สารบัญตาราง	ฎ
บทที่ 1 บทนำ	
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตโครงงาน	2
1.4 ขั้นตอนการพัฒนาระบบ	3
บทที่ 2 แนวคิดในการสร้างระบบควบคุมและติดตามการจราจรอัตโนมัติ	
2.1 รูปแบบการจราจรในความเป็นจริง	4
2.2 รูปแบบการจราจรในระบบควบคุมและติดตามการจราจรอัตโนมัติ	4
บทที่ 3 การวิเคราะห์ข้อมูลของระบบ	
3.1 ปริมาณการจราจรในชั่วโมงต่าง ๆ ของแต่ละถนน	9
3.2 ปริมาณรถสะสมในชั่วโมงต่าง ๆ ของแต่ละถนน	15
3.3 แผนที่ในบริเวณพิจารณา	15
3.4 ความสัมพันธ์ระหว่างทางแยกต่าง ๆ	15
3.5 วิธีการควบคุมสัญญาณไฟจราจรบริเวณทางแยก	16
3.6 รายละเอียดของแต่ละถนน	17
บทที่ 4 การออกแบบฐานข้อมูลและการออกแบบระบบควบคุมและติดตามการจราจรอัตโนมัติ	
4.1 การออกแบบฐานข้อมูลที่ใช้ในระบบควบคุมและติดตามการจราจรอัตโนมัติ	18
4.2 การออกแบบระบบควบคุมและติดตามการจราจรอัตโนมัติ	31

บทที่ 5 การพัฒนาระบบควบคุมและติดตามการจราจรอัตโนมัติ

5.1 เทคโนโลยีสำคัญที่ใช้ในการพัฒนาระบบ 46

- โปรแกรม MySQL 46
- ภาษา JAVA 55
- ภาษา XML 56

5.2 การประยุกต์หลักการจำลองมาใช้ในการพัฒนาระบบ 58

- แนะนำการจำลอง 58
- การจำลองคือเครื่องมือที่เหมาะสม 58
- การจำลองที่ไม่เหมาะสม 59
- ข้อดีและข้อเสียของการจำลอง 60
- ขั้นตอนในการศึกษาการจำลอง 61

บทที่ 6 สรุปผลการทำโครงงาน

6.1 ปัญหาและอุปสรรคในการทำโครงงาน 66

6.2 ประโยชน์ที่ได้รับ 66

6.3 ข้อเสนอแนะในการพัฒนาต่อไป 66

ภาคผนวก ก แบบเสนอหัวข้อโครงงานรายวิชา 2301399 (Project Proposal) 67

ภาคผนวก ข คู่มือการใช้โปรแกรม 74

ภาคผนวก ค Sequence Diagram และ Collaboration ของระบบ 93

ภาคผนวก ง การติดตั้งโปรแกรมจัดการฐานข้อมูล MySQL 112

ภาคผนวก จ ความรู้ทั่วไปในการใช้ MySQL 121

ภาคผนวก ฉ ตัวอย่างไฟล์โปรเจค KCN (.KCN File) 127

บรรณานุกรม 150

สารบัญรูปภาพ

รูปที่ 2-1: ตัวอย่างรูปแบบการจราจรในความเป็นจริง	4
รูปที่ 2-2: รูปตัวอย่างการคิดช่องการจราจรเฉลี่ย	5
รูปที่ 3-1:รูปตัวอย่างข้อมูลปริมาณการจราจรของทางสำนักการจราจรและขนส่งส่วนที่ 1	13
รูปที่ 3-2:รูปตัวอย่างข้อมูลปริมาณการจราจรของทางสำนักการจราจรและขนส่งส่วนที่ 2	14
รูปที่ 3-3: รูปแบบแผนที่ที่ใช้แสดงผลในระบบจำลอง	15
รูปที่ 3-4: รูปแสดงความสัมพันธ์ในแต่ละทางแยก	16
รูปที่ 3-5: รูปแสดงรายละเอียดในแต่ละถนน	17
รูปที่ 4-1: รูปภาพแสดงการเก็บข้อมูลลงในตารางความสัมพันธ์	21
รูปที่ 4-2: รูปภาพสรุปหมายเลขความสัมพันธ์	22
รูปที่ 4-3: รูปภาพแสดงความสัมพันธ์ในฐานข้อมูล	26
รูปที่ 4-4: รูปภาพแสดงรูปแบบข้อมูลของทางสำนักการจราจรและขนส่งกรณีทีเป็น 4 แยก	27
รูปที่ 4-5: รูปภาพแสดงรูปแบบข้อมูลของทางสำนักการจราจรและขนส่งกรณีทีเป็น 3 แยก	28
รูปที่ 4-6: รูปภาพแสดงรูปแบบข้อมูล CSV File ที่ถูกต้อง	29
รูปที่ 4-7: รูปภาพการแปลงข้อมูลให้อยู่ในรูปของ CSV File	29
รูปที่ 4-8: รูปภาพการแยกข้อมูลใน CSV File ตามความสัมพันธ์ๆ	30
รูปที่ 4-9: รูปภาพการแยกข้อมูลในแต่ละทิศทางของความสัมพันธ์ใด ๆ	30
รูปที่ 4-10: Use Case ของระบบ	31
รูปที่ 4-11: Use Case ของระบบในส่วนของผู้ดูแลฐานข้อมูล	32
รูปที่ 4-12: Use Case ของระบบในส่วนของผู้ศึกษา	32
รูปที่ 4-13: Sequence Diagram ของการเริ่มจำลองสถานการณ์จราจรใน USE CASE การควบคุมระบบ (Control System)	34-35
รูปที่ 4-14: Collaboration Diagram ของการเริ่มจำลองสถานการณ์จราจรใน USE CASE การควบคุมระบบ (Control System)	37
รูปที่ 4-15: Class Diagram แสดงความสัมพันธ์ระหว่างแต่ละ Class ในระบบ	38
รูปที่ 4-16: Activity Diagram ของ Automatic Control	41

รูปที่ 4-17: Activity Diagram ของ Junction	42
รูปที่ 4-18: Activity Diagram ของ Road	43
รูปที่ 4-19: รูปแสดงระยะทางของรถแต่ละกลุ่มก่อนจะมีการรวมกลุ่ม	44
รูปที่ 4-20: รูปแสดงการคำนวณในกรณีที่คำนึงถึงระยะทางของรถแต่ละกลุ่ม	44
รูปที่ 4-21: รูปแสดงการคำนวณในกรณีที่ไม่คำนึงระยะทางถึงของรถแต่ละกลุ่ม	45
รูปที่ 4-22: รูปแสดงการวาร์ปของกลุ่มรถ	45
รูปที่ 5-1: รูปแบบการจัดเก็บใน MySQL	47
รูปที่ 5-2: โครงสร้างของ Database ของ MySQL	47
รูปที่ 5-3: ผลจากการหา File ทั้งหมดใน database	50
รูปที่ 5-4**:** รูปแสดงความสามารถของ Java บนระบบปฏิบัติการ	55
รูปที่ 5-5: ขั้นตอนการสร้างโมเดลและการจำลอง	62
รูปที่ ก-1: รูปแสดงขอบเขตการจราจรที่ใช้ในการศึกษา	69
รูปที่ ก-2: การแบ่งช่องจราจร	70
รูปที่ ก-3: โครงสร้างโปรแกรม	71
รูปที่ ก-4: หน้าจอของโปรแกรม	72
รูปที่ ข-1: รูปภาพของหน้าจอโปรแกรม	75
รูปที่ ข-2: รูปภาพของเมนูหลัก	76
รูปที่ ข-3: รูปภาพของเครื่องมือลัด	76
รูปที่ ข-4: รูปภาพของแท็ป Monitoring	76
รูปที่ ข-5: รูปภาพของแท็ปข้อมูลสี่แยก	77
รูปที่ ข-6: รูปภาพของแท็ปรายละเอียดสี่แยก	78
รูปที่ ข-7: รูปภาพของแท็ปข้อมูลถนน	78
รูปที่ ข-8: รูปภาพของแท็ปรายละเอียดถนน	79
รูปที่ ข-9: รูปภาพของแท็ป Statistic	79
รูปที่ ข-10: รูปภาพของแท็ป Database	80
รูปที่ ข-11: รูปภาพของการสร้างโปรเจคใหม่	81
รูปที่ ข-12: รูปภาพของหน้าจอเมื่อสร้างโปรเจคใหม่เสร็จ	81
รูปที่ ข-13: รูปภาพของการเปิดโปรเจคใหม่	82
รูปที่ ข-14: รูปภาพของการบันทึกไฟล์ข้อมูลแบบ Save	83

รูปที่ ข-15: รูปภาพของการบันทึกไฟล์ข้อมูลแบบ Save As	84
รูปที่ ข-16: รูปภาพของLogin เข้าสู่ระบบแก้ไขฐานข้อมูล	87
รูปที่ ข-17: รูปภาพหน้าจอระบบแก้ไขฐานข้อมูล	87
รูปที่ ข-18: รูปภาพแสดงการเลือกชื่อตารางที่ต้องการแก้ไข	88
รูปที่ ข-19: รูปภาพแสดงข้อมูลในตารางที่เลือก	89
รูปที่ ข-20: รูปภาพแสดงการยืนยันในการเปลี่ยนแปลงข้อมูล	90
รูปที่ ข-21: รูปภาพแสดงข้อมูลถูกบันทึกลงฐานข้อมูลเป็นที่เรียบร้อย	90
รูปที่ ข-22: รูปภาพแสดงการเลือกชื่อไฟล์ที่ต้องการจะแปลง	91
รูปที่ ข-23: รูปภาพแสดง CSV File ถูกแปลงลงฐานข้อมูลเป็นที่เรียบร้อย	92
รูปที่ ค-1: Collaboration Diagram ของ New Project	93
รูปที่ ค-2: Sequence Diagram ของ New Project	93
รูปที่ ค-3: Collaboration Diagram ของ Open Project	94
รูปที่ ค-4: Sequence Diagram ของ Open Project	94
รูปที่ ค-5: Collaboration Diagram ของ Save	95
รูปที่ ค-6: Sequence Diagram ของ Save	95
รูปที่ ค-7: Collaboration Diagram ของ Save As	96
รูปที่ ค-8: Sequence Diagram ของ Save As	96
รูปที่ ค-9: Collaboration Diagram ของ Close	97
รูปที่ ค-10: Sequence Diagram ของ Close	97
รูปที่ ค-11: Collaboration Diagram ของ Previous Turn	98
รูปที่ ค-12: Sequence Diagram ของ Previous Turn	98
รูปที่ ค-13: Collaboration Diagram ของ Next Turn	99
รูปที่ ค-14: Sequence Diagram ของ Next Turn	99
รูปที่ ค-15: Collaboration Diagram ของ Fit Map	100
รูปที่ ค-16: Sequence Diagram ของ Fit Map	100
รูปที่ ค-17: Collaboration Diagram ของ Zoom In	101
รูปที่ ค-18: Sequence Diagram ของ Zoom In	101
รูปที่ ค-19: Collaboration Diagram ของ Zoom Out	102
รูปที่ ค-20: Sequence Diagram ของ Zoom Out	102

รูปที่ ค-21: Collaboration Diagram ของ Set Light Control of Junction	103
รูปที่ ค-22: Sequence Diagram ของ Set Light Control of Junction	103
รูปที่ ค-23: Collaboration Diagram ของ Update Junction Detail	104
รูปที่ ค-24: Sequence Diagram ของ Update Junction Detail	104
รูปที่ ค-25: Collaboration Diagram ของ View Junction Tab	105
รูปที่ ค-26: Sequence Diagram ของ View Junction Tab	105
รูปที่ ค-27: Collaboration Diagram ของ View Junction Detail Tab	106
รูปที่ ค-28: Sequence Diagram ของ View Junction Detail Tab	106
รูปที่ ค-29: Collaboration Diagram ของ View Road Tab	107
รูปที่ ค-30: Sequence Diagram ของ View Road Tab	107
รูปที่ ค-31: Collaboration Diagram ของ View Road Detail Tab	108
รูปที่ ค-32: Sequence Diagram ของ View Road Detail Tab	108
รูปที่ ค-33: Collaboration Diagram ของ Use Database Management	109
รูปที่ ค-34: Sequence Diagram ของ Use Database Management	109
รูปที่ ค-35: Collaboration Diagram ของ Update Database	110
รูปที่ ค-36: Sequence Diagram ของ Update Database	110
รูปที่ ค-37: Collaboration Diagram ของ Import CSV file	111
รูปที่ ค-38: Sequence Diagram ของ Import CSV File	111
รูปที่ ง-1: รูปแสดงการ Download MySQL for Windows	112
รูปที่ ง-2: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 1	113
รูปที่ ง-3: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 2	113
รูปที่ ง-4: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 3	114
รูปที่ ง-5: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 4	114
รูปที่ ง-6: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 5	115
รูปที่ ง-7: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 6	115
รูปที่ ง-8: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 7	116
รูปที่ ง-9: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 8	116
รูปที่ ง-10: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 9	117
รูปที่ ง-11: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 10	117

รูปที่ ง-12: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 11	117
รูปที่ ง-13: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 12	117
รูปที่ ง-14: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 13	118
รูปที่ ง-15: รูปแสดงการ Download MySQL for Linux	118

สารบัญตาราง

ตารางที่ 3.1: รายชื่อและหมายเลขของทางแยกที่อยู่ภายในขอบเขตพิจารณา	10
ตารางที่ 3.2: รายชื่อและหมายเลขของทางแยกที่อยู่ภายนอกขอบเขตพิจารณา	11
ตารางที่ 3.3: ตารางแสดงข้อมูลที่มีในขอบเขตที่พิจารณา	12
ตารางที่ 3.4: ตารางแสดงข้อมูลที่มีนอกขอบเขตที่พิจารณา	12
ตารางที่ 4.1: Schema ของ Junction Table	19
ตารางที่ 4.2: Schema ของ Road Table	20
ตารางที่ 4.3: Schema ของ Relation Table	20
ตารางที่ 4.4: Schema ของ Statcar Table	23
ตารางที่ 4.5: Schema ของ Tuning Table	23
ตารางที่ 4.6: Schema ของ Projdata Table	24
ตารางที่ 4.7: Schema ของ Carincome Table	25
ตารางที่ 4.8: Schema ของ Historyaction Table	25
ตารางที่ 4.9: Schema ของ User Table	25
ตารางที่ 4.10: ตัวอย่างข้อมูลที่บันทึกลง Database ในตาราง Statcar	30
ตารางที่ 4.11: คำอธิบาย USE CASE	33
ตารางที่ ก.1: ตารางระยะเวลาการดำเนินงาน	71
ตารางที่ จ.1: ตารางแสดงชนิดของข้อมูลใน MySQL	123
ตารางที่ จ.2: ตารางแสดง Relational operator	124
ตารางที่ จ.3: ตารางแสดง Bulletin operator	124
ตารางที่ จ.4: ตารางแสดงคำสั่ง MySOL ที่ใช้บ่อย	125

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

การเดินทางในปัจจุบันของผู้คนส่วนใหญ่จะนิยมใช้รถยนต์เป็นยานพาหนะเพื่อไปติดต่อทำธุระต่างๆ ไม่ว่าจะเป็นเดินทางไปทำงาน การรับส่งสินค้า หรือแม้เพียงต้องการจะเดินทางไปที่ใกล้ๆสักแห่งก็ยังคงต้องพึ่งพารถยนต์ จึงส่งผลให้จำนวนรถยนต์บนท้องถนนสายต่างๆมีปริมาณมาก ซึ่งก่อให้เกิดปัญหาที่เกี่ยวข้องตามมาด้วย ปัญหาที่เด่นชัดมากที่สุด คือ ปัญหาการจราจรที่ติดขัด เพราะมีผลกระทบต่อปัจจัยหลายๆด้าน เช่น ความเครียดของบุคคลบนท้องถนน การสูญเสียเวลาโดยเปล่าประโยชน์ การก่อมลพิษทางอากาศ เป็นต้น

จากปัญหาที่กล่าวมานี้ ทำให้การควบคุมสัญญาณไฟจราจรบนท้องถนนเป็นสิ่งที่สำคัญ ที่จะแก้ไขปัญหาการจราจรที่ติดขัด ให้มีความคล่องตัวมากขึ้น แต่เป็นที่ทราบกันดีว่าการควบคุมสัญญาณไฟจราจรที่ดีนั้น ต้องมีการศึกษาและวางแผนหาวิธีการควบคุมให้เหมาะสมกับสภาพการณ์จราจรบริเวณนั้นในช่วงเวลาต่าง ๆ เช่น ถนนที่มีโรงเรียนอยู่ละแวกนั้น ในช่วงเวลา 7:30 - 8:30 น. หรือ 15:00 – 17.00 น. จะมีปริมาณรถหนาแน่นเป็นพิเศษ เพราะมีการรับส่งเด็กนักเรียนที่โรงเรียน ดังนั้นการเปิดสัญญาณไฟจราจรในช่วงนี้ก็ควรมีรูปแบบที่แตกต่างกับการเปิดสัญญาณไฟจราจรในช่วงเวลาปกติ เป็นต้น การศึกษาและวางแผนเช่นนี้จึงสามารถช่วยลดปัญหาการจราจรที่ติดขัดในบริเวณนั้นให้มีความคล่องตัวมากขึ้น แต่เราจะทราบได้อย่างไร ว่าวิธีการควบคุมสัญญาณไฟจราจรมีความเหมาะสมกับสภาพการณ์จราจรบริเวณนั้นจริง การที่เราจะไปทดลองวิธีการเหล่านั้นกับสถานการณ์จริงเป็นสิ่งที่ทำได้ยาก และไม่สามารถมองเห็นภาพรวมของระบบการจราจรบริเวณที่พิจารณาได้ชัดเจน อีกทั้งต้องใช้ค่าใช้จ่ายในการศึกษาและวางแผนเป็นจำนวนมาก

ในการศึกษาวิธีการแก้ปัญหานั้น ขั้นต้นได้ทำการศึกษาระบบ SCOOT ที่ใช้กันอยู่ในหลายประเทศ เนื่องจากสามารถลดปัญหาการจราจรได้ในระดับหนึ่ง อีกทั้งยังช่วยลดภาระค่าใช้จ่ายในเรื่องบุคลากรในการควบคุมเพราะระบบสามารถทำงานได้ด้วยตัวเอง ซึ่งทางระบบ SCOOT นี้มีวิธีการทำงานดังนี้ ในแต่ละแยกจะมีอุปกรณ์ที่ใช้นับปริมาณความหนาแน่นของรถแล้วส่งข้อมูลเหล่านี้ไปยังศูนย์คอมพิวเตอร์ เพื่อประมวลหาช่วงเวลาที่เหมาะสมในการเปิด-ปิดสัญญาณไฟในแต่ละครั้งแล้วจึงส่งกลับไปใช้งานในแยกนั้น ๆ หากคำนึงถึงหลักการทำงานและประสิทธิภาพของระบบ SCOOT นี้แล้ว ดูจะสามารถช่วยในการจัดการกับปัญหาจราจรในกรุงเทพมหานครได้ดี

พอสมควร แต่หากลองในมุมที่ต่างกันออกไปจะเห็นได้ว่าเรายังจำเป็นต้องเสียงบประมาณอีกเป็นจำนวนมากเพื่อติดตั้งอุปกรณ์ที่ใช้นับปริมาณความหนาแน่นของรถ

ดังนั้นทางคณะผู้จัดทำจึงคิดที่จะทำโปรแกรมควบคุมและติดตามการจราจรอัตโนมัติขึ้นมาเพื่อให้สามารถศึกษาและวางแผนหาวิธีการควบคุมสัญญาณไฟจราจรที่มีความเหมาะสมกับสภาพการณ์จราจรบริเวณที่ต้องการจะศึกษาในช่วงเวลาต่าง ๆ ซึ่งทางระบบจะจำลองสถานการณ์การจราจรในลักษณะต่างๆ โดยนำข้อมูลทางสถิติจากสำนักการจราจรและขนส่ง กรุงเทพมหานครมาทำการวิเคราะห์และออกแบบข้อมูลเพื่อใช้ร่วมกับทฤษฎี Poisson distribution simulation ในการจำลองระบบจราจรและแสดงผลการเคลื่อนตัวของรถในสถานการณ์จำลองอันเนื่องมาจากวิธีการควบคุมสัญญาณไฟจราจรที่กำหนดไว้ โดยระบบที่จัดทำขึ้นนี้จะคลอบคลุมบริเวณสวนจิตรลดา (อยู่ในเขตดุสิต กรุงเทพมหานคร)

สถานการณ์การจราจรที่จำลองขึ้นในระบบนี้ไม่คำนึงถึงกรณีที่มีอุบัติเหตุหรือการซ่อมถนนเกิดขึ้นในระบบ รวมไปถึงบริเวณห้ามจอด บริเวณห้ามเลี้ยว สิ่งกีดขวาง ฯลฯ ในกรณีที่จำนวนของช่องการจราจรมีขนาดไม่สม่ำเสมอก็จะใช้ค่าเฉลี่ยของช่องการจราจรในการคำนวณ ส่วนกรณีของผู้ขับขี่รถยนต์ในระบบจำลองนี้จะสมมติให้มีวินัยการจราจรอย่างเคร่งครัด (ผู้ขับขี่ทุกคนไม่สามารถแซงกัน ปาดหน้าหรือแทรกแถวกันได้) ข้อจำกัดของระบบนี้ คือ แผนที่ที่ถูกสร้างขึ้นในระบบอาจไม่สมจริงและการแสดงปริมาณการจราจรนั้นถูกแสดงอยู่ในรูปของอัตราส่วนระหว่างปริมาณรถในขณะนั้นเทียบกับปริมาณรถที่ถนนจะรับได้ เช่น บนถนนนี้มีปริมาณรถอยู่ 20% เป็นต้น

ทางคณะผู้จัดทำหวังเป็นอย่างยิ่งว่า ผู้ใช้งานจะได้รับประโยชน์จากระบบที่สร้างขึ้นนี้ โดยที่ผู้ใช้งานนั้นจะสามารถศึกษาและหาวิธีการที่ใช้ในการแก้ไขปัญหาการจราจรในบริเวณพิจารณาได้อย่างเหมาะสม รวมทั้งสามารถที่จะเรียนรู้และทำความเข้าใจการ Formulate ปัญหาการจราจรตามกรณีศึกษาได้

1.2 วัตถุประสงค์

วัตถุประสงค์ของโครงงานนี้ คือ สร้างระบบจำลองสถานการณ์การจราจรที่สามารถแสดงสภาพการจราจรอันเป็นผลมาจากการควบคุมสัญญาณไฟจราจรที่ได้กำหนดไว้ เพื่อให้ผู้ใช้สามารถศึกษาหาวิธีการควบคุมสัญญาณไฟจราจรที่เหมาะสมกับแต่ละถนนในขอบเขตศึกษาได้

1.3 ขอบเขตโครงงาน

1. ขอบเขตการจราจรที่นำมาศึกษามีอาณาเขตในเขตกรุงเทพมหานคร บริเวณสวนจิตรลดา
2. สภาพการณ์จราจรที่ศึกษาไม่คำนึงถึงกรณีที่เกิดจากอุบัติเหตุ การซ่อมถนน สิ่งกีดขวางการจราจรต่าง ๆ ฯลฯ
3. สถานการณ์ที่ศึกษาจะพิจารณาในกรณีที่ผู้ขับรถมีวินัยการจราจรอย่างเคร่งครัดเท่านั้น

4. ปริมาณการจราจรแสดงอยู่ในรูปแบบของ Scale (ไม่ใช่ Object)
5. ค่าเฉลี่ยของช่องทางจราจรจะใช้ในการคำนวณกรณีที่ช่องทางจราจรมีรูปแบบไม่แน่นอน
6. แผนที่ที่ถูกสร้างขึ้นในโปรแกรมอาจไม่ตรงกับความเป็นจริง
7. สะพานข้ามแยก ทางด่วน ทางเดินเฉพาะรถประจำทาง ไม่ได้นำมาพิจารณาในระบบ

1.4 ขั้นตอนการพัฒนาระบบ

1. ศึกษาแนวคิดและระบบที่เกี่ยวข้องกับการจราจร เช่น การควบคุมสัญญาณไฟในระบบ SCOOT เป็นต้น
2. รวบรวมรายละเอียดของแต่ละถนน รวมทั้งข้อมูลต่าง ๆ ที่อยู่ภายในขอบเขตศึกษา เช่น ปริมาณการจราจรในทางแยกต่าง ๆ รูปแบบของช่องทางจราจร ฯลฯ
3. ศึกษาวิธีการจำลองสถานการณ์เพื่อใช้ในการสร้างแบบจำลองของระบบโดยการนำทฤษฎี Poisson distribution simulation มาประยุกต์ใช้ เช่น การคำนวณปริมาณรถที่วิ่งออกมาจากแต่ละถนน เป็นต้น
4. กำหนดขอบเขตโครงงาน
5. ออกแบบระบบ (รูปแบบโปรแกรม, รูปแบบฐานข้อมูล, รูปแบบหน้าจอ)
6. พัฒนาระบบ
7. ทดสอบระบบ
8. จัดทำเอกสารประกอบ

บทที่ 2

แนวคิดในการสร้างระบบควบคุมและติดตามการจราจรอัตโนมัติ

2.1 รูปแบบการจราจรในความเป็นจริง

รูปแบบของระบบการจราจรในความเป็นจริงนั้น มีปัญหาที่เกิดขึ้นมากมายในระบบ ไม่ว่าจะเป็นมารยาทในการขับรถยนต์ของผู้ขับขี่บนท้องถนน อาทิเช่น การเปลี่ยนช่องทางการจราจรกะทันหัน (การปาดหน้า) การขับรถคร่อมช่องทางการจราจร ฯลฯ หรือ อาจจะเป็นเหตุการณ์ต่าง ๆ ที่ทำให้เกิดผลกระทบกับการจราจรในระบบ เช่น การเกิดอุบัติเหตุต่าง ๆ การซ่อมถนน การมีสิ่งกีดขวางการจราจร เป็นต้น แต่ปัญหาที่มีผลต่อการจำลองสถานการณ์การจราจรมากที่สุดนั้นน่าจะเป็นปัญหาที่เกิดจากความจริงที่ว่า ปริมาณการจราจรที่วิ่งเข้าสู่ถนนเส้นหนึ่ง ๆ นั้นไม่ได้หมายความว่าจะต้องมีปริมาณรถยนต์ที่วิ่งออกมาจากถนนนั้นในปริมาณที่เท่ากัน เพราะผู้ขับขี่สามารถที่จะขับรถเข้าซอยเพื่อจอดหรือเลี่ยงไปใช้ถนนอื่น ๆ หรือกลับรถก็ได้

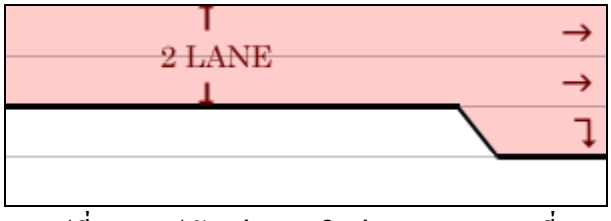
รูปที่ 2-1: ตัวอย่างรูปแบบการจราจรในความเป็นจริง

2.2 รูปแบบการจราจรในระบบควบคุมและติดตามการจราจรอัตโนมัติ

จากสิ่งที่ได้กล่าวมาข้างต้นเป็นส่วนหนึ่งที่ทำให้ระบบการจราจรนั้นมีความยุ่งยากและซับซ้อนต่อการจำลองสถานการณ์การจราจรเป็นอย่างยิ่ง ดังนั้นระบบควบคุมและติดตามการจราจรอัตโนมัติจึงได้ปรับเปลี่ยนรูปแบบการจราจรที่ซับซ้อนเหล่านี้ให้อยู่ในรูปแบบที่ง่ายต่อการจำลองสถานการณ์การจราจรในระบบ ซึ่งในระบบจำลองที่สร้างขึ้นมีผลมาจากสมมติฐานดังต่อไปนี้

- ปริมาณรถยนต์ที่วิ่งเข้าถนนในแต่ละถนนกับปริมาณรถยนต์ที่วิ่งออกจะต้องมีค่าเท่ากัน

- ทิศทางของรถยนต์ที่วิ่งออกจากถนนนั่นจะมีอยู่ 3 ทิศทางคือ ทางซ้าย ทางตรงและทางขวา
- ปริมาณของรถยนต์ที่ออกไปในแต่ละทิศทาง (ซ้าย ตรง ขวา) นั้นมาจากอัตราส่วนที่คำนวณได้จากข้อมูลปริมาณการจราจรของสำนักการจราจรและขนส่งกรุงเทพมหานคร
- ความเร็วของรถยนต์ที่วิ่งบนท้องถนนจะเป็นความเร็วรถเฉลี่ย
- เวลาที่รถออกตัวหลังการเปลี่ยนไฟจราจรให้เป็นไฟเขียวนั้นจะสมมติให้รถสามารถออกตัวได้ทันทีและมีอัตราเร็วของรถเท่ากับอัตราเร็วเฉลี่ย (ไม่มี Time Delay ในช่วงออกตัว)
- รถที่ออกจากแต่ละถนนจะออกเป็นกลุ่มของรถ ไม่ใช่รถเป็นคันๆ
- จำนวนช่องการจราจรบนถนนจะใช้เป็นจำนวนช่องการจราจรเฉลี่ย เช่น ปลายถนนมีจำนวนเลน 3 เลนแต่ช่วงกลาง ๆ ของถนนส่วนใหญ่เป็น 2 ช่องการจราจรก็จะคิดเป็น 2 ช่องการจราจรสำหรับถนนนั้น ดังตัวอย่างในรูปที่ 2.2



รูปที่ 2-2: รูปตัวอย่างการคิดช่องการจราจรเฉลี่ย

- ปริมาณรถที่เข้าสู่ระบบที่สร้างขึ้นนั้นจะวิ่งมาจากบริเวณนอกขอบเขตของระบบ (จุดขอบ) โดยอัตราปริมาณรถที่เข้ามานั้นจะถูกกำหนดขึ้นจากผู้ศึกษาเอง เช่น กำหนดให้มีรถเข้าในจุดขอบนี้ 150 คันต่อนาที เป็นต้น

ในการจำลองรูปแบบระบบควบคุมและติดตามการจราจรอัตโนมัติที่ได้กล่าวมานี้จะใช้ทฤษฎีทาง Simulation เข้ามาใช้ในการจำลองระบบเพื่อสร้างความสมจริงในระบบจำลองให้ใกล้เคียงกับความจริงมากขึ้น โดยการทำ Simulation นี้สร้างขึ้นมาจากทฤษฎี Poisson distribution โดยมีรายละเอียดของทฤษฎีดังนี้

ทฤษฎีปัวซองค์

ถ้าตัวเลขในเหตุการณ์หนึ่งที่ได้มาในแต่ละช่วงเวลาเป็นตัวเลขแบบสุ่มและเป็นอิสระจากช่วงเวลาอื่นๆ จะได้ว่าความเป็นไปได้ของจำนวนนั้นสามารถอธิบายได้โดยทฤษฎีปัวซองค์ กระบวนการสร้างตัวอย่างจากทฤษฎีนี้จะต้องใช้ค่า λ ซึ่งเป็นค่าเฉลี่ยของจำนวนที่ต้องการสุ่มจะได้ว่าความเป็นไปได้ของค่า x คือ

$$f_x(\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}, 0 < x < \infty; \lambda > 0$$

เป็นที่รู้กันว่า ถ้าช่วงเวลาระหว่างเหตุการณ์มาจาก $\exp(\frac{1}{\lambda})$ และตัวเลขในช่วงเวลานั้นมาจาก $f(\lambda)$ เพราะฉะนั้นในทางคณิตศาสตร์จะเขียนได้ว่า

$$\sum_{i=0}^{x} T_i \le 1 \le \sum_{i=0}^{x+1} Ti,$$

โดยที่ T_i , $i = 0,1,\ldots,$X+1มาจาก $\exp(\frac{1}{\lambda})$

จาก $T_i = -(\frac{1}{\lambda}) \ln U_i$

สุดท้ายจะได้

$$\prod_{i=0}^{x} U_i \ge e^{-\lambda} \ge \prod_{i=0}^{x+1} U_i, x = 0,1,\ldots$$

ซึ่งจะได้วิธีคิดออกมาดังนี้

1 $A \leftarrow 1$ $(g_k = 1)$.

2 $K \leftarrow 0$.

3 Generate U_k from U(0,1).

4 $A \leftarrow U_k A$ $(g_{k+1} = g_k U_k)$.

5 If $A < e^{-\lambda}$, deliver X = K.

6 $K \leftarrow K + 1$.

7 Go to step 3.

ในขั้นต้นการสร้างระบบควบคุมและติดตามการจราจรนี้ได้เริ่มพิจารณาลักษณะพื้นฐานของการจราจรบนท้องถนนว่ามีลักษณะอย่างไร ซึ่งหลังการวิเคราะห์จึงสรุปได้ว่า

- ถนนใด ๆ จะมีรถเข้ามาสู่ถนนนี้ในทุกรอบของการเปิด-ปิดสัญญาณไฟจราจร
- รถที่จะออกจากถนนใด ๆ จะมีทิศทางที่รถนั้น ๆ สามารถไปได้มากสุดเท่ากับจำนวนของถนนที่ติดกันอยู่ (จำนวนของแยก - 1) เช่น ห้าแยก ก็จะมีทิศทางที่รถออกจากถนนนั้นไปได้มากสุด คือ 4 ทิศทาง เป็นต้น

และจากสมมติฐานที่ได้กล่าวไปในข้างต้นที่ว่า

- ปริมาณรถยนต์ที่วิ่งเข้าถนนในแต่ละถนนกับปริมาณรถยนต์ที่วิ่งออกจะต้องมีค่าเท่ากัน
- ปริมาณของรถยนต์ที่ออกไปในแต่ละทิศทางนั้นมาจากอัตราส่วนที่คำนวณได้จากข้อมูลปริมาณการจราจรของสำนักการจราจรและขนส่งกรุงเทพมหานคร

ทำให้เราได้กำหนดรูปแบบของการจราจรบนถนนใด ๆ ในระบบให้มีรูปแบบดังนี้ คือ จำนวนรถที่เข้ามาในถนนใด ๆ จะถูกปล่อยออกไปตามอัตราส่วนที่คำนวณขึ้น เมื่อสัญญาณไฟเป็นแดงเมื่อใดจำนวนรถที่ปล่อยไปไม่หมดก็จะสะสมค้างอยู่ในถนนดังกล่าว แต่เนื่องจากการใช้อัตราส่วนในการคำนวณปริมาณรถที่จะออกไปเป็นค่าคงที่ในทุก ๆ รอบสัญญาณไฟ ในแต่ละรอบสัญญาณไฟก็จะปล่อยรถเป็นจำนวนเท่าเดิม (เช่น ถนนนี้เปิดไฟเขียว 1 นาทีสามารถคำนวณรถที่จะออกไปได้เท่ากับ 55 คัน ถ้าไม่มีการปรับระยะเวลาการเปิดไฟเขียวใหม่ ในรอบที่เปิดไฟเขียวครั้งต่อไปก็จะมีรถออก 55 คันเท่าเดิม) จึงทำให้ขาดความสมจริง ซึ่งในความเป็นจริงแล้วแต่ละรอบสัญญาณไฟปริมาณรถที่ออกไปนั้นจะมีค่าไม่คงที่ เราจึงใช้ทฤษฎีป๊วซองค์เข้ามาช่วยในจุดนี้ให้การจำลองมีความสมจริงมากยิ่งขึ้น โดยเรานำหลักการของทฤษฎีป๊วซองค์มาประยุกต์ใช้ซึ่งมีขั้นตอนดังนี้

1. กำหนดจำนวนรถที่สามารถขับผ่านไฟเขียวในรอบการทำงานนี้ (หนึ่งรอบการทำงานเท่ากับระยะเวลาจริงประมาณ 1 นาที) ซึ่งจะคำนวณได้จากสถิติของสำนักการจราจรและขนส่ง กรุงเทพมหานคร เช่น จากสถิติมีรถออกจากถนนนี้ 600 คันต่อชั่วโมง ดังนั้นใน 1 นาทีจะมีรถออกจากถนนนี้ 10 คัน เป็นต้น
2. นำปริมาณของรถยนต์ที่ออกไปในแต่ละทิศทาง (จากสถิติของสำนักการจราจรและขนส่ง) มาคำนวณหาอัตราส่วนของแต่ละทิศทางเพื่อหาความเป็นไปได้รถคันหนึ่ง ๆ ที่จะวิ่งเข้าไปในแต่ละทิศทาง เช่น มีรถเลี้ยวซ้าย 2 คัน ตรง 5 คัน และขวา 3 คัน จะได้อัตราส่วนเท่ากับ 20 : 50 : 30

3. นำอัตราส่วนของทุกทิศทางสำหรับทางแยกนั้นมาทำการสุ่มค่าตามทฤษฎีป๊วซองค์เพื่อหาอัตราส่วนของทุกทิศทางใหม่ ดังนั้นอัตราส่วนที่คำนวณได้จะมีการเปลี่ยนแปลงตลอดในทุก ๆ รอบการทำงานซึ่งจะส่งผลให้เกิดความสมจริงในการจำลองเพิ่มมากขึ้น
4. คำนวณอัตราส่วนออกมาเป็นเปอร์เซ็นต์
5. คำนวณหาตัวเลขสุ่มของรถแต่ละคันแล้วนำมาเปรียบเทียบกับเปอร์เซ็นต์ของแต่ละทิศทางเพื่อตัดสินว่ารถคันนี้จะวิ่งไปในทิศทางใด
6. สุดท้ายจะได้จำนวนรถที่สามารถขับผ่านไฟเขียวในรอบการทำงานนี้ในแต่ละทิศทางสำหรับทางแยกนั้น ๆ

เพื่อความเข้าใจในขั้นตอนดังกล่าวสามารถพิจารณาได้จากตัวอย่างดังต่อไปนี้

สมมติให้มีรถออกจากทางแยกในรอบการทำงานนี้เป็นจำนวน 4 คัน และทางแยกนั้นเป็น 3 แยกซึ่งมีปริมาณของรถยนต์ที่ออกไปในแต่ละทิศทางซึ่งผ่านการคำนวณอัตราส่วนแล้วได้เป็น 60 : 40 นำอัตราส่วนมาคำนวณตัวเลขโดยสุ่มด้วยทฤษฎีป๊วซองค์ ได้ค่าอัตราส่วนใหม่เป็น 52 : 48 จะได้ว่าจะมีความเป็นไปได้ที่รถแต่ละคันที่จะออกไปในทิศทางที่ 1 เท่ากับ 52% และมีความเป็นไปได้ที่รถแต่ละคันที่จะออกไปในทิศทางที่ 2 จำนวน 48 % (ในโปรแกรมจะกำหนดช่วงเลข 1- 100 โดยให้ช่วงเลข 1-52 เป็นช่วงของรถที่จะวิ่งเข้าไปในทางที่ 1 ซึ่งจะคลอบคลุม 52 ช่องแทนความหมายของความเป็นไปได้ในทิศทางนี้ คือ 52% และช่วงเลข 53-100 เป็นช่วงของรถที่จะวิ่งเข้าไปในทางที่ 2 ซึ่งจะคลอบคลุม 48 ช่องแทนความหมายของความเป็นไปได้ในทิศทางที่ 2 นี้ คือ 48%) หลังจากนั้นสุ่มค่าให้กับรถคันที่ 1 ได้ 80 สุ่มค่าให้กับรถคันที่ 2 ได้ 17 สุ่มค่าให้กับรถคันที่ 3 ได้ 52 และสุ่มค่าให้กับรถคันที่ 4 ได้ 45 นั่นหมายความว่าจะมีรถวิ่งไปในทิศทางที่ 1 จำนวน 3 คัน และจะมีรถวิ่งไปในทิศทางที่ 2 จำนวน 1 คัน

จะเห็นว่าอัตราส่วนใหม่ที่คิดตามทฤษฎีป๊วซองค์ จะช่วยให้ค่าที่ได้ออกมามีการเปลี่ยนแปลงและดูสมจริงมากขึ้นในทุกรอบการทำงานตามอัตราส่วนที่ได้มา วิธีการนี้จะใช้กับทุก ๆ ทางแยก ทั้งที่มีและไม่มีตัวควบคุมสัญญาณไฟจราจร ในกรณีของทางแยกที่ไม่มีตัวควบคุมสัญญาณไฟจราจรหรือมีช่องทางเลี้ยวซ้ายผ่านตลอด ก็จะกำหนดว่าให้ทิศทางเหล่านั้นเหมือนเปิดสัญญาณไฟเขียวอยู่ตลอดเวลา

หลังจากที่ทราบถึงแนวคิดในการคำนวณปริมาณรถที่จะออกในแต่ละถนนของทุกทิศทางแล้วก็ทำให้เราทราบถึงปริมาณรถที่เข้ามาในแต่ละถนนได้ด้วย (รถที่ออกจากถนนในทิศทางหนึ่งก็จะเป็นรถที่เข้าไปในอีกถนนหนึ่ง) ขั้นต่อไปก็คือการวิเคราะห์ข้อมูลที่ใช้ในระบบซึ่งกล่าวอยู่ในบทถัดไป

บทที่ 3

การวิเคราะห์ข้อมูลของระบบ

จากการวิเคราะห์ถึงรูปแบบของระบบจราจรทั่ว ๆ ไปในกรุงเทพมหานคร รวมไปถึงการศึกษาทฤษฎีและวิธีการที่ใช้ในการสร้างระบบจราจรจำลองนั้น ทางคณะผู้จัดทำสามารถรวบรวมข้อมูลที่สำคัญและจำเป็นต่อการสร้างสถานการณ์จำลองขึ้นได้ดังนี้

3.1 ปริมาณการจราจรในชั่วโมงต่าง ๆ ของแต่ละถนน

ข้อมูลในส่วนนี้จะเป็นส่วนสำคัญที่ใช้ในการคำนวณหาอัตราส่วนของปริมาณของรถในแต่ละถนนว่ามีรถเลี้ยวซ้าย รถขับตรงและรถเลี้ยวขวาในชั่วโมงนี้ประมาณกี่คัน เพื่อกำหนดรูปแบบการเคลื่อนตัวของรถแต่ละถนนในระบบให้มีอัตราการเคลื่อนตัวที่เหมาะสมและใกล้เคียงกับความเป็นจริงมากที่สุด โดยข้อมูลเหล่านี้จะอ้างอิงมาจากข้อมูลทางสถิติที่ผ่านมาของทางสำนักการจราจรและขนส่ง กรุงเทพมหานคร รูปแบบข้อมูลที่นำมาใช้นั้นเป็นปริมาณการจราจรของแต่ละทางแยกบริเวณเขตดุสิต (ซึ่งเป็นเขตที่อยู่ในขอบเขตศึกษา) ในชั่วโมงต่าง ๆ ตั้งแต่เวลา 07.00 – 19.00 น. ซึ่งมีทางแยกที่เกี่ยวข้องกับขอบเขตพิจารณาเป็นจำนวนทั้งสิ้น 33 ทางแยก ซึ่งสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ

1. กลุ่มของทางแยกที่อยู่ภายในขอบเขตพิจารณาจำนวน 18 ทางแยก
2. กลุ่มทางแยกภายนอกขอบเขตพิจารณาแต่มีความสัมพันธ์กับทางแยกภายในเขตพิจารณาจำนวน 15 ทางแยก

รายละเอียดของทั้ง 2 กลุ่มนั้นได้นำไปแสดงไว้ในหน้าถัดไปในตารางที่ 3.1 และ 3.2 ตามลำดับ

1. กลุ่มของทางแยกที่อยู่ภายในขอบเขตพิจารณา (จำนวน 18 ทางแยก)

ชุดข้อมูลของทางสำนักการจราจรและขนส่ง กรุงเทพมหานคร ที่ได้ขอมานั้น ข้อมูลทางแยกในลำดับที่ 15,16 และ 17 มีเพียงข้อมูลหมายเลขของทางแยกเท่านั้น ข้อมูลในส่วนอื่น ๆ นั้นไม่พบในชุดข้อมูลที่ขอมา ชื่อทางแยกจึงตั้งจากทางคณะผู้จัดทำโปรแกรมเอง จึงอาจไม่ตรงกับข้อมูลทางสำนักการจราจรและขนส่ง ส่วนทางแยกลำดับที่ 18 นั้นไม่มีข้อมูลอยู่เลย ชื่อทางแยกและหมายเลขทางแยกจึงตั้งจากทางคณะผู้จัดทำโปรแกรมเองทั้งหมด

ตารางที่ 3.1: รายชื่อและหมายเลขของทางแยกที่อยู่ภายในขอบเขตพิจารณา

ลำดับที่	หมายเลขทางแยก (เป็นหมายเลขเดียวกันกับทางสำนักการจราจรและขนส่ง)	ชื่อของทางแยก
1	14	การเรือน
2	16	ศรีย่าน
3	17	ซังฮี้
4	21	อู่ทองใน
5	28	ราชวัตร
6	31	พิชัย
7	62	ขัตติยานี
8	65	สุโขทัย
9	69	ราชวิถี
10	85	สวนรื่น
11	86	เกษะโกมล
12	100	ดุสิต
13	129	บางกระบือ
14	149	ร่วมจิตร
15*	348	สามแยกหน้าสำนักงานเขตดุสิต
16*	351	สามแยกพิชัย
17*	424	สี่แยกหน้าซอยองครักษ์ ๖
18*	901	สามแยกข้างรัฐสภา

2. กลุ่มของทางแยกภายนอกขอบเขตพิจารณาแต่มีความสัมพันธ์กับทางแยกภายในเขตพิจารณา (จำนวน 15 ทางแยก)

ชุดข้อมูลของทางสำนักการจราจรและขนส่ง กรุงเทพมหานคร ที่ขอมาได้นั้น ข้อมูลในทางแยกตั้งแต่ลำดับที่ 11-15 นั้นไม่มีข้อมูลอยู่เลย ชื่อทางแยกและหมายเลขของทางแยกจึงตั้งจากทางคณะผู้จัดทำโปรแกรมเองจึงอาจไม่ตรงกับข้อมูลของทางสำนักการจราจรและขนส่ง

ตารางที่ 3.2: รายชื่อและหมายเลขของทางแยกที่อยู่ภายนอกขอบเขตพิจารณา

ลำดับที่	หมายเลขทางแยก (เป็นหมายเลขเดียวกันกับทางสำนักการจราจรและขนส่ง)	ชื่อของทางแยก
1	15	วัดเบญจมบพิตร
2	54	อุภัย
3	71	สี่เสาเทเวศร์
4	84	อู่ทองนอก
5	97	เกียกกาย
6	139	ลานพระรูป
7	233	สวรรคโลก
8	234	นครไชยศรี (Local Road)
9	235	เทอดดำริ - เศรษฐศิริ
10	349	สี่แยกหน้าถนนระนอง 1
11*	902	สวรรคโลก-สุคันธาราม
12*	903	ทางขึ้นสะพานกรุงธนบุรี
13*	904	ท่าน้ำสามเสน
14*	905	ท่าน้ำพายัพ
15*	906	ท่าน้ำเขียวไข่กา

เนื่องจากข้อมูลที่ทางผู้จัดทำขอจากสำนักการจราจรและขนส่งในแต่ละปีนั้นมีข้อมูลไม่ครบทุกทางแยกส่งผลให้คณะผู้จัดทำต้องรวบรวมข้อมูลในหลาย ๆ ปี เพื่อที่จะทำให้ตัวระบบมีข้อมูลที่เพียงพอในทุก ๆ ทางแยกที่เกี่ยวข้องกับการคำนวณ โดยข้อมูลที่ได้มานั้นเป็นข้อมูลในช่วงเวลา 3 ปี ตั้งแต่ปี 2542 ถึง 2544 ซึ่งสามารถสรุปข้อมูลที่มีในแต่ละปีได้ดังนี้

ตารางที่ 3.3: ตารางแสดงข้อมูลที่มีในขอบเขตที่พิจารณา

ปี \ แยก	14	16	17	21	28	31	62	65	69	85	86	100	129	149	**348**	**351**	**424**	**901**
2542		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔				
2543			✔			✔				✔								
2544	✔	✔	✔		✔	✔	✔	✔	✔		✔	✔	✔	✔				
ทางแยกที่ไม่มีข้อมูลอยู่เลย ได้แก่ 348,351,424 และ 901																		

ตารางที่ 3.4: ตารางแสดงข้อมูลที่มีนอกขอบเขตที่พิจารณา

ปี \ แยก	15	54	71	84	97	139	233	234	235	**349**	**902**	**903**	**904**	**905**	**906**
2542	✔	✔			✔	✔	✔	✔	✔						
2543	✔	✔	✔		✔	✔									
2544			✔	✔	✔		✔								
ทางแยกที่ไม่มีข้อมูลอยู่เลย ได้แก่ 902,903,904,905 และ 906															

หมายเหตุ เครื่องหมาย ✔ หมายถึง ทางแยกนั้นมีข้อมูล
ส่วนช่องในตารางที่ว่าง หมายถึง ทางแยกนั้นไม่มีข้อมูล

รูปแบบข้อมูลที่ได้มาจากสำนักการจราจรและขนส่งเป็นข้อมูลในส่วนของปริมาณรถที่วิ่งเข้าไปในทางแยกในแต่ละทิศทางต่อเวลาหนึ่งชั่วโมง (คันต่อชั่วโมง) ซึ่งสามารถนำมาใช้ในระบบได้ ซึ่งข้อมูลที่จะสามารถนำมาใช้ในระบบได้นั้นต้องผ่านการแปลงให้อยู่ในรูปแบบที่ระบบสามารถนำไปใช้งานได้ โดยจะอธิบายอยู่ในบทที่ 4 ในหัวข้อ *วิธีการแปลงรูปแบบข้อมูลที่ได้จากสำนักการจราจรและขนส่งให้อยู่ในรูปแบบของฐานข้อมูลที่ใช้บนระบบ* ตัวอย่างของข้อมูลในแต่ละทางแยกที่ขอมาได้จากสำนักการจราจรและขนส่งกรุงเทพมหานครนั้นแสดงในรูปที่ 3-1 และ 3-2 ตามลำดับ

3.2 ปริมาณรถสะสมในชั่วโมงต่าง ๆ ของแต่ละถนน

ปริมาณรถสะสมจะเป็นตัวกำหนดถึงปริมาณของรถที่ค้างอยู่ในแต่ละถนนตอนเริ่มระบบจำลองสถานการณ์จราจรครั้งแรกเพื่อพยายามที่จะสร้างสถานการณ์ให้มีความใกล้เคียงกับสถานการณ์จริงมากที่สุด

3.3 แผนที่ในบริเวณพิจารณา

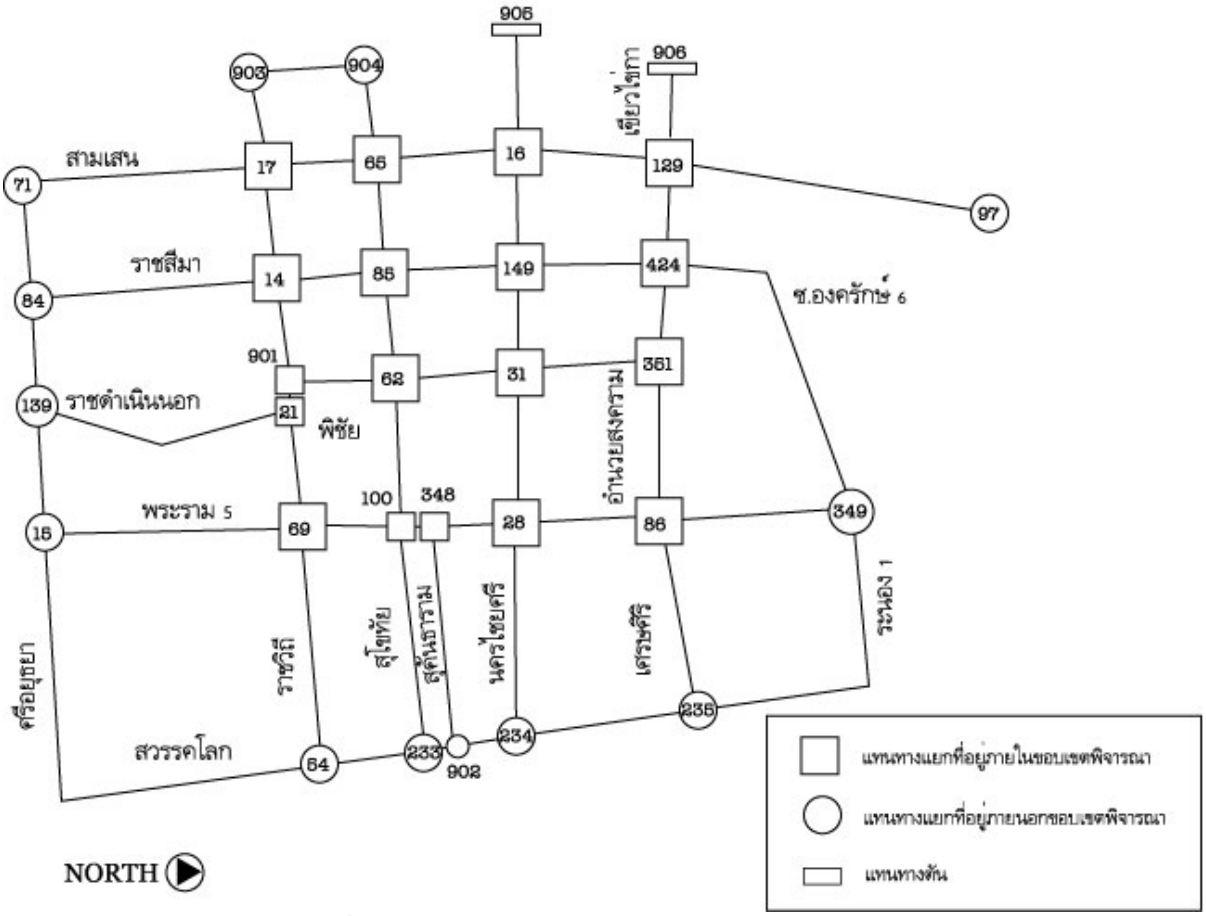
แผนที่ในบริเวณพิจารณาจะถูกนำมาเปลี่ยนให้อยู่ในรูปแบบของแผนที่ที่ใช้แสดงในระบบจำลองซึ่งทำหน้าที่แสดงผลข้อมูลในสถานการณ์จำลองว่ามีการเปลี่ยนแปลงเป็นอย่างไร เพื่อให้สามารถเข้าใจและทำให้เห็นภาพรวมของการจราจรในระบบจำลอง ซึ่งข้อมูลในส่วนนี้จะขึ้นอยู่กับลักษณะของพื้นที่ที่เราเข้าไปเก็บข้อมูลด้วยว่าเป็นอย่างไร

รูปที่ 3-3: รูปแบบแผนที่ที่ใช้แสดงผลในระบบจำลอง

3.4 ความสัมพันธ์ระหว่างทางแยกต่าง ๆ

การประมวลผลข้อมูลเพื่อหาปริมาณของรถที่จะวิ่งออกไปนั้นจำเป็นที่จะต้องรู้ว่ากลุ่มของรถที่วิ่งออกไปจะต้องไปอยู่รวมกับปริมาณรถที่สะสมค้างอยู่ในถนนใด หากเราไม่ได้คำนึงถึงความสัมพันธ์ระหว่างทางแยกต่าง ๆ แล้ว ก็จะทำให้เกิดความสับสนว่าทางแยกใดเชื่อมอยู่กับถนนใดได้ ดังนั้นข้อมูลในส่วนนี้จึงมีความจำเป็นอย่างยิ่งต่อการสร้างระบบจำลองนี้ขึ้น ซึ่งการกำหนดความสัมพันธ์ระหว่างทางแยกต่าง ๆ นั้นได้ทำการสมมติให้ทางแยกแต่ละแยกเป็นจุด โดยมีเส้นลากระหว่างจุดแทนความหมายของถนนที่เชื่อมระหว่างทางแยก 2 ทาง

แยก นั่นทำให้สามารถสรุปความสัมพันธ์ระหว่างทางแยกที่อยู่ในขอบเขตพิจารณาได้ดังรูปที่ 3-4



รูปที่ 3-4: รูปแสดงความสัมพันธ์ในแต่ละทางแยก

จากรูปจะเห็นได้ว่าแต่ละทางแยกจะมีหมายเลขกำกับไว้ (ตรงกับหมายเลขของทางสำนักการจราจรและขนส่ง) ซึ่งข้อมูลนี้จะได้มาจากรูปแบบของแผนที่ในบริเวณพิจารณานั่นเอง ส่วนรูปแบบในการจัดเก็บข้อมูลความสัมพันธ์นี้จะนำเสนออย่างละเอียดในบทที่ 4 หัวข้อ การออกแบบฐานข้อมูลในตาราง Relation

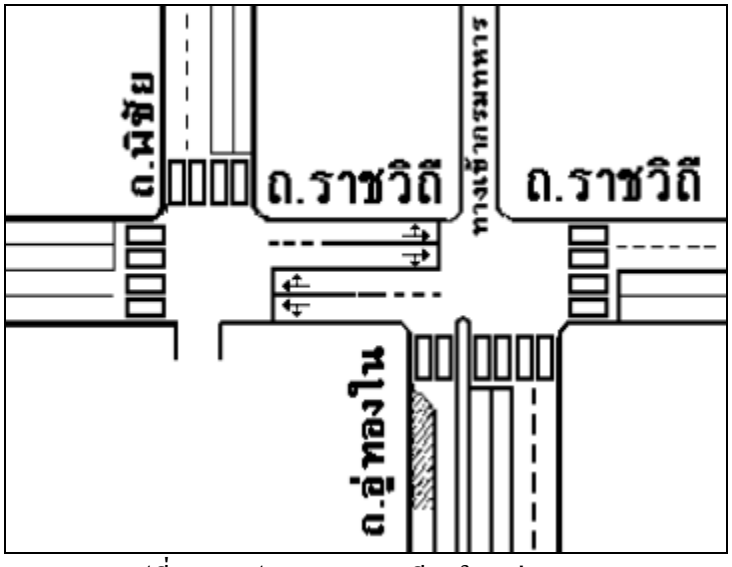
3.5 วิธีการควบคุมสัญญาณไฟจราจรบริเวณทางแยก

วิธีการควบคุมสัญญาณไฟที่กำหนดให้กับแต่ละทางแยก ย่อมมีผลต่อการเคลื่อนตัวของรถในระบบ หากเรามีการเปิดสัญญาณไฟให้กับทางใดทางหนึ่งมากไปก็อาจจะส่งผลให้อีก

ทางหนึ่งมีรถติดสะสมอยู่ในถนนเพิ่มมากขึ้น ดังนั้นวิธีการควบคุมสัญญาณไฟจึงเป็นข้อมูลที่สำคัญที่ใช้ในการกำหนดสถานการณ์ให้กับระบบจำลอง

3.6 รายละเอียดของแต่ละถนน

รายละเอียดของแต่ละถนนจะประกอบไปด้วยส่วนที่กำหนดปริมาณรถสูงสุดที่แต่ละถนนสามารถรองรับได้ (โดยคำนวณจากจำนวนช่องทางการจราจรคูณความยาวของถนนหารด้วยความยาวรถเฉลี่ย) เพื่อใช้ในการแสดงผลแต่ละถนนว่ามีการจราจรติดขัดแค่ไหน และยังมีส่วนที่เป็นข้อมูลเสริมให้กับผู้ใช้ได้เข้าใจในรูปแบบของช่องทางการจราจรของถนนที่พิจารณาอยู่อีกด้วย (เช่น มีช่องทางเลี้ยวซ้ายอยู่ 1 ช่อง ขวาอีก 1 ช่อง ตรงไป 2 ช่องทาง เป็นต้น) ซึ่งตัวอย่างรายละเอียดของถนนแสดงอยู่ในรูปที่ 3-5



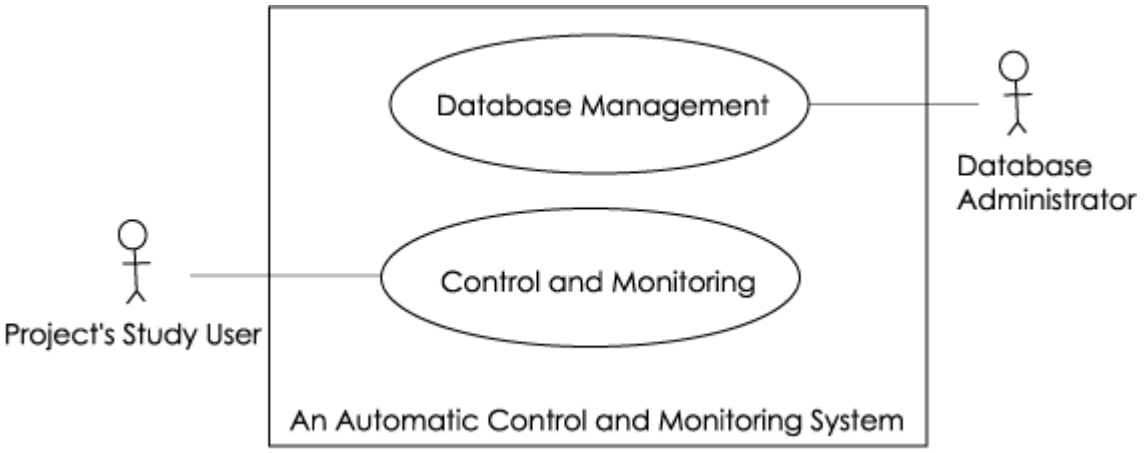
รูปที่ 3-5: รูปแสดงรายละเอียดในแต่ละถนน

ด้วยข้อมูลพื้นฐานเหล่านี้ก็ทำให้ระบบสามารถจำลองสถานการณ์การจราจรอย่างง่าย ๆ ขึ้นมาได้ ข้อมูลเหล่านี้ยังเป็นจุดกำเนิดของข้อมูลอื่น ๆ ซึ่งสามารถคำนวณได้จากข้อมูลที่มีอยู่ เพื่อใช้ในการทำงานของระบบต่อไปได้อีกด้วย

4.2 การออกแบบระบบควบคุมและติดตามการจราจรอัตโนมัติ

จากแนวคิดในการสร้างระบบและการวิเคราะห์ข้อมูลที่ได้กล่าวถึงในช่วงบทที่ผ่านมานั้นทำให้ทราบว่าการจะสร้างระบบควบคุมและติดตามการจราจรอัตโนมัติที่มีขีดความสามารถในการจำลองสถานการณ์การจราจรขึ้นจากข้อมูลทางสถิติของทางสำนักการจราจรและขนส่ง กรุงเทพมหานคร ได้นั้นต้องใช้ข้อมูลและหลักทฤษฎีต่างๆที่เกี่ยวข้องโดยเฉพาะหลักการสุ่มข้อมูลแบบปัวซองค์มาประยุกต์ใช้เพื่อให้สามารถจำลองข้อมูลให้มีความเหมือนจริงและแสดงสถานการณ์การจราจรที่เปลี่ยนแปลงไปอันเป็นผลมาจากการควบคุมสัญญาณไฟจราจรในสถานการณ์จำลองที่สร้างขึ้น

ทางคณะผู้จัดทำได้ทำการรวบรวมข้อสรุปต่าง ๆ ที่ได้ในช่วงการวิเคราะห์มากำหนดรูปแบบของระบบที่ต้องการจะพัฒนาขึ้นว่าให้ระบบมีความสามารถที่จะทำอะไรได้บ้าง โดยได้ออกแบบตัวระบบด้วยการใช้ภาษา UML (Unified Modeling Language) ซึ่งเป็นภาษามาตรฐานที่ใช้ในการสร้างโมเดลของระบบงานต่าง ๆ เพื่อความสะดวกในการพัฒนาระบบเชิงวัตถุและเพื่อให้เกิดความเข้าใจตรงกันในรูปแบบของระบบที่ได้วางไว้ ซึ่งจากข้อสรุปนำมาสร้างเป็น USE CASE ของระบบที่แสดงอยู่ในรูปที่ 4-10

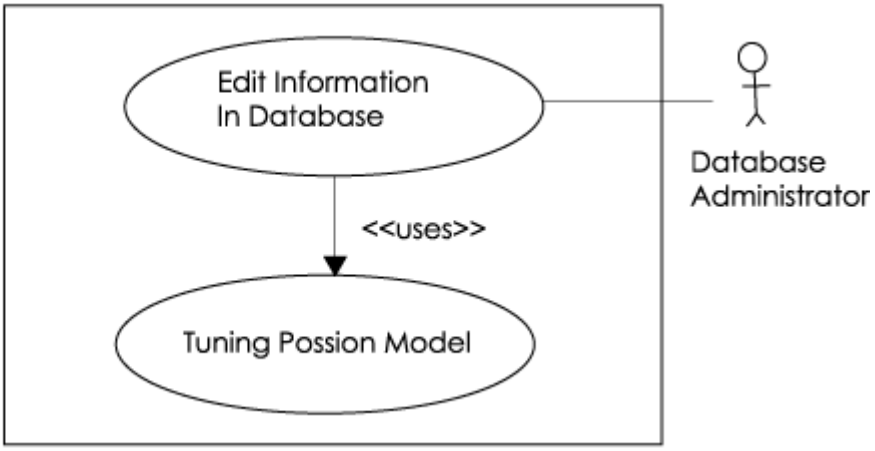


รูปที่ 4-10: Use Case ของระบบ

จากรูปภาพนี้ แสดงให้เห็นถึงผู้เกี่ยวข้องกับระบบซึ่งมีอยู่ 2 กลุ่ม คือ

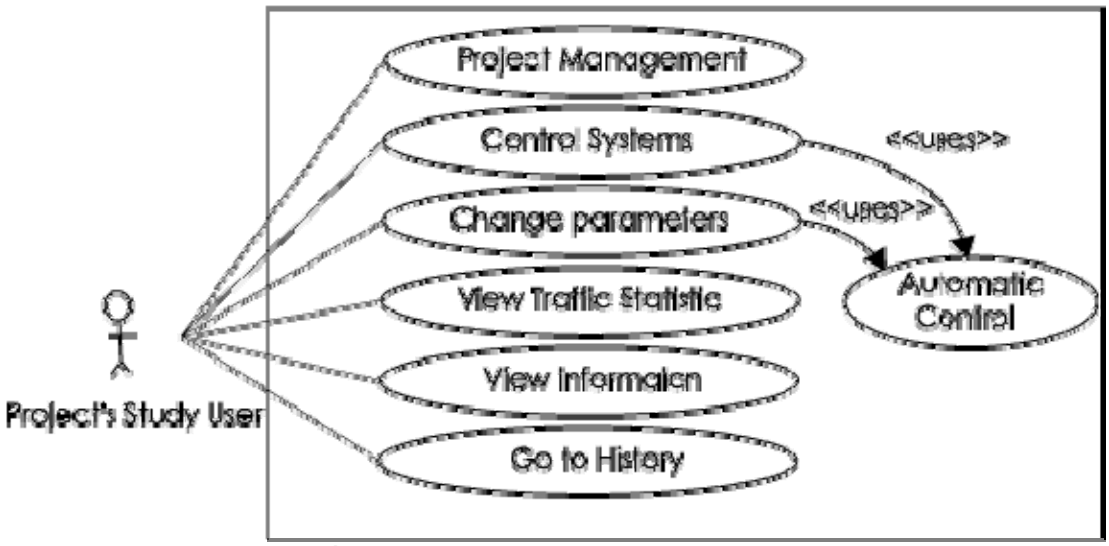
- กลุ่มของผู้ศึกษาสถานการณ์การจราจรจำลองซึ่งมีความสามารถในการควบคุมและติดตามสภาพการณ์การจราจรที่ตัวเองสร้างขึ้นมาได้
- กลุ่มของผู้ดูแลฐานข้อมูลของระบบซึ่งเป็นแหล่งข้อมูลของระบบทั้งหมด โดยมีสิทธิในการแก้ไขข้อมูลที่สำคัญต่าง ๆ ที่ใช้ในระบบได้

ในส่วนของผู้ดูแลฐานข้อมูลของระบบ หากข้อมูลที่มีการแก้ไขในฐานข้อมูลนั้นมีความเกี่ยวข้องในการคำนวณค่าป๊วซองค์แล้ว ทางระบบจะมีการคำนวณค่าภายในตาราง Tuning ใหม่เองโดยอัตโนมัติ (ตาราง Tuning ใช้ในการคำนวณปริมาณรถที่จะออกจากถนนในแต่ละทิศทาง) ดังที่แสดงอยู่ในรูปที่ 4-11



รูปที่ 4-11: Use Case ของระบบในส่วนของผู้ดูแลฐานข้อมูล

ในส่วนของผู้ที่ศึกษาสถานการณ์การจราจรจำลองนั้นจะสามารถเรียกใช้การทำงานตัวระบบผ่านทาง User Interface ได้ 6 หมวด ซึ่งได้อธิบายอยู่ในตารางที่ 4.15



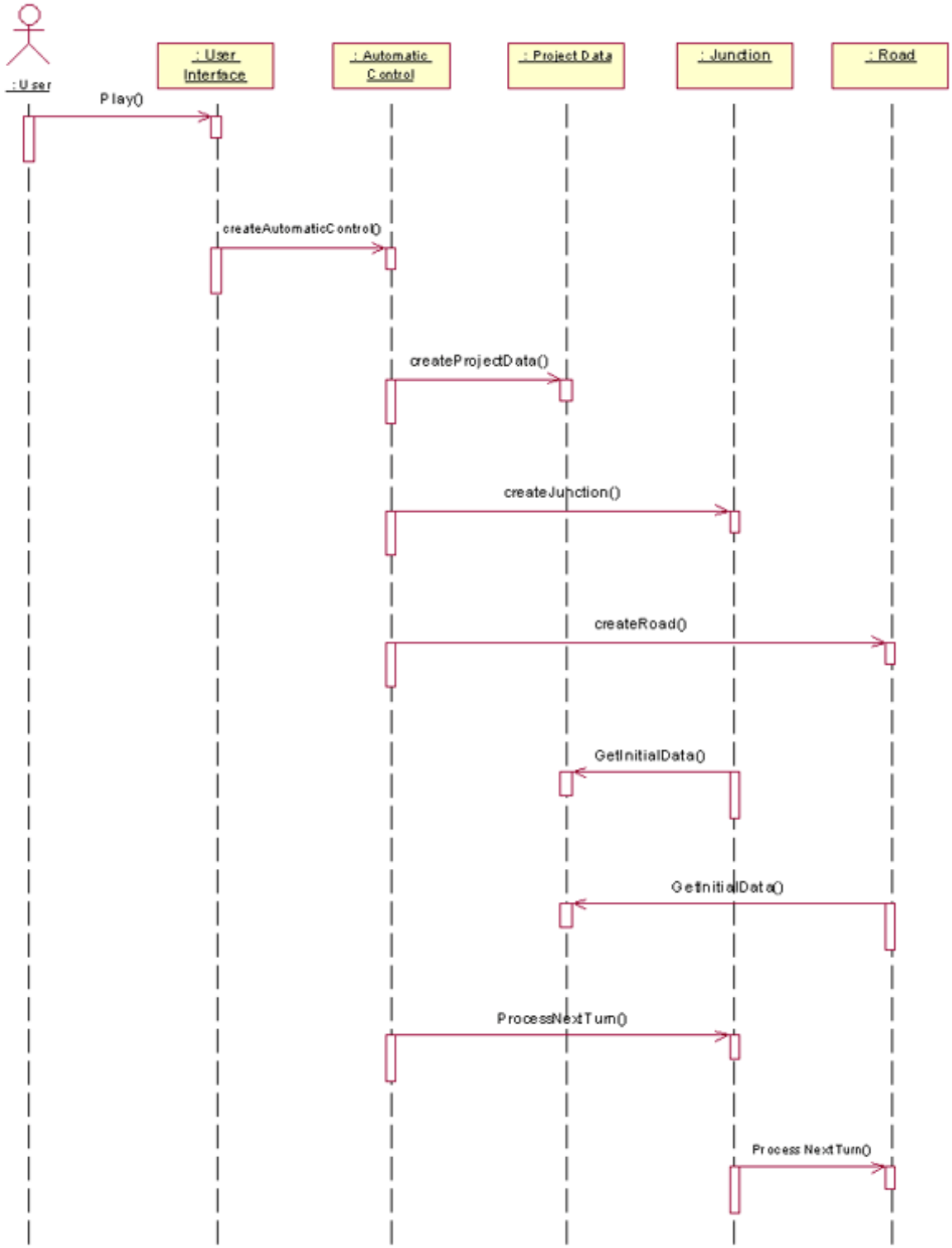
รูปที่ 4-12: Use Case ของระบบในส่วนของผู้ศึกษา

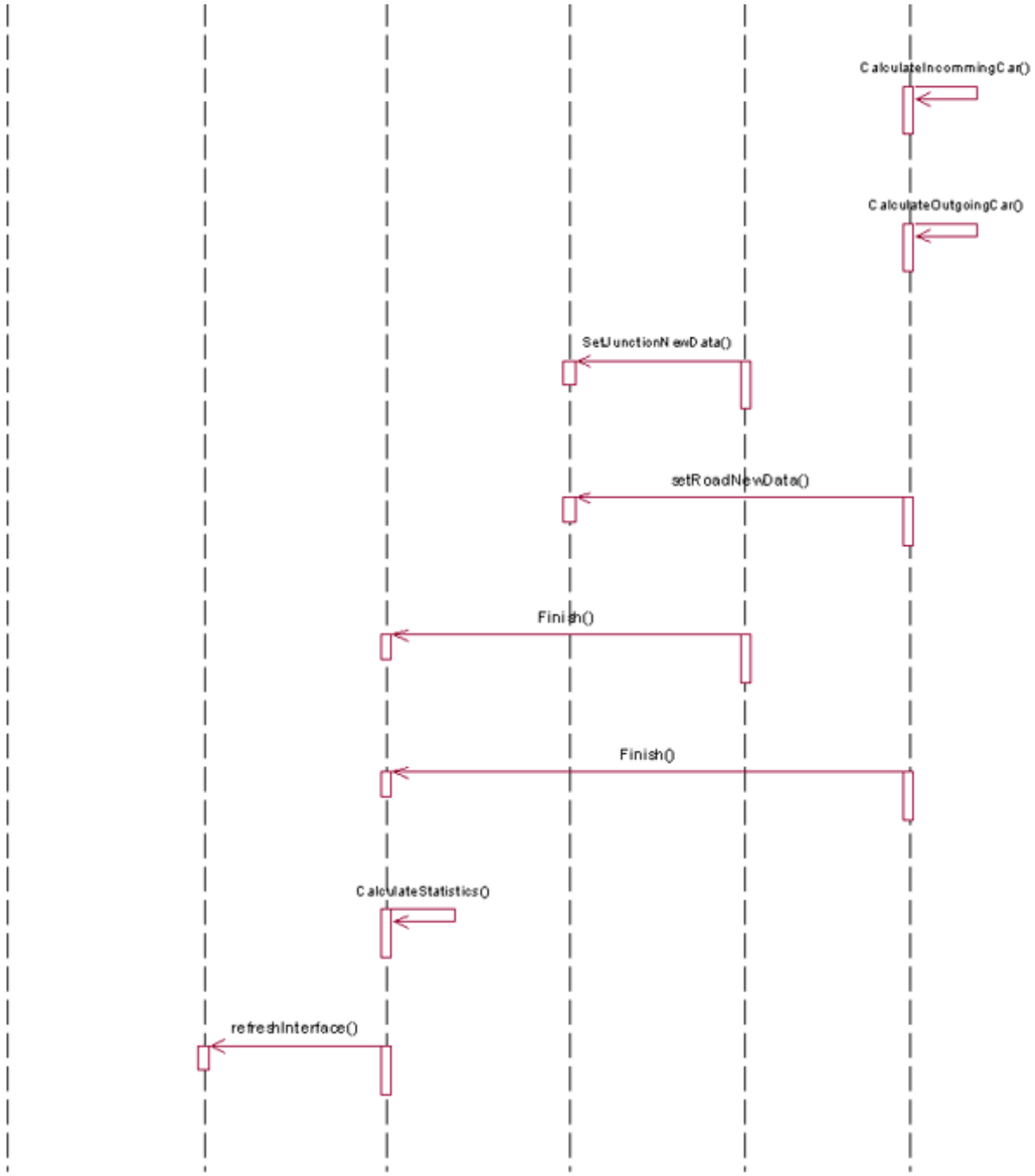
คำอธิบาย USE CASE (USE CASE DESCRIPTION)

ตารางที่ 4.11: คำอธิบาย USE CASE

Project Management	ผู้ศึกษาสามารถจัดการกับสถานการณ์จำลองที่สร้างขึ้นในระบบได้ อาทิเช่น การเปิดไฟล์ของสถานการณ์จำลองที่สร้างขึ้นแล้วเพื่อมาศึกษาต่อ (Open) หรือ การบันทึกข้อมูลของสถานการณ์จำลองที่ศึกษามา (Save) เป็นต้น
Control System	ผู้ศึกษาสามารถสั่งการให้ระบบเริ่มการจำลองหรือหยุดการจำลองของระบบก็ได้ ซึ่งเมื่อมีคำสั่งเริ่มการจำลอง ระบบก็จะรับคำสั่งให้เริ่มทำงานโดยเรียกใช้การจำลองสถานการณ์อัตโนมัติ (Automatic Control) และจะทำงานไปจนกว่าจะมีการสั่งให้หยุดการจำลองเท่านั้น
Change Parameter	ผู้ศึกษาสามารถที่จะเปลี่ยนค่าบางค่าในระบบเพื่อเพิ่มทางเลือกในการศึกษาสถานการณ์จำลอง ที่สร้างขึ้นให้มีความแตกต่างกัน เช่น การปรับระยะเวลาที่เปิดสัญญาณไฟเขียว หรือ การปรับจำนวนรถที่วิ่งเข้ามาในระบบให้มีมากขึ้น เป็นต้น
View Traffic Statistic	ผู้ศึกษาสามารถที่จะดูสถิติต่าง ๆ ที่เกี่ยวข้องกับสถานการณ์จำลอง อาทิเช่น สามารถดูปริมาณรถในแต่ละถนนเพื่อนำมาเปรียบเทียบผลจากการเปิดสัญญาณไฟจราจรในลักษณะต่าง ๆ เป็นต้น
View Information	ผู้ศึกษาสามารถดูรายละเอียดของทางแยกและถนนได้ว่ามีรูปแบบและความสัมพันธ์อย่างไร
Go to History	ผู้ศึกษาสามารถย้อนหลังกลับไปในรอบการทำงานที่ผ่านมาได้เพื่อปรับเปลี่ยนวิธีการควบคุมการจราจรในสถานการณ์จำลองที่มีผลต่อสถานการณ์จราจร เพื่อหาวิธีควบคุมการจราจรที่เหมาะสมได้
Automatic Control	ส่วนนี้จะเป็นผู้รับผิดชอบในส่วนที่เป็นการคำนวณข้อมูลทั้งหมดเพื่อนำมาสร้างสภาพการจราจรที่เป็นผลจากการควบคุมสัญญาณไฟ

หลังจากที่ได้กำหนดความสามารถของระบบซึ่งมีรายละเอียดอยู่ในรูปภาพ USE CASE จึงสามารถนำมาออกแบบส่วนของ Sequence Diagram ที่แสดงถึงลำดับการทำงานระหว่าง Object ต่างๆ ที่มีในระบบเมื่อมีการส่ง Message ระหว่างกัน แต่เนื่องจากในระบบมี Sequence Diagram สำหรับแต่ละ USE CASE เป็นจำนวนมากจึงขอยกตัวอย่าง USE CASE ที่สำคัญเพียงบางส่วนมาแสดง คือ ส่วนของการควบคุมระบบ (Control System) โดย Sequence Diagram ที่อยู่ใน USE CASE นี้ เช่น การเริ่มจำลองสถานการณ์จราจร (play) ซึ่งมีรายละเอียดดังนี้


: User
: User Interface
: Automatic Control
: Project Data
: Junction
: Road
Play()
createAutomaticControl()
createProjectData()
createJunction()
createRoad()
GetInitialData()
GetInitialData()
ProcessNextTurn()
Process NextTurn()



รูปที่ 4-13: Sequence Diagram ของการเริ่มจำลองสถานการณ์จราจร

ใน USE CASE การควบคุมระบบ (Control System)

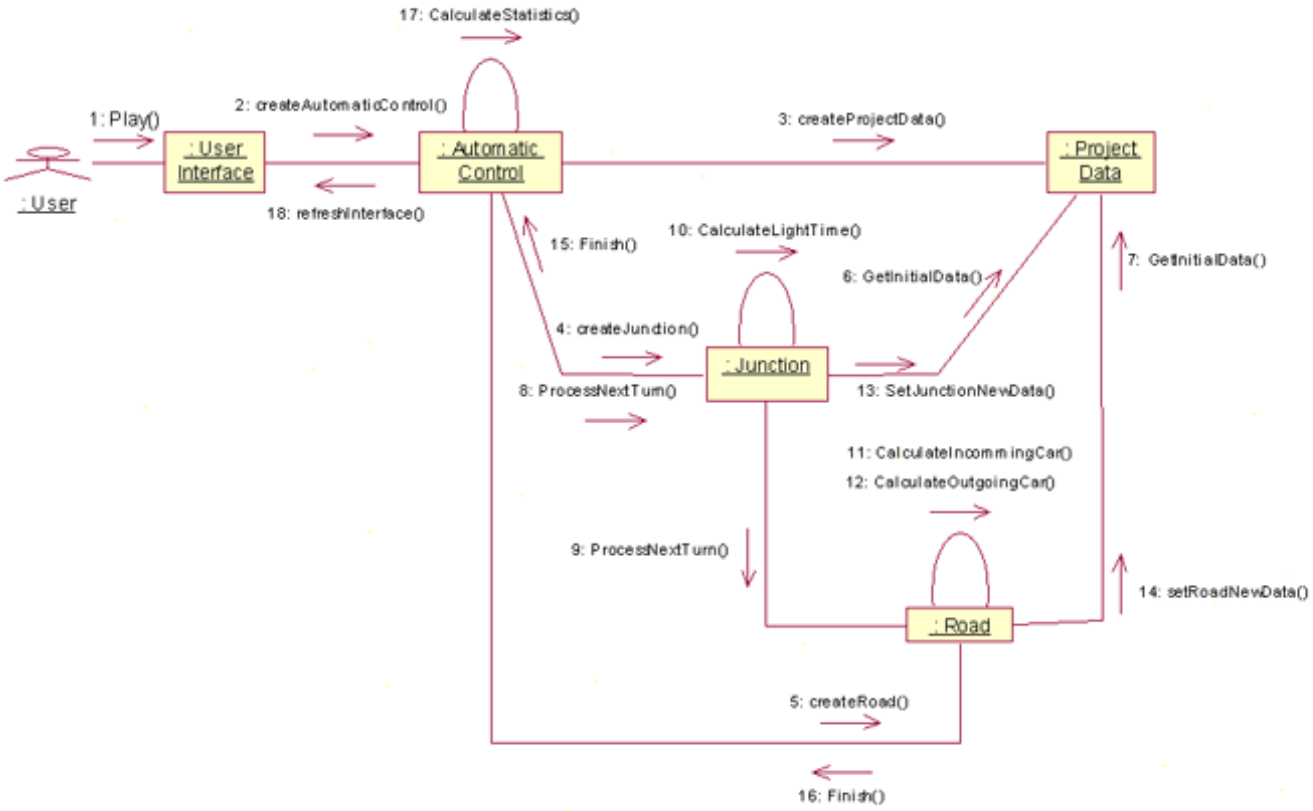
จากรูปภาพนี้จะเห็นได้ว่า เมื่อผู้ใช้โปรแกรมสั่งให้ระบบเริ่มการจำลองสถานการณ์ Class User Interface จะทำการส่ง Message ไปบอกยัง Object Automatic Control ให้ทำการสร้างสถานการณ์จำลองอัตโนมัติขึ้นมาโดยสร้างจากข้อมูลทางสถิติที่ได้เก็บเอาไว้ในฐานข้อมูล ซึ่งในส่วนนี้จะคอยควบคุมการทำงานในแต่ละรอบว่าจะต้องทำสิ่งใดบ้าง และต้องทำสิ่งใดก่อน-หลัง ซึ่งเริ่มจากการส่ง Message ต่อไปยัง Object Junction เพื่อกำหนดสถานะของทางแยกในทุก ๆ ทางแยก ซึ่งมีดังต่อไปนี้

- ลำดับการเปิดสัญญาณไฟ (Light order)
- ชนิดของการเปิดสัญญาณไฟ (Type of Light Control)
- สถานะของสัญญาณไฟ (Light state)
- ระยะเวลาการเปิดสัญญาณไฟ (Green Light Interval)
- ระยะเวลาที่เหลืออยู่ก่อนที่จะเปลี่ยนสัญญาณไฟ (Green Light Remaining Time)

ในแต่ละทางแยกนั้นจะมีสถานะของสัญญาณไฟเป็นของแต่ละทางแยกเองและเป็นตัวคอยควบคุมการเปิด ปิดสัญญาณไฟจราจรในแต่ละทางแยกนั้น เมื่อสิ้นสุดการกำหนดสถานะนี้แล้ว Object Junction จะ Message ต่อไปยัง Object Road ที่มีหน้าที่ควบคุมการเคลื่อนที่ของรถในแต่ละถนนว่าจะมีรถเข้าและออกจากถนนนี้เป็นจำนวนเท่าใด ในส่วนนี้เองที่ทำให้สถานการณ์จำลองมีการเปลี่ยนแปลงในรูปแบบต่าง ๆ อันเป็นผลมาจากการควบคุมสัญญาณไฟใน Object Junction หลังจากเสร็จสิ้นการคำนวณนี้ ผลลัพธ์ที่ถูกคำนวณของทุก ๆ ถนนจะถูกส่งไปเก็บรวมกันยัง Object ProjData ที่มีหน้าที่ในการจัดการและรวบรวมข้อมูลทั้งหมดที่ใช้บนสถานการณ์จำลองนี้ เช่นเดียวกับข้อมูลสัญญาณไฟที่จะถูกส่งไปเก็บใน Object ProjData เช่นกัน เมื่อ Object Automatic Control ได้รับสัญญาณการสิ้นสุดการคำนวณในรอบการทำงานนี้ทั้งจาก Object Junction และ Object Road แล้ว Object Automatic Control จะทำการคำนวณหาข้อมูลทางสถิติที่มีการเปลี่ยนแปลงไปในทุกรอบการทำงาน เพื่อให้ผู้ศึกษาสามารถเปรียบเทียบผลลัพธ์ที่ได้จากการควบคุมสัญญาณไฟในรูปแบบต่าง ๆ หลังจากที่ได้คำนวณสถิติก็จะส่ง Message ตัวสุดท้ายไปยัง Object User Interface อีกครั้งเพื่อให้แสดงผลอยู่ในรูปของกราฟิกเพื่อให้ผู้ศึกษาสามารถเรียนรู้และเข้าใจรวมทั้งเห็นภาพของสถานการณ์ที่เกิดขึ้นนั่นเอง

เนื่องจาก Sequence Diagram ที่ได้ออกแบบสามารถแปลงให้อยู่ในรูปแบบของ Collaboration Diagram เพื่อแสดงการส่ง Messageระหว่างแต่ละ Object โดยจะเน้นถึงโครงสร้างปฏิสัมพันธ์กันของ Object ต่าง ๆ ด้วย โดยรายละเอียดได้นำมาแสดงอยู่ในรูปภาพทางด้านล่าง

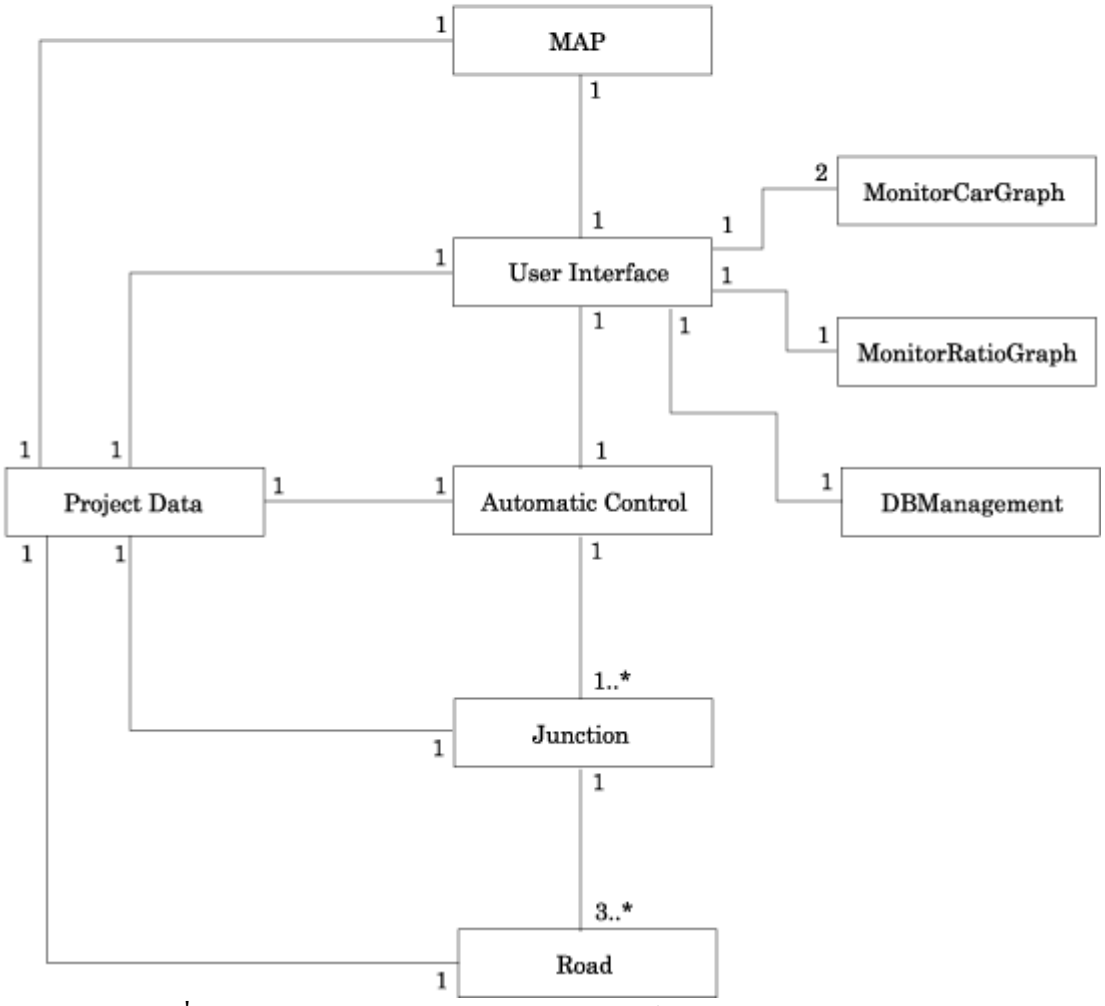
Collaboration Diagram ของการเริ่มจำลองสถานการณ์จราจร



รูปที่ 4-14: Collaboration Diagram ของการเริ่มจำลองสถานการณ์จราจร
ใน USECASE การควบคุมระบบ (Control System)

ในขั้นตอนต่อมานั้น ทางคณะผู้จัดทำได้รวบรวมรายละเอียดของแต่ละ Object ที่ปรากฏอยู่ใน Sequence Diagram ข้างต้นมาใช้ในการออกแบบ Object ของแต่ละ Object รวมถึงความสัมพันธ์ระหว่างแต่ละ Object สรุปรวมเอาไว้อยู่ใน Object Diagram โดยภายในนี้จะมีรายละเอียดของตัวแปรต่าง ๆ ที่ใช้ใน Object นั้นซึ่งรวมไปถึง Method ที่ใช้ในแต่ละ Object อีกด้วย เพื่อจะได้ง่ายต่อการเข้าใจภาพรวมการทำงานของระบบนี้ Object Diagram ที่ออกแบบสมบูรณ์แล้วมีรายละเอียดดังรูปที่ 4-15

CLASS ASSOCIATION



รูปที่ 4-15: Class Diagram แสดงความสัมพันธ์ระหว่างแต่ละ Class ในระบบ

จาก Class Diagram นี้จะเห็นถึงภาพรวมของระบบว่ามีความสัมพันธ์ในการทำงานกันอย่างไร ซึ่งDiagram นี้จะแสดงเพียงความสัมพันธ์ (ในส่วนรายละเอียดของคุณสมบัติ (Attribute) กับความสามารถ (Operation) ของ Class จะนำไปแสดงอยู่ในภาคผนวก) โดยอธิบายเป็นขั้นตอนง่าย ๆ ได้ดังนี้ คือ เมื่อผู้ใช้ได้เรียกใช้ระบบให้เริ่มการทำงานนั้น ทุก Class จะสร้าง Instance ที่จำเป็นในการ

ทำงานขึ้นมาเพื่อพร้อมที่คอยรับคำสั่งจากผู้ใช้จากส่วนติดต่อผู้ใช้ (User Interface) ส่วนติดต่อผู้ใช้นี้ประกอบไปด้วย 4 ส่วนประกอบไปด้วย

- **ส่วนที่แสดงกราฟ** โดย Class MonitorCarGraph และ MonitorRatioGraph จะเป็นผู้รับผิดชอบการแสดงผลแบบกราฟิกในรูปแบบของกราฟเส้นที่ข้อมูลมีการเปลี่ยนแปลงอยู่เสมอในทุก ๆ รอบการทำงาน
- **ส่วนที่แสดงแผนที่** Class ที่รับผิดชอบการแสดงรูปแผนที่ในขอบเขตศึกษา คือ Class MAP จะคอยคำนวณพิกัดและมาตราส่วนที่มีการเปลี่ยนแปลงในแผนที่ในทุกรอบการทำงาน เช่นเดียวกับในส่วนของกราฟ
- **ส่วนที่แสดงข้อมูลของแต่ละทางแยกหรือถนน** จะมี Class ProjData ซึ่งเป็นผู้ดูแลข้อมูลทั้งหมดในการจำลองสถานการณ์นี้ในทุก ๆ รอบการทำงาน (ในโปรเจคนั้น) จะเป็นผู้คอยจัดหาข้อมูลที่ผู้ใช้ต้องการมานำแสดงบนหน้าจอให้รวดเร็วและถูกต้อง
- **ส่วนที่จัดการฐานข้อมูล** ในส่วนของการแก้ไขข้อมูลในฐานข้อมูลนั้นจะมี Class DBManagement ที่เป็นตัวดูแลและจัดการข้อมูลที่อยู่ในฐานข้อมูลทั้งหมดซึ่งมีความสามารถในการเปลี่ยนแปลงข้อมูลเท่านั้นเพราะข้อมูลในส่วนนี้มีความสำคัญต่อระบบทั้งหมด การแก้ไขข้อมูลที่ผิดพลาดอาจก่อให้เกิดความเสียหายต่อระบบได้ ดังนั้นส่วนนี้จึงสงวนไว้สำหรับผู้ดูแลฐานข้อมูลเท่านั้น

เมื่อผู้ใช้มีการติดต่อกับส่วนติดต่อผู้ใช้ (User Interface) Class นี้ก็จะเป็นผู้ตัดสินใจและกระจายงานไปให้กับส่วนที่รับผิดชอบนั้น ๆ ทำงานต่อไป เช่น การแก้ไขฐานข้อมูล User Interface ก็จะส่งงานให้กับส่วนที่จัดการฐานข้อมูลที่อยู่ใน Class DBManagement รับผิดชอบต่อ เมื่อเสร็จสิ้นการทำงาน User Interface ก็จะรวบรวมผลลัพธ์ที่ได้กลับมาแสดงต่อผู้ใช้ ในส่วนของ Class Automatic Control นั้นจะคอยรับผิดชอบในส่วนการจำลองสถานการณ์การจราจรซึ่งเป็นส่วนที่ทำงานอยู่ด้านหลัง ซึ่งทำหน้าที่คำนวณผลลัพธ์จากชุดข้อมูลที่ผู้ใช้กำหนดให้และส่งไปยัง Class MAP เพื่อคำนวณกราฟิกใหม่ที่จะใช้แสดงในแต่ละรอบการทำงาน

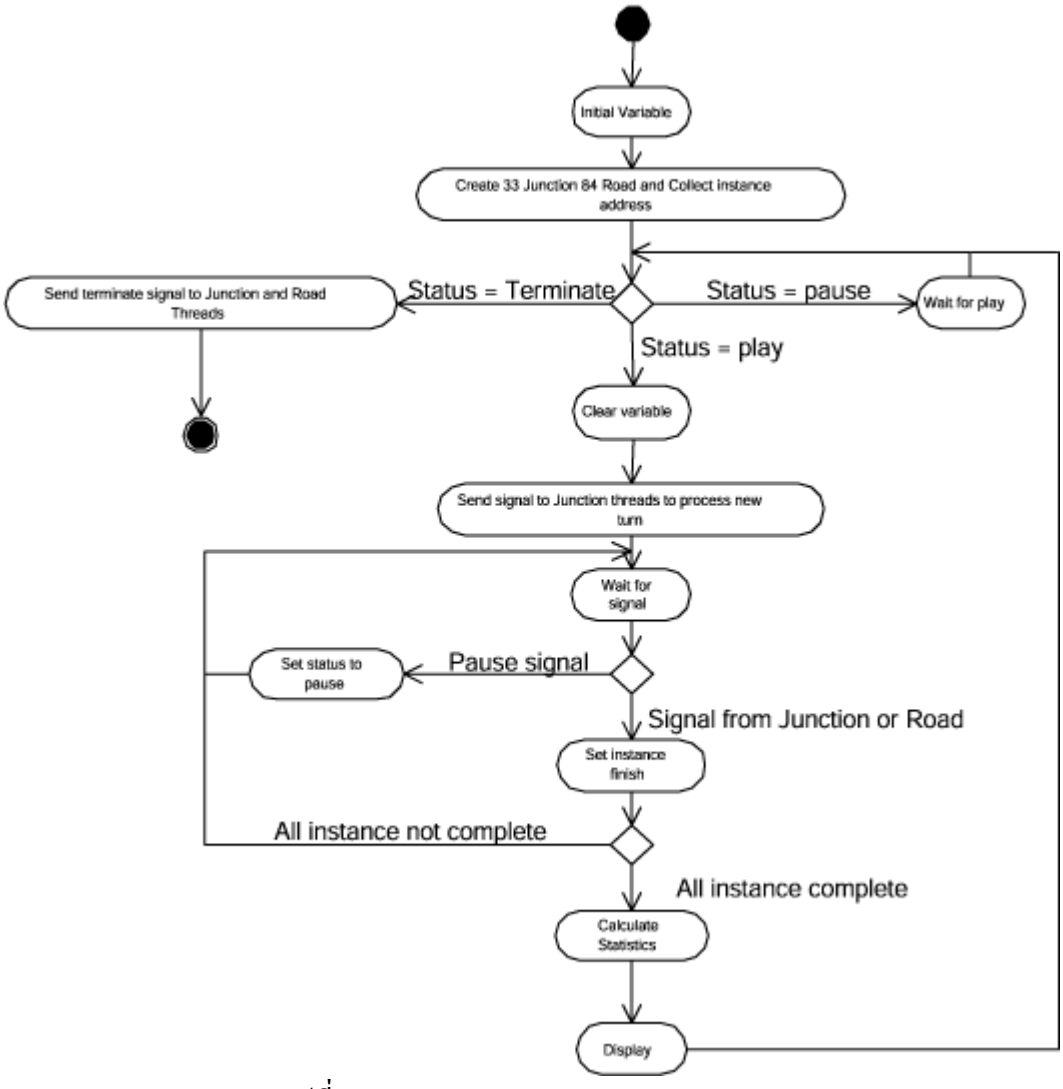
ส่วนของ Class Automatic Control นี้จะอธิบายหลักการทำงานภายใน Class เพิ่มเติมในช่วงของ Activity Diagram ส่วน Class Junction และ Class Road จะทำงานร่วมกับส่วนของ Class Automatic Control เพื่อคำนวณปริมาณการจราจรที่เปลี่ยนไปอันเป็นผลมาจากการควบคุมสัญญาณไฟจราจร ซึ่งการหาปริมาณการจราจรจะขึ้นกับส่วนของ Class Road เป็นผู้รับผิดชอบ ส่วนการควบคุม

สัญญาณไฟจราจรนั้น Class Junction จะเป็นคนดูแลการเปิด-ปิดสัญญาณไฟจราจร รวมทั้งป้องกันการกำหนดค่าให้ตัวควบคุมไฟจราจรที่เกิดจากความผิดพลาด เช่น การป้องกันการเปิดไฟเขียวให้กับถนน 2 ถนนที่ตัดกันในเวลาเดียวกัน ซึ่งการทำงานใน 2 ส่วนนี้จะอธิบายเพิ่มเติมในช่วงของ Activity Diagram เช่นกัน

หลังจากที่ Class Diagram ได้เสนอมุมมองในแง่ภาพรวมของระบบรวมทั้งแสดงถึงความสัมพันธ์ของการทำงานระหว่าง Class ไปแล้วนั้น ก็ถึงเวลาที่เราจะมาพิจารณาถึงขั้นตอนในการทำงานของระบบ ซึ่งสามารถแสดงให้เข้าใจถึงหลักการทำงานด้วย Activity Diagram

Activity Diagram เป็น Diagram ที่แสดงถึงขั้นตอนในการทำงานที่ต้องมีการตัดสินใจซึ่งเกิดภายใน Object ที่เกี่ยวข้องในการทำงานแต่เนื่องจากในระบบนั้นมีขั้นตอนการทำงานมากมายในแต่ละ Object ดังนั้นจึงจะขอยกตัวอย่าง Activity Diagram เพียงบางส่วนเอามาแสดง ซึ่ง Diagram ที่จะนำมานั้นอยู่ในส่วนของ Automatic Control, Junction และ Road ซึ่งมีรายละเอียดดังต่อไปนี้

Automatic Control

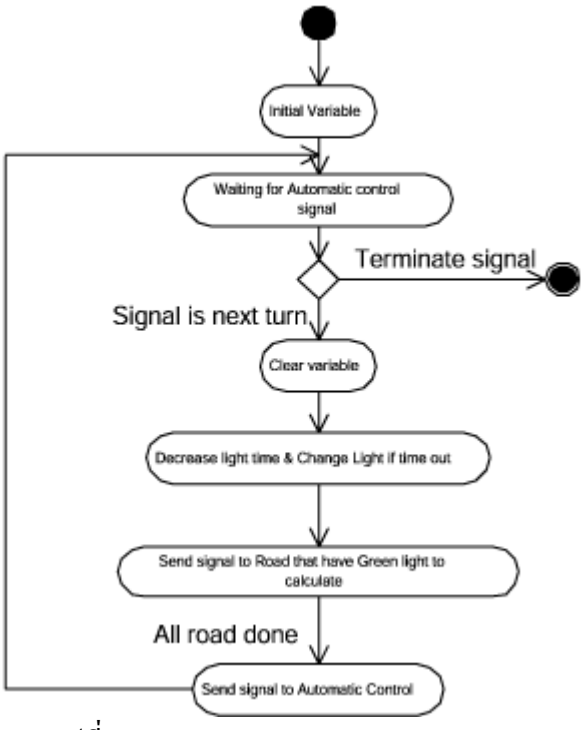


รูปที่ 4-16: Activity Diagram ของ Automatic Control

หลักการทำงานของ Automatic Control จะทำงานแบบทำซ้ำโดยที่ 1 รอบการทำงานจะเท่ากับเวลาที่ใช้ในระบบจำลอง 1 หน่วยเวลา (อาจจะเป็น 1 วินาที หรือ 1 นาทีก็ได้) เมื่อตอนที่ Object ของ Class Automatic Control ถูกสร้างขึ้นนั้น จะมีการสร้าง Object ของ Class Junction และ Class Road ที่เกี่ยวข้องตามมาด้วย หลังจากนั้นตัว Automatic Control จะตรวจสอบสถานะของระบบ ซึ่งมีอยู่ 2 สถานะ ได้แก่ สถานะเริ่ม (Play) กับสถานะหยุดการจำลอง (Stop) โดยที่หากสถานะของระบบนั้นเป็น Stop ส่วนของ Automatic Control จะรอจนกระทั่งสถานะเปลี่ยนเป็น Play จึงค่อยเริ่มการทำงาน เมื่อเริ่มทำงานส่วนของ Automatic Control จะเตรียมตัวแปรต่างๆให้

พร้อมในการทำงาน จากนั้นจะสั่ง Class Junction ให้เริ่มทำงาน (รายละเอียดอยู่ใน Activity Diagram ของ Junction) ระหว่างนี้ Automatic Control จะคอยรอรับสัญญาณ (Signal) ที่จะส่งกลับมาจาก Junction และ Road ทั้งหมดเพื่อแสดงการทำงานของ Class ทั้งสองว่าทำทุก Junction และ Road เสร็จสมบูรณ์ หลังจากนั้น Automatic Control ก็จะทำการคิดค่าสถิติใหม่จากข้อมูลที่ถูกคำนวณขึ้นจาก Class ทั้งสองแล้วนำข้อมูลทั้งหมดส่งไปให้ User Interface ใช้ในการแสดงผล ขั้นตอนการทำงานของ Class Junction และ Road เมื่อได้รับ Message จาก Automatic Control จะแสดงอยู่ในรูปถัดไป

Junction Thread

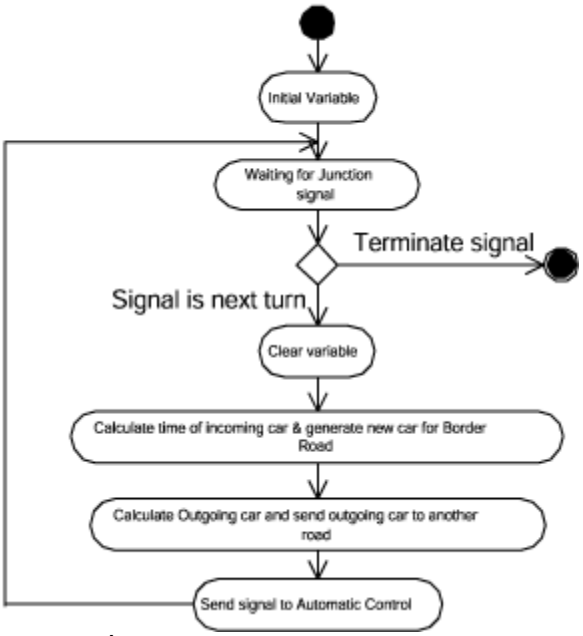


รูปที่ 4-17: Activity Diagram ของ Junction

หลักการทำงานของ Junction เมื่อได้รับ Message จากส่วน Automatic Control ที่ส่งมาให้แล้ว Class Junction นั้นจะเริ่มการทำงาน โดยหน้าที่ของ Junction คือ การคำนวณเวลาการเปิดปิดสัญญาณไฟจราจรใน 1 รอบการทำงานนั้นและจะส่ง Message ไปให้กับ Object Road ที่รอสัญญาณไฟจากทางแยก (Junction) ที่ทำงานอยู่ (ในระบบมีหลายทางแยก) ซึ่งในแต่ละแยกจะมีข้อมูลต่างๆของตัวเองและสามารถทำงานได้โดยอัตโนมัติ เมื่อทำเสร็จก็จะมีการส่ง Message กลับไปยัง Automatic Control เพื่อ

แจ้งว่า Junction นี้ทำงานเสร็จและหยุดรอ Message จาก Automatic Control ที่จะส่งมาบอกให้ทำงานในรอบต่อไป

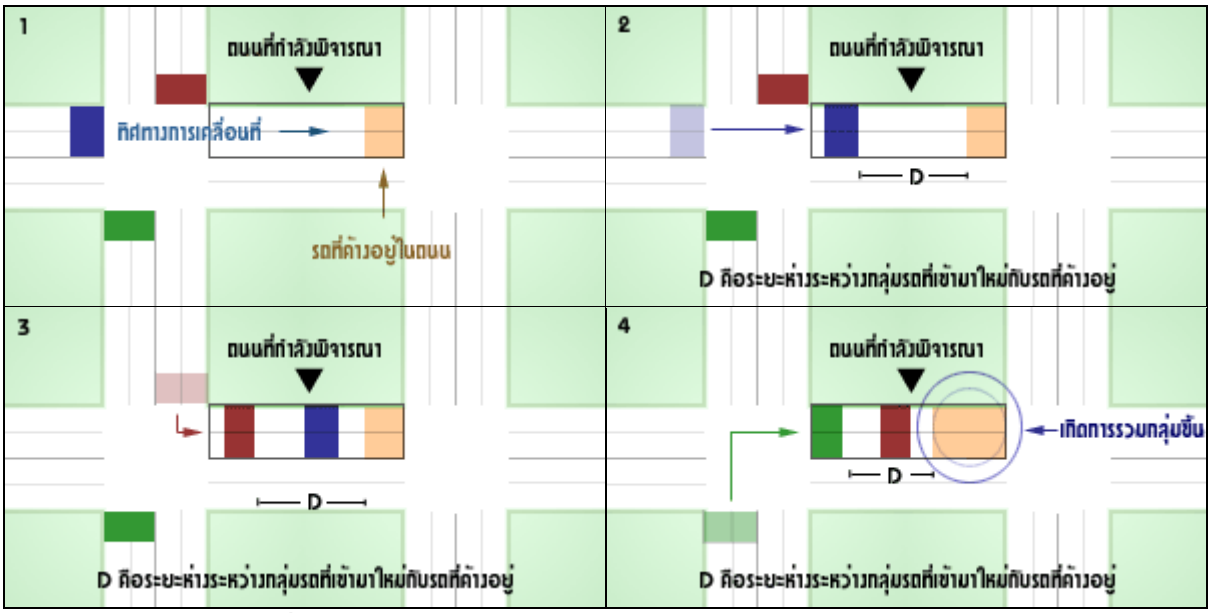
Road Thread



รูปที่ 4-18: Activity Diagram ของ Road

หลักการทำงานของ Road จะเริ่มต้นทำงานเมื่อ Junction ส่ง Message มาให้กับ Class Road เมื่อได้รับแล้ว Road จะทำการคำนวณปริมาณรถที่จะเข้ามาในถนนนี้จากถนนอื่นๆ ที่อยู่ติดกัน หลังจากนั้นก็จะคำนวณรถที่จะออกจากถนนนี้และส่งข้อมูลของรถที่ออกจากถนนนี้ไปยังอีกถนนถนนอื่นๆ ที่อยู่ติดกัน เพื่อรอที่จะนำไปรวมกับปริมาณของรถในถนนเหล่านั้น จึงจะเสร็จสมบูรณ์ใน 1 รอบการทำงาน ท้ายที่สุด Road ก็จะส่ง Message ไปยัง Automatic Control เพื่อแจ้งว่า Object Road นี้ทำงานเสร็จแล้วและรอที่จะทำงานในรอบถัดไป

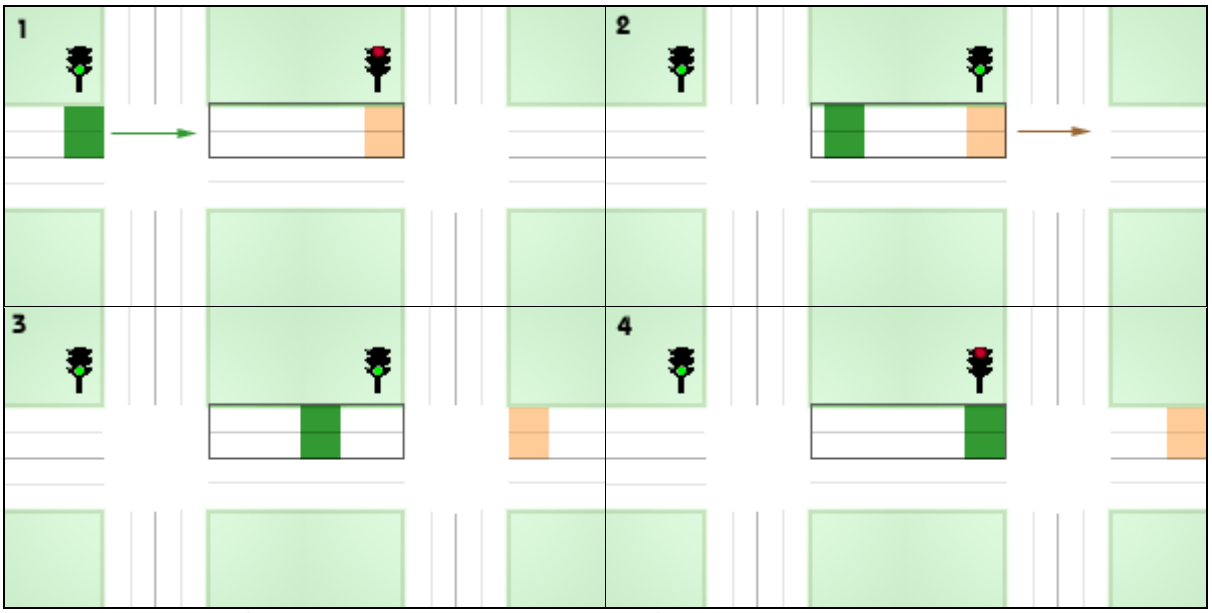
ในการคำนวณปริมาณรถที่จะเข้ามาในถนนที่พิจารณานั้น กลุ่มของรถที่เข้ามาจากถนนอื่น ๆ ที่อยู่ติดกันจะไม่สามารถเอาจำนวนของรถในแต่ละกลุ่มมารวมกันได้ทันทีเนื่องจากรถแต่ละกลุ่มจะมีระยะทางที่ต้องวิ่งก่อนถึงท้ายแถวของรถที่ค้างอยู่ในถนนที่พิจารณาอยู่ไม่เท่ากัน (ระยะทางระหว่างกลุ่มรถที่เคลื่อนมาได้ในรอบการทำงานนี้จนถึงตำแหน่งท้ายแถวของถนนนี้) ทำให้ต้องมีการคำนวณระยะทางใหม่ให้กับรถแต่ละกลุ่มที่มาจากคนละทิศทางกันในทุก ๆ รอบการทำงานจนกว่าจะมีการรวมกับกลุ่มรถชุดอื่น ๆ ดังที่แสดงในรูปที่ 4-19



รูปที่ 4-19: รูปแสดงระยะทางของรถแต่ละกลุ่มก่อนจะมีการรวมกลุ่ม

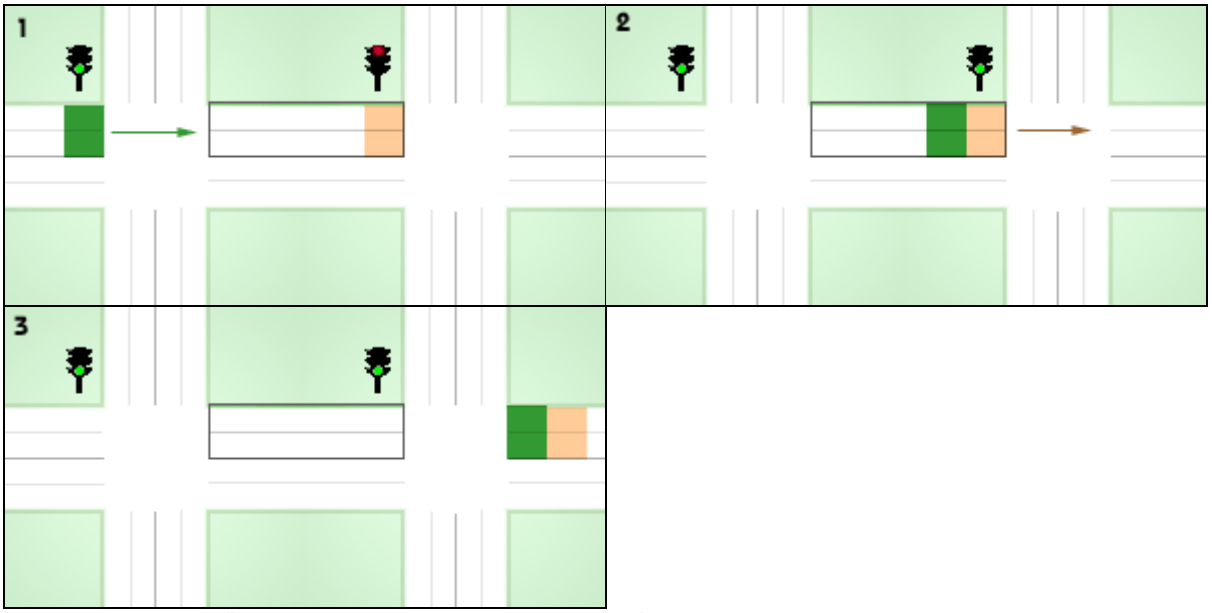
จากรูปจะเห็นได้ว่ากลุ่มรถที่มาจากแต่ละทิศทางนั้นจะไม่สามารถนำกลุ่มรถมารวมกันได้ทันที เนื่องจากจะก่อให้เกิดความผิดพลาดในการคำนวณได้ โดยสามารถพิจารณาได้ในกรณีต่อไปนี้

สมมติให้มีกลุ่มรถที่มาใหม่กำลังวิ่งไปสู่ถนนที่เรากำลังพิจารณาซึ่งมีกลุ่มรถที่ค้างอยู่ในถนน (รูปที่ 1) ต่อมาถนนนั้นมีการปล่อยรถออกไปในรอบสัญญาณไฟนั้น (รูปที่ 2) แต่รถกลุ่มใหม่ที่กำลังตามมานั้นมาไม่ทัน (รูปที่ 3) เนื่องจากไฟแดงก่อนที่จะตามกลุ่มหน้าทัน (รูปที่ 4) หากเราคำนึงถึงระยะทางของรถแต่ละกลุ่มก็จะทำให้การคำนวณมีความถูกต้องและทำให้สถานการณ์ที่สร้างมีความใกล้เคียงกับความเป็นจริง ดังที่แสดงในรูปที่ 4-20



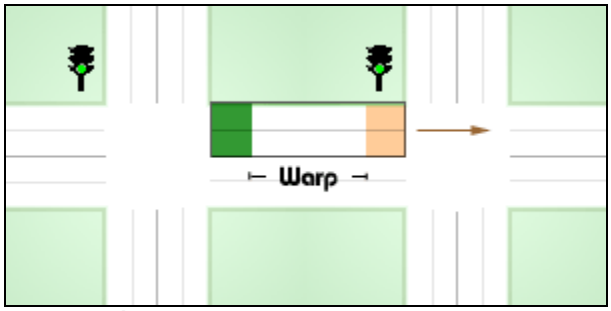
รูปที่ 4-20: รูปแสดงการคำนวณในกรณีที่คำนึงถึงระยะทางของรถแต่ละกลุ่ม

แต่หากในกรณีนี้เราไม่ได้คำนึงถึงระยะทางของรถแต่ละกลุ่ม นั่นคือ รวมรถทั้ง 2 กลุ่มเข้าไปด้วยกันตั้งแต่ที่รถกลุ่มใหม่เข้ามาในถนนนี้ การคำนวณก็จะทำให้รถทั้ง 2 กลุ่มนั้นผ่านไฟแดงไปด้วยกัน ทั้งที่ขัดกับความจริงที่รถกลุ่มใหม่นั้นควรจะต้องรอสัญญาณไฟ เนื่องจากยังตามกลุ่มด้านหน้าไม่ทัน ดังที่แสดงในรูปที่ 4-21



รูปที่ 4-21: รูปแสดงการคำนวณในกรณีที่ไม่คำนึงระยะทางถึงของรถแต่ละกลุ่ม

ในกรณีหลังนี้ทำให้เกิดความผิดพลาดในการคำนวณขึ้นเนื่องจากระยะห่างของรถแต่ละกลุ่มนั้นหายไป (เสมือนว่ารถกลุ่มนั้นวาร์ปได้) ทำให้ปริมาณรถที่คำนวณมีค่าไม่ถูกต้อง ดังรูปที่ 4-22



รูปที่ 4-22: รูปแสดงการวาร์ปของกลุ่มรถ

ด้วยเหตุนี้จึงทำให้เราต้องคำนึงถึงระยะทางระหว่างกลุ่มรถที่มาจากคนละทิศทางกันในทุกๆ รอบการทำงานเพื่อให้การคำนวณมีความถูกต้องและใกล้เคียงกับความจริงมากที่สุด

หลังจากที่ได้ทราบถึงการออกแบบระบบแล้วว่ามีรูปแบบการทำงานในแต่ละส่วนอย่างไร แต่นั่นก็เป็นเพียงการออกแบบจากแนวคิดทั้งหมด ซึ่งอาจจะมีบางส่วนที่ต้องมีการแก้ไขและปรับปรุงให้เหมาะสมกับเทคนิคที่นำมาใช้ในการสร้างระบบในความเป็นจริงอีกครั้งหนึ่ง โดยเทคโนโลยีที่นำมาใช้ในการสร้างในตัวระบบนี้ได้ถูกรวบรวมเอาไว้ในบทถัดไป

บทที่ 4

การออกแบบฐานข้อมูลและระบบควบคุมและติดตามการจราจรอัตโนมัติ

เนื้อหาในบทนี้ถูกจัดแบ่งออกเป็น 2 ส่วน โดยที่ส่วนแรกของบทนั้นจะกล่าวถึงการออกแบบตัวฐานข้อมูลที่ใช้ในระบบควบคุมและติดตามการจราจรอัตโนมัติว่ามีรูปแบบและความสัมพันธ์ในฐานข้อมูลเป็นอย่างไร หลังจากนั้นจะเป็นส่วนของการออกแบบตัวระบบซึ่งจะกล่าวถึงการออกแบบระบบการควบคุมและติดตามการจราจรอัตโนมัติว่ามีรูปแบบและหลักการทำงานเป็นอย่างไร

4.1 การออกแบบฐานข้อมูลที่ใช้ในระบบการควบคุมและติดตามการจราจรอัตโนมัติ

การออกแบบตัวฐานข้อมูลในระบบนั้น ได้ออกแบบตามหลักการที่สำคัญต่าง ๆ เพื่อช่วยจัดการข้อมูลที่มีอยู่เป็นจำนวนมากให้เกิดความเป็นระเบียบ มีความรวดเร็วในการดึงข้อมูลและง่ายต่อการที่จะนำข้อมูลไปใช้ รวมทั้งยังต้องช่วยประหยัดหน่วยความจำในเครื่องคอมพิวเตอร์ที่ทำงานบนระบบ เนื่องจากตัวระบบนั้นมีการทำงานและบันทึกข้อมูลเป็นจำนวนมากอยู่ตลอดเวลา ดังนั้นการออกแบบจึงต้องออกแบบให้สามารถรองรับกับตัวระบบเพื่อให้ระบบสามารถทำงานได้โดยไม่เกิดปัญหาระหว่างการทำงานและใช้ทรัพยากรที่มีอยู่อย่างจำกัดให้เกิดความคุ้มค่าที่สุด ซึ่งในส่วนของการออกแบบฐานข้อมูลประกอบไปด้วยหัวข้อย่อยต่าง ๆ ดังต่อไปนี้

- รูปแบบฐานข้อมูลที่สร้างขึ้นมาใช้ในระบบ
- รูปภาพสรุปความสัมพันธ์ในฐานข้อมูล
- วิธีการแปลงรูปแบบข้อมูลที่ได้จากสำนักการจราจรและขนส่งให้อยู่ในรูปแบบของฐานข้อมูลที่ใช้บนระบบ

4.1.1 รูปแบบฐานข้อมูลที่ใช้ในระบบ

จากการวิเคราะห์ข้อมูลที่จำเป็นต้องใช้ในการสร้างระบบเพื่อจำลองสถานการณ์การจราจรนั้น จะสังเกตได้ว่าในระบบมีสิ่งที่เกี่ยวข้อง(Entity) ได้แก่ ทางแยก (Junction) ถนน (Road) ความสัมพันธ์ระหว่างทางแยกกับถนน (Relation) และปริมาณการจราจรในแต่ละถนน (Statcar) เพื่อให้สามารถจำลองสถานการณ์ต่าง ๆ ขึ้นมาได้ การออกแบบฐานข้อมูลของระบบจึงสร้างมาจาก Entity พื้นฐานและความสัมพันธ์ระหว่าง Entity เหล่านี้ ฐานข้อมูลจึงประกอบไปด้วยตารางทั้งหมด 10 ตาราง ซึ่งประกอบไปด้วยตารางต่าง ๆ ต่อไปนี้

1) ตาราง Junction

หน้าที่ของตาราง : จัดเก็บข้อมูลของทางแยกทั้งหมดที่ปรากฏอยู่ในระบบ มีทั้งหมด 6 เขตข้อมูล ซึ่งประกอบไปด้วยเขตข้อมูลต่าง ๆ ดังนี้

ตารางที่ 4.1: Schema ของ Junction Table

Field Name	**Data Type**	**Description**
jid	smallint(6) not null	เก็บหมายเลขทางแยก ซึ่งกำหนดให้มีหมายเลขเดียวกับของสำนักการจราจรและขนส่ง เพื่อให้ข้อมูลสามารถเข้ากันได้ (Compatible) กับข้อมูลของสำนักการจราจรและขนส่งและยังทำหน้าที่เป็น Primary Key ของตารางนี้ด้วย
name	Tinytext	เก็บชื่อของทางแยกที่ตรงกันกับของสำนักการจราจรและขนส่ง
connect_num	tinyint(1) not null	เก็บจำนวนของทางแยกอื่น ๆ ที่เชื่อมกับทางแยกนี้ จากเขตข้อมูลนี้จะสามารถบอกได้ว่าทางแยกเป็นทางแยกแบบใด (สามแยกหรือสี่แยก)
bound	enum('true', 'false') not null	เก็บสถานะที่บอกว่าทางแยกนี้เป็นจุดที่อยู่นอกขอบเขตพิจารณาหรือไม่ โดยที่ค่า true นั้น หมายถึง ทางแยกนั้นเป็นจุดนอกขอบเขตพิจารณา ส่วนค่า False นั้นจะหมายถึง ทางแยกนั้นเป็นจุดที่อยู่ในขอบเขตที่พิจารณา
have_light	enum('yes', 'no') not null	เก็บสถานะของทางแยกนั้นว่ามีสัญญาณไฟคอยควบคุมการจราจรอยู่บริเวณทางแยกหรือไม่ โดยค่า Yes นั้น หมายถึง ทางแยกนั้นมีสัญญาณไฟควบคุมอยู่ ส่วนค่า No หมายถึง ไม่มีสัญญาณไฟ
junc_image	Blob	เก็บรูปภาพกราฟิกที่แสดงรูปแบบช่องทางการจราจรบริเวณทางแยก

2) ตาราง Road

หน้าที่ของตาราง : จัดเก็บข้อมูลของถนนที่ปรากฏอยู่ในระบบ โดยมีเขตข้อมูลทั้งหมด 2 เขตดังนี้

ตารางที่ 4.2: Schema ของ Road Table

Field Name	**Data Type**	**Description**
rid	smallint(4) not null	เก็บหมายเลขของถนน ซึ่งเขตข้อมูลนี้ทำหน้าที่เป็น Primary Key ของตารางนี้ด้วย
name	tinytext	เก็บชื่อของถนนที่ใช้บนระบบ

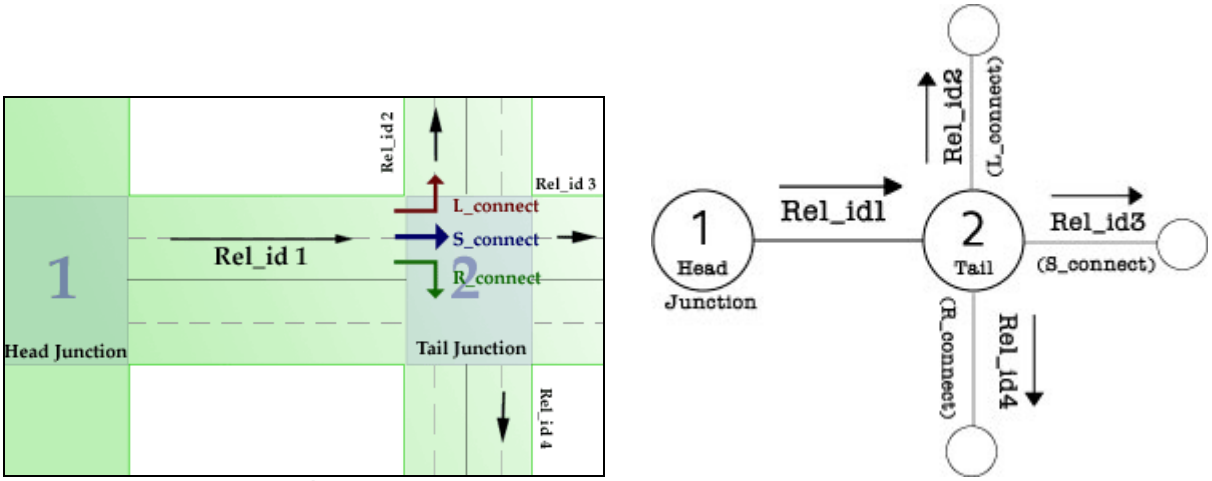
3) ตาราง RELATION

หน้าที่ของตาราง : จัดเก็บความสัมพันธ์ระหว่างทางแยกกับถนนในระบบ ประกอบไปด้วยเขตข้อมูลทั้งหมด 11 เขตดังนี้

ตารางที่ 4.3: Schema ของ Relation Table

Field Name	**Data Type**	**Description**
rel_id	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ระหว่างสองทางแยกโดยมีถนนเป็นตัวเชื่อมตรงกลางระหว่างทางแยกทั้งสอง
head_junc	smallint(6) not null	เก็บหมายเลขทางแยกของหัวถนน
tail_junc	smallint(6) not null	เก็บหมายเลขทางแยกของท้ายถนน
rid	smallint(4)	เก็บหมายเลขของถนนที่อยู่ระหว่างทางแยกทั้งสอง
lane_num	tinyint(2) not null	เก็บจำนวนช่องทางการจราจรของถนนที่อยู่ระหว่างทางแยกทั้งสอง
l_connect	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ถัดไปในทิศเลี้ยวซ้ายจากความสัมพันธ์นี้
s_connect	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ถัดไปในทิศทางตรงต่อจากความสัมพันธ์นี้
r_connect	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ถัดไปในทิศเลี้ยวขวาจากความสัมพันธ์นี้
rel_length	int(11) not null	เก็บความยาวของถนนส่วนที่อยู่ระหว่างทางแยกทั้งสอง
direction	enum('North', 'South', 'West', 'East') not null	เก็บทิศทางของถนนเทียบกับทิศทางในแผนที่
lane_image	Blob	เก็บกราฟิกที่แสดงรูปแบบช่องทางการจราจรของความสัมพันธ์นี้

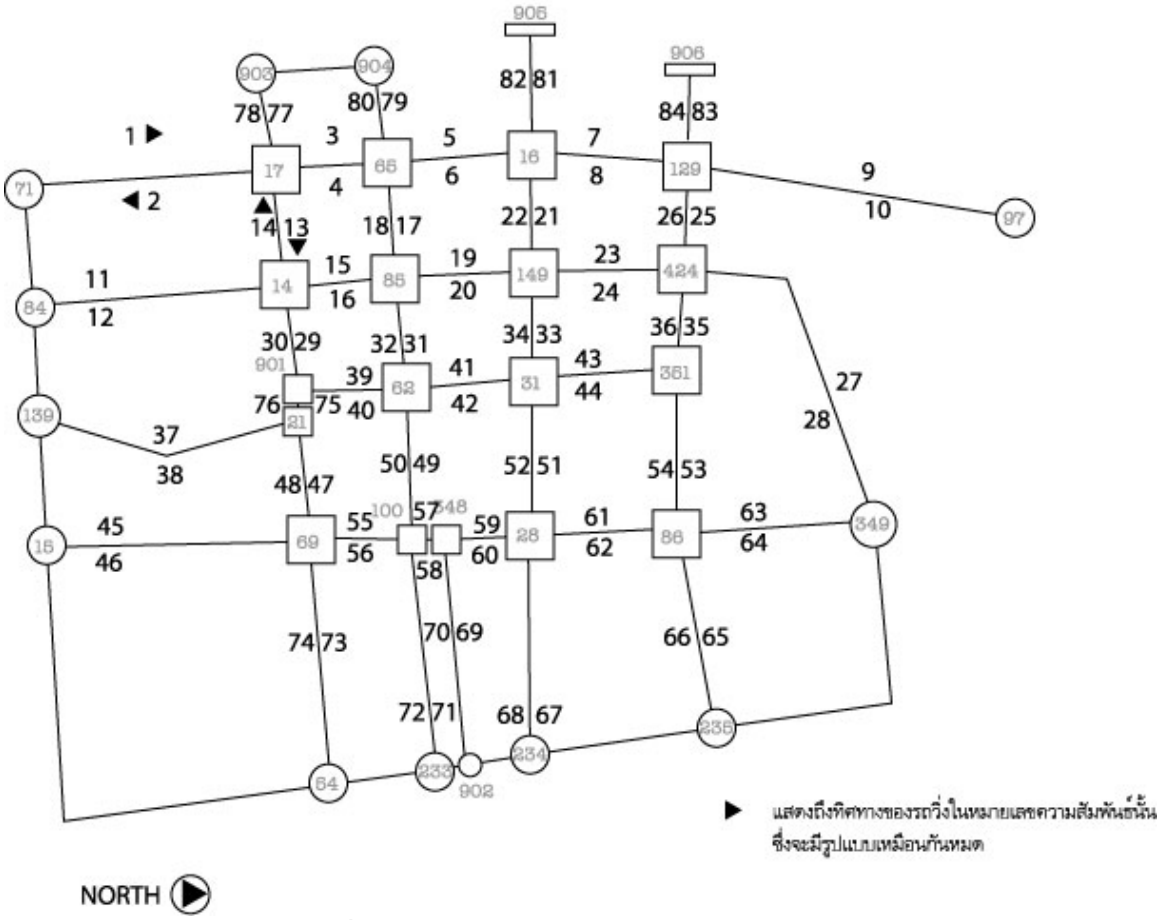
เพื่อให้เกิดความเข้าใจในวิธีการเก็บข้อมูลในตาราง Relation นี้ ลองดูตัวอย่างการเก็บข้อมูลความสัมพันธ์ระหว่างสองทางแยก (ทางแยกหมายเลข 1 และหมายเลข 2) โดยมีถนนเป็นตัวเชื่อมทางแยกทั้งสอง



รูปที่ 4-1: รูปภาพแสดงการเก็บข้อมูลลงในตารางความสัมพันธ์

จากรูปภาพนี้ จะเห็นได้ว่ามีทางแยกอยู่ 2 ทางแยก คือ ทางแยกหมายเลข 1 ซึ่งเป็นหัวถนน (Head_Junc) และทางแยกหมายเลข 2 ซึ่งเป็นท้ายถนน (Tail_Junc) โดยมีความสัมพันธ์หมายเลข 1 (Rel_id1) เชื่อมทางแยกทั้งสองไว้ด้วยกัน ซึ่งความสัมพันธ์นี้เป็นความสัมพันธ์แบบทิศทางเดียว คือ จากทางแยกหมายเลข 1 ไปสู่ทางแยกหมายเลข 2 เท่านั้น ถ้าเป็นทิศทางกลับกันจากทางแยกหมายเลข 2 มายังทางแยกหมายเลข 1 นั้นหมายเลขความสัมพันธ์จะเป็นหมายเลขใหม่ (ห้ามซ้ำกับที่มีอยู่) สำหรับความสัมพันธ์หมายเลข 1 นี้จะมีความสัมพันธ์ถัดไปในทิศเลี้ยวซ้าย ตรงและเลี้ยวขวา คือ rel_id 2, rel_id 3, rel_id 4 ตามลำดับ และจะมีทิศทางของถนนเทียบกับทิศทางในแผนที่เป็นทิศเหนือ

จากการเก็บข้อมูลความสัมพันธ์ในรูปแบบนี้ ทางคณะผู้จัดจึงได้กำหนดหมายเลขความสัมพันธ์ที่เชื่อมระหว่างทางแยกต่าง ๆ ได้ดังที่แสดงในรูปที่ 4-2



รูปที่ 4-2: รูปภาพสรุปหมายเลขความสัมพันธ์

จากรูปจะเห็นได้ว่ามีทั้งหมด 84 ความสัมพันธ์ โดยที่ความสัมพันธ์แต่ละอันจะบอกถึงทิศทางว่ามาจากทางแยกและไปสู่ทางแยกใด โดยมีตำแหน่งของหมายเลขเหล่านั้นเทียบกับเส้น (ถนน) เป็นตัวบอกทิศทาง ซึ่งมีอยู่ 4 ตำแหน่งดังนี้

- ถ้าหมายเลขอยู่ข้างบนเส้น จะมีทิศวิ่งสู่ทิศเหนือ เช่น หมายเลข 1 จะมีทิศทางมาจากทางแยกหมายเลข 71 ไปสู่ทางแยกหมายเลข 17
- ถ้าหมายเลขอยู่ข้างล่างเส้น จะมีทิศวิ่งสู่ทิศใต้ เช่น หมายเลข 2 จะมีทิศทางมาจากทางแยกหมายเลข 17 ไปสู่ทางแยกหมายเลข 71
- ถ้าหมายเลขอยู่ด้านซ้ายของเส้น จะมีทิศวิ่งสู่ทิศตะวันตก เช่น หมายเลข 14 จะมีทิศทางมาจากทางแยกหมายเลข 14 ไปสู่ทางแยกหมายเลข 17
- ถ้าหมายเลขอยู่ด้านขวาของเส้น จะมีทิศวิ่งสู่ทิศตะวันออก เช่น หมายเลข 13 จะมีทิศทางมาจากทางแยกหมายเลข 17 ไปสู่ทางแยกหมายเลข 14

4) ตาราง STATCAR

หน้าที่ของตาราง : รวบรวมข้อมูลสถิติปริมาณการจราจรถนนที่ใช้ในระบบในชั่วโมงต่าง ๆ ซึ่งใช้ในการคำนวณตาราง Tuning ต่อไป มีทั้งหมด 7 เขตข้อมูลดังนี้

ตารางที่ 4.4: Schema ของ Statcar Table

Field Name	**Data Type**	**Description**
year	year(2) not null	เก็บปีที่ข้อมูลทางสถิติชุดนี้ได้ทำการบันทึกเอาไว้
rel_id	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ระหว่างสองทางแยกโดยมีถนนเป็นตัวเชื่อมตรงกลางระหว่างทางแยกทั้งสอง
period	char(11) not null	เก็บช่วงเวลาของแต่ละชั่วโมงตั้งแต่ 7.00 จนถึง 19.00 น. เช่น ช่วง 7.00 – 7.59 น. เป็นต้น
Recorddate	Date	เก็บวัน เดือน ปี ที่ทางสถิติชุดนี้ได้ทำการบันทึกเอาไว้
l_carnum	smallint(6) not null	เก็บปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วเลี้ยวซ้ายเมื่อถึงทางแยกที่เป็นท้ายถนน(Tail_Junc)
s_carnum	smallint(6) not null	เก็บปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วตรงต่อไปเมื่อถึงทางแยกที่เป็นท้ายถนน(Tail_Junc)
r_carnum	smallint(6) not null	เก็บปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วเลี้ยวขวาเมื่อถึงทางแยกที่เป็นท้ายถนน(Tail_Junc)

5) ตาราง Tuning

หน้าที่ของตาราง : จัดเก็บสัดส่วนของรถที่จะเลี้ยวซ้าย ตรงและเลี้ยวขวา ซึ่งใช้ในการสุ่มค่าตามทฤษฎีปัวซองค์เพื่อหาปริมาณรถที่จะวิ่งผ่านแต่ละทิศทางในแต่ละถนน ซึ่งมี 5 เขตข้อมูลดังนี้

ตารางที่ 4.5: Schema ของ Tuning Table

Field Name	**Data Type**	**Description**
rel_id	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ระหว่างสองทางแยกโดยมีถนนเป็นตัวเชื่อมตรงกลางระหว่างทางแยกทั้งสอง
period	char(11) not null	เก็บช่วงเวลาของแต่ละชั่วโมงตั้งแต่ 7.00 จนถึง 19.00 น. เช่น ช่วง 7.00 – 7.59 น. เป็นต้น
l_percent	float not null	เก็บอัตราส่วนของปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วเลี้ยวซ้ายเมื่อถึงทางแยกที่เป็นท้ายถนนเทียบกับปริมาณการจราจรในความสัมพันธ์นี้ทั้งหมด

s_percent	float not null	เก็บอัตราส่วนของปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วตรงต่อไปเมื่อถึงทางแยกที่เป็นท้ายถนนเทียบกับปริมาณการจราจรในความสัมพันธ์นี้ทั้งหมด
r_percent	float not null	เก็บอัตราส่วนของปริมาณการจราจรที่ผ่านความสัมพันธ์นี้แล้วเลี้ยวขวาเมื่อถึงทางแยกที่เป็นท้ายถนนเทียบกับปริมาณการจราจรในความสัมพันธ์นี้ทั้งหมด

6) ตาราง Projdata

หน้าที่ของตาราง : เก็บชุดข้อมูลที่ใช้ในการทำงานของระบบทุก ๆ รอบตั้งแต่ตอนเริ่มระบบจนถึงสถานะล่าสุด ซึ่งมีทั้งหมด 11 เขตข้อมูลดังนี้

ตารางที่ 4.6: Schema ของ Projdata Table

Field Name	**Data Type**	**Description**
Turn	bigint (20) not null	เก็บหมายเลขของรอบการทำงานนับตั้งแต่ตอนสถานะเริ่มต้นจนถึงสถานะล่าสุด
rel_id	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ระหว่างสองทางแยกโดยมีถนนเป็นตัวเชื่อมตรงกลางระหว่างทางแยกทั้งสอง
car_out	smallint(6) not null	เก็บจำนวนรถที่ออกจากถนนนั้น
car_remain	smallint(6) not null	เก็บจำนวนรถที่ยังเหลือค้างในถนน
light_order	tinyint(4) not null	เก็บลำดับการเปิดสัญญาณไฟของถนนว่าให้เปิดสัญญาณไฟให้กับถนนใดก่อน
light_type	enum('single', 'pair') not null	เก็บชนิดของการเปิดสัญญาณไฟของถนน ซึ่งมีอยู่ 2 แบบ คือ แบบ Single และแบบ Pairโดยที่การเปิดสัญญาณไฟแบบ Single นั้นจะเป็นการเปิดไฟเขียวให้กับถนนใด ๆ ได้เพียงถนนเดียวในช่วงเวลาหนึ่งแต่ถ้าเป็นการเปิดสัญญาณไฟแบบ Pair การเปิดไฟเขียวจะเปิดให้กับถนนใด ๆ กับถนนในทิศทางตรงข้ามได้พร้อมกัน
light_state	enum('green', 'red') not null	บอกสถานะของสัญญาณไฟตอนนี้ว่าเป็นไฟเขียวหรือไฟแดง
light_length	smallint(6) not null	บอกระยะเวลาการเปิดสัญญาณไฟเขียวว่าให้เปิดนานเท่าใด
light_remain	smallint(6) not null	เก็บระยะเวลาที่เหลืออยู่ก่อนที่จะเปลี่ยนสัญญาณไฟจากเขียวเป็นแดง
tf_flow	Float	บอกอัตราการเคลื่อนตัวของรถในถนนนี้
tf_ratio	smallint(6))	บอกอัตราการเปลี่ยนแปลงของปริมาณรถในถนนนี้

7) ตาราง Carincome

หน้าที่ของตาราง : เก็บปริมาณของรถที่วิ่งเข้าสู่ถนนใด ๆ มีเขตข้อมูลอยู่ 4 เขตดังนี้

ตารางที่ 4.7: Schema ของ Carincome Table

Field Name	Data Type	Description
Turn	bigint (20) not null	เก็บหมายเลขของรอบการทำงานนับตั้งแต่ตอนสถานะเริ่มต้นจนถึงสถานะล่าสุด
rel_id	smallint(6) not null	เก็บหมายเลขความสัมพันธ์ระหว่างสองทางแยกโดยมีถนนเป็นตัวเชื่อมตรงกลางระหว่างทางแยกทั้งสอง
distance	int(11)	เก็บระยะทางของรถกลุ่มนี้ว่าวิ่งเข้ามาในถนนนี้ไปแล้วเท่าใด
carvolume	int(11)	เก็บปริมาณของรถที่วิ่งเข้าสู่ถนนนี้ในรอบการทำงานปัจจุบัน

8) ตาราง Historyaction

หน้าที่ของตาราง : เก็บชนิดของคำสั่งควบคุมระบบในแต่ละรอบการทำงาน มีอยู่ 2 เขตข้อมูลดังนี้

ตารางที่ 4.8: Schema ของ Historyaction Table

Field Name	Data Type	Description
Turn	bigint(11)	เก็บหมายเลขของรอบการทำงานนับตั้งแต่ตอนสถานะเริ่มต้นจนถึงสถานะล่าสุด
action	enum('auto', 'play', 'stop')	เก็บชนิดของคำสั่งควบคุมการจำลองสถานการณ์จราจร ซึ่งมีอยู่ 3 แบบ ได้แก่ เริ่มการจำลอง (Play), หยุดการจำลอง (Stop) , เริ่มการจำลองอัตโนมัติ (auto)

9) ตาราง User

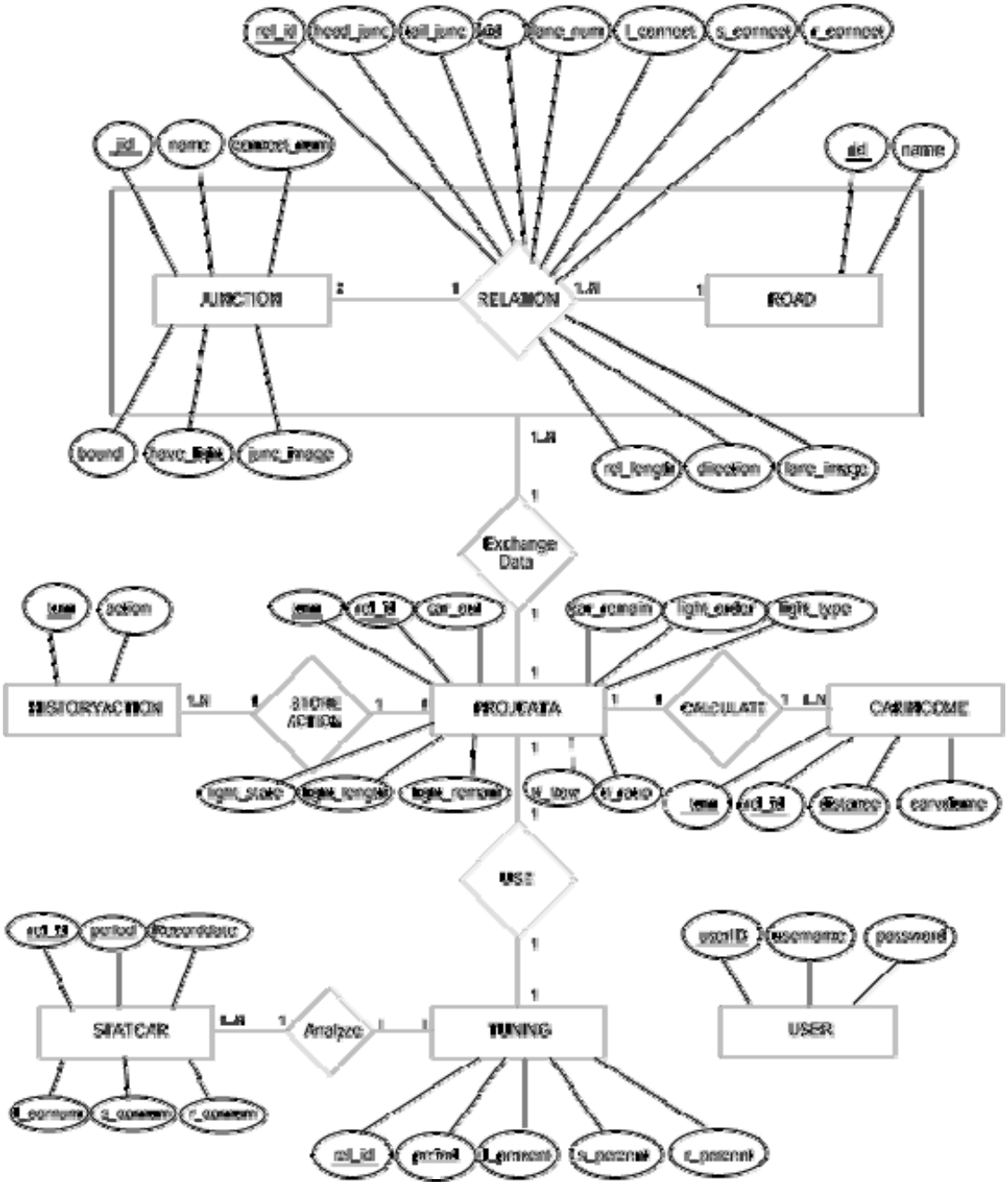
หน้าที่ของตาราง : เก็บรายชื่อผู้มีสิทธิแก้ไขข้อมูลในฐานข้อมูล ซึ่งมีอยู่ 3 เขตข้อมูลดังนี้

ตารางที่ 4.9: Schema ของ User Table

Field Name	Data Type	Description
userID	smallint(6)	หมายเลขของผู้มีสิทธิแก้ไขฐานข้อมูล
username	tinytext	ชื่อของผู้มีสิทธิแก้ไขฐานข้อมูล
password	tinytext	รหัสผ่านผู้มีสิทธิแก้ไขฐานข้อมูล

4.1.2 รูปภาพสรุปความสัมพันธ์ในฐานข้อมูล

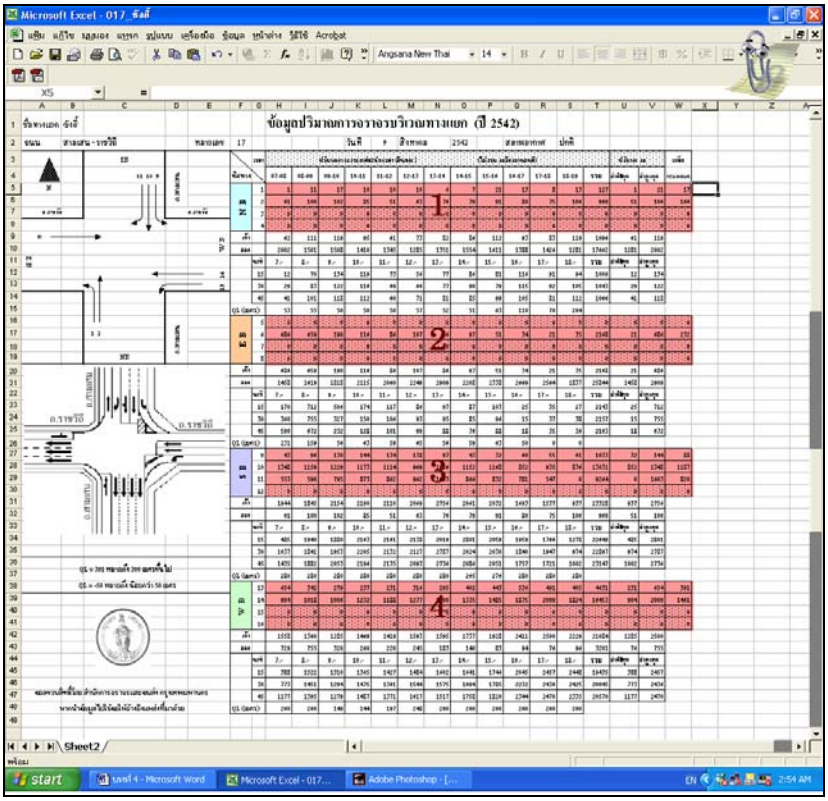
รายละเอียดของ Entity และความสัมพันธ์ระหว่างแต่ละ Entity ที่ถูกสร้างขึ้นในระบบจำลอง สามารถที่จะสรุปได้ดังที่ปรากฏอยู่ในรูปภาพสรุปความสัมพันธ์ในฐานข้อมูลทางด้านล่างนี้



รูปที่ 4-3: รูปภาพแสดงความสัมพันธ์ในฐานข้อมูล

4.1.3 วิธีการแปลงรูปแบบข้อมูลที่ได้จากสำนักการจราจรและขนส่งให้อยู่ในรูปแบบของฐานข้อมูลที่ใช้บนระบบ

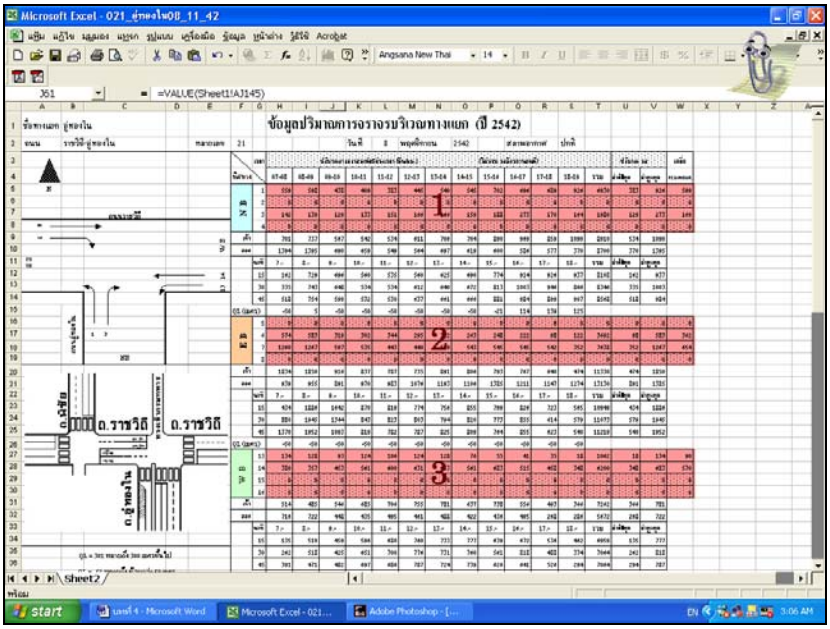
รูปแบบข้อมูลของทางสำนักการจราจรและขนส่งที่นำมาใช้ในระบบนั้นยังไม่สามารถนำมาประยุกต์ใช้ได้โดยตรงเพราะรูปแบบการเก็บข้อมูลในชุดหนึ่ง ๆ ของทางสำนักการจราจรและขนส่งนั้นอยู่ในลักษณะของปริมาณการจราจรบริเวณทางแยกในชั่วโมงต่าง ๆ นั่นหมายความว่า ข้อมูล 1 ชุด ประกอบไปด้วยข้อมูลชุดย่อยที่บอกถึงปริมาณรถในแต่ละถนนที่วิ่งเข้าสู่ทางแยกนั้น เช่น ในกรณีที่เป็นสี่แยกจะมีข้อมูลปริมาณการจราจรจำนวน 4 ชุดย่อย ดังที่แสดงในรูปที่ 4-4



รูปที่ 4-4: รูปภาพแสดงรูปแบบข้อมูลของทางสำนักการจราจรและขนส่งกรณีที่เป็น 4 แยก

จากรูปภาพตัวอย่าง แถบสีแดง คือ ข้อมูลปริมาณการจราจรที่วิ่งเข้าสู่ทางแยกในชั่วโมงต่าง ๆ ที่สามารถนำมาใช้ได้กับระบบ ส่วนข้อมูลที่เหลือนั้นไม่จำเป็นต้องใช้ในการจำลองสถานการณ์การจราจรในระบบ

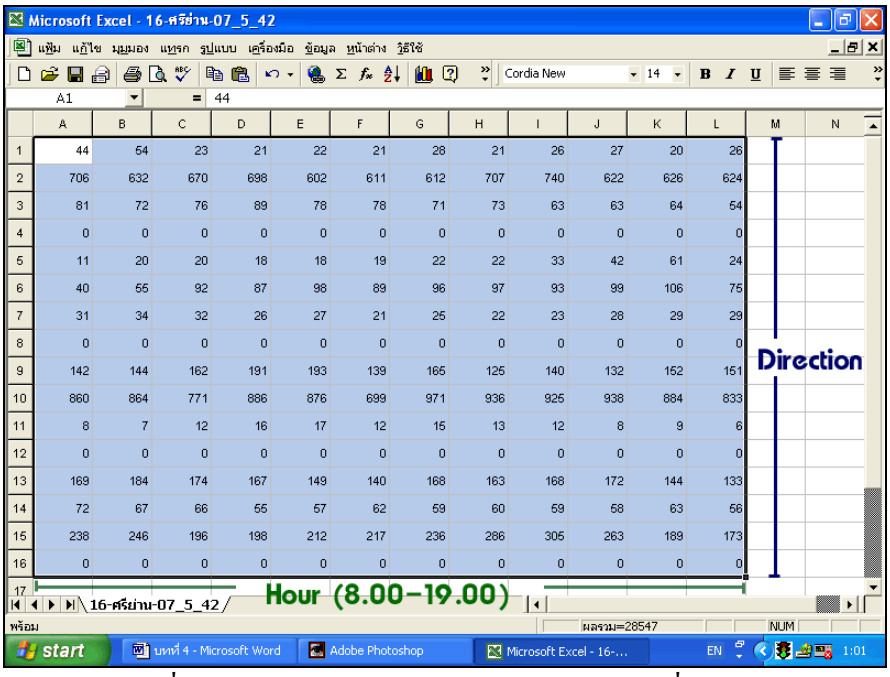
ส่วนในกรณีที่เป็นสามแยกจะมีข้อมูลปริมาณการจราจรจำนวน 3 ชุดย่อย ดังที่แสดงในรูปที่ 4-5



รูปที่ 4-5: รูปภาพแสดงรูปแบบข้อมูลของทางสำนักการจราจรและขนส่งกรณีทีเป็น 3 แยก

แต่เนื่องจากรูปแบบข้อมูลในฐานข้อมูลนั้นเก็บอยู่ในรูปแบบของถนนแต่ละถนนว่ามีปริมาณการจราจรเป็นอย่างไรในแต่ละชั่วโมง (ไม่ใช่ปริมาณการจราจรของทั้งทางแยกเหมือนกับรูปแบบข้อมูลของทางสำนักการจราจรและขนส่ง) ทำให้จำเป็นต้องมีการแปลงรูปแบบข้อมูลของทางสำนักการจราจรและขนส่งให้อยู่ในรูปแบบที่ฐานข้อมูลกำหนดเอาไว้เพื่อที่จะนำข้อมูลเหล่านั้นที่มีอยู่เป็นจำนวนมากมาใช้ร่วมกับระบบที่สร้างขึ้น

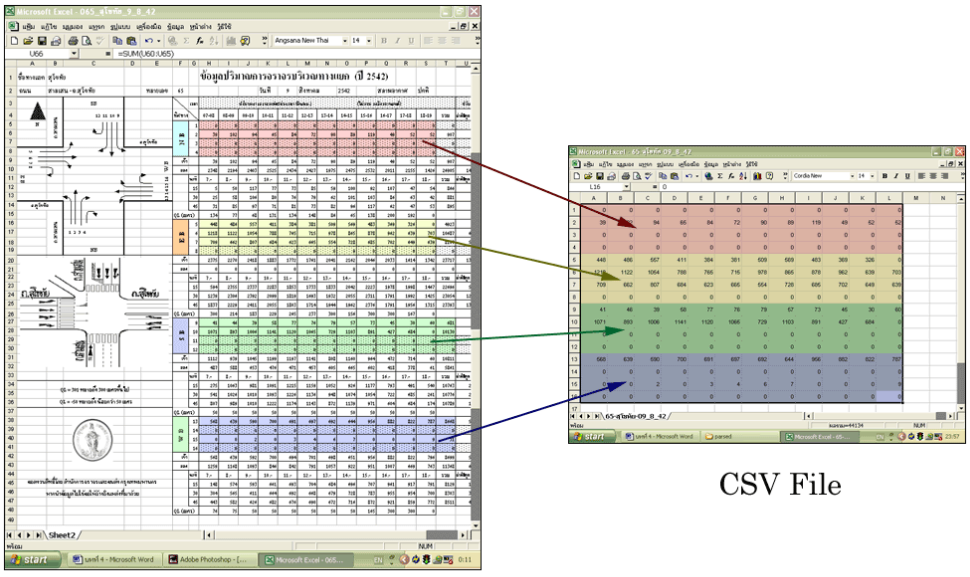
การแปลงนี้สามารถทำได้ด้วยการใช้ คำสั่ง Import CSV File ในส่วนของ Database Management ซึ่งจะทำการแปลงข้อมูลที่ได้เลือกไว้ให้มาอยู่ในรูปแบบเดียวกันที่ใช้บนฐานข้อมูลและจะทำการบันทึกข้อมูลเพิ่มเติมลงไปให้อัตโนมัติอีกด้วย ซึ่งรูปแบบของ CSV ที่สามารถแปลงได้ จะมีรูปแบบเป็นตารางข้อมูลปริมาณการจราจรเฉพาะส่วนที่เป็นปริมาณรถที่วิ่งเข้าสู่แยกจากแต่ละทิศทางในชั่วโมงต่าง ๆ เท่านั้น (เฉพาะแถบสีแดง) โดยมีขนาดกว้าง 11 ช่อง ยาว 16 ช่อง โดยหน่วยของข้อมูลในแนวนอน คือ ชั่วโมงต่าง ๆ ตั้งแต่ 8.00 – 19.00 น. (8.00 อยู่ช่องซ้ายสุด) ส่วนหน่วยข้อมูลในแนวตั้ง คือ ทิศทางต่าง ๆ ในทางแยกนั้น ตัวอย่างของข้อมูล CSV File ที่ถูกต้องแสดงดังรูปที่ 4-6



รูปที่ 4-6: รูปภาพแสดงรูปแบบข้อมูล CSV File ที่ถูกต้อง

ขั้นตอนในการแปลงทั้งหมด พอที่จะสรุปได้เป็นข้อ ๆ ดังนี้

1. การแปลงจะเริ่มจากการนำข้อมูลที่อยู่ในรูปแบบของสำนักการจราจรและขนส่ง สมมติวันที่เก็บสถิติ คือ 16-01-42 (เป็น Excel File) มาบันทึกให้อยู่ในรูปแบบที่เป็น CSV File ขนาดกว้าง 11 ช่อง ยาว 16 ช่อง

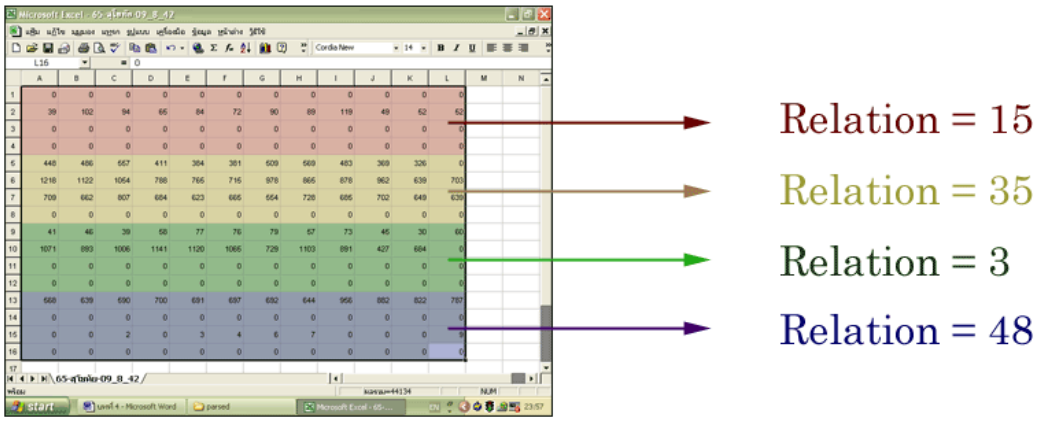


CSV File

Excel File

รูปที่ 4-7: รูปภาพการแปลงข้อมูลให้อยู่ในรูปของ CSV File

2. แยกข้อมูลชุดย่อยเหล่านั้น (แถบสีต่าง ๆ) ออกจากกันแล้วค่อยทำการตรวจสอบว่าข้อมูลชุดย่อยแต่ละชุดนั้นตรงกับความสัมพันธ์ใดใน Database



รูปที่ 4-8: รูปภาพการแยกข้อมูลใน CSV File ตามความสัมพันธ์

3. เอาข้อมูลย่อยนั้นมาตรวจสอบปริมาณการจราจรในแต่ละทิศทางทุก ๆ ชั่วโมง ในรูปที่ 4-9 แสดงให้เห็นว่าปริมาณการจราจรมีเพียงแค่ตรงไปเท่านั้น เพราะทิศทางอื่นเป็น 0 ในทุก ๆ ชั่วโมง

รูปที่ 4-9: รูปภาพการแยกข้อมูลในแต่ละทิศทางของความสัมพันธ์ใด ๆ

4. สุดท้ายก็จะได้ข้อมูลที่สามารถบันทึกลงในตาราง Statcar ในฐานข้อมูล

ตารางที่ 4.10: ตัวอย่างข้อมูลที่บันทึกลง Database ในตาราง Statcar

Year	rel_id	period	recorddate	l_carnum	s_carnum	r_carnum
42	15	8.00-9.00	16-01-42	0	39	0
42	15	9.00-10.00	16-01-42	0	102	0
42	15	10.00-11.00	16-01-42	0	94	0
…	…	…	…	…	…	…

บทที่ 5

การพัฒนาระบบควบคุมและติดตามการจราจรอัตโนมัติ

หลังจากที่ได้ทำการออกแบบระบบไปแล้ว ขั้นต่อไปก็คือการพัฒนาระบบขึ้นมา ซึ่งการพัฒนานี้จำเป็นต้องมีการพิจารณาหาเครื่องมือมาใช้ในช่วงการพัฒนาเพื่อให้ระบบที่สร้างขึ้นมีประสิทธิภาพและที่สำคัญคือ เครื่องมือเหล่านี้จะต้องสามารถนำไปประยุกต์ใช้กับเทคนิคที่ได้ออกแบบเอาไว้ในการออกแบบได้

ส่วนแรกของบทนี้ก็จะกล่าวถึงเทคโนโลยีที่ใช้ในการสร้างระบบเพื่อทำให้ทราบถึงรายละเอียดของเครื่องมือที่นำมาใช้ ในส่วนถัดไปจะกล่าวถึงการประยุกต์หลักการจำลองมาใช้ในการสร้างสถานการณ์จราจรจำลองบนระบบซึ่งจะอธิบายถึงการนำเอาทฤษฎีทางการจำลองมาประยุกต์เป็นเทคนิคมาใช้ในการสร้างระบบ

5.1 เทคโนโลยีสำคัญที่ใช้ในการพัฒนาระบบนั้นมีดังนี้

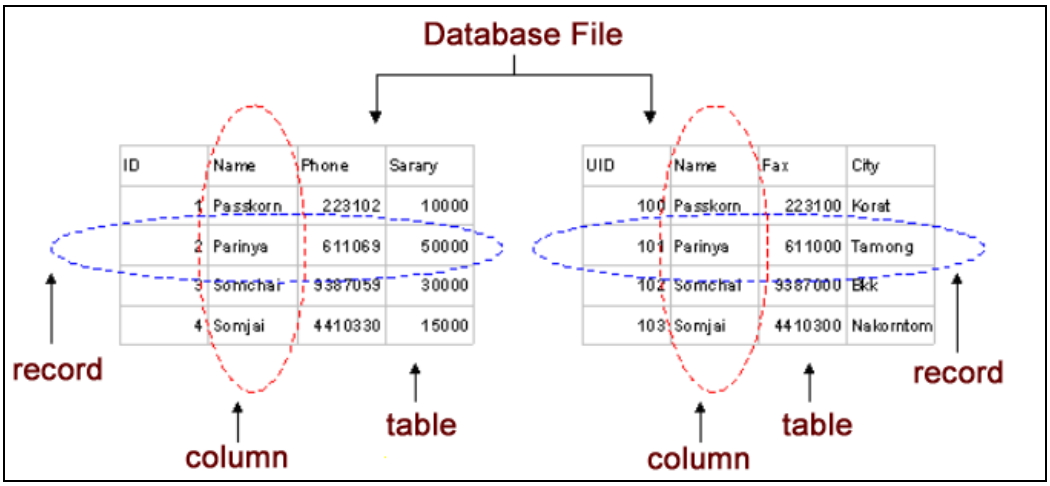
- **โปรแกรม MySQL** ใช้ในการจัดการฐานข้อมูลที่ใช้ในระบบ
- **ภาษา JAVA** ใช้ในการสร้างโปรแกรมส่วนใหญ่ในระบบ
- **ภาษา XML** ใช้ในส่วนของการอ่านและบันทึกข้อมูลในเครื่องที่ระบบติดตั้งอยู่

โดยรายละเอียดของเทคโนโลยีในแต่ละหัวข้อนั้นจะอธิบายได้ดังต่อไปนี้

โปรแกรม MySQL

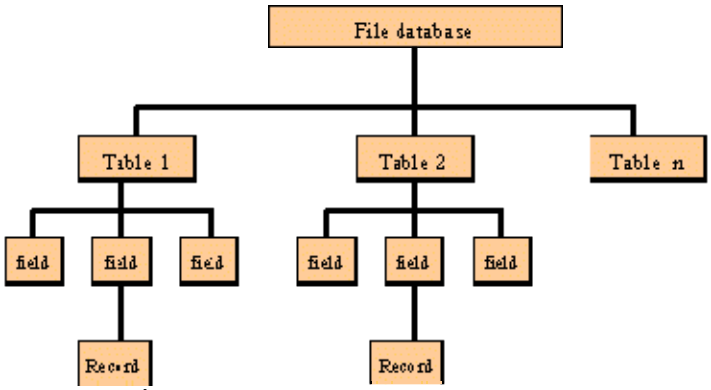
ทางคณะผู้จัดทำได้เลือกโปรแกรม MySQL มาใช้ในการพัฒนาฐานข้อมูลบนระบบ เนื่องจาก MySQL เป็น Freeware ทำให้ไม่มีปัญหาในด้านลิขสิทธ์ที่อาจเป็นปัญหาต่อการพัฒนาระบบในอนาคต และประสิทธิภาพของทางตัว MySQL เองก็ไม่ได้ด้อยไปกว่าโปรแกรมดูแลฐานข้อมูลอื่น ๆ เลย

การจัดเก็บข้อมูลที่มีอยู่เป็นจำนวนมากในระบบจำเป็นจะต้องมีรูปแบบในการจัดเก็บที่เหมาะสม ซึ่งรูปแบบการจัดเก็บใน MySQL สามารถแบ่งได้ตามภาพข้างล่างดังนี้



รูปที่ 5-1: รูปแบบการจัดเก็บใน MySQL

ข้อมูลทั้งหมดที่จัดเก็บเข้าเป็นพวกเดียวกันเราเรียกว่า Database file ภายใน Database file ก็จะประกอบไปด้วย Table ต่างๆ ที่จัดเก็บข้อมูล ภายใน Table ก็จะประกอบไปด้วย field ซึ่งจะบ่งบอกลักษณะของข้อมูล (Type) ส่วนของ Record จะเป็นส่วนของข้อมูลที่ถูกจัดเข้าด้วยกันตามลักษณะการแบ่งพวกของ field โครงสร้าง MySQL database สามารถสรุปมาแสดงเป็นรูปได้ดังนี้



รูปที่ 5-2: โครงสร้างของ Database ของ MySQL

จะเห็นได้ว่าโครงสร้างของ Database ของ Mysql ก็จะไม่ต่างจากโครงสร้าง database ของโปรแกรมdatabase ตระกูลอื่นโดยเริ่มจาก สร้าง file database ขึ้นมาก่อนแล้วจึง สร้าง table โดย design ตามที่เราต้องการ จากนั้นก็จะได้ field ต่างๆ เมื่อแต่ละ field มีข้อมูลก็จะถูกรวมเป็น record ต่อไป

ลักษณะของ MySQL นั้นทำงานในลักษณะ Client Server ทำงานบนระบบ Telnet บน Unix System และบน Win32 ทั่วไปบนระบบเครือข่าย Internet และ Intranet นั้นหมายความว่า เราสามารถเรียกใช้ MySQL ได้ทั่วโลกกรณีเป็น Internet และ ทั่วบริเวณที่เป็น Intranet และยังสามารถเรียกใช้บน Web Browser ได้กรณีใช้ language เป็น Interface ในการเชื่อม language ที่ใช้เป็น Interface อาทิเช่น ภาษา PHP ภาษา C เป็นต้น MySQL นั้นยังเป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) RDBMS คือ สามารถทำงานกับตารางข้อมูลหลายตารางพร้อมๆ กัน โดยสามารถแสดงความสัมพันธ์ของตารางเหล่านั้นด้วย field ที่ใช้ร่วมกัน

MySQL ใน Version ที่ใช้กับระบบนั้นยังมีข้อจำกัดอยู่บางประการที่ทำให้ไม่สะดวกในการทำงาน เช่น ไม่สนับสนุน Query บางรูปแบบ คือ ไม่สามารถใช้ SubQuery หรือ NestQuery ได้ รวมทั้งไม่มีการเช็คข้อมูลที่เป็น Foreign Key ซึ่งเป็นส่วนที่ทำให้ข้อมูลมีความสอดคล้องและถูกต้องในข้อมูลที่มีความสัมพันธ์กันอยู่ สิ่งเหล่านี้ทำให้ผู้ดูแลฐานข้อมูลเป็นคนที่ต้องคอยเช็คและตรวจสอบความถูกต้องดังกล่าวเอง ซึ่งเมื่อ MySQL ออก Version ใหม่ออกมาก็คงจะช่วยในจุดนี้ได้พอสมควร

หลังจากที่เราได้ติดตั้ง MySQL บนระบบแล้ว (วิธีการติดตั้งแทรกอยู่ในภาคผนวก) ก็จะพูดถึงตัวอย่างการใช้คำสั่งที่จำเป็นในการจัดการฐานข้อมูลโดยเริ่มจากการสร้างฐานข้อมูลกันก่อน

รูปแบบคำสั่ง

CREATE DATABASE databasename

คำอธิบาย

CREATE DATABASE : เป็นคำสั่งที่ใช้สร้าง file database

databasename :เป็นชื่อ file database ที่เรากำหนดขึ้น

ตัวอย่าง

mysql>CREATE DATABASE datatest;

จากคำสั่งเราต้องการสร้าง file database ที่ชื่อว่า test

จากคำสั่งข้างต้นโปรดสังเกตเมื่อสิ้นสุดคำสั่งจะต้องมีเครื่องหมาย ; ทุกครั้ง ถ้าไม่มีเครื่องหมาย ; โปรแกรมจะตีความหมายว่ายังไม่รับคำสั่งจนกว่าจะพบเครื่องหมาย ; เมื่อสร้าง file database ได้แล้วก็ต้องรู้วิธีในการลบ file database ด้วยโดยกระทำได้ดังนี้

รูปแบบคำสั่ง

DROP DATABASE datatest

คำอธิบาย

DROP DATABASE : เป็นคำสั่งที่ลบ database

datatest : เป็นชื่อ file database ที่ต้องการลบ

ตัวอย่าง

```
mysql>DROP DATABASE datatest;
ให้ลบ database ที่ชื่อ datatest
```

ผลลัพธ์ที่ได้ดังภาพข้างล่าง

```
mysql> DROP DATABASE datatest;
Query OK, 0 rows affected (0.00 sec)
```

MySQL จะรายงานการออกมาว่าสามารถลบได้เรียบร้อยแล้ว แต่ถ้าลบไม่ได้จะมี syntax error ขึ้นมาดังเช่น ภาพ error ในบทการสร้าง file database สำหรับใน syntax error นั้น ถ้าสังเกตให้ดี มันจะมีประโยชน์กับเรามาก เพราะว่ามันจะรายงานในสิ่งที่ผิดพลาดออกมาให้เราดู เช่น ใช้คำสั่งผิด หรือไม่พบ file ที่ต้องการลบ ดังเช่นตัวอย่างข้างล่างผู้เขียนลบ file database ไปแล้วแต่ยังคงใช้คำสั่งลบ file database ซ้ำก็จะได้ผลลัพธ์ดังนี้

```
mysql> DROP DATABASE datatest;
ERROR 1008: Can't drop database 'datatest'. Database doesn't exist
mysql>
```

สำหรับการลบ file database นั้น MySQL จะยอมให้ผู้ที่เป็นสร้าง file database เป็นผู้ลบเท่านั้น เช่นถ้า user nick เป็นคนสร้าง user jib จะไปลบไม่ได้ เป็นเรื่องของสิทธิการใช้งาน

ต่อมาหากต้องการจะหา File ทั้งหมดใน database จะมีรูปแบบการใช้งานดังนี้

รูปแบบคำสั่ง

SHOW DATABASES

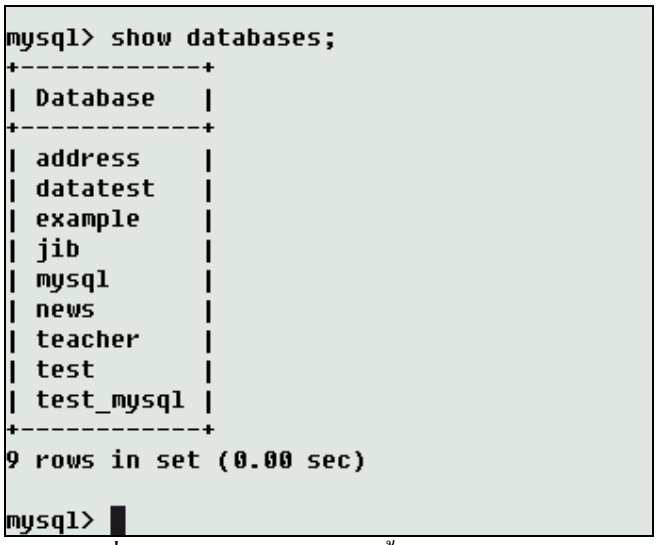
คำอธิบาย

SHOW DATABASES : เป็นคำสั่งที่ใช้แสดง database ทั้งหมดที่มี

ตัวอย่าง

```
mysql>SHOW DATABASES;
ให้แสดง database ทั้งหมดที่มีใน Program MySQL นี้
```

ด้วยคำสั่งดังกล่าวจำทำให้ทราบว่าในขณะนี้มี file database อะไรบ้าง และรวมแล้วมีกี่ file บนเครื่องดังที่ได้แสดงดังภาพทางด้านล่าง



รูปที่ 5-3: ผลจากการหา File ทั้งหมดใน database

เมื่อมีฐานข้อมูลแล้วก็ถึงช่วงการสร้าง Table โดยที่ Table ต้องมี Option เป็นส่วนประกอบของโครงสร้าง ซึ่งมีดังต่อไปนี้

Table Option	Description
AUTO_INCREMENT	The next auto increment value you want to set for your table (MyISAM)
AVG_ROW_LENGTH	An approximation of the average row length for your table. You only need to set this for tables with variable size records.
CHECKSUM	Set this to 1 if you want MySQL to maintain a checksum for all rows (makes the table a little slower to update but makes it easier to corrupted tables) (MyISAM)
COMMENT	A 60 character comment for your table
MAX_ROWS	Max number of rows you plan to store in the table
MIN_ROWS	Minimum number of rows you plan to store in the table
PACK_KEYS	Set this to 1 if you want to have smaller index. This usually makes updates slower and reads faster (MyISAM, ISAM).
PASSWORD	Encrypt the .frm le with a password. This option doesn't do anything in the standard MySQL version.
DELAY_KEY_WRITE	Set this to 1 if want to delay key table updates until the table is closed (MyISAM).
ROW_FORMAT	De nes how the rows should be stored (for the future).

PRIMARY KEY	Difference Record must not equals
NOT NULL	Must have any thing in record not null

การสร้าง Table มีรูปแบบคำสั่งดังนี้

รูปแบบคำสั่งสร้าง table

CREATE TABLE data1 (field1, filed2, filed3, filedn)

หรือ

create_definition:

col_name type [NOT NULL | NULL] [DEFAULT default_value]

[AUTO_INCREMENT][PRIMARY KEY]

[reference_definition]

or PRIMARY KEY (index_col_name,...)

or KEY [index_name] (index_col_name,...)

or INDEX [index_name] (index_col_name,...)

or UNIQUE [INDEX] [index_name] (index_col_name,...)

คำอธิบาย

CREATE TABLE : เป็นคำสั่งสร้าง table

data1 : เป็นชื่อ table ที่ต้องการสร้าง

field1 : เป็นชนิดของ column

ตัวอย่าง

```
mysql>CREATE TABLE  phonebook(
-> name VARCHAR(25),
-> email VARCHAR(30),
-> phone INT,
-> ID INT NOT NULL AUTO_INCREMENT,
-> PRIMARY KEY(ID));
```

หากมีการกดปุ่มลูกศรชี้ขึ้นโปรแกรมจะทวนคำสั่งจากคำสั่งสุดท้ายไปหาคำสั่งแรกตามลำดับซึ่งจะสร้างความสะดวกให้คุณ โดยไม่ต้องพิมพ์คำสั่งซ้ำๆอีก เมื่อสร้าง table ได้ก็ต้องลบได้เช่น แต่ก่อนลบจะต้องมั่นใจก่อนว่า จะลบ table จริงๆ มิฉะนั้นแล้วข้อมูล ก็จะหายไปในทันที ทางที่ดีควร backup ไว้ก่อน รูปแบบคำสั่งเป็นดังนี้

รูปแบบคำสั่งการลบ table ใน file database

DROP TABLE tablename

คำอธิบาย

DROP TABLE : เป็นคำสั่งลบ table

tablename : เป็นชื่อ table ที่ต้องการลบ

ตัวอย่าง

DROP TABLE phonebook;

หลังจากลบ table phonebook แล้ว โปรแกรม จะรายงานว่า Query OK คือได้ทำการลบเสร็จแล้ว และเมื่อใช้คำสั่ง SHOW TABLE ก็ไม่พบ table ใดๆ (กรณีที่มี table เดียว) ในส่วนของการเพิ่มข้อมูลเข้าสู่ Table มีวิธีการอย่างคร่าวๆ ดังนี้

รูปแบบคำสั่งการเพิ่มข้อมูลลงใน table

INSERT INTO tablename (field1,field2,fieldn) VALUES ('field1', 'field2', 'field2');

คำอธิบาย

INSERT INTO : เป็นคำสั่งเพิ่มข้อมูลเข้าสู่ table

tablename : เป็นชื่อ table ที่ต้องการเพิ่มข้อมูล

(field1,field2,fieldn) : เป็นชื่อ field ที่ต้องการเพิ่มข้อมูล

VALUES : กลุ่มของข้อมูลที่ต้องการเพิ่ม

('field1', 'field2', 'fieldn') : เป็นข้อมูลของแต่ละ field ที่ต้องการเพิ่ม

field 1 : เป็น field ที่ 1 ของข้อมูลที่ต้องการเพิ่ม (จากที่ได้ออกแบบ Table ไว้)

field 2 : เป็น field ที่ 2 ของข้อมูลที่ต้องการเพิ่ม

field n : เป็น field ที่ n ของข้อมูลที่ต้องการเพิ่ม

ตัวอย่าง

mysql>INTER INTO phonebook (name,email,phone) VALUES

mysql>('ภาสกร เรืองรอง',

mysql>'ccpasskn@hotmail.com'

mysql>223102, NULL);

จะเห็นว่าในคำสั่ง INSERT INTO หรือคำสั่งอื่นๆที่ยาวมาก ถ้าเรา Key ข้อมูลโดยไม่ Enter ขึ้นบรรทัดใหม่เลย จะทำให้บรรทัดที่ Key นั้นยาวมากจนล้นและทำให้ดูยาก MySQL เปิดโอกาสให้ Key ข้อมูลไปก่อน แต่ MySQL จะยังไม่รับคำสั่งจนกว่าจะเครื่องหมาย ; ดังนั้นคุณสามารถ Key คำสั่งที่ต้องการแล้วกด Enter ขึ้นบรรทัดใหม่ไปได้เรื่อยๆ จนครบคำสั่งแล้วจึงใส่เครื่องหมาย ; ปิดท้ายจากนั้น Enter อีกครั้ง MySQL ก็จะรับคำสั่งไปทำการ Proces ถ้าไม่สามารถทำได้ก็จะรายงาน

บรรทัดที่ผิดพลาดมาให้ เช่น

ERROR 1064: You have an error in your SQL syntax near '223102,NULL)' at line 4

เมื่อเราเพิ่มข้อมูลเข้าไปเรื่อยๆแล้วถ้าข้อมูลไม่มากนัก เราสามารถเรียกดูได้ไม่ยากโดยการเรียกดูทั้งหมดและใช้ตาเลือกดูข้อมูลที่ต้องการแต่ถ้าข้อมูลเพิ่มขึ้นเรื่อยๆ การสืบค้นข้อมูลโดยวิธีดังกล่าวไม่สามารถทำได้ MySQL จึงมีระบบการสืบค้นข้อมูลมาเพื่อให้สามารถสืบค้นโดยการสร้างเงื่อนไขตรงความต้องการ โดยมีวิธีการอย่างคร่าวๆ ดังนี้

- กรณีที่สืบค้นข้อมูลโดยไม่มีเงื่อนไข มีรูปแบบคำสั่งดังต่อไปนี้

รูปแบบคำสั่งสืบค้นข้อมูลโดยขอดูข้อมูลทั้งหมด table

SELECT * FROM tablename;

รูปแบบคำสั่งสืบค้นข้อมูลโดยขอดูข้อมูลบางส่วน(บาง field)ทั้งหมด table

SELECT fieldname FROM tablename;

คำอธิบาย

SELECT : คำสั่งเลือกข้อมูล

fieldname : field ที่ต้องการ ถ้าต้องการทุก field ให้ใช้เครื่องหมาย "*"

FROM : จาก table อะไร

tablename : table ที่ต้องการสืบค้น

ตัวอย่าง

```
mysql>SELECT * FROM phonebook;
mysql>SELECT name,email FROM phonebook;
```

- กรณีที่สืบค้นข้อมูลโดยมีเงื่อนไข (Operator)

การสืบค้นข้อมูลโดยมีเงื่อนไขจำเป็นต้องใช้ Operator เข้าช่วย การใช้ Operator สามารถใช้ได้ในโอกาสและสถานการณ์ที่แตกต่างกัน สามารถแยกการใช้ได้ดังนี้

1. การสืบค้นโดยใช้ Operator WHERE
2. การสืบค้นโดยใช้ Operator LIKE
3. การสืบค้นโดยใช้ Operator AND
4. การสืบค้นโดยใช้ Operator OR
5. การสืบค้นโดยใช้ Operator >
6. การสืบค้นโดยใช้ Operator >=
7. การสืบค้นโดยใช้ Operator <
8. การสืบค้นโดยใช้ Operator <=
9. การสืบค้นโดยใช้ Operator <>

ซึ่งสามารถยกตัวอย่างการใช้งานบางประเภท ซึ่งรูปแบบที่ยกมา คือ การสืบค้นโดยใช้ Operator WHERE โดยมีรูปแบบคำสั่งดังต่อไปนี้

รูปแบบคำสั่งสืบค้นข้อมูลโดยใช้ Operator WHERE

SELECT fieldname FROM tablename WHERE (fieldname='namerecord');

คำอธิบาย

SELECT : คำสั่งเลือกข้อมูล

fieldname : field ที่ต้องการ ถ้าต้องการทุก field ให้ใช้เครื่องหมาย "*"

FROM : จาก table อะไร

WHERE : ข้อมูลที่ต้องการสืบค้น

(fieldname=namerecord) : field และ Record ที่ต้องการ

ตัวอย่าง

mysql>SELECT * FROM phonebook WHERE name='passkorn';

ในกรณีที่มีการแก้ไขข้อมูลมีรูปแบบคำสั่งดังต่อไปนี้

รูปแบบคำสั่งการแก้ไขข้อมูล

UPDATE tablename SET fieldname='datanew' WHERE fieldname='dataold;

คำอธิบาย

UPDATE : เป็นคำสั่งแก้ไขข้อมูล

tablename : เป็นชื่อ table ที่ต้องการแก้ไขข้อมูลภายใน table

fieldname='datanew' : เป็นชื่อ field และข้อมูลใหม่ที่ต้องการแก้ไขข้อมูล

WHERE : คำสั่งค้นหาข้อมูล

fieldname='dataold; : เป็นชื่อ field และข้อมูลเก่าที่ต้องถูกแก้ไข

ตัวอย่าง

mysql>UPDATE phonebook SET name='somjai' WHERE name='somsee';

หลังจากที่สามารถทำการแก้ไขข้อมูลได้ ขั้นตอนถัดไป คือ การลบข้อมูลซึ่งมีความจำเป็นต้องใช้ความระวังอย่างสูง เนื่องจาก MySQL จะไม่ทวนคำถามก่อนว่าจะยืนยันการลบหรือไม่ ข้อมูลที่ลบไปแล้วไม่สามารถ Undo กลับมาได้ ถ้าต้องการเรียกคืนจะต้องสร้างขึ้นใหม่ รูปแบบคำสั่งการลบข้อมูลมีดังต่อไปนี้

รูปแบบคำสั่งการลบข้อมูล

DELETE FROM tablename WHERE (fieldname='datadel');

คำอธิบาย

DELETE FROM : เป็นคำสั่งลบข้อมูล

tablename : เป็นชื่อ table ที่ต้องการลบข้อมูลภายใน table

WHERE : คำสั่งค้นหาข้อมูล

fieldname='datadel' : เป็นชื่อ field และข้อมูลที่ต้องการลบ

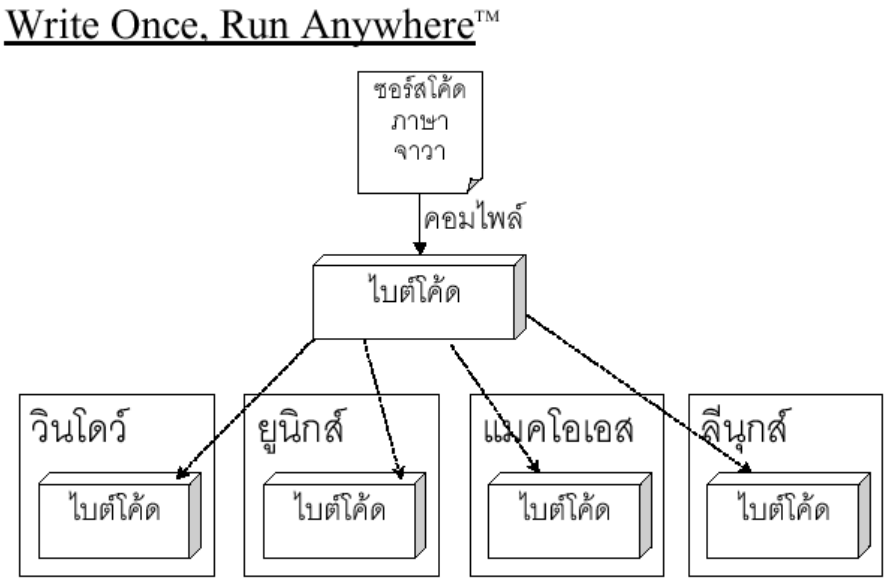
ตัวอย่าง

```
mysql >DELETE FROM phonebook WHERE (name='somsee');
```

นั่นก็เป็นเพียงตัวอย่างในการจัดการกับฐานข้อมูล MySQL ใน MySQL ยังเตรียมคำสั่งไว้สำหรับใช้งานอีกเป็นจำนวนมาก โดยสามารถหารายละเอียดเพิ่มเติมจากหนังสือที่เกี่ยวกับ MySQL ได้หรืออาจจะดูวิธีการใช้งานจาก Internet ก็ได้เช่นกัน

JAVA

Java เป็นภาษาเชิงวัตถุ (Object Oriented Program) ที่ได้รับความนิยมไปทั่วโลก ซึ่งเหมาะสมในการนำมาใช้กับออกแบบด้วย UML ที่ได้กล่าวไว้แล้วในบทที่ 4 อย่างยิ่ง และจากคุณสมบัติของมันที่ไม่ขึ้นอยู่กับระบบใดระบบหนึ่ง ทำให้มันสามารถ ทำงานได้บนหลากหลายระบบปฏิบัติการ บนคอมพิวเตอร์และอุปกรณ์หลากหลายประเภท หรือที่เราได้ยินเสมอๆ ว่า Write Once, Run Anywhere ซึ่งคุณสมบัติอันนี้เป็นคุณสมบัติที่สำคัญที่สุดอันหนึ่งของจาวา เพื่อให้โปรแกรมที่ได้พัฒนาขึ้นมี ความสามารถที่ทำงานได้ทุกระบบปฏิบัติการ ยกตัวอย่างเช่น ระบบปฏิบัติการวินโดว์ ยูนิกส์ แมคโอเอส และลีนุกซ์ เป็นต้น ดังรูปที่ 5-4



รูปที่ 5-4: รูปแสดงความสามารถของ Java บนระบบปฏิบัติการ

เหตุผลอีกประการหนึ่งคือการนำกลับมาใช้ใหม่ ทำให้ลดเวลาในการพัฒนาโปรแกรมลง ด้วยเหตุผลเหล่านี้เองทางคณะผู้จัดทำจึงเลือกใช้ภาษาจาวาเป็นภาษาในการพัฒนาระบบควบคุมและติดตามการจราจรอัตโนมัติ

XML

XML (eXtensible Markup Language) เป็นภาษาที่สามารถทำให้ผู้ใช้สร้างเอกสาร ที่ปรับเปลี่ยนการแสดงผลข้อมูลในรูปแบบที่หลากหลาย เช่น เนื้อหาของหนังสือพิมพ์ที่ถูกรวบรวมไว้ในเอกสารที่มีโครงสร้างแบบ XML เราจะสามารถนำเอกสารเหล่านั้น ไปพิมพ์ลงบนกระดาษ หรือ นำเสนอผ่านจอภาพคอมพิวเตอร์ ผลลัพธ์จากทั้งสองกรณีจะทำให้ผู้รับข้อมูล ได้ข่าวสารที่เหมือนกันจาก XML แต่ต่างกันตรงที่การแสดงผลเท่านั้น สิ่งนี้เองทำให้ XML มีความโดดเด่นและน่าสนใจ ที่ให้การพัฒนาและมีศักยภาพในส่วนของโครงสร้างข้อมูลจากหลากหลายแอพพลิเคชันมานำเสนอบนเครื่องเดสก์ทอป ด้วย XML จะทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอพพลิเคชันต่างๆ จะเข้าสู่มาตรฐานเดียวกัน

XML จะมีการกำหนดรายละเอียดของเนื้อหาเอกสารที่เรียกว่า Document Type Definition (DTD) ซึ่งเป็นตัวกำหนดโครงสร้างของเอกสารXMLโดยDTDจะทำหน้าที่ตรวจสอบความถูกต้องของเอกสาร XML ว่าสอดคล้องกับข้อตกลงที่กำหนดไว้ใน DTD หรือไม่ ดังนั้น XML เหมาะสมเป็นอย่างยิ่งที่จะใช้เป็นตัวแทนในการอธิบายข้อมูล เพื่องานติดต่อสื่อสารหรือแลกเปลี่ยนข้อมูล โดยสรุปได้ดังนี้

XML เป็นเพียง Text File จึงทำให้ง่ายต่อการพัฒนาส่วนโปรแกรมเพื่อมารองรับ เพื่อส่งข้อมูลสื่อสารระหว่างโปรแกรม อีกทั้งตัวอักษรในเอกสารจะถูกเข้ารหัสโดยมาตรฐานที่ได้รับการยอมรับ เช่น UTF-8 ดังนั้นจึงสามารถที่จะนำไปใช้ได้ในการแลกเปลี่ยนข้อมูลระหว่างระบบ (platform) ได้อีกด้วย

XML มีโครงสร้างเชิงวัตถุ (object oriented) จึงง่ายต่อการทำความเข้าใจของมนุษย์ และด้วยการออกแบบที่เป็นมาตรฐานทำให้ง่ายแก่แปลความหรือใช้งานในโปรแกรมด้วยเช่นกัน

ทางคณะผู้จัดทำจึงได้นำ XML มาใช้บนระบบควบคุมและติดตามการจราจรอัตโนมัติในส่วนของการบันทึกข้อมูลลงในไฟล์โปรเจค เพื่อความสะดวกในการแลกเปลี่ยนข้อมูลเนื่องจากมาตรฐานของ XML เป็นที่ยอมรับอย่างกว้างขวางทำให้สามารถนำข้อมูลในไฟล์โปรเจคของระบบไปประยุกต์ใช้ร่วมกับระบบอื่น ๆ ในรูปแบบที่ต้องการได้ อาทิเช่น การนำข้อมูลในไฟล์โปรเจคไปแสดงผลกราฟิกในอีกระบบหนึ่ง หรือการนำข้อมูลไปคำนวณค่าอื่น ๆ ต่อไปเป็นต้น

ตัวอย่างบางส่วนของไฟล์โปรเจค

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE kcnSystemFile [
<!ELEMENT kcnSystemFile (current, history) >
<!ELEMENT current (data_c, incomming_c)>
<!ELEMENT data_c (road_c*)>
<!ELEMENT road_c EMPTY>
<!ELEMENT incomming_c (order_c*)>
<!ELEMENT order_c EMPTY>
<!ELEMENT history (turn*)>
<!ELEMENT turn (data_h, incomming_h)>
<!ATTLIST turn id ID #REQUIRED  action CDATA "0">
<!ELEMENT data_h (road_h)*>
<!ELEMENT incomming_h (order_h)*>
<!ELEMENT road_h EMPTY >
<!ELEMENT order_h EMPTY >
<!ATTLIST road_c  id ID #REQUIRED  rel_id CDATA "0"  car_out CDATA "0"  car_remain
CDATA "0"  light_order CDATA "0"  light_type CDATA "0"  light_state CDATA "0"
light_length CDATA "0"  light_remain CDATA "0"  tf_flow CDATA "0"  tf_ratio CDATA "0">
<!ATTLIST order_c id ID #REQUIRED  rel_id CDATA "0"  distance CDATA "0"  carvolume
CDATA "0">
<!ATTLIST road_h   id ID #REQUIRED  rel_id CDATA "0" car_out CDATA "0"  car_remain
CDATA "0"  light_order CDATA "0"  light_type CDATA "0"  light_state CDATA "0"
light_length CDATA "0"  light_remain CDATA "0"  tf_flow CDATA "0"  tf_ratio CDATA "0">
<!ATTLIST order_h  id ID #REQUIRED  rel_id CDATA "0"  distance CDATA "0"  carvolume
CDATA "0">]>
…
```

ตัวอย่างไฟล์โปรเจคบางส่วนนี้จะแสดงการใช้ DTD ในการกำหนดโครงสร้างเอกสาร เพื่อใช้ตรวจสอบความถูกต้องของไฟล์โปรเจคที่ใช้งาน ส่วนตัวอย่างไฟล์โปรเจคที่เหลือจะแสดงอยู่ในภาคผนวก ฉ

5.2: การประยุกต์หลักการจำลองมาใช้ในการพัฒนาระบบ

5.2.1 แนะนำการจำลอง

การจำลองคือการทำตามของการดำเนินการของการประมวลผลความเป็นจริง หรือระบบ ไม่ว่าทำด้วยตัวเองหรือคอมพิวเตอร์ การจำลองเกี่ยวข้องกับข้อมูลเก่าที่สร้างขึ้นของระบบและการสังเกตของข้อมูลเก่าที่สร้างขึ้นเพื่อดึงข้อสรุปเกี่ยวกับลักษณะเฉพาะของระบบจริง

พฤติกรรมของระบบถูกศึกษาโดยการพัฒนาโมเดลการจำลอง. โมเดลรวบรวมรูปแบบการสมมติฐานเกี่ยวกับการดำเนินการของระบบ การสมมติฐานเหล่านี้เป็นรูปแบบเกี่ยวกับคณิตศาสตร์ ตรรกะ และวัตถุที่สนใจของระบบ. เมื่อพัฒนาจนสมบูรณ์ โมเดลสามารถศึกษาคำถาม “what if”ในระบบจริงได้หลากหลายขึ้น การเปลี่ยนแปลงของระบบจะถูกจำลองขึ้นเพื่อทำนายผลกระทบต่อประสิทธิภาพระบบ การจำลองสามารถใช้ศึกษาระบบในส่วนที่ต้องการก่อนจะเกิดขึ้นจริง ดังนั้นสร้างแบบการจำลองสามารถใช้เป็นเครื่องมือวิเคราะห์เพื่อทำนายผลของการเปลี่ยนแปลงต่อระบบจริง และเครื่องมือออกแบบเพื่อทำนายประสิทธิภาพของระบบใหม่ต่อสถานการณ์ที่กำหนดขึ้น

ในตัวอย่างหลายๆอันโมเดลสามารถพัฒนาด้วยวิธีที่เกี่ยวกับคณิตศาสตร์ เช่นทฤษฎีอนุพันธ์แคลคูลัสความน่าจะเป็น พีชคณิต หรือเทคนิคที่เกี่ยวกับคณิตศาสตร์อื่น ผลลัพธ์ประกอบด้วยพารามิเตอร์เป็นตัวเลข และตัวเลขนี้เป็นมาตรฐานการวัดผลของระบบ อย่างไรก็ตามระบบจริงจะสลับซับซ้อนมาก และโมเดลของระบบเหล่านี้ไม่มีทางแก้ด้วยคณิตศาสตร์ได้ ตัวเลขของพารามิเตอร์เหล่านี้ถูกใช้จำลองเพื่อเลียนแบบพฤติกรรมของระบบ จากการจำลองระบบข้อมูลต่างที่ได้จะเหมือนว่าระบบจริงกำลังถูกสังเกต และข้อมูลที่ถูกจำลองใช้ประเมินมาตรฐานการวัดผลของระบบ.

5.2.2 การจำลองคือเครื่องมือที่เหมาะสม

สิ่งที่เป็นจุดประสงค์พิเศษของการจำลองคือ ความสามารถของการคำนวณที่ดี และลดค่าใช้จ่ายต่อการดำเนินการ และการจำลองที่พัฒนาออกมา ทำให้การจำลองเป็นเครื่องมือที่ใช้อย่างกว้างขวางที่สุดอันหนึ่ง และได้รับการยอมรับในการวิเคราะห์ระบบ การจำลองสามารถใช้สำหรับจุดมุ่งหมายดังนี้:

1. การจำลองทำให้การศึกษาการปฏิสัมพันธ์ภายในของระบบที่สลับซับซ้อนหรือระบบที่แยกออกภายในระบบที่สลับซับซ้อนทำได้ง่ายขึ้น.
2. การเปลี่ยนแปลงทางสารสนเทศและสิ่งแวดล้อมสามารถเลียนแบบและผลของการเปลี่ยนแปลงพฤติกรรมของโมเดลสามารถสังเกตได้.

3. ความรู้ที่ได้มาในการออกแบบโมเดลการจำลองบางทีจะมีคุณค่ามากเช่นการปรับปรุงในระบบที่กำลังวินิจฉัย.
4. การเปลี่ยนข้อมูลการจำลองและการสังเกตผลลัพธ์ ผลที่มีสำคัญซึ่งจะมาจากค่าตัวแปรและตัวแปรติดต่อกันอย่างไร.
5. การจำลองสามารถใช้เป็นเครื่องมือศึกษาศาสตร์เพื่อเสนับสนุนกระบวนการวิเคราะห์คำตอบ
6. การจำลองสามารถใช้ทดลองออกแบบการพัฒนา หรือเตรียมตัวให้พร้อมสำหรับบางอย่างที่อาจจะเกิดขึ้น
7. การจำลองสามารถใช้เพื่อตรวจสอบคำตอบที่ผ่านการวิเคราะห์.
8. โมเดลการจำลองที่ออกแบบสามารถใช้เรียนรู้ซึ่งไม่มีค่าใช้จ่าย
9. ภาพเคลื่อนไหวแสดงระบบด้วยการดำเนินการที่ถูกเลียนแบบซึ่งแผนสามารถแสดงฝให้เห็นด้วยภาพได้
10. เพราะว่าระบบร่วมสมัย(โรงงานประกอบเวเฟอร์)สลับซับซ้อนมากดังนั้นการปฏิสัมพันธ์ภายในสามารถถูกจัดการโดยใช้การจำลองเท่านั้น

5.2.3 การจำลองที่ไม่เหมาะสม

ส่วนนี้อ้างอิงจากบทความของ Banks and Gibson [1997], ได้กำหนดกฎ 10 ข้อเพื่อตัดสินใจว่าการจำลองไม่เหมาะสมเมื่อไหร่

กฎที่ 1 บอกการจำลองไม่ควรใช้เวลาที่ปัญหาสามารถแก้โดยใช้สามัญสำนึก ตัวอย่าง

กฎที่ 2 การจำลองไม่ควรใช้ถ้าปัญหาสามารถถูกแก้ด้วยการวิเคราะห์ได้

กฎที่ 3 การจำลองไม่ควรใช้ถ้าการทดลองโดยตรงง่าย

กฎที่ 4 ไม่ใช้การจำลองถ้าค่าใช้จ่ายสูงเกินรายได้ การจำลองคงจะไม่เหมาะสม

กฎที่ 5 และ 6 การจำลองไม่ควรสร้างถ้าทรัพยากรหรือเวลามีไม่พอ เช่น เงินทุน หรือ เวลา

กฎที่ 7 การจำลองต้องใช้ข้อมูลซึ่งบางครั้งข้อมูลจำนวนมาก ถ้าข้อมูลมีไม่พอก็ไม่ควรจำลอง

กฎที่ 8 เกี่ยวกับความสามารถตรวจสอบและประสิทธิภาพโมเดล ถ้าไม่มีเวลาหรือบุคลากรเพียงพอก็ไม่ควรจำลอง

กฎที่ 9 ถึงความต้องการของการจำลองไม่สมเหตุสมผลมากเกินไป ทำให้ไม่เหมาะสมกับการจำลอง

กฎที่ 10 การจำลองพฤติกรรมระบบสลับซับซ้อนหรือให้คำจำกัดความการจำลองไม่เหมาะสม เช่น พฤติกรรมมนุษย์ซึ่งสลับซับซ้อนมากเกินไปสำหรับโมเดลบางครั้งทำให้ไม่เหมาะสมกับการจำลอง

5.2.4 ข้อดีและข้อเสียของการจำลอง

ข้อมูลออกจากการจำลองควรจะสอดคล้องผลที่ควรถูกเป็นจากระบบจริง เป็นไปได้ที่การพัฒนาโมเดลของระบบจำลองไม่มีสมมติฐานที่กำกวม ของโมเดลที่แก้ไขได้ทางคณิตศาสตร์ (อย่างเช่นการแบ่งกระจายตามสถิติที่เหมือนกันสำหรับค่าที่สุ่มขึ้นมาทุกครั้ง) สำหรับเหตุผลเหล่านี้การจำลองใช้เป็นเทคนิคของการเลือกในแก้ไขปัญหาบ่อยๆ

ในทางกลับกันโมเดลเหมาะสมที่สุดคือโมเดลการจำลองจะทำงานมากกว่าแก้ปัญหา ข้อมูลและลักษณะเฉพาะของโมเดลจะทำงานและสังเกตพฤติกรรมการจำลอง การประมวลผลข้อมูลที่เปลี่ยนและลักษณะเฉพาะโมเดลทำให้เกิดผลลัพธ์ออกมา

การจำลองมีข้อดีและข้อเสียเช่นกัน ข้อดีข้อเสียเหล่านี้แสดงโดย Pegden, Shannon, และ Sadowski [1995] ข้อดีคือ

1. นโยบายใหม่ ขั้นตอนการปฏิบัติ กฎการตัดสินใจ ขั้นตอนที่เป็นองค์กร สามารถสำรวจโดยไม่ทำให้การดำเนินการวุ่นวายขณะที่อยู่ระหว่างการดำเนินการของระบบจริง.
2. การออกแบบฮาร์ดแวร์ ระบบการขนส่ง สามารถทดลองโดยไม่จำเป็นต้องใช้ทรัพยากรจริง
3. สามารถทดสอบความเป็นไปได้เหตุการณ์บางอย่างที่จะทำให้เป็นจริง.
4. สามารถลดหรือเพิ่มเวลา การพิจารณาโดยการเพิ่มหรือลดความเร็วของเหตุการณ์ที่สังเกตได้
5. สามารถดูความสัมพันธ์ตัวแปรได้อย่างชัดเจน
6. การศึกษาการจำลองช่วยให้เข้าใจว่าระบบทำงานอย่างไร แทนที่จะคิดว่าระบบปฏิบัติการเดี่ยวทำงานอย่างไร
7. สามารถตอบคำถาม\'What-if\' Ibis มีประโยชน์อย่างยิ่งในการออกแบบของระบบใหม่

ความเสียเปรียบคือต่อไปนี้:

1. การสร้างแบบจำลองต้องการอบรมพิเศษ ซึ่งการจำลองเหมือนเป็นศาสตร์แขนงหนึ่งที่ต้องใช้เวลาการศึกษานาน และเรียนรู้จากประสบการณ์
2. ผลการจำลองแปลออกมายาก เพราะว่าผลการจำลองส่วนใหญ่คือค่าตัวแปรสุ่ม ซึ่งจะยากในการตัดสินใจว่าคำตอบเป็น ผลของความสัมพันธ์ซึ่งกันและกันของระบบหรือการสุ่ม.
3. สร้างแบบการจำลองและวิเคราะห์ใช้เวลามากและแพง การประหยัดทรัพยากรอาจจะเกิดผลแค่โมเดลการจำลองหรือวิเคราะห์ที่ไม่เพียงพอในการทำงาน

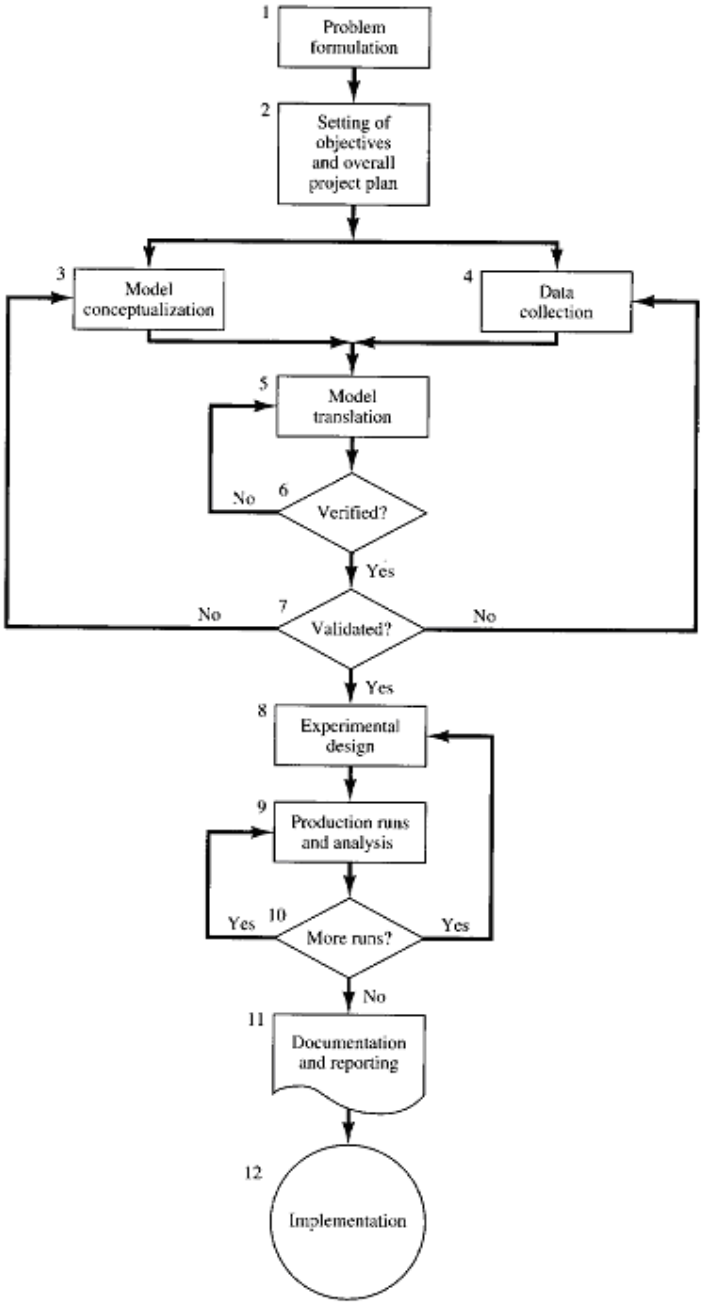
แต่ข้อเสียของการจำลองเหล่านี้ ได้มีการพัฒนาแก้ไขตามลำดับดังต่อไปนี้

1. ซอฟต์แวร์การจำลองสำเร็จรูป ที่พัฒนาสำเร็จจะมีโมเดลที่ต้องการ เพียงใส่ข้อมูลโมเดลสามารถทำงานได้ โมเดลแบบนั้นมีข้อความทั่วไปว่า"simulators" หรือ "templates"
2. ซอฟต์แวร์การจำลองสำเร็จรูปพัฒนาความสามารถวิเคราะห์ผลภายในโปรแกรมสำเร็จเพื่อปฏิบัติการวิเคราะห์ที่สมบูรณ์
3. การจำลองจะสามารถทำงานเร็วขึ้น และมีการพัฒนาการจำลองเป็นฮาร์ดแวร์สำเร็จรูปที่ทำงานได้รวดเร็วกว่ามาก .

5.2.5 ขั้นตอนในการศึกษาการจำลอง

ในรูปที่ 5-5 ได้แสดงถึงขั้นตอนการสร้างโมเดลและการจำลอง ขั้นในการศึกษาการจำลองมีดังต่อไปนี้

1. **Problem formulation** การศึกษาทุกอย่างควรเริ่มต้นด้วยการปัญหา ถ้าปัญหานี้มาจากผู้กำหนดนโยบายหรือผู้ที่เห็นปัญหา ต้องอธิบายปัญหาเข้าใจอย่างชัดเจน ถ้าปัญหามาจากผู้วิเคราะห์ต้องแน่ในว่าปัญหานี้ผู้กำหนดนโยบายเข้าใจ และเห็นด้วย ในบางทีปัญหาต้องเรียบเรียงใหม่อยู่เสมอเมื่อศึกษาไปเรื่อยๆ
2. **Setting of objective and overall project plan** วัตถุประสงค์จะชี้คำถามที่ดูผลลัพธ์ได้จากการจำลอง ซึ่งการจำลองควรจะสร้างโดยคำนึงถึงปัญหาและวัตถุประสงค์ที่กำหนดขึ้นมา สมมติว่าการจำลองถูกสร้างขึ้นมา การวางแผนระบบโดยรวมควรรวมระบบรูปแบบอื่น และกระบวนประเมินผลด้วย นอกจากนั้นควรแสดงถึงผู้ใช้ที่เกี่ยวข้องกับระบบนี้
3. **Model conceptualization** การสร้างโมเดลของระบบคล้ายกับศิลปะรวมกับวิทยาศาสตร์ [Pritsker, 1998] ได้พูดถึงขั้นตอนนี้ "แม้ว่าการกำหนดการสร้างโมเดลที่ทำให้ประสบความสำเร็จและเหมาะสมในปัญหาเป็นไปไม่ได้มี แต่อย่างน้อยเป็นคำแนะแนวทั่วไปที่สามารถทำตามได้" [Morris, 1967] ศิลปะการสร้างโมเดลขึ้นอยู่กับสามอย่างคือ หนึ่ง ความสามารถการกำหนดความสำคัญของปัญหา สองการเลือกและปรับเปลี่ยนสมมติฐานที่เป็นลักษณะเฉพาะของระบบ สามการกำหนดโมเดลจนกระทั่งได้ผลลัพธ์ที่คาดไว้ ดังนั้นควรจะเริ่มที่โมเดลธรรมดาและสร้างโมเดลที่ซับซ้อนยิ่งกว่าตามมา อย่างไรก็ตามโมเดลไม่จำเป็นซับซ้อนเพื่อดำเนินการให้ลุล่วงจุดมุ่งหมายที่ตั้งใจ ถ้าทำเกินความจำเป็นจะทำให้ค่าใช้จ่ายเพิ่มขึ้นเพียงอย่างเดียว



รูปที่ 5-5: ขั้นตอนการสร้างโมเดลและการจำลอง

4. **Data collection** เป็นสิ่งที่อยู่ระหว่างการสร้างโมเดลและชุดข้อมูลที่จำเป็น [Shannon, 1975] เมื่อความซับซ้อนของโมเดลเปลี่ยน ข้อมูลก็เปลี่ยนตาม ดังนั้นถ้าข้อมูลมีจำนวนมากและต้องใช้เวลามากในการจำลองจึงควรเริ่มให้เร็วที่สุดซึ่งโดยปกติจะเริ่มที่การสร้างโมเดล

5. **Model Translation** เพราะว่าความเป็นจริงส่วนใหญ่จะถูกแทนเป็นโมเดล ซึ่งต้องการการเก็บข้อมูลมากมายและการคำนวณ โมเดลจะถูกแปลความหมายเป็นรูปแบบของคอมพิวเตอร์ ซึ่งจะเรียกว่า โปรแกรม แม้บางทีสามารถทำได้ง่ายหรืออาจจะไม่ต้องเขียนโปรแกรมเลย ผู้ออกแบบโมเดลจะต้องเลือกภาษาที่ใช้ในการจำลอง ภาษาการจำลองมีประสิทธิภาพและยืดหยุ่น อย่างไรก็ตามถ้าปัญหาตรงกับคำตอบด้วยโปรแกรมจำลองทำให้เวลาการพัฒนาโมเดลถูกทำให้น้อยลงอย่างมาก
6. **Verify?** การตรวจสอบเกี่ยวข้องกับโปรแกรมที่เตรียมพร้อมสำหรับโมเดลการจำลองโปรแกรมคอมพิวเตอร์กำลังปฏิบัติหน้าที่ปฏิบัติงานอย่างเหมาะสมหรือไม่? เพราะว่าโมเดลที่สลับซับซ้อนมันยากที่จะแปลโมเดลสำเร็จทั้งหมดโดยไม่มีเครื่องมือตรวจสอบที่ดี ถ้าพารามิเตอร์ข้อมูลและโมเดลถูกต้อง นั่นคือการตรวจสอบสำเร็จ ส่วยใหญ่สามัญสำนึกถูกใช้ในขั้นตอนนี้จนเสร็จ
7. **Validated?** Validation คือการตรวจสอบโมเดลที่แสดงนั้นถูกต้องเหมือนความเป็นจริง การประมวลผลเปรียบเทียบโมเดลกับพฤติกรรมของระบบซึ่งมีอยู่ในเวลานี้และการใช้ไม่สอดคล้องระหว่างกันเพื่อพัฒนาโมเดล การประมวลผลอันนี้ถูกทำซ้ำจนกระทั่งความละเอียดโมเดลถูกตัดสินใจว่าที่รับได้
8. **Experimental design** วิธีการจำลองต้องถูกกำหนด บ่อยๆการตัดสินใจที่กำลังเกี่ยวพันวิธีการจำลองคือหน้าที่ของการทำงานที่สำเร็จและวิเคราะห์ สำหรับการออกแบบระบบเพื่อการจำลองแต่ละอัน จะต้องตัดสินใจเกี่ยวกับเวลาในการเริ่มต้น เวลาของการจำลอง
9. **Production Runs and Analysis** ใช้สำหรับการประเมินประสิทธิภาพของระบบที่ถูกจำลองขึ้น
10. **More runs?** ขึ้นอยู่กับการวิเคราะห์ความสมบูรณ์ของการทำงานโดยตรวจสอบจากการทำงานเพิ่มเติมและการทดลองเพิ่มเติมที่ตามมา
11. **Documentation & Reporting** มีเอกสารอยู่ 2 แบบคือ โปรแกรมและความก้าวหน้า เอกสารโปรแกรมจำเป็นเมื่อต้องใช้เหตุผลมาก ถ้าโปรแกรมถูกนำไปใช้อีกครั้งโดยนักวิเคราะห์คนเดิมหรือไม่ใช่ก็ตาม ทุกคนต้องเข้าใจว่าโปรแกรมทำงานอย่างไร นอกจากนี้ถ้าโปรแกรมจำเป็นต้องถูกปรับเปลี่ยนโดยนักวิเคราะห์คนเดิมหรือไม่ใช่ ส่วนนี้จะทำได้ง่ายถ้ามีเครื่องมือที่เหมาะสม เหตุผลอีกอันของการทำให้เป็นเอกสารโปรแกรมเมื่อผู้ใช้โมเดลสามารถเปลี่ยนพารามิเตอร์เพื่อกำหนดความสัมพันธ์ระหว่างพารามิเตอร์ใส่ข้อมูลและการประเมินประสิทธิภาพของผลลัพธ์

[Musselman, 1998] กล่าวถึงความสำคัญรายงานความคืบหน้าที่ให้ข้อมูลการทำงานที่ผ่านมาและถูกเขียนของโครงการการจำลอง Musselman เสนอให้มีการรายงานบ่อยๆ (รายเดือนอย่างน้อย) เพื่อที่ไม่เกี่ยวข้องในการดำเนินการประจำวันก็ยังสามารถรักษาสภาพเดิมกำลังก้าวไปพร้อมกัน

ผลของวิเคราะห์ควรรายงานอย่างชัดเจนกะทัดรัดในรายงานส่วนสุดท้าย เพราะให้ผู้ใช้โมเดลพิจารณาไตร่ตรองปัญหาท้ายสุดของระบบ และทดลองเปรียบเทียบการเลือกคำตอบที่ถูกของปัญหา

12. **Implementation** ความสำเร็จของขั้นนี้ขึ้นอยู่กับ 11 ขั้นที่ผ่านมาถูกทำงานอย่างไร และยังเกี่ยวกับผู้ทำการวิเคราะห์เกี่ยวข้องกับผู้ใช้โมเดลที่สูงสุดโดยตลอดการจำลองอย่างไร ถ้าผู้ใช้โมเดลกับเกี่ยวข้องโดยตลอดและเข้าใจวิสัยของโมเดลและผลนั้นความเป็นไปได้ของการทำให้เกิดผลที่ดี [Pritsker, 1995] ในทางกลับกันถ้าโมเดลและการสมมติที่ออกมาไม่ถูกสื่อสารอย่างเหมาะสมการทำให้เกิดผลร้ายต่อโมเดลการจำลอง

การประมวลผลการจำลองการสร้างโมเดล แผนที่ 1.3 สามารถแยกออกเป็น 3 ขั้น ขั้นแรกประกอบด้วยขั้นตอนที่ 1 (Problem formulation) และ ขั้นตอนที่ 2 (Setting of objective and overall project plan) ซึ่งเป็นช่วงระยะเวลาของคิดค้น รายงานระยะแรกของปัญหาระยะแรกจะสับสนจำเป็นต้องถูกตั้งใหม่ ตัวชี้บนมาตราส่วนของเครื่องมือวัดใหม่เหล่านี้ น่าจะเกิดในขั้นนี้ หรืออาจจะสงสัยภายหลัง หรือระหว่างขั้นอื่นอีก (กล่าวคือนักวิเคราะห์น่าจะจำเป็นต้องเริ่มทำการประมวลผลใหม่)

ขั้นที่ 2 เกี่ยวโยงกับการสร้างแบบจำลองและการรวมรวมข้อมูล ขั้นตอนที่ 3 (Model conceptualization) 4 (Data collection) 5 (Model Translation) 6 (Verify?) และ 7 (Validation) ตัดส่วนใดส่วนหนึ่งออกของระหว่างขั้นอันนี้สามารถมีการเกี่ยวพันที่อันตราย

ขั้นที่ 3 เกี่ยวพันกับการทำงานของโมเดล เกี่ยวข้องกับขั้น 8 (Experimental design) 9 (More runs?)และ10 (Additional Runs) ขั้นอันนี้ต้องมีแผนที่ถูกคิดข้อเสนอโดยตลอดเกี่ยวกับการทดลองด้วยโมเดลการจำลอง ค่าตัวแปรผลคือการประเมินที่บรรจุค่าสุ่มผิด และดังนั้นการวิเคราะห์ทางสถิติที่เหมาะสมเป็นที่ต้องการ

ภาคผนวก ก

แบบเสนอหัวข้อโครงงานรายวิชา 2301399 Project Proposal

ปีการศึกษา 2545

ชื่อโครงงานภาษาไทย **ระบบควบคุมและติดตามการจราจรอัตโนมัติ**

ชื่อโครงงานภาษาอังกฤษ An Automatic Traffic Control And Monitoring System

ผู้ดำเนินการ

หัวหน้าโครงงาน : 1. นายณัฐพล พงษ์อดุลยสุข รหัสประจำตัวนิสิต 423 21007 23

2. นายกิตติพงศ์ เตชะพานิชกุล รหัสประจำตัวนิสิต 423 20173 23

3. นายจักกนาท วิวัฒนาวรสิน รหัสประจำตัวนิสิต 423 20442 23

อาจารย์ที่ปรึกษา **ผศ.ดร. พีระพนธ์ โสพัศสถิตย์**

ความเป็นมาและมูลเหตุจูงใจในการเสนอโครงงาน

การเดินทางในปัจจุบันมีรูปแบบหลากหลายวิธี ไม่ว่าจะเป็นการเดินทาง ทางน้ำ ทางบก ทางอากาศ แต่การเดินทางที่นิยมที่สุดคือ การเดินทางทางบกโดยใช้รถยนต์เป็นยานพาหนะ จึงทำให้การควบคุมรถยนต์บนท้องถนนมีความสำคัญ ดังนั้นปัญหาที่เกี่ยวข้องกับการควบคุมนี้จึงมีความสำคัญตามไปด้วย ปัญหาที่เด่นชัดมากที่สุด คือ ปัญหาการจราจรที่ติดขัด ซึ่งส่งผลกระทบต่อปัจจัยหลายๆด้าน เช่น ความเครียดของบุคคลบนท้องถนน การสูญเสียเวลาโดยเปล่าประโยชน์ การก่อมลพิษทางอากาศ เป็นต้น

จากปัญหาที่กล่าวมานี้ พบว่าเกิดจากการควบคุมสัญญาณไฟจราจร ที่ขาดการวางแผนให้มีความเหมาะสมกับจำนวนรถบนท้องถนนในขณะนั้น หากเราได้มีการศึกษาถึงปัญหารวมทั้งวิธีการควบคุมสัญญาณไฟจราจรที่เหมาะสม แล้ว จะทำให้เราสามารถนำมาประยุกต์ใช้กับสถานการณ์จริงได้ แต่การที่เราจะไปศึกษาถึงวิธีการควบคุมสัญญาณจราจรและผลกระทบที่เกิดขึ้นจากวิธีการควบคุมในสถานการณ์จริง ทำให้การมองเห็นภาพรวมของระบบการจราจรนั้นเป็นไปได้ยาก

ด้วยเหตุผลเหล่านี้จึงเป็นที่มาของการสร้างระบบควบคุมและติดตามการจราจรอัตโนมัติ ที่สามารถนำปัญหาและวิธีที่ใช้ในการควบคุมสัญญาณไฟ มาจำลองให้เกิดขึ้นในระบบจำลองนี้ แทนที่จะศึกษาจากสถานการณ์จริงที่อาจต้องใช้ระยะเวลาและค่าใช้จ่ายมากกว่า เพื่อให้ได้วิธีการที่เหมาะสมที่จะนำมาใช้กับสถานการณ์นั้น

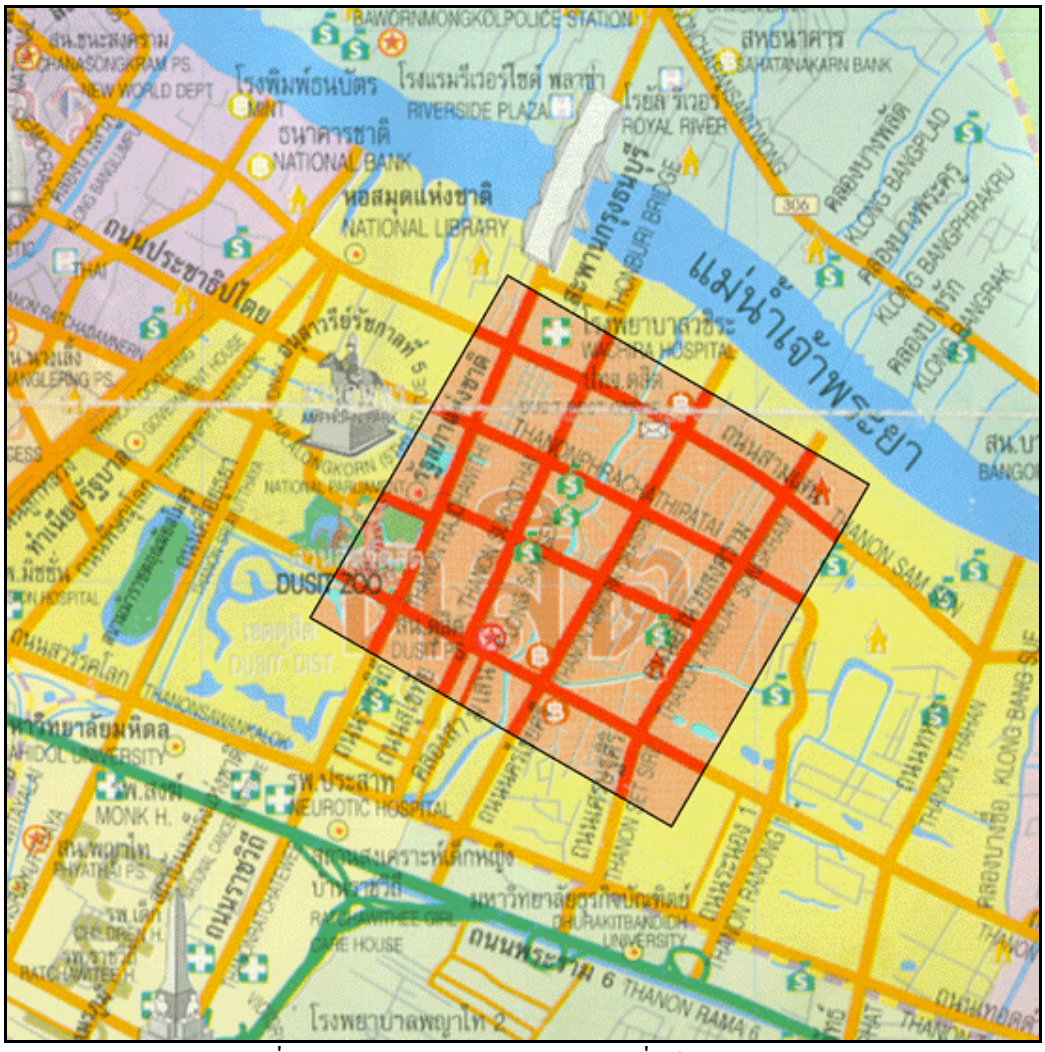
วัตถุประสงค์ของโครงงาน

สร้างระบบจำลองสถานการณ์การจราจรในลักษณะต่างๆ โดยนำข้อมูลจากการเก็บสถิติมาทำการวิเคราะห์ และประมวลผลเพื่อแสดงผลกระทบที่เกิดขึ้นจากการควบคุมสัญญาณไฟจราจรที่มีต่อการเคลื่อนตัวของรถในสถานการณ์จำลองที่สร้างขึ้น

ขอบเขตโครงงาน

1. ขอบเขตการจราจรที่นำมาใช้ในการศึกษานั้น จะมีอาณาเขตในเขตกรุงเทพมหานคร บริเวณสวนจิตรลดา (ดังรูปที่ ก-1)
2. สภาพการณ์จราจรนั้น นำมาศึกษาโดยไม่คำนึงถึงกรณีที่เกิดจากปัจจัยภายนอก เช่น อุบัติเหตุ การซ่อมถนน บริเวณห้ามจอด บริเวณห้ามเลี้ยว สิ่งกีดขวาง ฯลฯ
3. สถานการณ์ที่นำมาศึกษาจะพิจารณาในกรณีที่ผู้ขับรถมีวินัยการจราจรอย่างเคร่งครัด (ผู้ขับขี่ทุกคนไม่สามารถแซงกัน ปาดหน้าหรือแทรกแถวกันได้)
4. ปริมาณรถบอกอยู่ในรูปของ Scale ไม่ใช่ Object
5. กรณีที่จำนวนของช่องการจราจรมีขนาดไม่สม่ำเสมอจะใช้ค่าเฉลี่ยของช่องการจราจรในการคำนวณ
6. แผนที่ที่ถูกสร้างขึ้นในโปรแกรมอาจไม่ตรงตามความเป็นจริง (จะมีแค่ระยะทางและทิศทางของถนนเท่านั้นที่สมจริง เช่น ทางโค้งจะแสดงผลเป็นเส้นตรง)
7. สะพานข้ามแยก ทางด่วน ทางเดินเฉพาะรถประจำทาง จะไม่ได้นำมาพิจารณาในระบบ

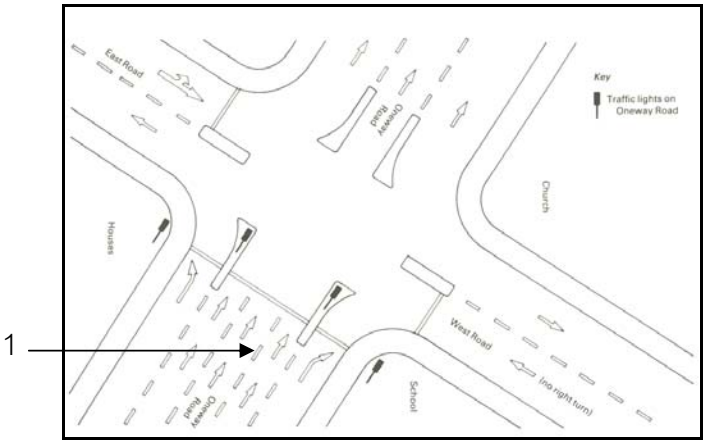
ขอบเขตการจราจรที่นำมาใช้ในการศึกษา



รูปที่ ก-1: รูปแสดงขอบเขตการจราจรที่ใช้ในการศึกษา

วิธีการดำเนินงาน

1. ศึกษาและทำความเข้าใจหลักการและทฤษฎีที่เกี่ยวกับการแก้ไขปัญหาจราจร เช่น การจำลองข้อมูลจะนำหลักการและทฤษฎีของป๊วซองมาประยุกต์ใช้ เพราะว่าการเก็บตัวอย่างนั้นเก็บมาเพียงบางช่วงเวลาเนื่องจากการเก็บข้อมูลจากสถานการณ์จริงให้ครบถ้วนนั้นทำได้ยากและใช้เวลานาน
2. ศึกษาและหาวิธีการที่ใช้จำลองข้อมูลการจราจร เช่น วิธีการแบ่งช่องจราจร ถนนที่สร้างขึ้นในระบบจำลองจะมีช่องจราจรเช่นเดียวกับถนนตามความเป็นจริงดังรูปที่ ก-2 ถนนหมายเลข 1 นั้นมีช่องจราจรเลี้ยวซ้ายและเลี้ยวขวาอย่างละ 1 ช่องและมีช่องจราจรตรงไปอีก 3 ช่อง



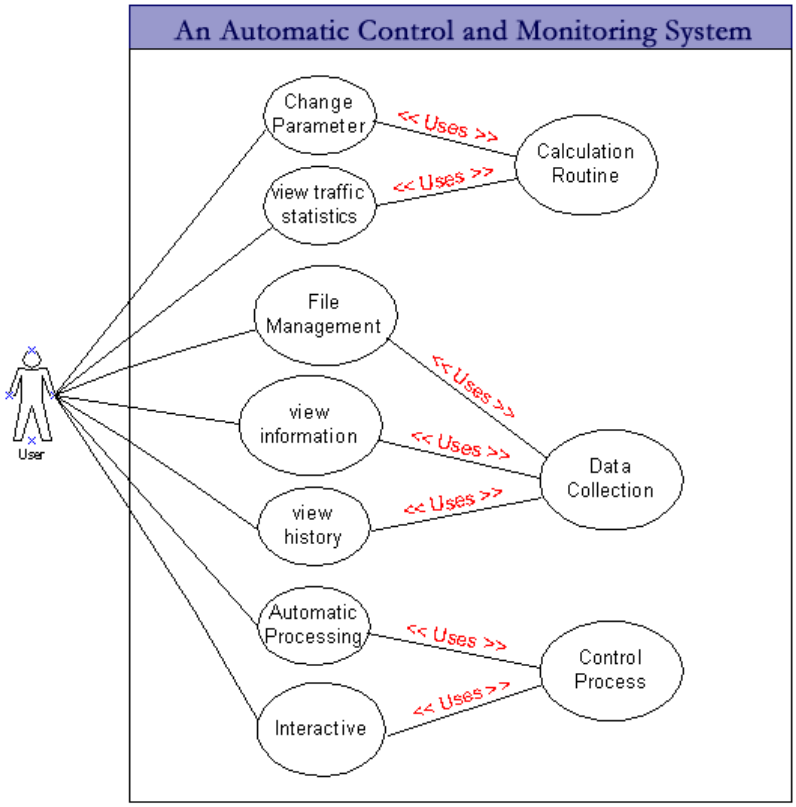
รูปที่ ก-2: การแบ่งช่องจราจร

3. รวบรวมรายละเอียดของแต่ละถนนรวมทั้งข้อมูลต่าง ๆ ที่อยู่ในขอบเขตที่ใช้ศึกษา โดยจะการรวบรวมข้อมูลเป็น 2 ชุด ได้แก่
   - ข้อมูลที่ใช้ระหว่างการพัฒนาระบบ เช่น ปริมาณรถ อัตราส่วนทิศทางรถ เป็นต้น เพื่อใช้ในการกำหนดพารามิเตอร์ของระบบจำลอง
   - ข้อมูลที่ใช้ในการทดสอบระบบ เพื่อใช้ในการตรวจสอบระบบว่าทำงานได้อย่างถูกต้อง
4. ศึกษาวิธีการจำลองสถานการณ์เพื่อการสร้างแบบจำลองของระบบ โดยจะนำข้อมูลที่ได้จากการเก็บในข้อ 3 มาเข้ากระบวนการป๊วซองโมเดลเพื่อสร้างข้อมูลที่ใช้ในระบบจำลอง เช่น การสร้างปริมาณรถของถนนในแต่ละช่วงเวลา
5. กำหนดขอบเขตโครงงาน ดังที่กล่าวไว้แล้วในหัวข้อขอบเขตโครงงาน
6. ออกแบบระบบ (โปรแกรม, ฐานข้อมูล, หน้าจอ)
7. พัฒนาระบบ โดยมีโครงสร้างของโปรแกรมดังรูปที่ ก-3 และมีรูปแบบหน้าจอดังรูปที่ ก-4
8. ทดสอบระบบ โดยจะนำข้อมูลสถิติการจราจรอย่างละเอียด (มาจากสำนักงานคณะกรรมการจัดการจราจร) มาตรวจสอบกับระบบจำลองที่สร้างขึ้น
9. จัดทำเอกสารประกอบ

ระยะเวลาการดำเนินงาน

ตารางที่ ก.1: ตารางระยะเวลาการดำเนินงาน

ขั้นตอนการดำเนินงาน	เดือน								
	มิ. ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
ศึกษาระบบจราจรจากสถานการณ์จริง									
ศึกษาวิธีการที่ใช้จำลองข้อมูลการจราจร									
รวบรวมรายละเอียดของแต่ละถนน									
ศึกษาวิธีการจำลองสถานการณ์									
กำหนดขอบเขตโครงงาน									
ออกแบบระบบ									
พัฒนาระบบ									
ทดสอบระบบ									
จัดทำเอกสารประกอบ									

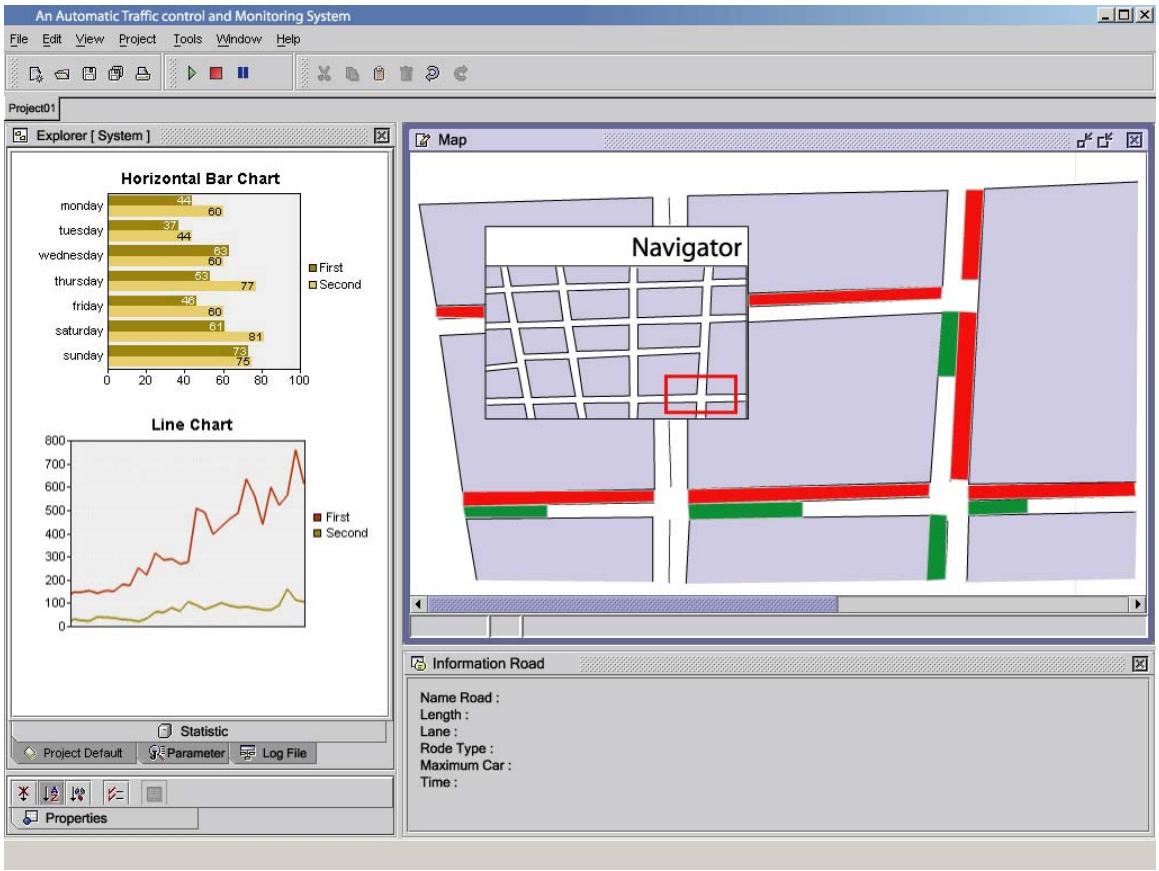


รูปที่ ก-3: โครงสร้างโปรแกรม

ประโยชน์ที่คาดว่าจะได้รับ

1. ศึกษาผลกระทบตัวสภาพจราจรจากเวลาและวิธีในการเปิด – ปิดไฟแดง การจำลองข้อมูลการจราจร แสดงผลของสภาพจราจรได้ ข้อมูลที่เกี่ยวข้องและแสดงสถิติสภาพจราจรจากผลกระทบที่เกิดจากการเปิด – ปิดไฟได้
2. เข้าใจการ Formulate ปัญหาการจราจรตามกรณีศึกษา
3. สามารถหาวิธีการที่ใช้แก้ไขปัญหาที่เหมาะสมกับปัญหานั้น ๆ ได้
4. สามารถออกแบบและสร้างฐานข้อมูลได้ และเขียนโปรแกรมในการจัดการกับฐานข้อมูลได้
5. ได้เรียนรู้ถึงภาษาต่างๆ ที่ใช้ในการพัฒนา เช่น
   - C on Unix
   - OpenGL

หน้าจอของโปรแกรม



รูปที่ ก-4: หน้าจอของโปรแกรม

อุปกรณ์ที่ใช้

1. อุปกรณ์คอมพิวเตอร์ และโปรแกรมที่ใช้ในการพัฒนา
   - CPU: Pentium 4 1.6 GHz
   - RAM 512 MB
   - HDD 40 GB
   - OS: Microsoft Windows 2000 Professional / FreeBSD 4.6 –Release
   - Microsoft Office 97/2000
   - OpenGL
   - MySQL
   - Internet Explorer
2. Disk 3.5” 1.44 MB
3. Printer
4. กระดาษ A4 ขนาด 70-80 แกรม
5. กล้องดิจิตอล
6. กล้องวิดีโอ

เอกสารอ้างอิง

[1] Horton, Thamus R.: “**Traffic Control Theory and Instrumentation”**, Plenum Press, New York (1965).

[2] Drew, Donald R.: **“Traffic Flow Theory and Control”**, McGraw-Hill, New York.

[3] Law, Averill M., and W. David Kelton: **“Simulation Modeling and Analysis”**, McGraw-Hill, New York (1991).

[4] Traffic: **“Traffic in Towns”**, Her Majesty’ Stationery Office (1963).

ภาคผนวก ข

คู่มือการใช้โปรแกรม

ก่อนจะเริ่มใช้โปรแกรมนั้น เราจำเป็นต้องทราบถึงความต้องการของระบบควบคุมและติดตามการจราจรอัตโนมัติเสียก่อน ซึ่งรายละเอียดของคุณสมบัติพื้นฐานสำหรับคอมพิวเตอร์ที่ใช้ในการติดตั้งระบบควบคุมและติดตามการจราจรอัตโนมัติมีดังต่อไปนี้

คุณสมบัติ Hardware

- CPU: Pentium III 733 MHz
- RAM: 128 MB
- HDD: 10 GB

คุณสมบัติ Software

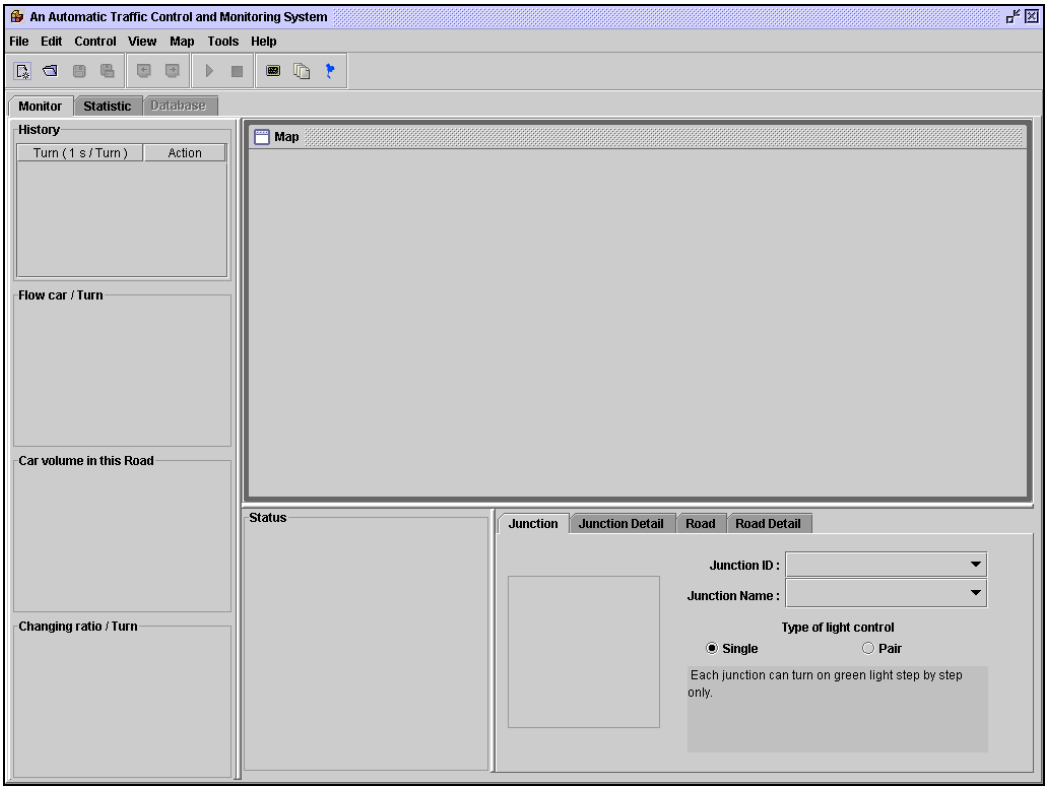
- Operating System:
  - Microsoft Windows 98
  - Microsoft Windows 2000
  - Microsoft Windows ME
  - Microsoft Windows XP
  - Microsoft Windows NT 4.0 + Service Pack 5
  - Linux Red Hat
  - Linux Mandrake 9.0
- JAVA SDK 1.4.0 ใช้สำหรับการ Run JAVA
- JSCI Class ใช้สำหรับช่วยในการแสดงผลในส่วนของกราฟ
- MySQL ใช้สำหรับดูแลและจัดการฐานข้อมูล
- MySQL Driver for JDBC เป็น Driver ที่ช่วยในการติดต่อระหว่าง JAVA กับ MySQL

ในส่วนนี้จะเริ่มเข้าสู่การใช้งานระบบควบคุมและติดตามการจราจรอัตโนมัติโดยเราจะแนะนำให้รู้จักกับส่วนต่างๆของโปรแกรมก่อน แล้วจะแนะนำ การจัดการเบื้องต้นเกี่ยวกับโปรเจค เช่น การสร้างโปรเจคใหม่ การบันทึกโปรเจค การเปิดโปรเจคเดิม การแก้ไขข้อมูลถนนและสี่แยกที่ใช้ในการจำลอง ร่วมไปถึงการแก้ไขข้อมูลต่างๆในฐานข้อมูล สำหรับผู้ดูแลระบบฐานข้อมูล

สิ่งแวดล้อมในการศึกษาสถานการณ์จราจรจำลอง

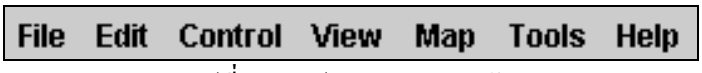
ในหัวข้อนี้จะแนะนำให้รู้จักกับหน้าจอ และส่วนต่างๆของโปรแกรมดังต่อไปนี้

- **หน้าจอหลัก** เมื่อเรียกใช้โปรแกรมระบบควบคุมและติดตามการจราจรอัตโนมัติ จะปรากฏจอภาพดังรูปที่ ข-1 ซึ่งประกอบไปด้วย 5 ส่วนคือ



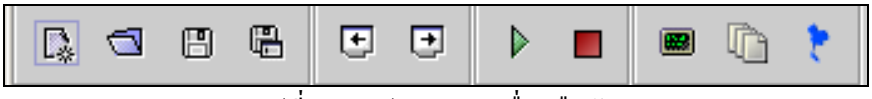
รูปที่ ข-1: รูปภาพของหน้าจอโปรแกรม

- **เมนูหลัก** เป็นส่วนหลักที่ใช้ในการทำงานคำสั่งต่างๆ เกือบทั้งหมดสามารถเรียกใช้ผ่านเมนูได้โดยมีลักษณะดังรูปที่ ข-2



รูปที่ ข-2: รูปภาพของเมนูหลัก

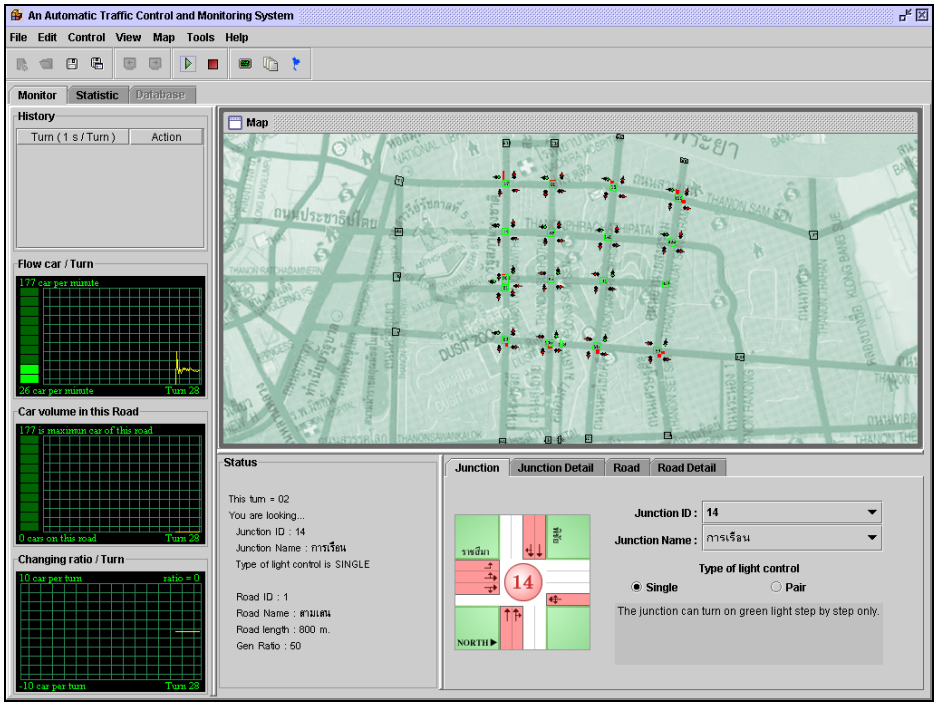
- **แถบเครื่องมือลัด** ประกอบไปด้วยปุ่มของคำสั่งที่ใช้งานบ่อยๆในการทำงาน เราสามารถใช้การคลิกเมาส์ที่ปุ่ม เพื่อเรียกคำสั่งเหล่านั้นออกมาทำงานแทนการเรียกจากเมนูหลัก ซึ่งมีลักษณะดังรูปที่ ข-3



รูปที่ ข-3: รูปภาพของเครื่องมือลัด

- **แท็ปหน้าจอ** ภายในแท็ปหน้าจอจะประกอบไปด้วยหน้าจอการทำงานย่อย โดยเราสามารถคลิกที่ชื่อแท็ปเพื่อเลือกหน้าจอย่อยเหล่านั้นให้ปรากฏขึ้นมา ซึ่งประกอบไปด้วย 4 แท็ปดังต่อไปนี้

1. **แท็ป Monitoring** ภายในแท็ปนี้จะประกอบไปด้วยส่วนต่างๆหลายส่วนที่ใช้ในการควบคุม และแสดงผลของการควบคุมสัญญาณไฟจราจร ซึ่งจะปรับปรุงข้อมูลทุกรอบของการทำงาน (turn) ดังรูปที่ ข-4 ส่วนต่างๆของแท็ป Monitoring มีดังนี้



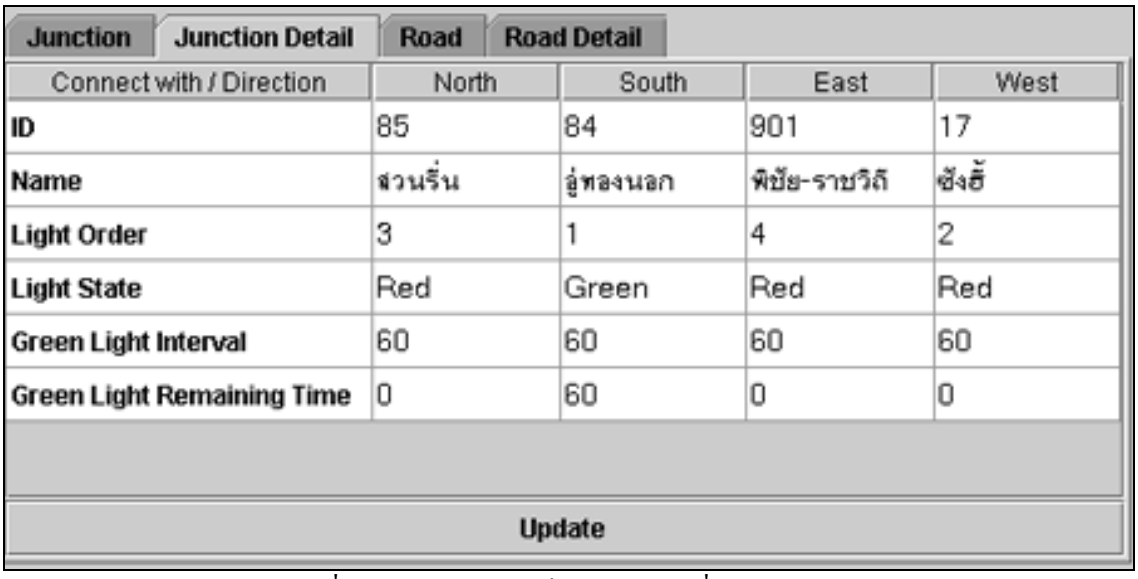
รูปที่ ข-4: รูปภาพของแท็ป Monitoring

ในแท็ป Monitoring นั้นจะมีส่วนประกอบย่อยดังต่อไปนี้

- ตาราง History เป็นตารางที่บอกจำนวนรอบการทำงาน (turn) ที่ได้ทำงานไปทั้งหมด โดยจะแสดงหมายเลขของรอบการทำงานคู่กับการกระทำ (action) ของรอบนั้นว่าได้มาจากการกระทำใด
- กราฟแสดงความเร็วของรถยนต์ที่ออกจากถนนนั้น(Flow car/turn) เป็นกราฟที่แสดงสถิติความเร็วของรถยนต์ที่เคลื่อนตัวออกจากถนนนั้น (คันต่อชั่วโมง) ในแต่ละรอบการทำงาน
- กราฟแสดงปริมาณรถยนต์บนถนน(Car volume) เป็นกราฟที่แสดงสถิติปริมาณรถยนต์ที่มีบนถนนในแต่ละรอบการทำงาน
- กราฟแสดงอัตราการเปลี่ยนแปลงรถยนต์บนถนน เป็นกราฟที่แสดงสถิติอัตราปริมาณรถยนต์ที่เพิ่มขึ้น ลดลงในแต่ละรอบการทำงาน
- แผนที่ถนน ในแผนที่จะแสดงจำนวนรถยนต์บนถนนในรูปของ Scale และสถานะของสัญญาณไฟจราจรของแต่ละถนน
- ช่องสถานะการทำงาน ใช้แสดงรายละเอียดทั้งหมดที่ผู้ใช้กระทำในแต่ละรอบการทำงาน
- แท็ปข้อมูล ซึ่งภายในจะมีแท็ปย่อยจำนวน 4 แท็ปดังต่อไปนี้
  1. แท็ปข้อมูลสี่แยก ภายในแท็ปนี้จะแสดงหมายเลขของสี่แยก ชื่อสี่แยก ชนิดของการเปิดไฟ รวมทั้งแสดงช่องการจราจรของแต่ละถนนที่เชื่อมอยู่กับสี่แยกนั้น ดังรูปที่ ข-5

รูปที่ ข-5: รูปภาพของแท็ปข้อมูลสี่แยก

2. แท็ปรายละเอียดสี่แยก ภายในแท็ปนี้จะแสดงข้อมูลของถนนที่เชื่อมอยู่กับสี่แยกที่ได้เลือกในแท็ปข้อมูลสี่แยก ในทางทิศเหนือ ใต้ ตะวันออก ตะวันตก ดังรูปที่ ข-6

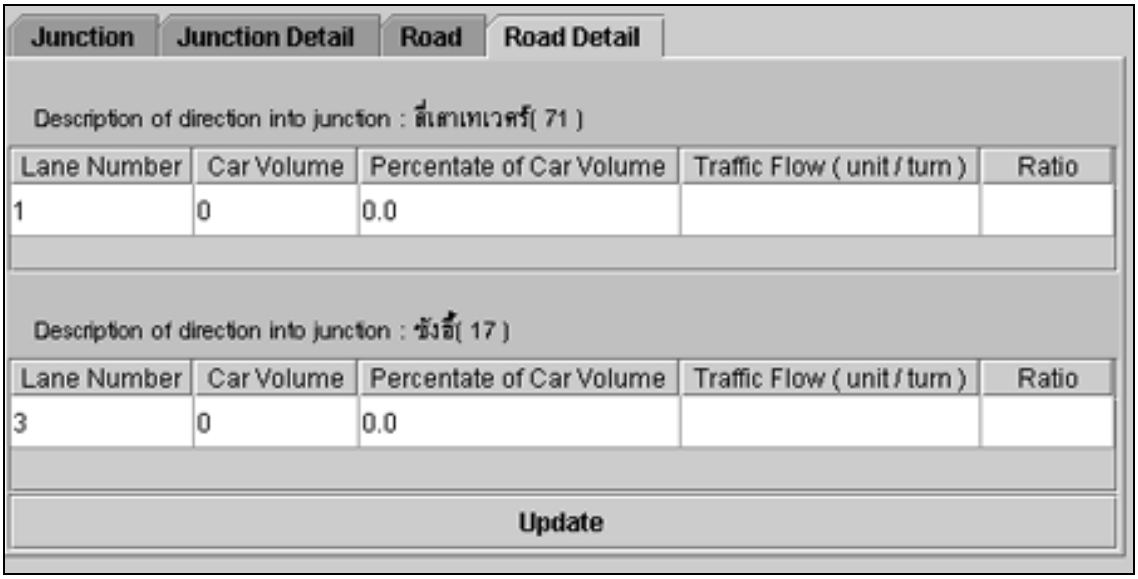


รูปที่ ข-6: รูปภาพของแท็ปรายละเอียดสี่แยก

3. แท็ปข้อมูลถนน ภายในจะแสดงหมายเลขของถนน ชื่อถนน ความยาวถนน และช่องทางการจราจร รวมทั้งช่องอัตราการให้กำเนิดรถยนต์ (คันต่อนาที) ซึ่งจะมีเฉพาะถนนที่มีทิศวิ่งจากจุดขอบเข้าสู่ระบบ ดังรูปที่ ข-7

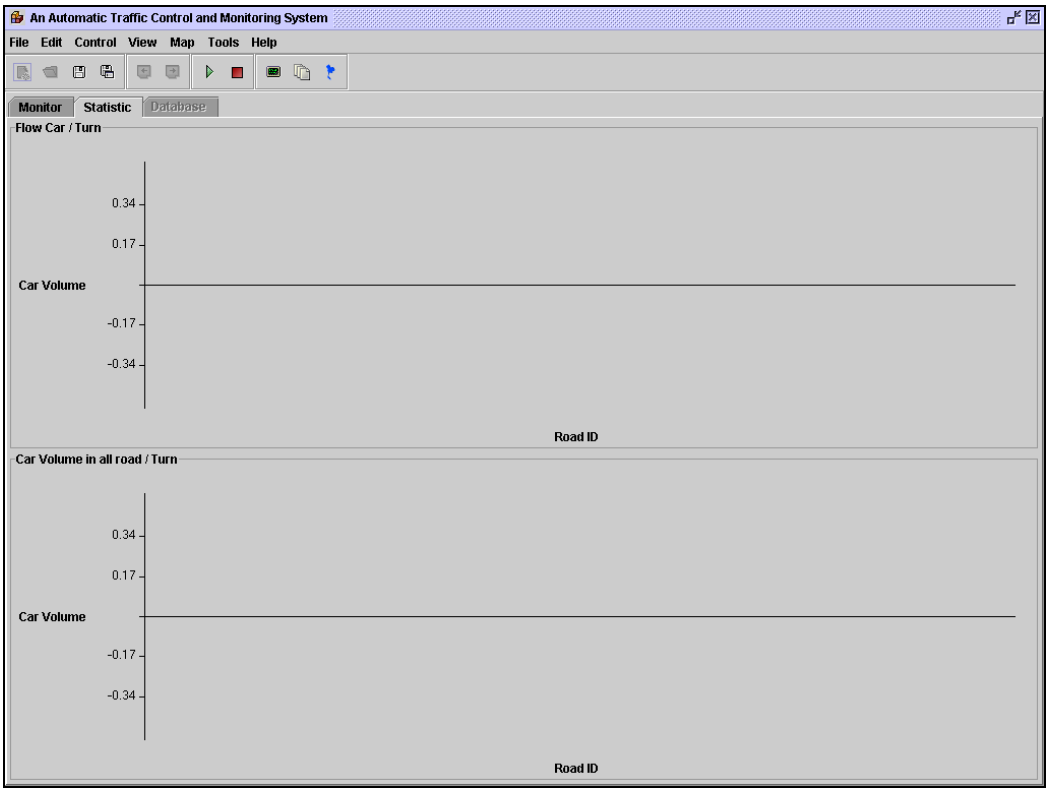
รูปที่ ข-7: รูปภาพของแท็ปข้อมูลถนน

4. แท็ปรายละเอียดถนน ซึ่งมีรายละเอียดดังรูปที่ ข-8



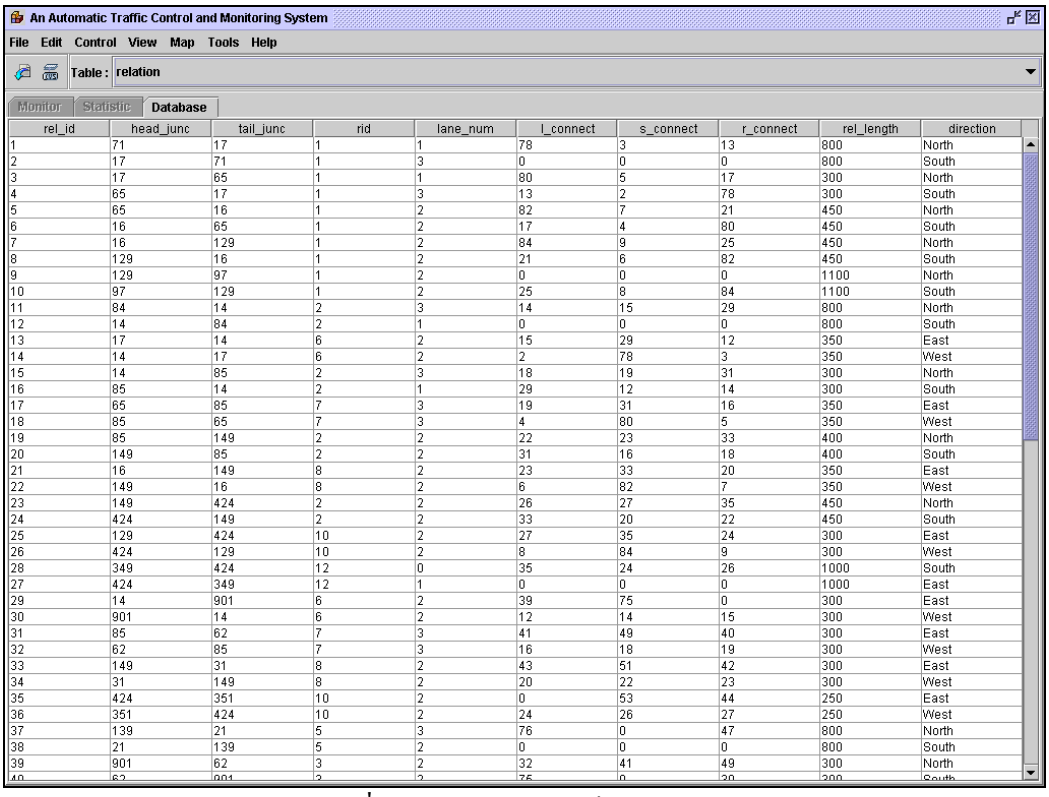
รูปที่ ข-8: รูปภาพของแท็ปรายละเอียดถนน

2. **แท็ป Statistic** จะแสดงกราฟสถิติโดยรวมของทุกถนน ซึ่งจะประกอบไปด้วย 2 กราฟ คือ กราฟ Flow car / turn และ กราฟจำนวนรถยนต์บนถนน ซึ่งจะปรับปรุงข้อมูลทุกรอบของการทำงาน (turn) ดังรูปที่ ข-9



รูปที่ ข-9: รูปภาพของแท็ป Statistic

3. **แท็ป Database** หน้าจอย่อยในแท็ปนี้จะใช้ในการแก้ไขข้อมูลในฐานข้อมูล แต่ผู้ที่จะใช้งานในแท็ปนี้ต้องเป็นผู้ดูแลฐานข้อมูลเท่านั้น ดังรูปที่ ข-10 ซึ่งการใช้งานในแท็ปนี้จะอธิบายภายหลังในหัวข้อ การทำงานและจัดการเกี่ยวกับฐานข้อมูล



rel_id	head_junc	tail_junc	rid	lane_num	l_connect	s_connect	r_connect	rel_length	direction
1	71	17	1	1	78	3	13	800	North
2	17	71	1	3	0	0	0	800	South
3	17	65	1	1	80	5	17	300	North
4	65	17	1	3	13	2	78	300	South
5	65	16	1	2	82	7	21	450	North
6	16	65	1	2	17	4	80	450	South
7	16	129	1	2	84	9	25	450	North
8	129	16	1	2	21	6	82	450	South
9	129	97	1	2	0	0	0	1100	North
10	97	129	1	2	25	8	84	1100	South
11	84	14	2	3	14	15	29	800	North
12	14	84	2	1	0	0	0	800	South
13	17	14	6	2	15	29	12	350	East
14	14	17	6	2	2	78	3	350	West
15	14	85	2	3	18	19	31	300	North
16	85	14	2	1	29	12	14	300	South
17	65	85	7	3	19	31	16	350	East
18	85	65	7	3	4	80	5	350	West
19	85	149	2	2	22	23	33	400	North
20	149	85	2	2	31	16	18	400	South
21	16	149	8	2	23	33	20	350	East
22	149	16	8	2	6	82	7	350	West
23	149	424	2	2	26	27	35	450	North
24	424	149	2	2	33	20	22	450	South
25	129	424	10	2	27	35	24	300	East
26	424	129	10	2	8	84	9	300	West
28	349	424	12	0	35	24	26	1000	South
27	424	349	12	1	0	0	0	1000	East
29	14	901	6	2	39	75	0	300	East
30	901	14	6	2	12	14	15	300	West
31	85	62	7	3	41	49	40	300	East
32	62	85	7	3	16	18	19	300	West
33	149	31	8	2	43	51	42	300	East
34	31	149	8	2	20	22	23	300	West
35	424	351	10	2	0	53	44	250	East
36	351	424	10	2	24	26	27	250	West
37	139	21	5	3	76	0	47	800	North
38	21	139	5	2	0	0	0	800	South
39	901	62	3	2	32	41	49	300	North

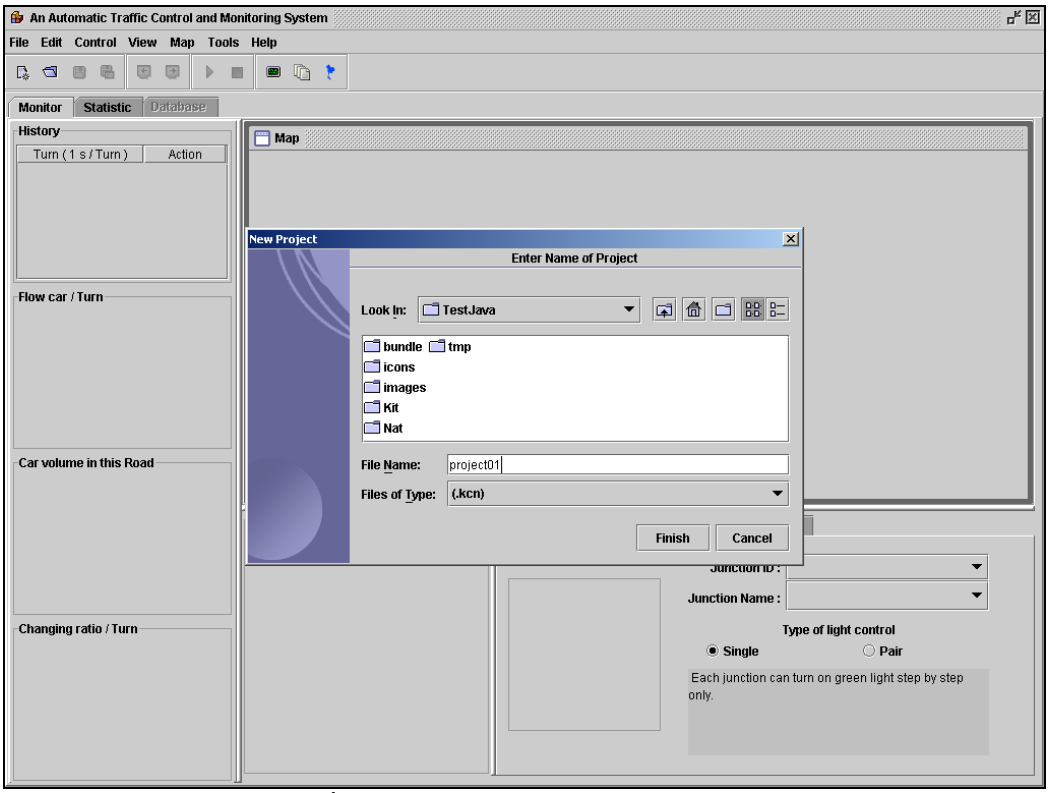
รูปที่ ข-10: รูปภาพของแท็ป Database

การทำงานและการจัดการกับโปรเจค

ไฟล์โปรเจค คือ ไฟล์ที่ใช้สำหรับเก็บรวบรวมข้อมูลต่างๆ ในการทำงานแต่ละรอบการทำงานเข้าไว้ด้วยกัน เพื่อใช้ในการจำลองสถานการณ์จราจร ในการทำงานกับโปรเจค เราสามารถเรียกคำสั่งต่างๆ จากเมนูหลักได้ทั้งหมด หรือเรียกจากแถบเครื่องมือลัด ซึ่งเราจะอธิบายการทำงานแต่ละอย่างตามหัวข้อต่อไปนี้

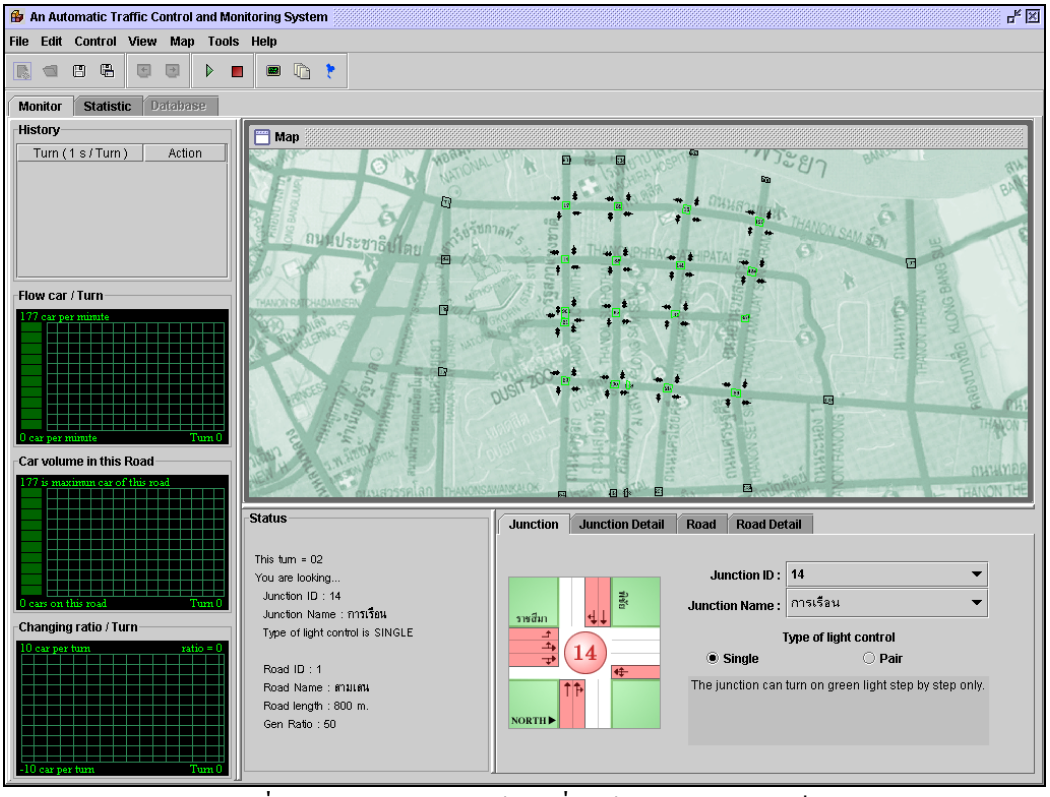
- **การสร้างโปรเจคใหม่**

เราสามารถสร้างโปรเจคขึ้นมาใหม่ได้โดยเลือกที่หัวข้อ File แล้วเรียกคำสั่งNew จากเมนูหลัก หรือกดรูป ที่แถบเมนูลัด เมื่อเรียกคำสั่งสร้างโปรเจคแล้วจะปรากฏไดอะล็อกขึ้นมาให้ใส่ชื่อโปรเจคพร้อมทั้งระบุโฟล์เดอร์ที่จะจัดเก็บดังรูปที่ ข-11



รูปที่ ข-11: รูปภาพของการสร้างโปรเจคใหม่

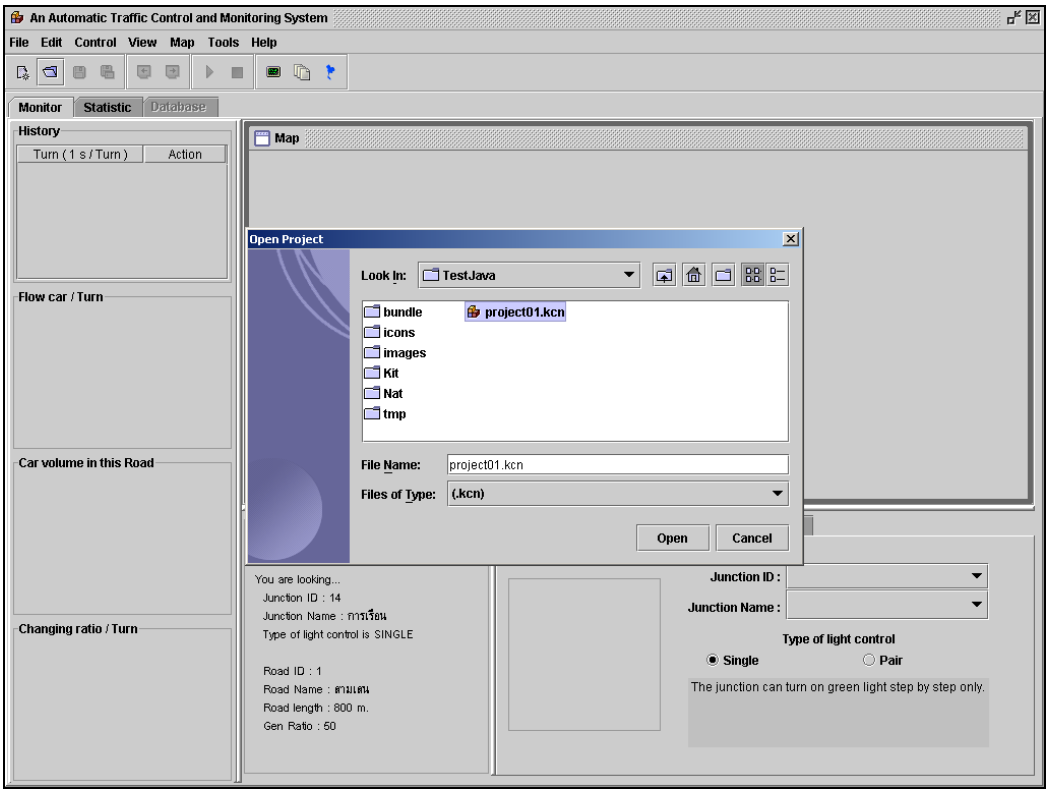
เมื่อสร้างโปรเจคใหม่เสร็จแล้วจะได้ไฟล์นามสกุล kcn และปรากฏหน้าจอขึ้นดังรูปที่ ข-12



รูปที่ ข-12: รูปภาพของหน้าจอเมื่อสร้างโปรเจคใหม่เสร็จ

- **การเปิดไฟล์เจค**

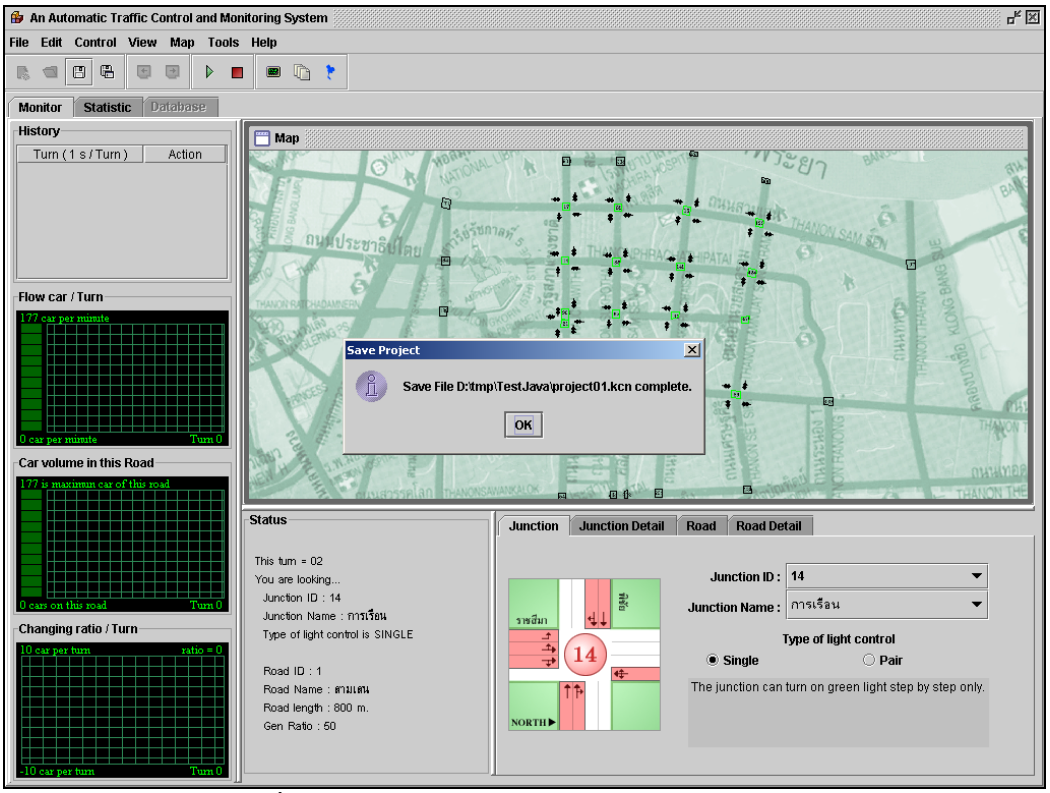
เราสามารถเปิดไฟล์ที่เราได้สร้างแล้วโดยเลือกที่หัวข้อ File แล้วเรียกคำสั่งOpen จากเมนูหลัก หรือกดรูป ที่แถบเมนูลัด เมื่อเรียกคำสั่งเปิดโปรเจคแล้วจะปรากฏไดอะล็อกขึ้นมาให้เลือกโปรเจค ดังรูปที่ ข-13



รูปที่ ข-13: รูปภาพของการเปิดโปรเจคใหม่

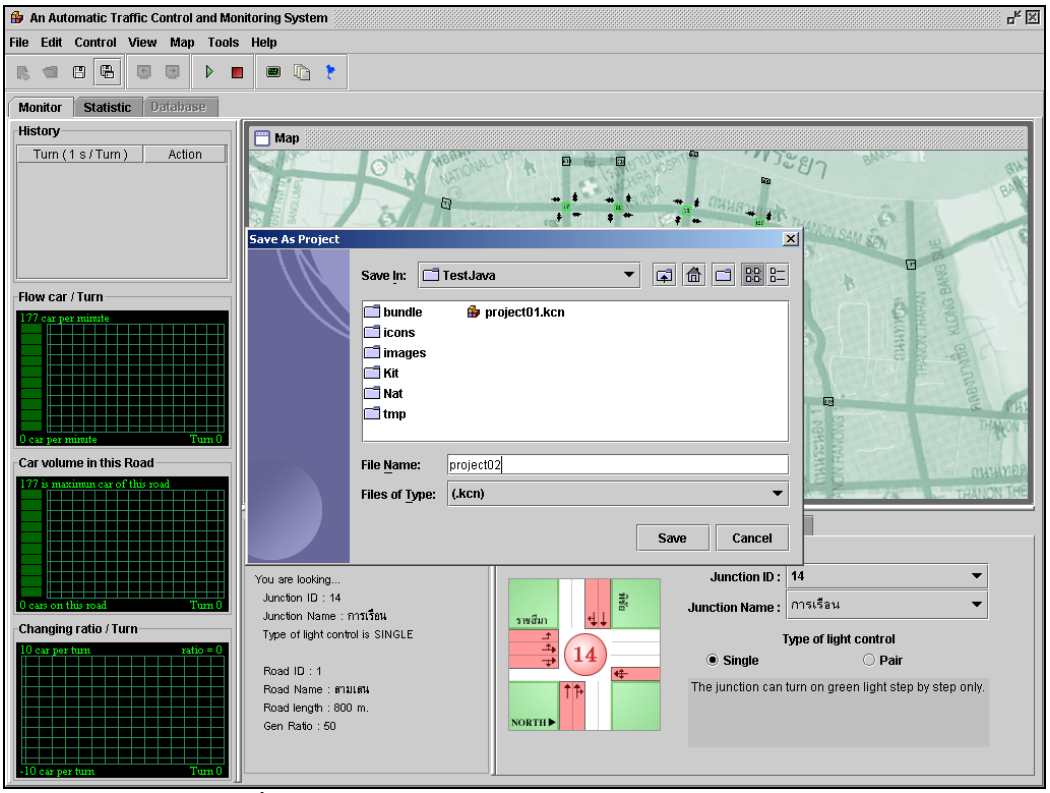
- **การบันทึกโปรเจค** การบันทึกข้อมูลสามารถทำได้ทั้งหมด 2 ลักษณะได้แก่

1. **Save** เป็นการสั่งให้บันทึกข้อมูล โดยจะเป็นการใช้ชื่อไฟล์ที่ได้จากการสร้างโปรเจคหรือการเปิดไฟล์โปรเจค ในการบันทึกข้อมูล เราสามารถทำการสั่งให้บันทึกข้อมูลด้วยวิธีการนี้โดยเลือกที่หัวข้อ File แล้วเรียกคำสั่งSave จากเมนูหลัก หรือกดรูป ที่แถบเมนูลัด เมื่อโปรแกรมทำการบันทึกข้อมูลเรียบร้อยแล้วจะปรากฏไดอะล็อกขึ้น ดังแสดงอยู่ในรูปที่ ข-14



รูปที่ ข-14: รูปภาพของการบันทึกไฟล์ข้อมูลแบบ Save

2. **SaveAs** เป็นการบันทึกไฟล์ข้อมูลที่ถูกแก้ไขด้วยชื่อไฟล์ใหม่ โดยที่จะมีไดอะล็อกให้เราใส่ชื่อไฟล์ที่เราต้องการจะเก็บข้อมูลโดยเลือกที่หัวข้อ File แล้วเรียกคำสั่ง SaveAs จากเมนูหลัก หรือกดรูป ที่แถบเมนูลัด ดังรูปที่ ข-15



รูปที่ ข-15: รูปภาพของการบันทึกไฟล์ข้อมูลแบบ Save As

- **การปิดโปรเจค** เราสามารถปิดโปรเจค เมื่อต้องการจะสร้างโปรเจค หรือเปิดโปรเจคใหม่ โดยเลือกที่หัวข้อ File แล้วเรียกคำสั่ง Close จากเมนูหลัก ในการปิดโปรเจคนี้จะมีการบันทึกข้อมูลให้อัตโนมัติ ในกรณีที่ยังไม่มีการบันทึกโปรเจคมาก่อน
- **การเริ่มและหยุดจำลองสถานการณ์จราจร**

  เราจะเริ่มจำลองสถานการณ์จราจรได้โดยเลือกที่หัวข้อ Control แล้วเรียกคำสั่งExecute จากเมนูหลัก หรือกดรูป ▶ ที่แถบเมนูลัด ระบบจะทำการจำลองสถานการณ์จราจรอัตโนมัติจนกว่าจะมีการสั่งหยุดการจำลองสถานการณ์โดยเลือกที่หัวข้อ Control แล้วเรียกคำสั่งStop จากเมนูหลัก หรือกดรูป ■ ที่แถบเมนูลัด และทุกรอบการทำงานของระบบจำลองจะมีการปรับปรุงข้อมูลในแต่ละแท็ปข้อมูล กราฟทางสถิติต่างๆ รวมทั้งการแสดงผลในแผนที่อีกด้วย
- **การใช้งานตาราง History**

  ตาราง History เป็นตารางที่บอกจำนวนรอบการทำงาน (turn) ที่ได้ทำงานไปทั้งหมด โดยจะแสดงเบอร์รอบการทำงานคู่กับการกระทำ(action)ของรอบนั้นว่าได้มาจากการกระทำใด ซึ่งเราสามารถที่กลับดูข้อมูลต่างๆของรอบการทำงานที่ผ่านมาแล้วโดยการคลิกที่เบอร์รอบการทำงานที่ต้องการกลับไป ซึ่งหน้าจอการทำงานต่างๆจะ

แสดงข้อมูลของรอบการทำงานนั้นอย่างอัตโนมัติ หรือเลือกหัวข้อ Edit แล้วเรียกคำสั่ง Previous Turn จากเมนูหลัก หรือกดรูป (Previous)เพื่อกลับไปยังรอบการทำงานที่ก่อนหน้า และเลือกหัวข้อ Edit แล้วเรียกคำสั่งNext Turn จากเมนูหลัก หรือกดรูป (Next) เพื่อไปยังรอบการทำงานที่ถัดจากรอบการทำงานนั้น เช่นถ้าปัจจุบันอยู่ที่รอบการทำงานที่5 เมื่อกดปุ่ม Previous 2 ครั้ง จะกลับไปยังรอบการทำงานที่ 3 และเมื่อกดปุ่ม Next 1 ครั้งจะกลับมาที่รอบการทำงานที่ 4 เป็นต้น

- **การใช้งานแผนที่ถนน**

  ในแผนที่จะแสดงจำนวนรถยนต์บนถนนในรูปของ Scale และสถานะของสัญญาณไฟจราจรของแต่ละถนน ซึ่งเราสามารถขยายแผนที่ออก (Zoom in) ขยายแผนที่เข้า (Zoom Out) และแสดงแผนที่ในขนาดที่เหมาะสมได้ (Fit Map) ดังนี้

- **ขยายแผนที่ออก(Zoom in)**

  เลือกหัวข้อ Map แล้วเรียกคำสั่งZoom-In จากเมนูหลัก หรือกดรูปเพื่อขยายแผนที่ออก

- **ขยายแผนที่เข้า(Zoom Out)**

  เลือกหัวข้อ Map แล้วเรียกคำสั่งZoom-Out จากเมนูหลัก หรือกดรูปเพื่อขยายแผนที่เข้า

- **แสดงแผนที่ในขนาดที่เหมาะสม(Fit Map)**

  เลือกหัวข้อ Map แล้วเรียกคำสั่งFit Map จากเมนูหลัก หรือกดรูป เพื่อให้แผนที่แสดงในขนาดที่เหมาะสม

การแก้ไขข้อมูลในแท็ปข้อมูล

การที่จะแก้ไขข้อมูลในแท็ปข้อมูลได้นั้น ตัวระบบต้องอยู่ในสถานะหยุดการทำงานเท่านั้นจึงสามารถแก้ไขข้อมูลได้ ซึ่งข้อมูลที่สามารถแก้ไขได้ของแต่ละแท็ปมีดังต่อไปนี้

- **แท็ปข้อมูลสี่แยก** ภายในแท็ปนี้เราสามารถเปลี่ยนลักษณะการเปิดสัญญาณไฟจราจรของสี่แยกที่สนใจได้ 2 ลักษณะคือ

  1. **การเปิดแบบ SINGLE** คือในช่วงเวลาหนึ่งจะเปิดสัญญาณไฟเขียวให้แก่ถนนเดียวเท่านั้น
  2. **การเปิดแบบ PAIR** คือในช่วงเวลาหนึ่งจะเปิดสัญญาณไฟเขียว 2 ถนนที่วิ่งสวนทางกันเท่านั้น

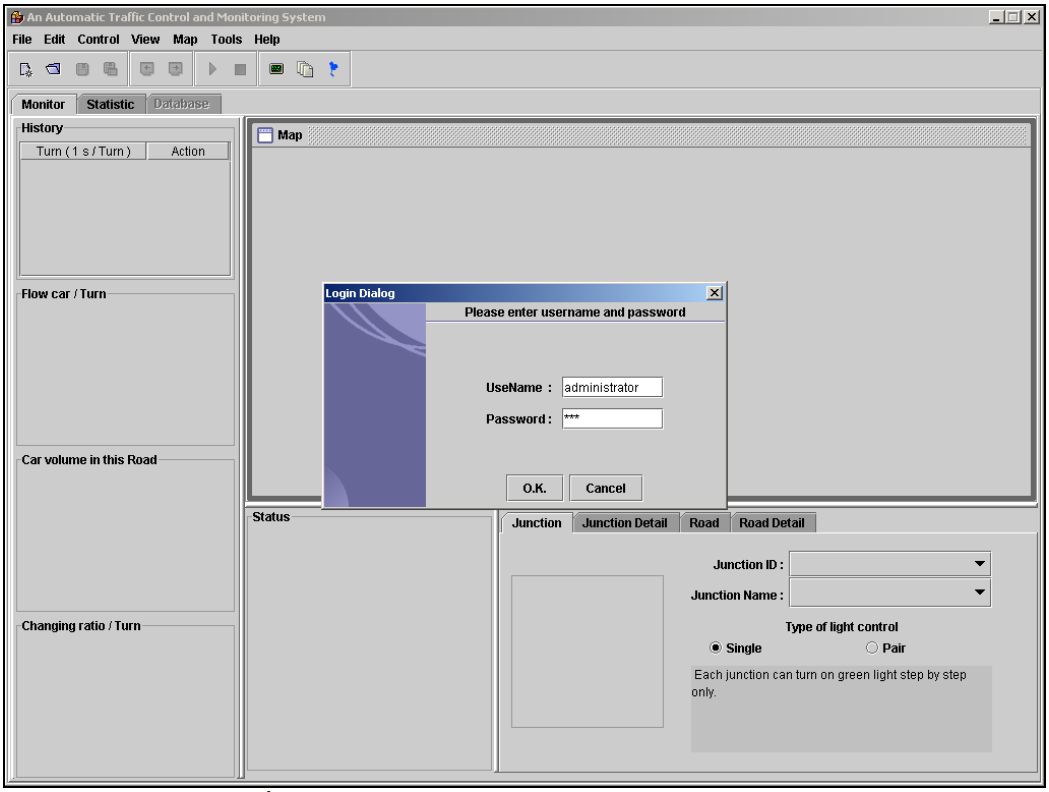
  ซึ่งถ้าต้องการเปลี่ยนลักษณะการเปิดสัญญาณไฟของสี่แยกเป็นแบบใด ก็ให้คลิกที่ลักษณะนั้น โปรแกรมจะทำการเปลี่ยนค่าข้อมูลของสี่แยกนั้นให้อัตโนมัติ

- **แท็ปรายละเอียดสี่แยก** ข้อมูลภายในแท็ปจะเป็นข้อมูลของถนนที่มีความสัมพันธ์กับข้อมูลของสี่แยกที่อยู่ในแท็ปข้อมูลสี่แยก ซึ่งข้อมูลที่สามารถแก้ไขได้มีดังนี้
    1. ลำดับการเปิดสัญญาณไฟจราจร(Light Order)
    2. สถานะของสัญญาณไฟจราจร(Light State)
    3. ระยะเวลาในการเปิดสัญญาณไฟเขียว(Green Light Interval)
    4. ระยะเวลาที่เหลือในการเปิดไฟเขียว(Green Light Remaining Time)

  เมื่อทำการแก้ไขข้อมูลเรียบร้อยแล้วต้องกดปุ่ม Update ที่อยู่ข้างล่างตารางเพื่อให้โปรแกรมทำข้อมูลที่เปลี่ยนแปลงไปใช้งาน
- แท็ปข้อมูลถนน ข้อมูลที่สามารถแก้ไขได้ภายในแท็ปคืออัตราการให้กำเนิดรถยนต์(คันต่อนาที) แต่จะแก้ไขได้เฉพาะถนนที่มีทิศวิ่งจากจุดขอบเข้าสู่ระบบเท่านั้น เมื่อทำการแก้ไขข้อมูลเรียบร้อยแล้วต้องกดปุ่ม Update ที่อยู่ข้างล่างเพื่อให้โปรแกรมทำข้อมูลที่เปลี่ยนแปลงไปใช้งาน
- แท็ปรายละเอียดถนน ในแท็ปนี้จะไม่สามารถแก้ไขข้อมูลใดๆได้ทั้งสิ้น

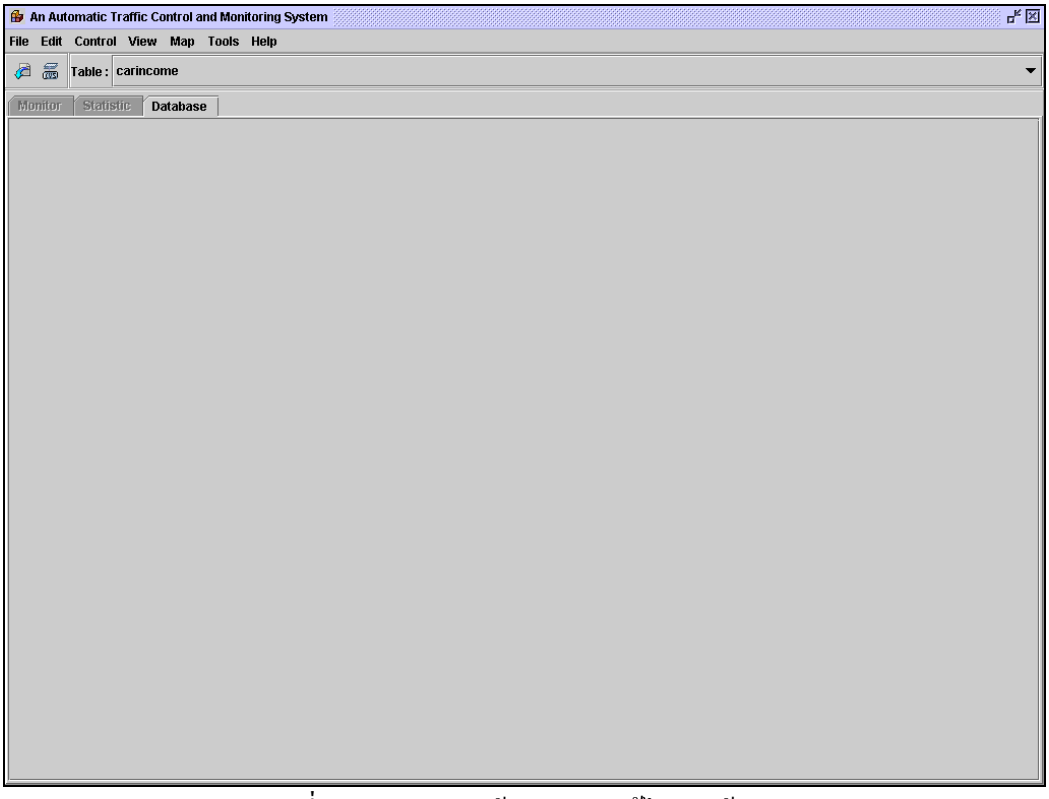
การทำงานและจัดการเกี่ยวกับฐานข้อมูล

ในการทำงานกับฐานข้อมูลนั้นผู้ที่จะเข้าใช้งานต้องเป็นผู้ดูแลฐานข้อมูลเท่านั้น ซึ่งก่อนการเข้าใช้งานต้องLogin เข้าสู่ระบบก่อนโดยเลือกที่หัวข้อ Tool >Database แล้วเรียกคำสั่งLogin จากเมนูหลัก เมื่อเรียกคำสั่งสร้างโปรเจคแล้วจะปรากฏไดอะล็อกขึ้นมาให้ใส่ชื่อผู้ใช้ และรหัสประจำตัว ดังรูปที่ ข-16



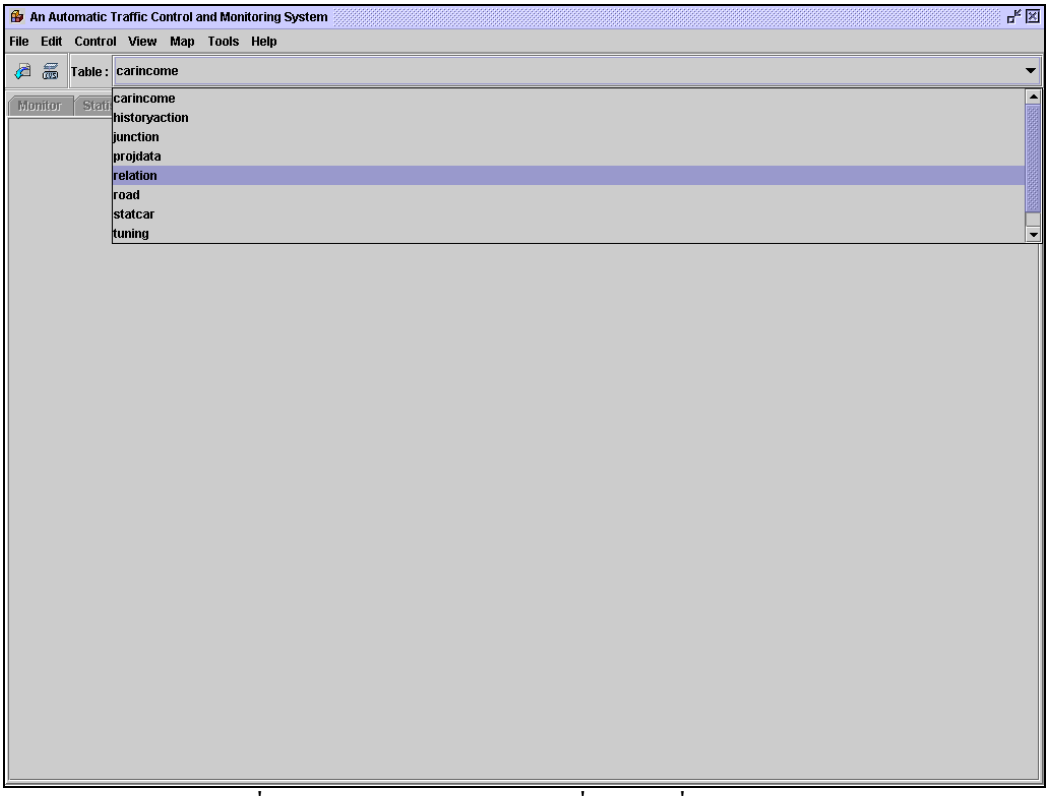
รูปที่ ข-16: รูปภาพของLogin เข้าสู่ระบบแก้ไขฐานข้อมูล

เมื่อการใส่ชื่อผู้ใช้ และรหัสประจำตัวถูกต้อง จะปรากฏหน้าจอขึ้นดังรูปที่ ข-17



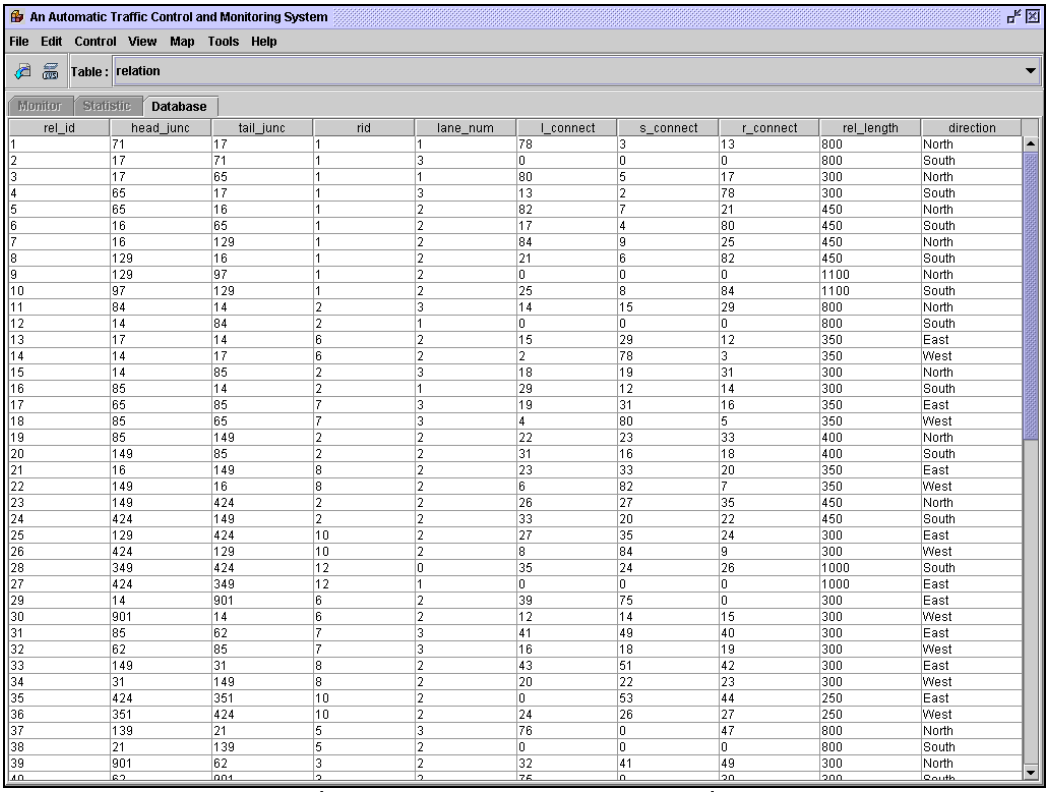
รูปที่ ข-17: รูปภาพหน้าจอระบบแก้ไขฐานข้อมูล

- **การแก้ไขข้อมูลในฐานข้อมูล** สามารถแก้ไขได้จากตารางโดยตรง โดยเลือกชื่อตารางที่ต้องการแก้ไขจากรายชื่อตารางทางด้านบนดังรูปที่ ข-18



รูปที่ ข-18: รูปภาพแสดงการเลือกชื่อตารางที่ต้องการแก้ไข

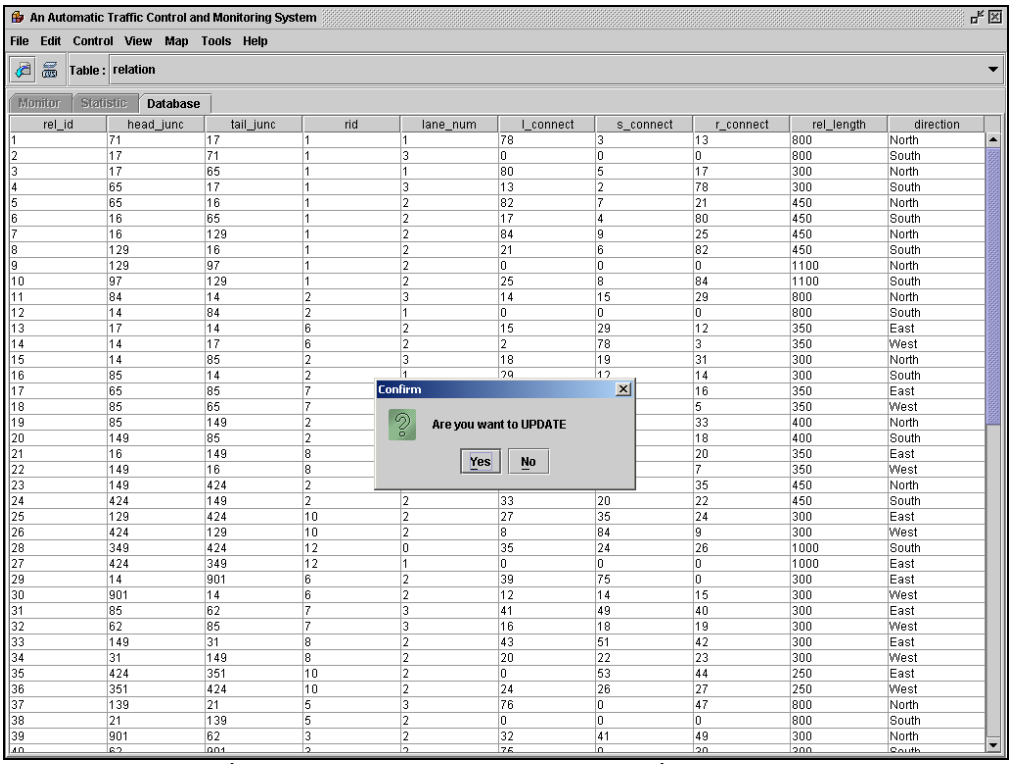
และเมื่อเลือกชื่อตารางแล้วจะปรากฏตารางขึ้นดังรูปที่ ข-19



rel_id	head_junc	tail_junc	rid	lane_num	l_connect	s_connect	r_connect	rel_length	direction
1	71	17	1	1	78	3	13	800	North
2	17	71	1	3	0	0	0	800	South
3	17	65	1	1	80	5	17	300	North
4	65	17	1	3	13	2	78	300	South
5	65	16	1	2	82	7	21	450	North
6	16	65	1	2	17	4	80	450	South
7	16	129	1	2	84	9	25	450	North
8	129	16	1	2	21	6	82	450	South
9	129	97	1	2	0	0	0	1100	North
10	97	129	1	2	25	8	84	1100	South
11	84	14	2	3	14	15	29	800	North
12	14	84	2	1	0	0	0	800	South
13	17	14	6	2	15	29	12	350	East
14	14	17	6	2	2	78	3	350	West
15	14	85	2	3	18	19	31	300	North
16	85	14	2	1	29	12	14	300	South
17	65	85	7	3	19	31	16	350	East
18	85	65	7	3	4	80	5	350	West
19	85	149	2	2	22	23	33	400	North
20	149	85	2	2	31	16	18	400	South
21	16	149	8	2	23	33	20	350	East
22	149	16	8	2	6	82	7	350	West
23	149	424	2	2	26	27	35	450	North
24	424	149	2	2	33	20	22	450	South
25	129	424	10	2	27	35	24	300	East
26	424	129	10	2	8	84	9	300	West
28	349	424	12	0	35	24	26	1000	South
27	424	349	12	1	0	0	0	1000	East
29	14	901	6	2	39	75	0	300	East
30	901	14	6	2	12	14	15	300	West
31	85	62	7	3	41	49	40	300	East
32	62	85	7	3	16	18	19	300	West
33	149	31	8	2	43	51	42	300	East
34	31	149	8	2	20	22	23	300	West
35	424	351	10	2	0	53	44	250	East
36	351	424	10	2	24	26	27	250	West
37	139	21	5	3	76	0	47	800	North
38	21	139	5	2	0	0	0	800	South
39	901	62	3	2	32	41	49	300	North

รูปที่ ข-19: รูปภาพแสดงข้อมูลในตารางที่เลือก

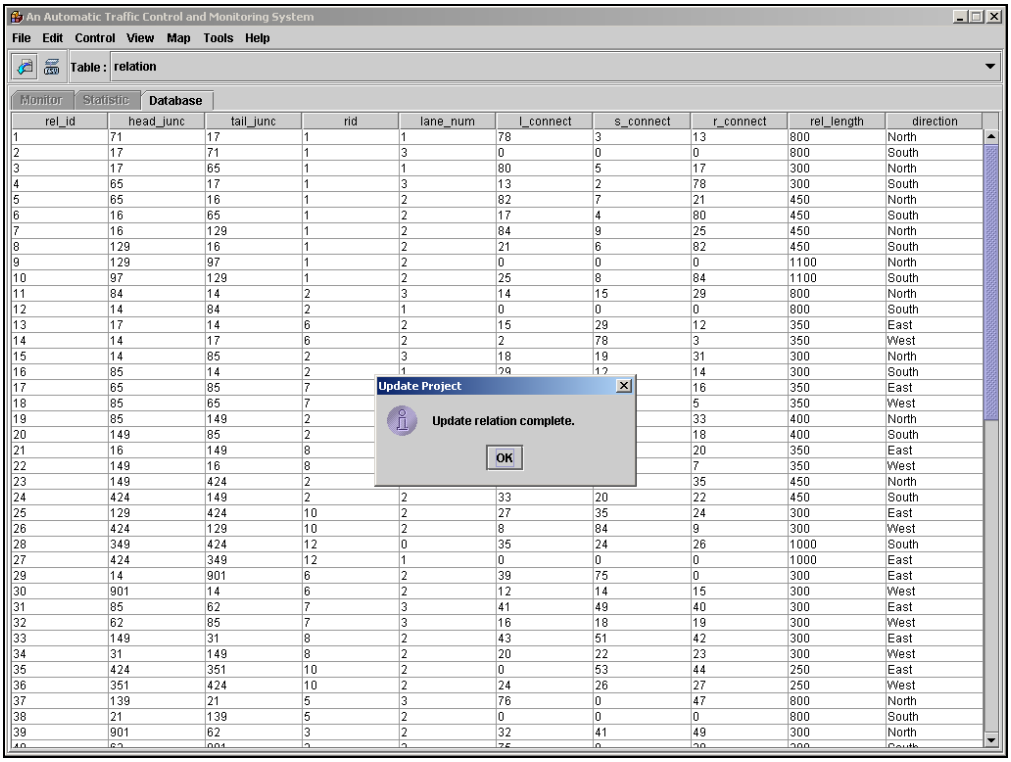
ซึ่งเมื่อเราแก้ไขข้อมูลในตารางเสร็จเรียบร้อยแล้ว ต้องโดยเลือกที่หัวข้อ Tool>Database แล้วเรียกคำสั่งUpdate จากเมนูหลัก หรือกดรูป ที่แถบเมนูลัด เพื่อให้โปรแกรมทำการบันทึกข้อมูลลงฐานข้อมูล และเมื่อเรียกคำสั่งแล้วจะปรากฏไดอะล็อกขึ้น ในการยืนยันความต้องการในการเปลี่ยนแปลงข้อมูล ดังรูปที่ ข-20



rel_id	head_junc	tail_junc	rid	lane_num	l_connect	s_connect	r_connect	rel_length	direction
1	71	17	1	1	78	3	13	800	North
2	17	71	1	3	0	0	0	800	South
3	17	65	1	1	80	5	17	300	North
4	65	17	1	3	13	2	78	300	South
5	65	16	1	2	82	7	21	450	North
6	16	65	1	2	17	4	80	450	South
7	16	129	1	2	84	9	25	450	North
8	129	16	1	2	21	6	82	450	South
9	129	97	1	2	0	0	0	1100	North
10	97	129	1	2	25	8	84	1100	South
11	84	14	2	3	14	15	29	800	North
12	14	84	2	1	0	0	0	800	South
13	17	14	6	2	15	29	12	350	East
14	14	17	6	2	2	78	3	350	West
15	14	85	2	3	18	19	31	300	North
16	85	14	2	1	29	12	14	300	South
17	65	85	7	3	76	0	16	350	East
18	85	65	7	3	76	0	5	350	West
19	85	149	2	3	18	19	33	400	North
20	149	85	2	3	19	18	18	400	South
21	16	149	8	2	43	51	20	350	East
22	149	16	8	2	51	43	7	350	West
23	149	424	2	3	18	19	35	450	North
24	424	149	2	2	33	20	22	450	South
25	129	424	10	2	27	35	24	300	East
26	424	129	10	2	8	84	9	300	West
28	349	424	12	0	35	24	26	1000	South
27	424	349	12	1	0	0	0	1000	East
29	14	901	6	2	39	75	0	300	East
30	901	14	6	2	12	14	15	300	West
31	85	62	7	3	41	49	40	300	East
32	62	85	7	3	16	18	19	300	West
33	149	31	8	2	43	51	42	300	East
34	31	149	8	2	20	22	23	300	West
35	424	351	10	2	0	53	44	250	East
36	351	424	10	2	24	26	27	250	West
37	139	21	5	3	76	0	47	800	North
38	21	139	5	2	0	0	0	800	South
39	901	62	3	2	32	41	49	300	North

รูปที่ ข-20: รูปภาพแสดงการยืนยันในการเปลี่ยนแปลงข้อมูล

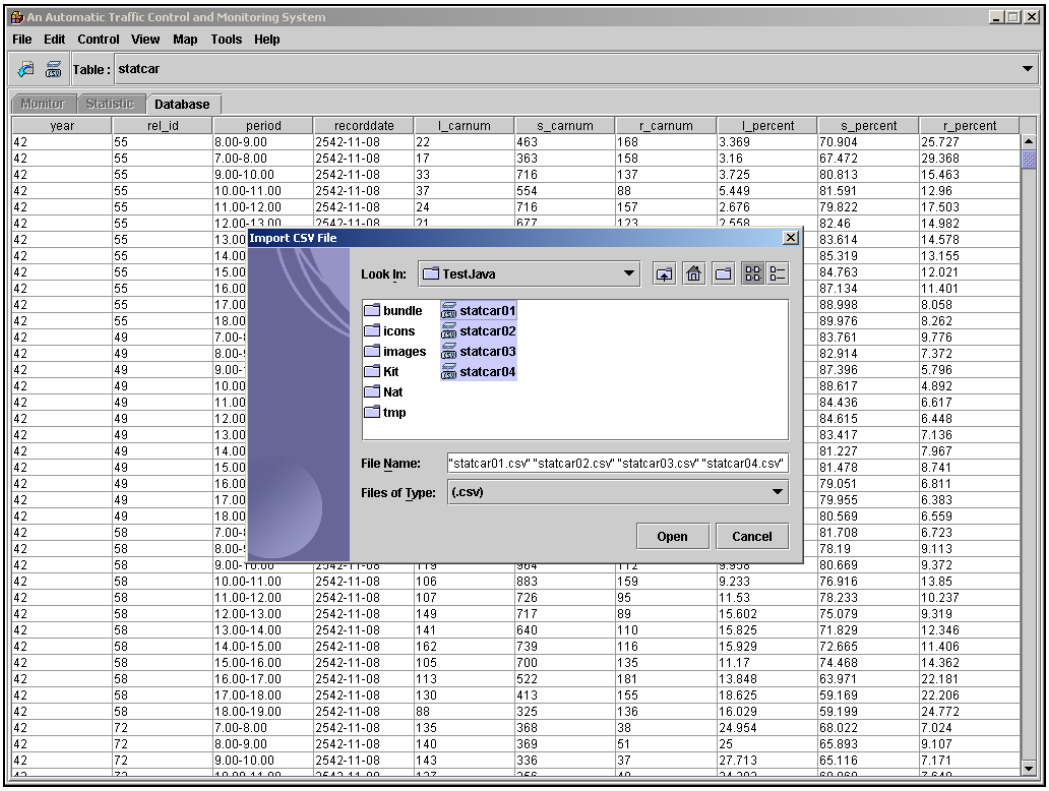
เมื่อโปรแกรมทำการบันทึกข้อมูลลงฐานข้อมูลเสร็จเรียบร้อยแล้วจะแสดงไดอะล็อกดังรูปที่ ข-21



rel_id	head_junc	tail_junc	rid	lane_num	l_connect	s_connect	r_connect	rel_length	direction
1	71	17	1	1	78	3	13	800	North
2	17	71	1	3	0	0	0	800	South
3	17	65	1	1	80	5	17	300	North
4	65	17	1	3	13	2	78	300	South
5	65	16	1	2	82	7	21	450	North
6	16	65	1	2	17	4	80	450	South
7	16	129	1	2	84	9	25	450	North
8	129	16	1	2	21	6	82	450	South
9	129	97	1	2	0	0	0	1100	North
10	97	129	1	2	25	8	84	1100	South
11	84	14	2	3	14	15	29	800	North
12	14	84	2	1	0	0	0	800	South
13	17	14	6	2	15	29	12	350	East
14	14	17	6	2	2	78	3	350	West
15	14	85	2	3	18	19	31	300	North
16	85	14	2	1	29	12	14	300	South
17	65	85	7	3	76	0	16	350	East
18	85	65	7	3	76	0	5	350	West
19	85	149	2	3	18	19	33	400	North
20	149	85	2	3	19	18	18	400	South
21	16	149	8	2	43	51	20	350	East
22	149	16	8	2	51	43	7	350	West
23	149	424	2	3	18	19	35	450	North
24	424	149	2	2	33	20	22	450	South
25	129	424	10	2	27	35	24	300	East
26	424	129	10	2	8	84	9	300	West
28	349	424	12	0	35	24	26	1000	South
27	424	349	12	1	0	0	0	1000	East
29	14	901	6	2	39	75	0	300	East
30	901	14	6	2	12	14	15	300	West
31	85	62	7	3	41	49	40	300	East
32	62	85	7	3	16	18	19	300	West
33	149	31	8	2	43	51	42	300	East
34	31	149	8	2	20	22	23	300	West
35	424	351	10	2	0	53	44	250	East
36	351	424	10	2	24	26	27	250	West
37	139	21	5	3	76	0	47	800	North
38	21	139	5	2	0	0	0	800	South
39	901	62	3	2	32	41	49	300	North

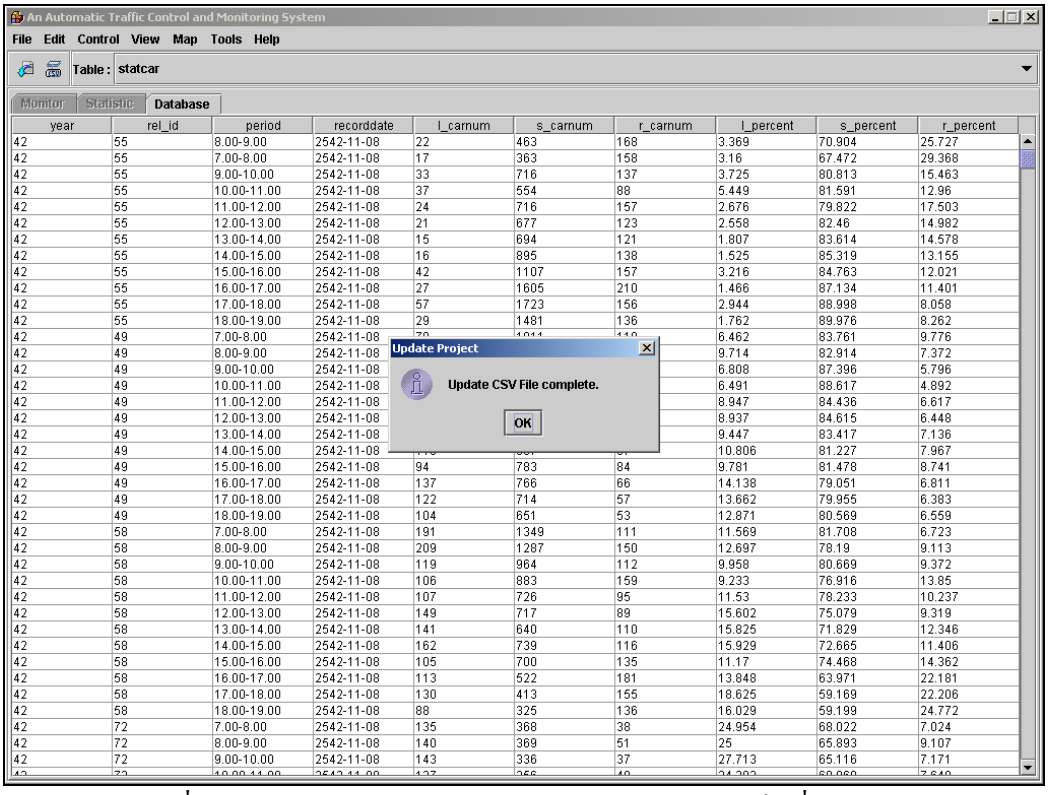
รูปที่ ข-21: รูปภาพแสดงข้อมูลถูกบันทึกลงฐานข้อมูลเป็นที่เรียบร้อย

- **การนำเข้า CSV File** เป็นการนำเอาข้อมูลปริมาณการจราจรของทางสำนักการจราจรและขนส่งที่ถูกบันทึกให้อยู่ในรูปแบบเอกสารที่มีนามสกุล CSV แล้ว (เป็น Format หนึ่งใน Excel) มาแปลงเป็นข้อมูลในตาราง Statcar (รูปแบบ CSV อธิบายไปแล้วในบทที่ 4 หัวข้อวิธีการแปลงรูปแบบข้อมูลที่ได้จากสำนักการจราจรและขนส่งให้อยู่ในรูปแบบของฐานข้อมูลที่ใช้บนระบบ) ซึ่งสามารถทำได้โดยการเลือกชื่อไฟล์ที่ต้องการจะแปลง ดังรูปที่ ข-22



รูปที่ ข-22: รูปภาพแสดงการเลือกชื่อไฟล์ที่ต้องการจะแปลง

เมื่อทำการเลือกไฟล์เสร็จแล้วก็ให้กดปุ่ม Open เพื่อเป็นการแปลงข้อมูลที่ได้เลือกเอาไว้ ถ้า CSV File ที่ทำการแปลงนั้นไม่มีข้อผิดพลาด หน้าจอจะแสดงไดอะล็อกดังรูปที่ ข-23

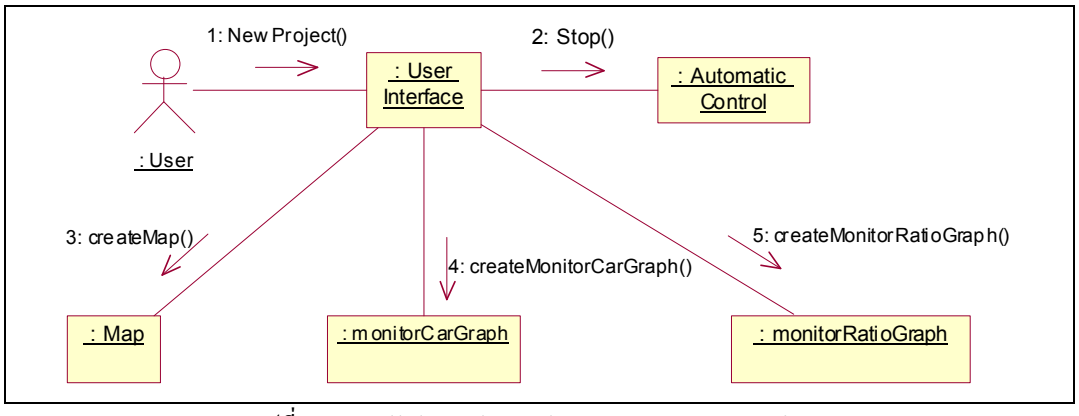


year	rel_id	period	recorddate	l_carnum	s_carnum	r_carnum	l_percent	s_percent	r_percent
42	55	8.00-9.00	2542-11-08	22	463	168	3.369	70.904	25.727
42	55	7.00-8.00	2542-11-08	17	363	158	3.16	67.472	29.368
42	55	9.00-10.00	2542-11-08	33	716	137	3.725	80.813	15.463
42	55	10.00-11.00	2542-11-08	37	554	88	5.449	81.591	12.96
42	55	11.00-12.00	2542-11-08	24	716	157	2.676	79.822	17.503
42	55	12.00-13.00	2542-11-08	21	677	123	2.558	82.46	14.982
42	55	13.00-14.00	2542-11-08	15	694	121	1.807	83.614	14.578
42	55	14.00-15.00	2542-11-08	16	895	138	1.525	85.319	13.155
42	55	15.00-16.00	2542-11-08	42	1107	157	3.216	84.763	12.021
42	55	16.00-17.00	2542-11-08	27	1605	210	1.466	87.134	11.401
42	55	17.00-18.00	2542-11-08	57	1723	156	2.944	88.998	8.058
42	55	18.00-19.00	2542-11-08	29	1481	136	1.762	89.976	8.262
42	49	7.00-8.00	2542-11-08				6.462	83.761	9.776
42	49	8.00-9.00	2542-11-08				9.714	82.914	7.372
42	49	9.00-10.00	2542-11-08				6.808	87.396	5.796
42	49	10.00-11.00	2542-11-08				6.491	88.617	4.892
42	49	11.00-12.00	2542-11-08				8.947	84.436	6.617
42	49	12.00-13.00	2542-11-08				8.937	84.615	6.448
42	49	13.00-14.00	2542-11-08				9.447	83.417	7.136
42	49	14.00-15.00	2542-11-08				10.806	81.227	7.967
42	49	15.00-16.00	2542-11-08	94	783	84	9.781	81.478	8.741
42	49	16.00-17.00	2542-11-08	137	766	66	14.138	79.051	6.811
42	49	17.00-18.00	2542-11-08	122	714	57	13.662	79.955	6.383
42	49	18.00-19.00	2542-11-08	104	651	53	12.871	80.569	6.559
42	58	7.00-8.00	2542-11-08	191	1349	111	11.569	81.708	6.723
42	58	8.00-9.00	2542-11-08	209	1287	150	12.697	78.19	9.113
42	58	9.00-10.00	2542-11-08	119	964	112	9.958	80.669	9.372
42	58	10.00-11.00	2542-11-08	106	883	159	9.233	76.916	13.85
42	58	11.00-12.00	2542-11-08	107	726	95	11.53	78.233	10.237
42	58	12.00-13.00	2542-11-08	149	717	89	15.602	75.079	9.319
42	58	13.00-14.00	2542-11-08	141	640	110	15.825	71.829	12.346
42	58	14.00-15.00	2542-11-08	162	739	116	15.929	72.665	11.406
42	58	15.00-16.00	2542-11-08	105	700	135	11.17	74.468	14.362
42	58	16.00-17.00	2542-11-08	113	522	181	13.848	63.971	22.181
42	58	17.00-18.00	2542-11-08	130	413	155	18.625	59.169	22.206
42	58	18.00-19.00	2542-11-08	88	325	136	16.029	59.199	24.772
42	72	7.00-8.00	2542-11-08	135	368	38	24.954	68.022	7.024
42	72	8.00-9.00	2542-11-08	140	369	51	25	65.893	9.107
42	72	9.00-10.00	2542-11-08	143	336	37	27.713	65.116	7.171

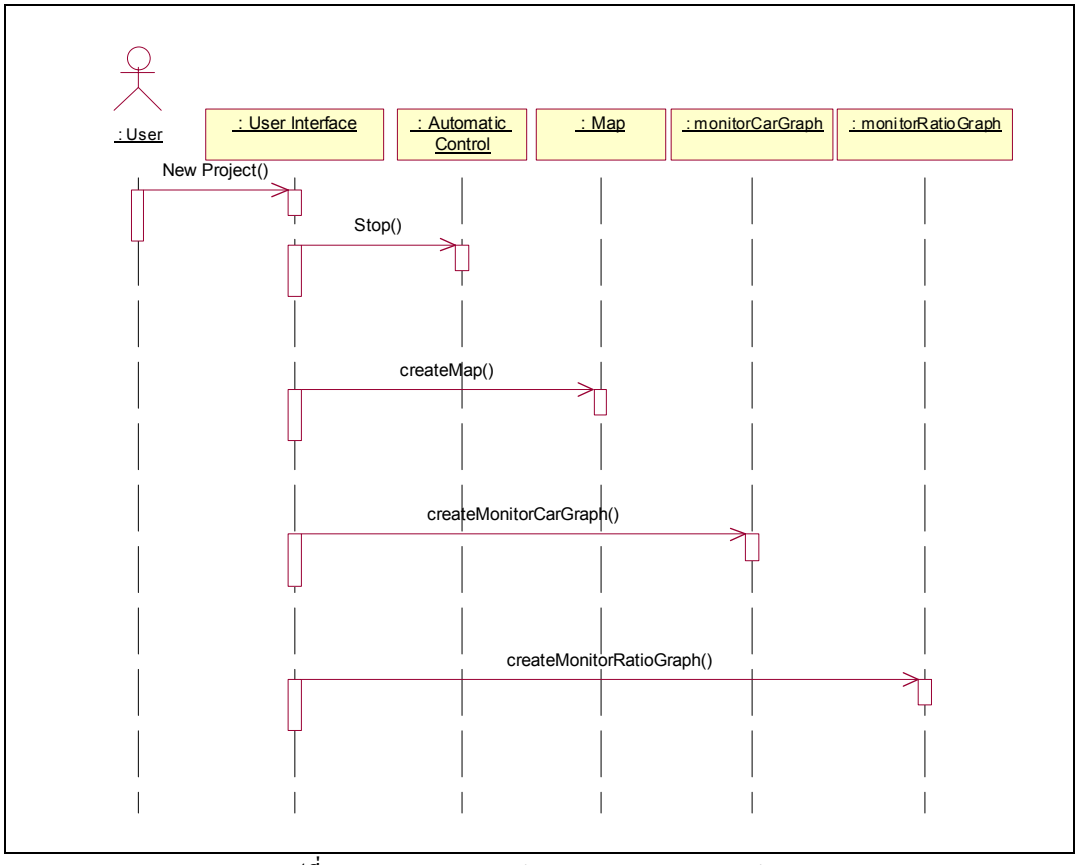
รูปที่ ข-23: รูปภาพแสดง CSV File ถูกแปลงลงฐานข้อมูลเป็นที่เรียบร้อย

ภาคผนวก ค

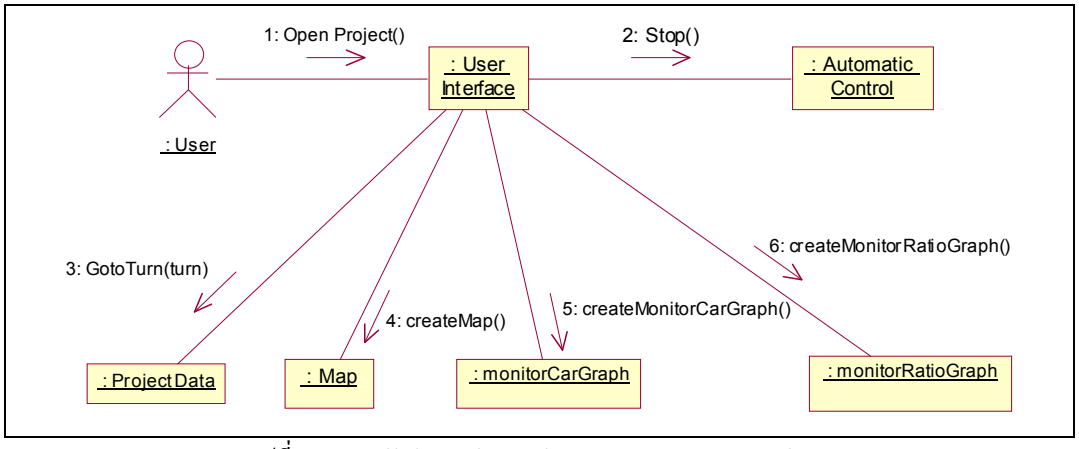
UML ระบบควบคุมและติดตามการจราจรอัตโนมัติ



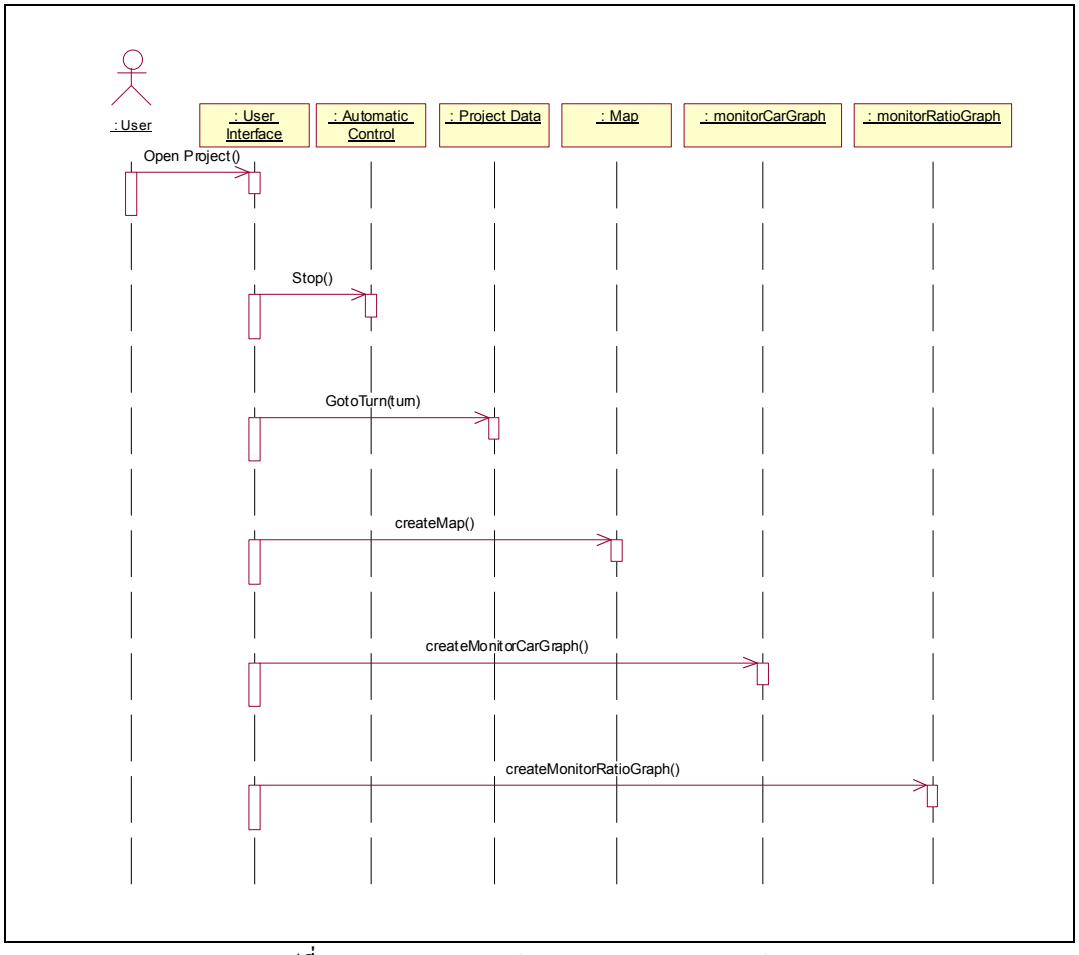
รูปที่ ค-1: Collaboration Diagramของ New Project



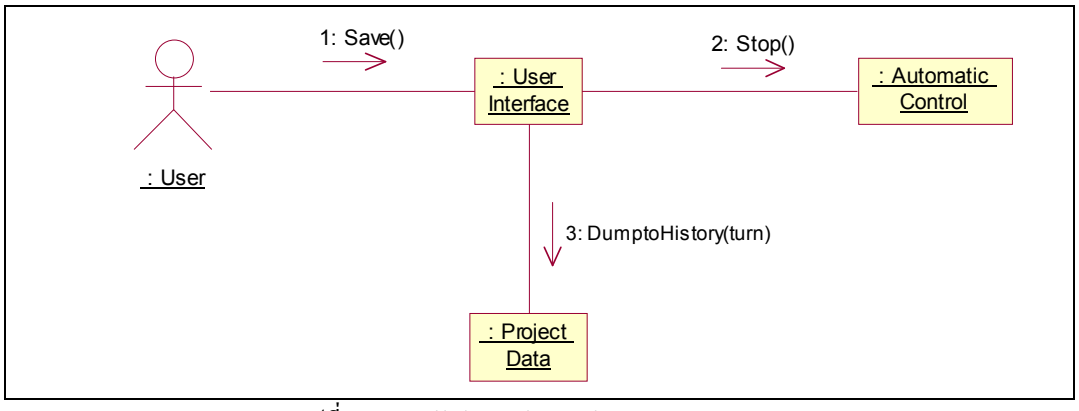
รูปที่ ค-2: Sequence Diagramของ New Project



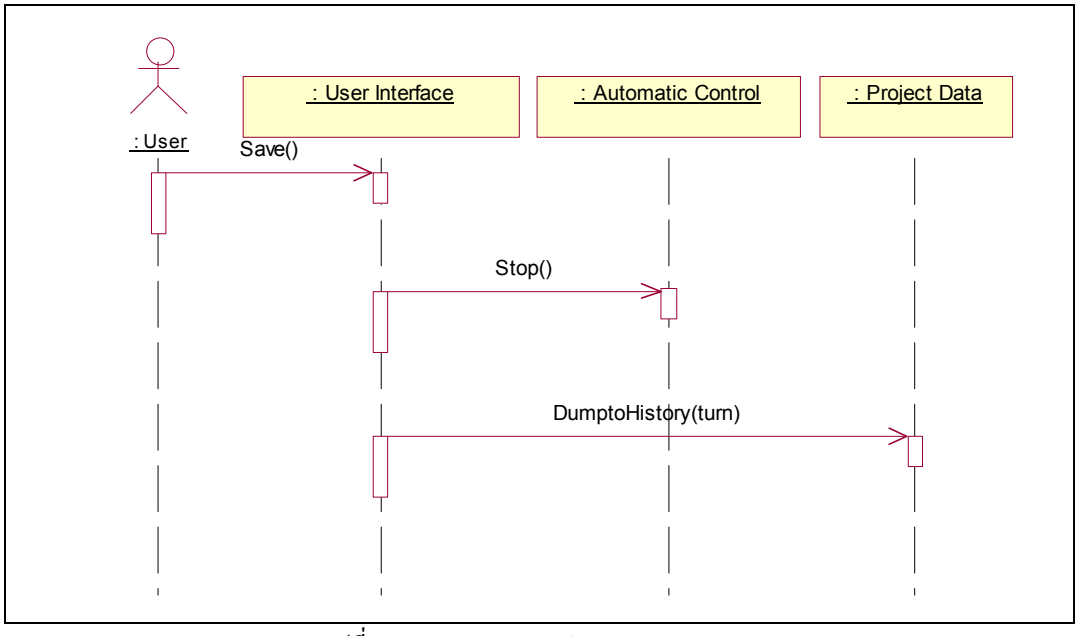
รูปที่ ค-3: Collaboration Diagramของ Open Project



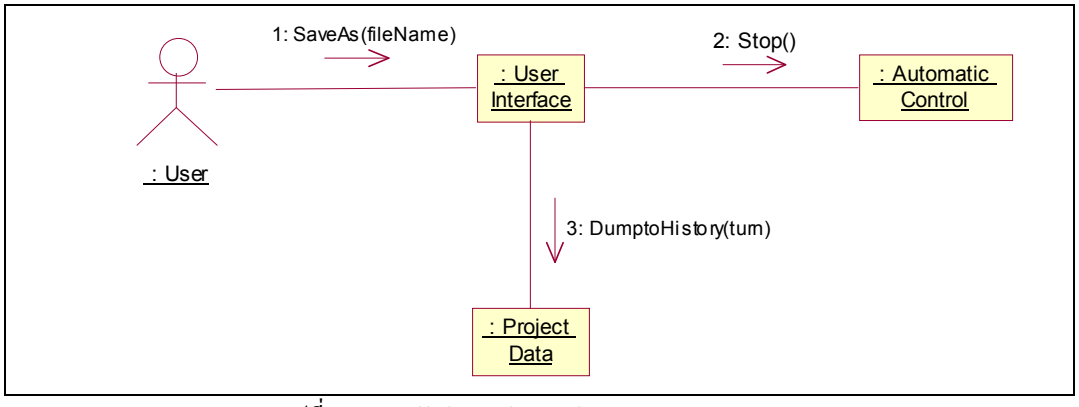
รูปที่ ค-4: Sequence Diagramของ Open Project



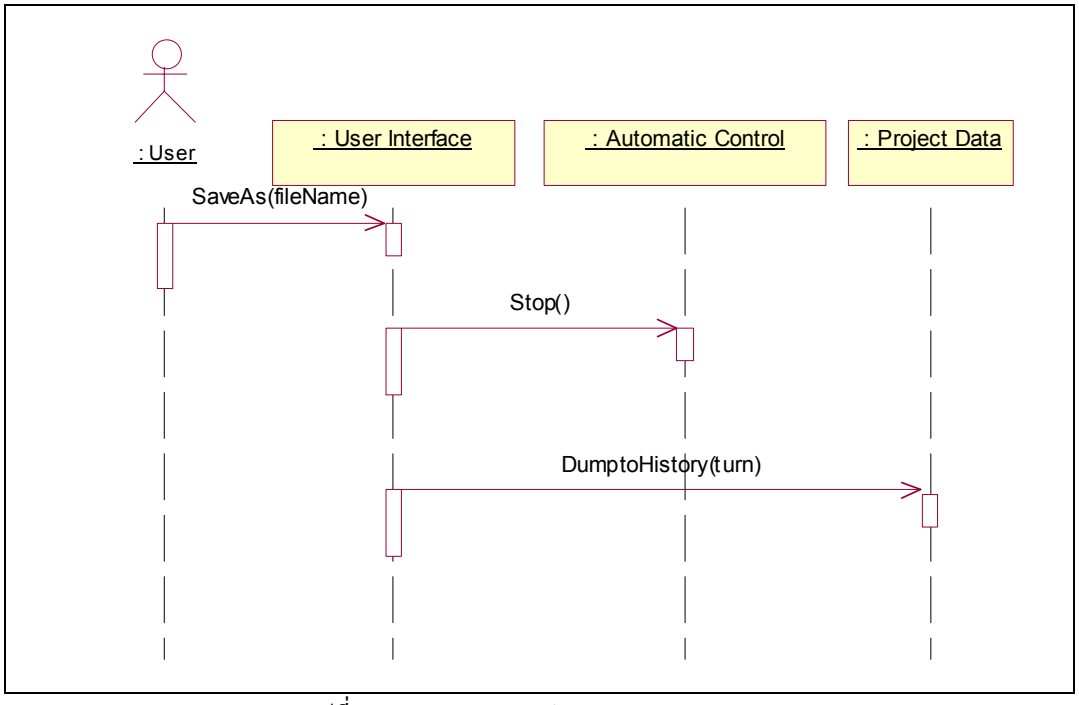
รูปที่ ค-5: Collaboration Diagramของ Save



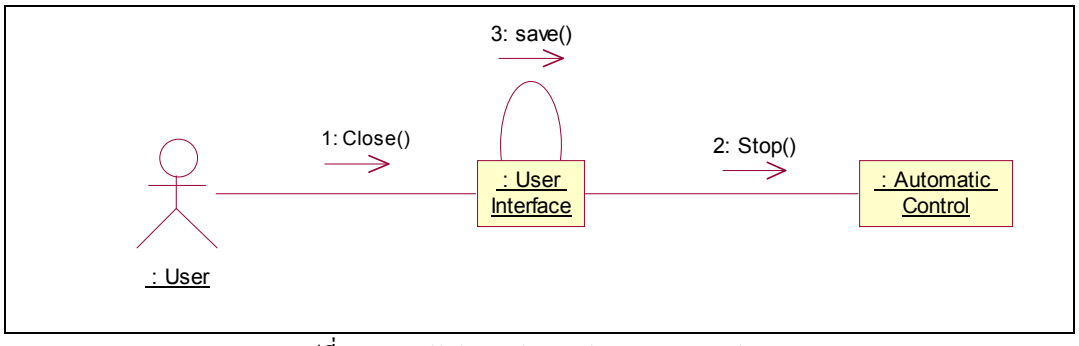
รูปที่ ค-6: Sequence Diagramของ Save



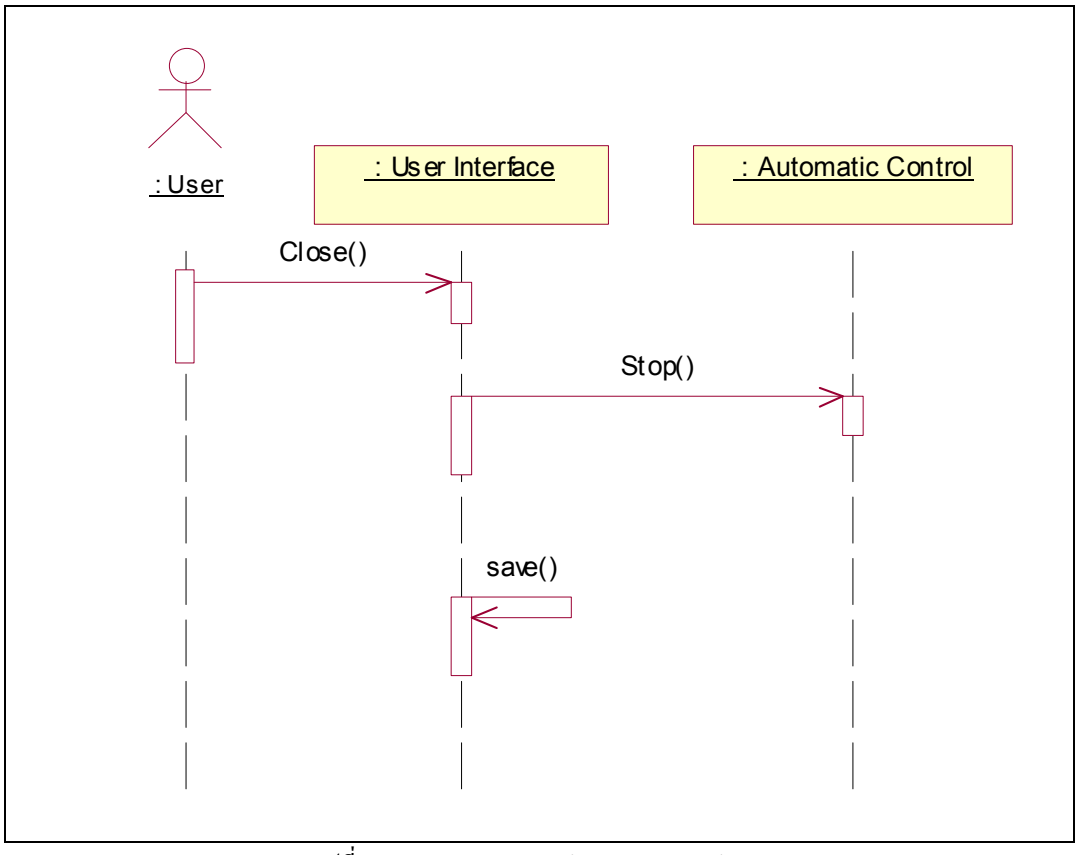
รูปที่ ค-7: Collaboration Diagramของ SaveAs



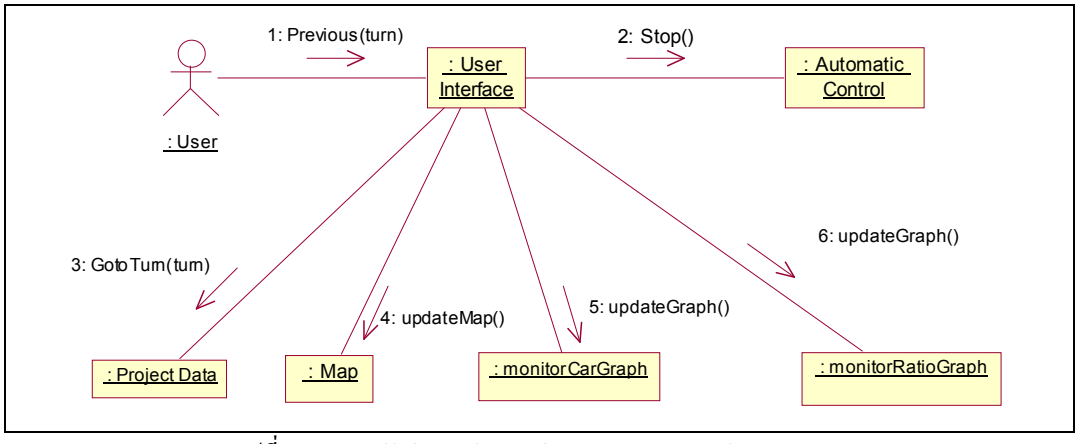
รูปที่ ค-8: Sequence Diagramของ SaveAs



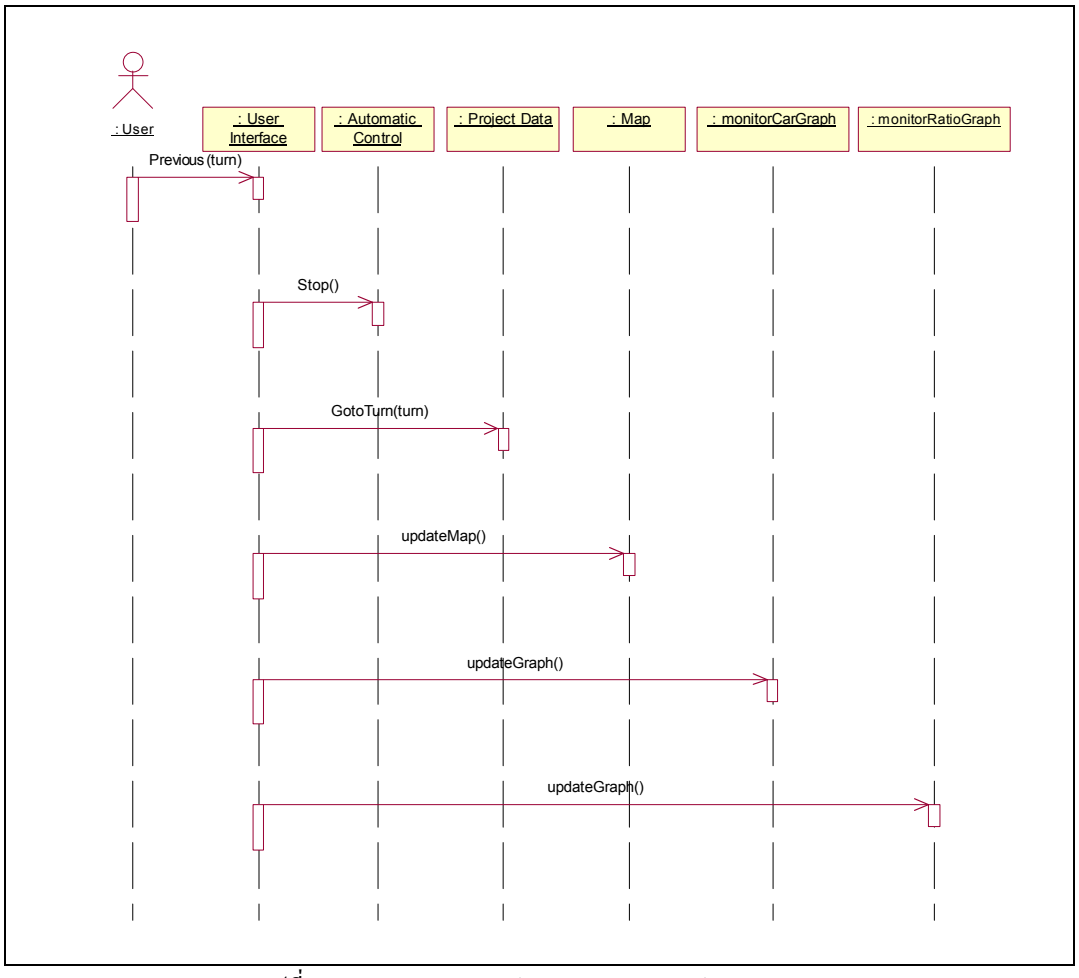
รูปที่ ค-9: Collaboration Diagramของ Close



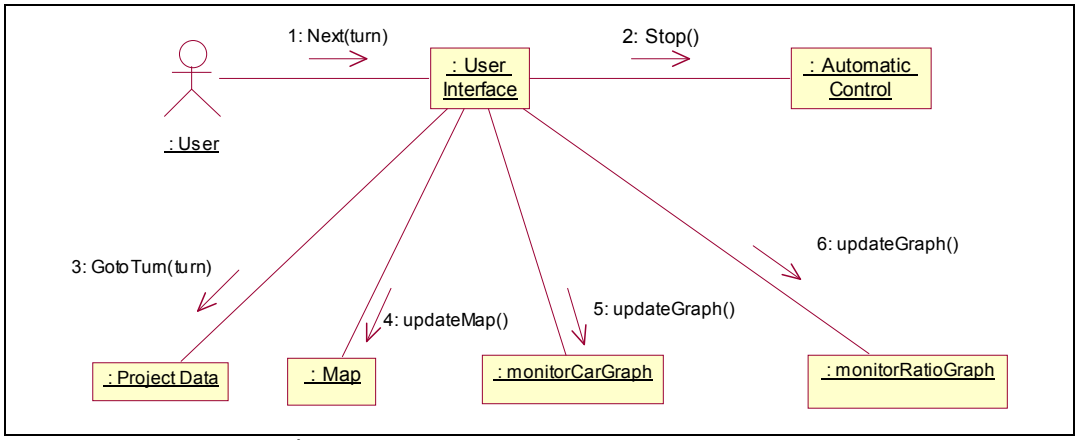
รูปที่ ค-10: Sequence Diagramของ Close



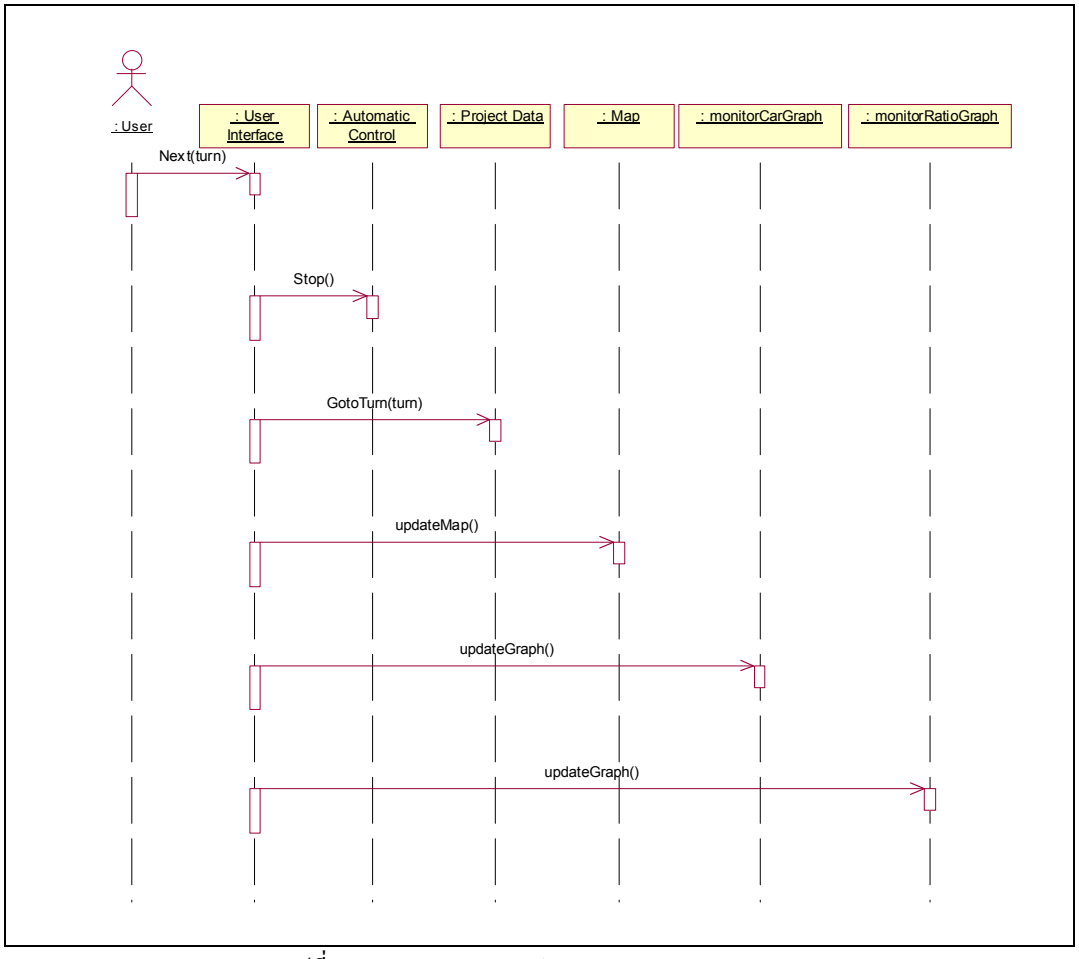
รูปที่ ค-11: Collaboration Diagramของ Previous Turn



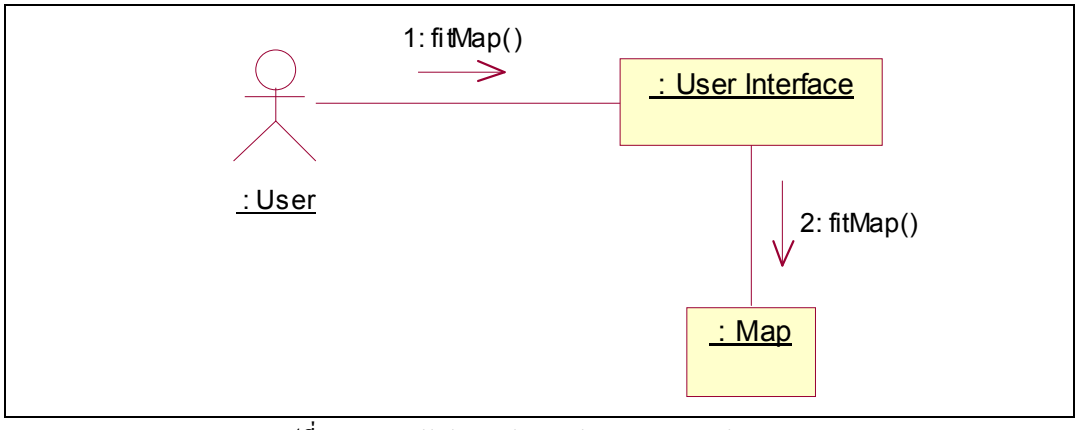
รูปที่ ค-12: Sequence Diagramของ Previous Turn



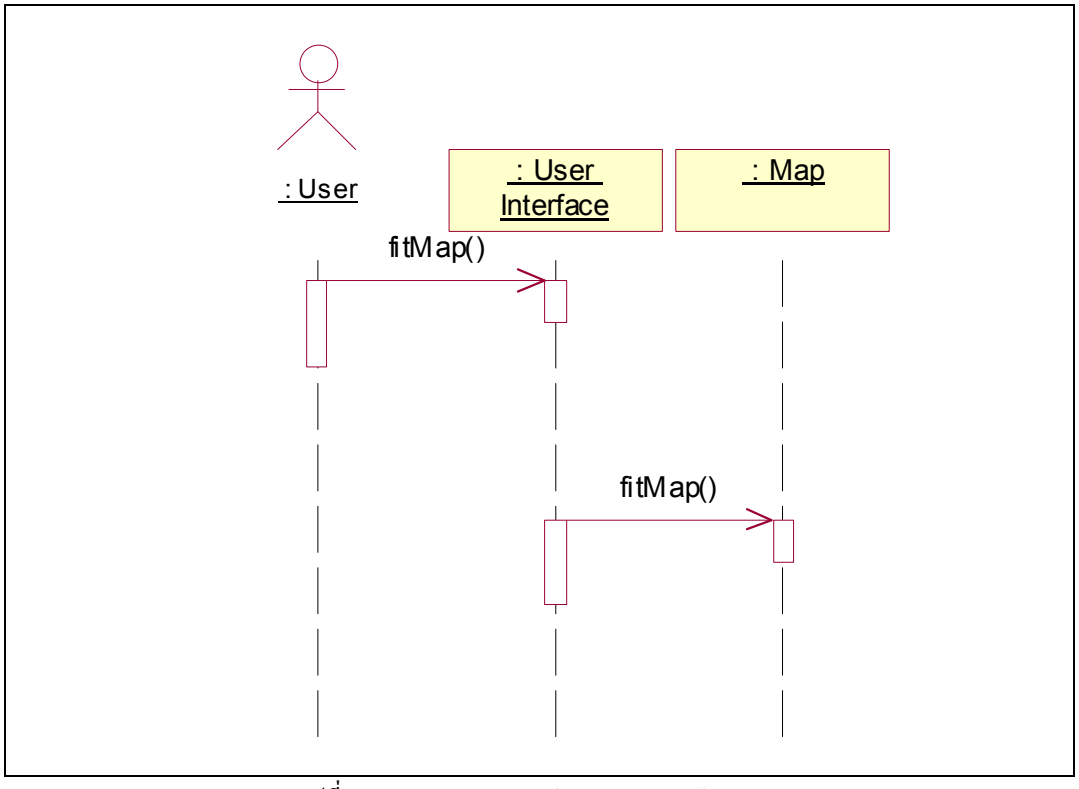
รูปที่ ค-13: Collaboration Diagramของ Next Turn



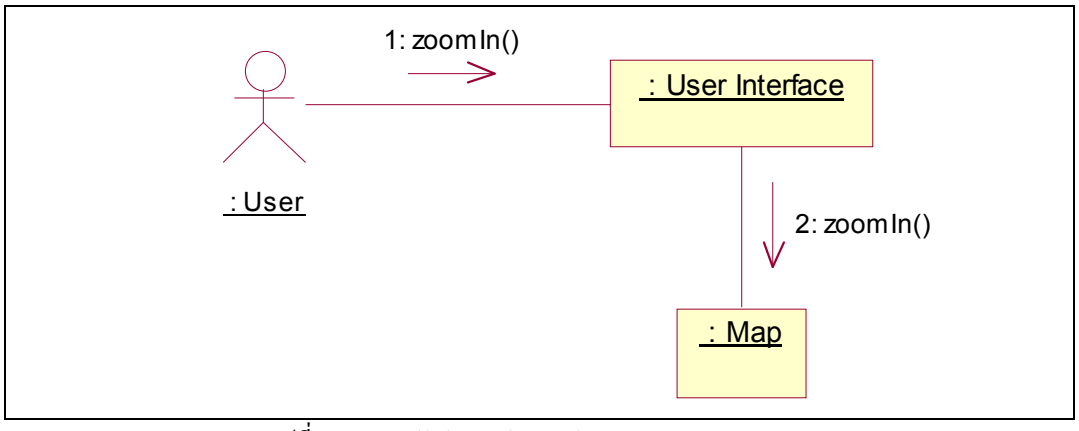
รูปที่ ค-14: Sequence Diagramของ Next Turn



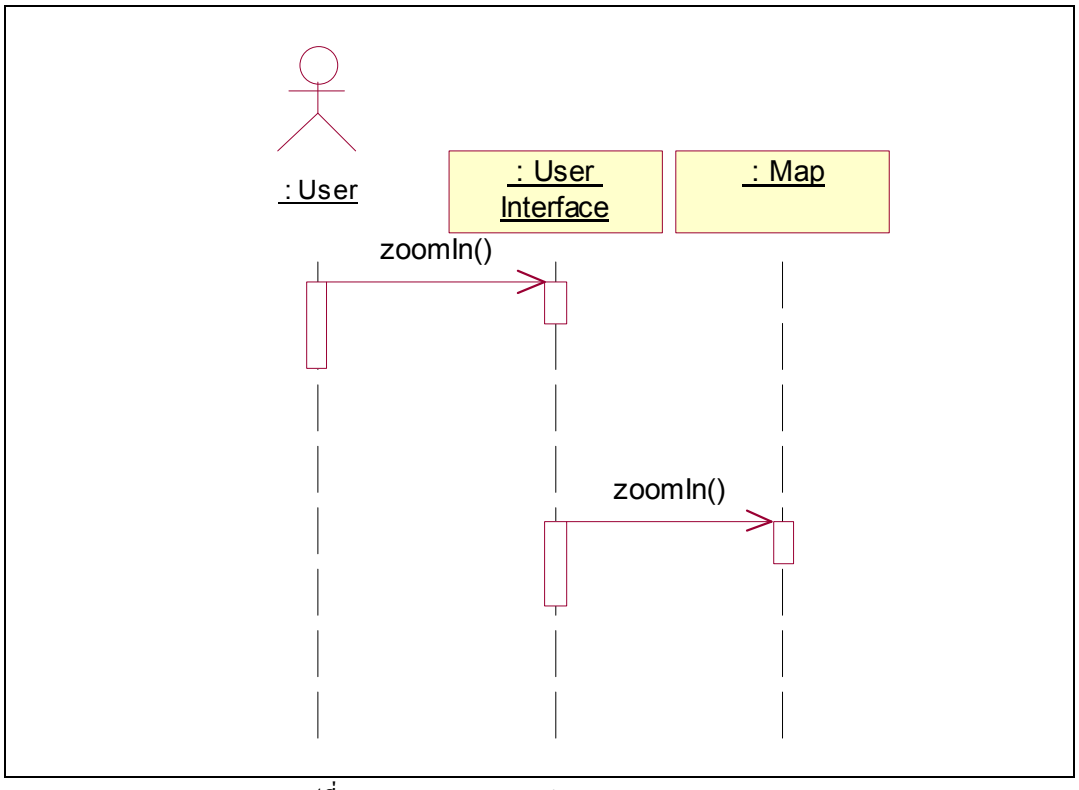
รูปที่ ค-15: Collaboration Diagramของ Fit Map



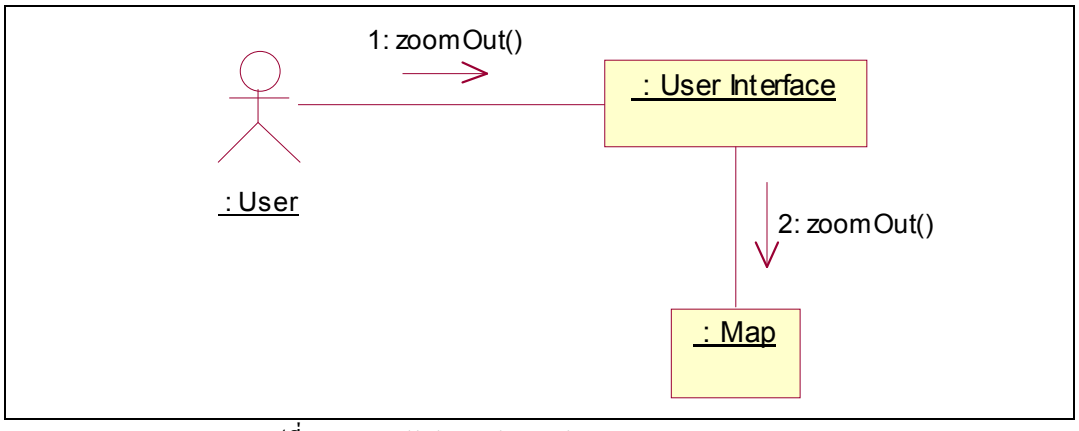
รูปที่ ค-16: Sequence Diagramของ Fit Map



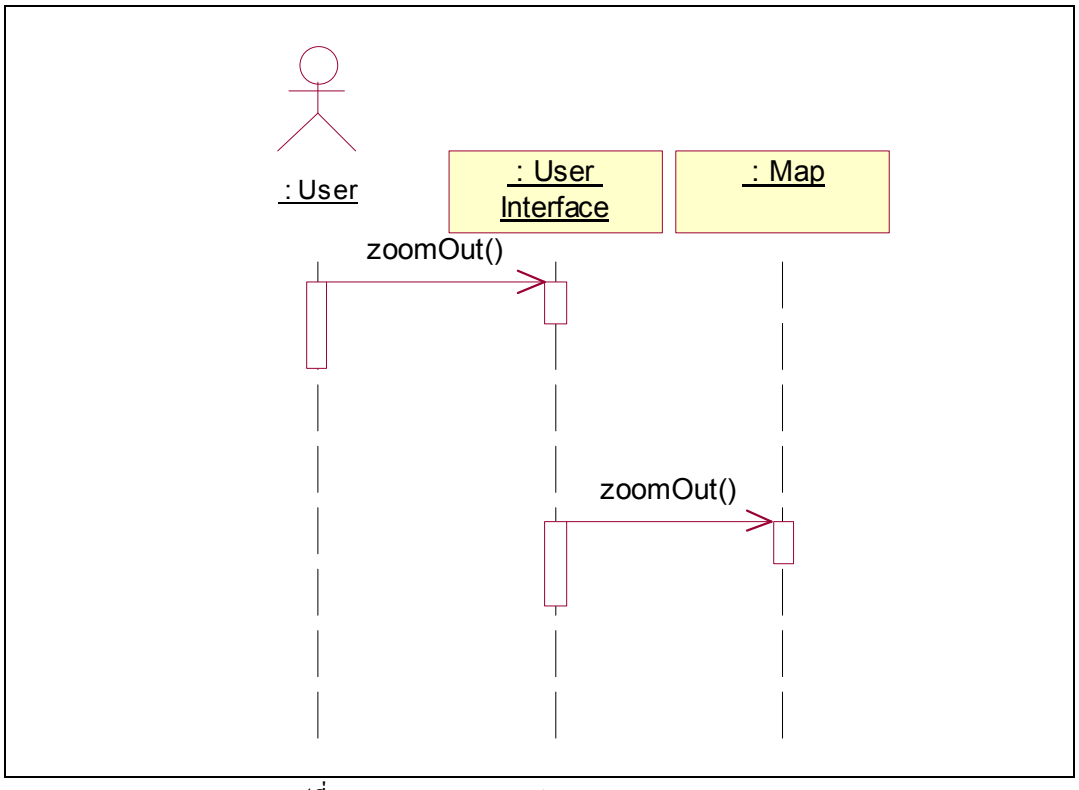
รูปที่ ค-17: Collaboration Diagramของ Zoom In



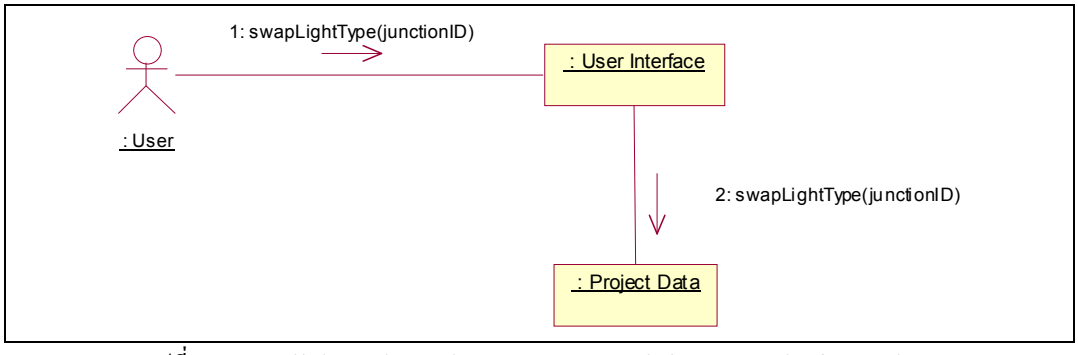
รูปที่ ค-18: Sequence Diagramของ Zoom In



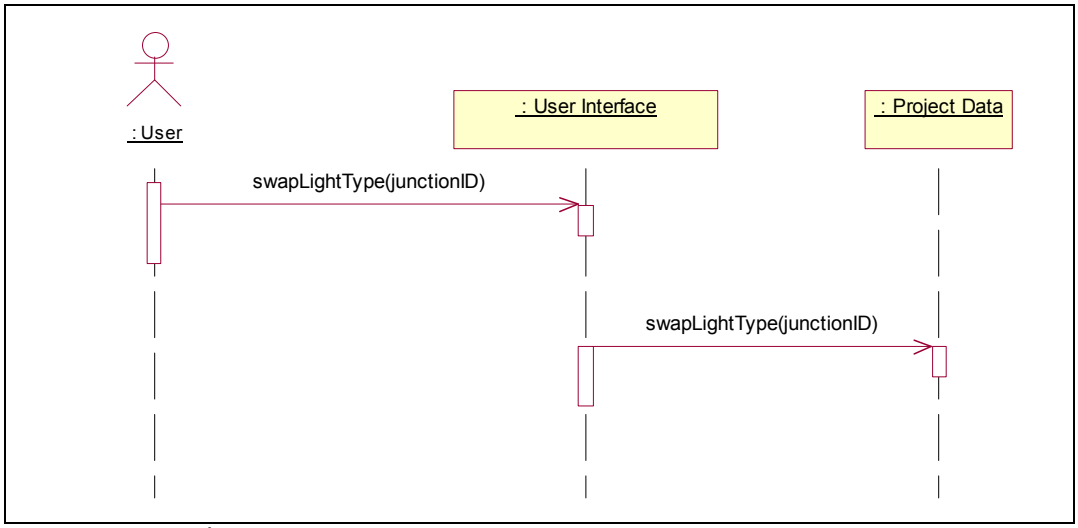
รูปที่ ค-19: Collaboration Diagramของ Zoom Out



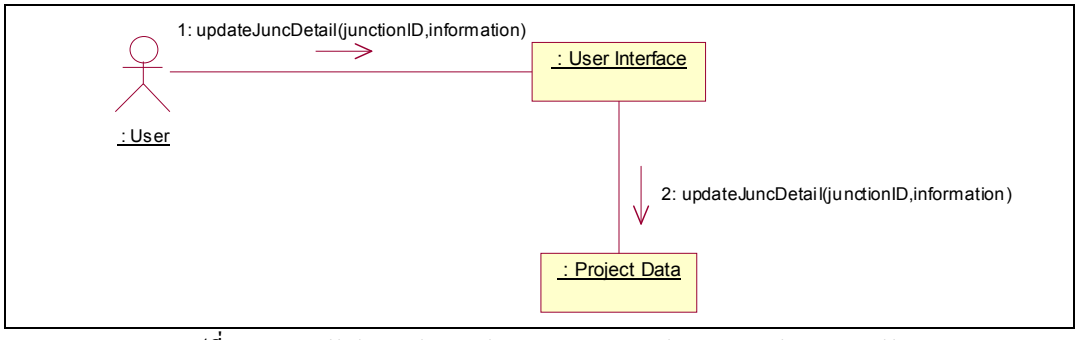
รูปที่ ค-20: Sequence Diagramของ Zoom Out



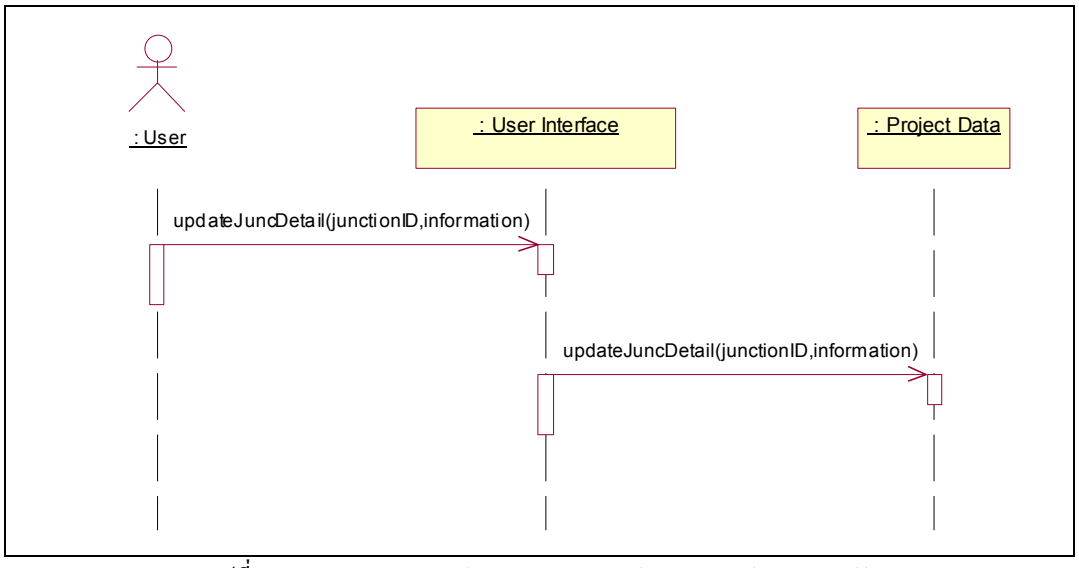
รูปที่ ค-21: Collaboration Diagramของ Set Light Control of Junction



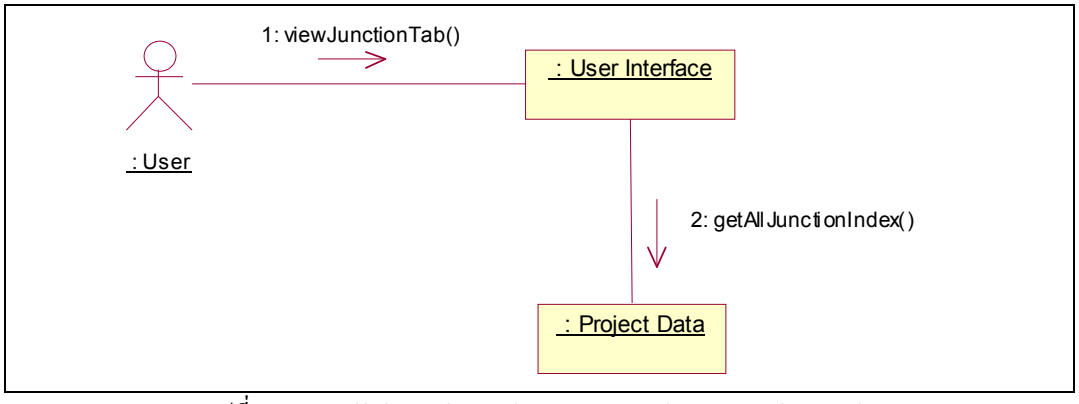
รูปที่ ค-22: Sequence Diagramของ Set Light Control of Junction



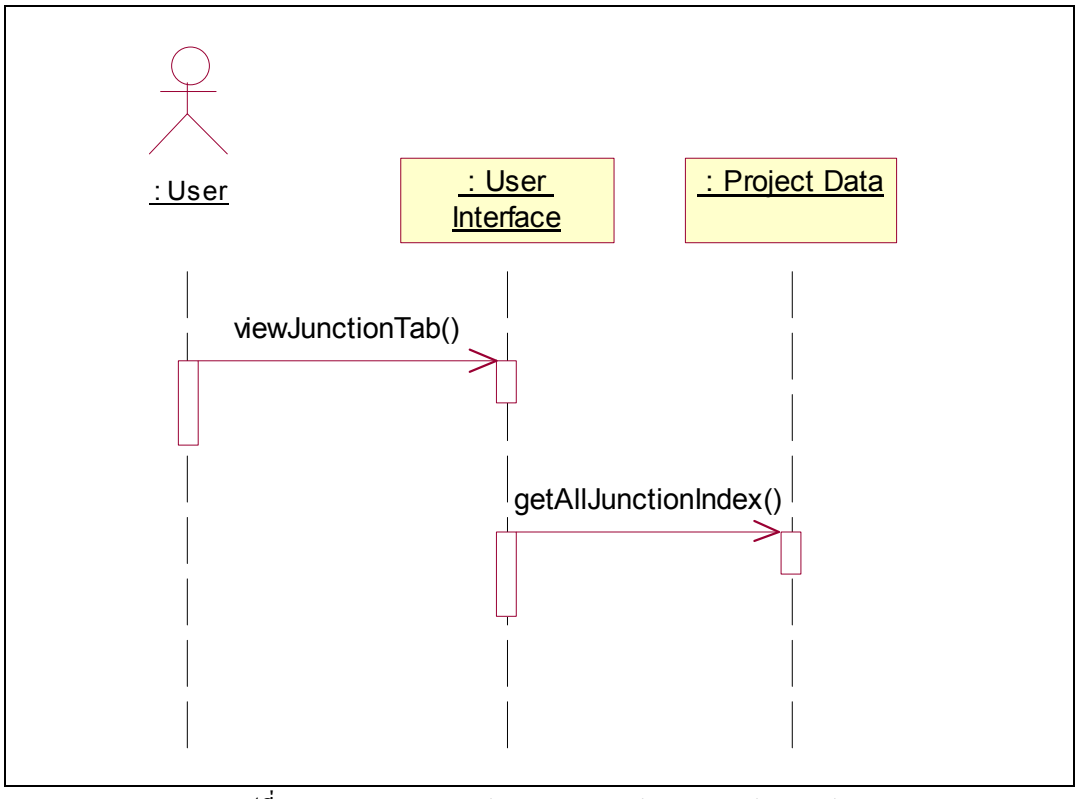
รูปที่ ค-23: Collaboration Diagram ของ Update Junction Detail



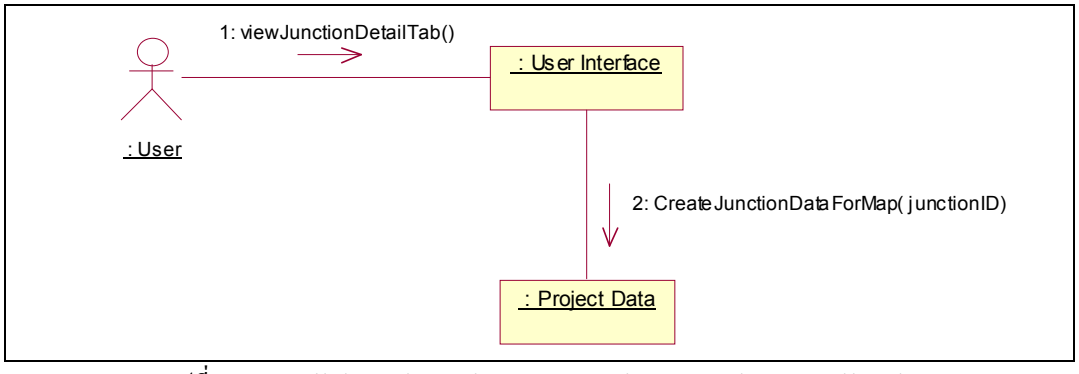
รูปที่ ค-24: Sequence Diagramของ Update Junction Detail



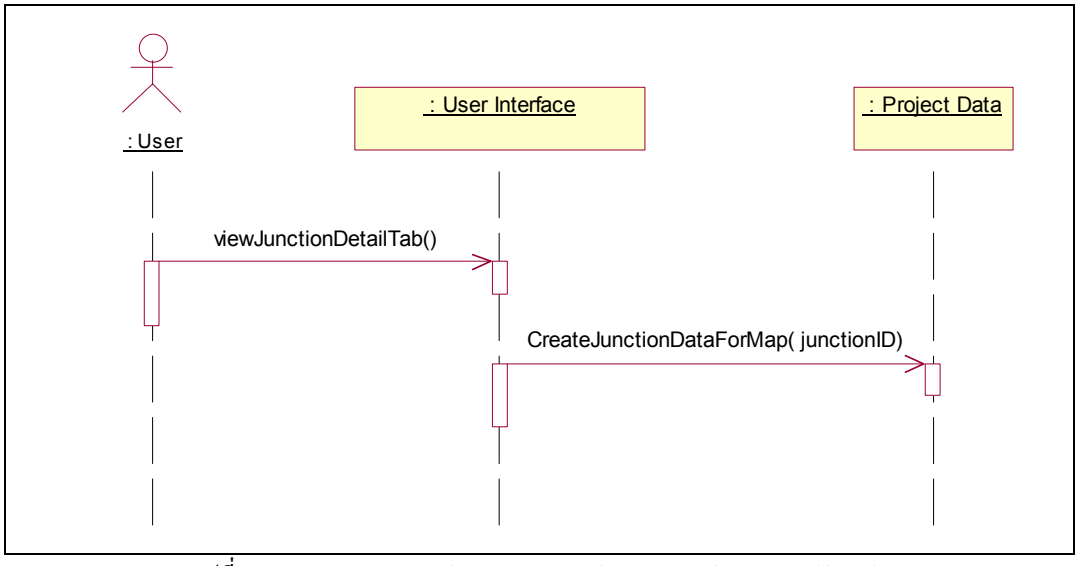
รูปที่ ค-25: Collaboration Diagramของ View Junction Tab



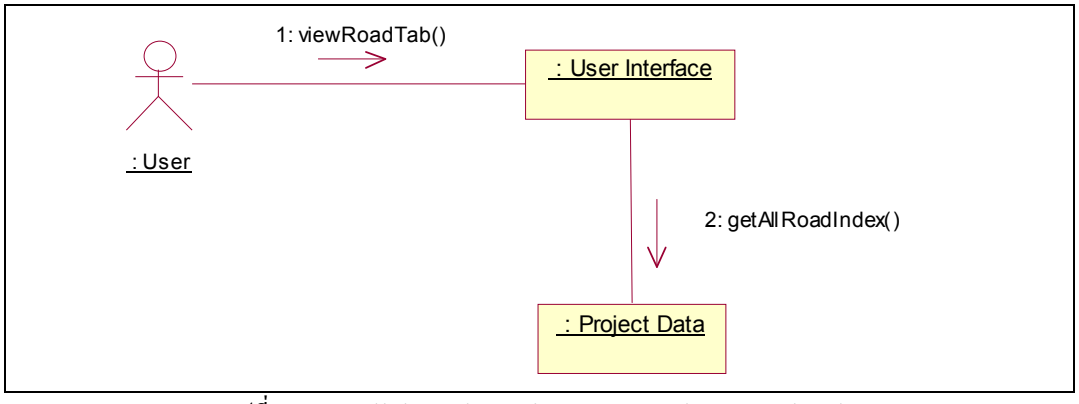
รูปที่ ค-26: Sequence Diagramของ View Junction Tab



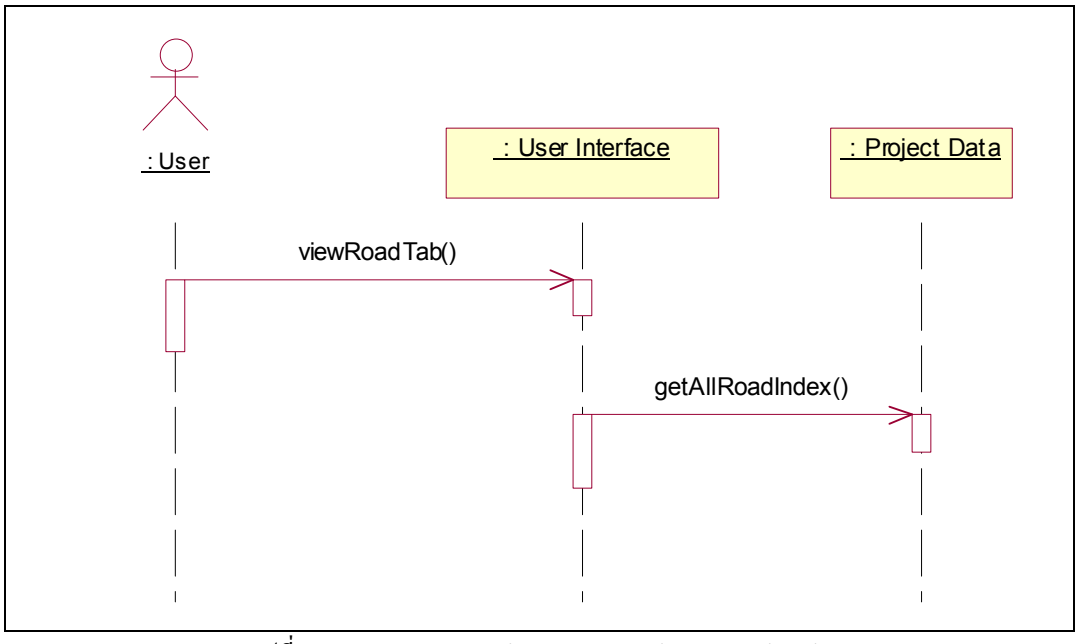
รูปที่ ค-27: Collaboration Diagramของ View Junction Detail Tab



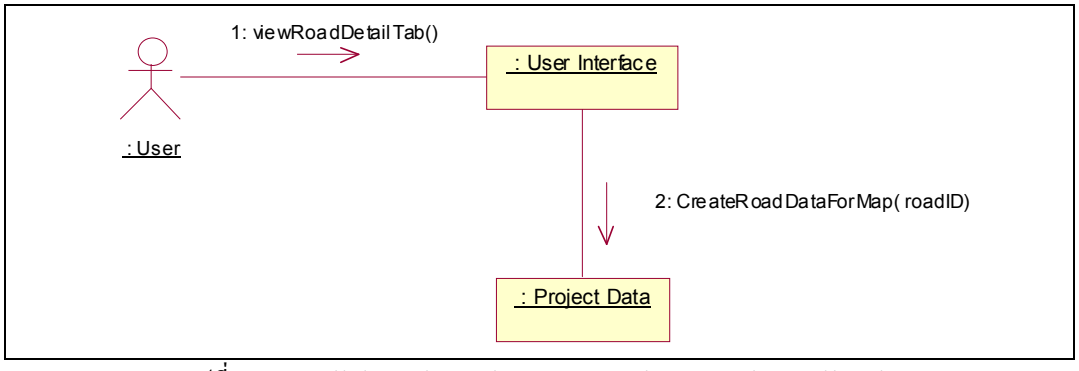
รูปที่ ค-28: Sequence Diagramของ View Junction Detail Tab



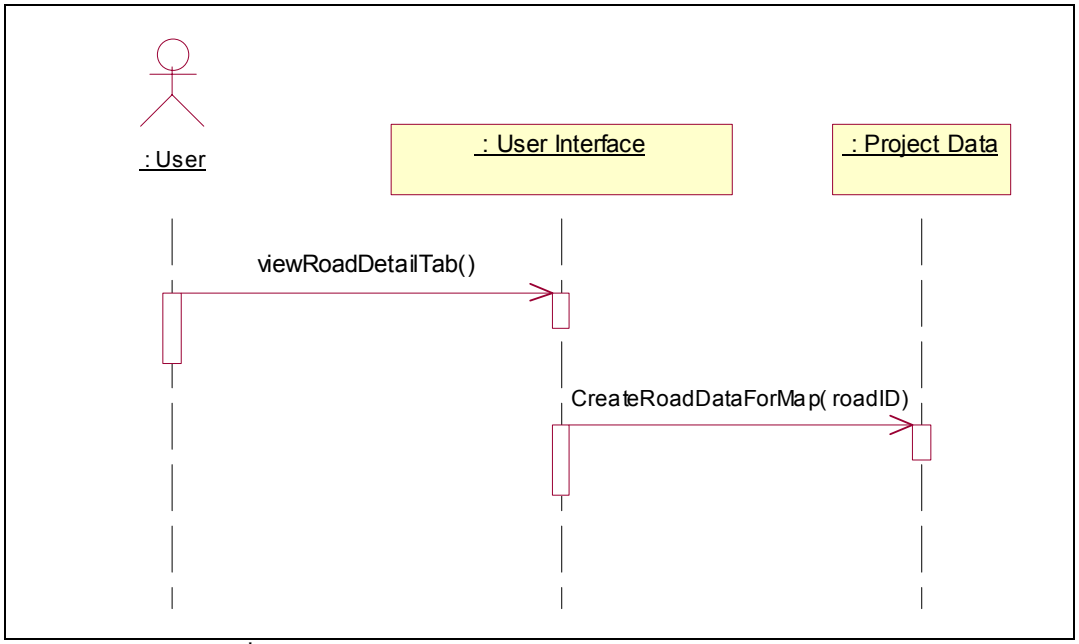
รูปที่ ค-29: Collaboration Diagramของ View Road Tab



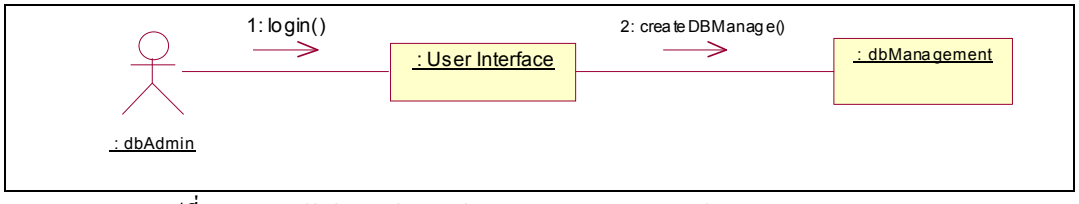
รูปที่ ค-30: Sequence Diagramของ View Road Tab



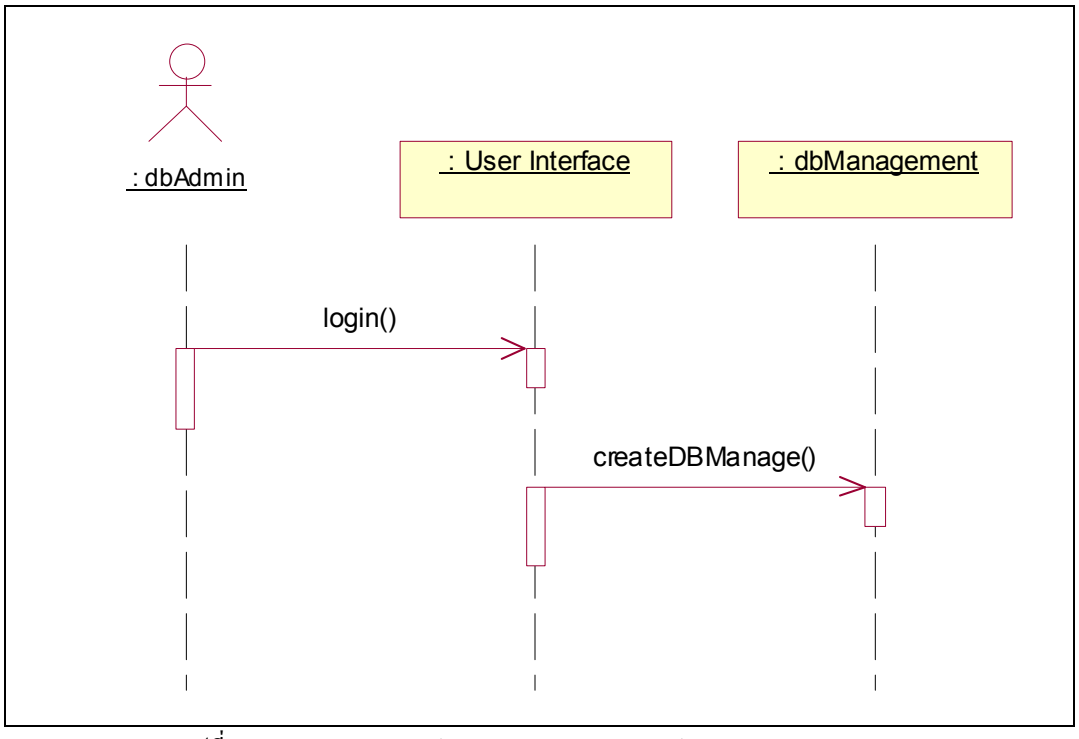
รูปที่ ค-31: Collaboration Diagramของ View Road Detail Tab



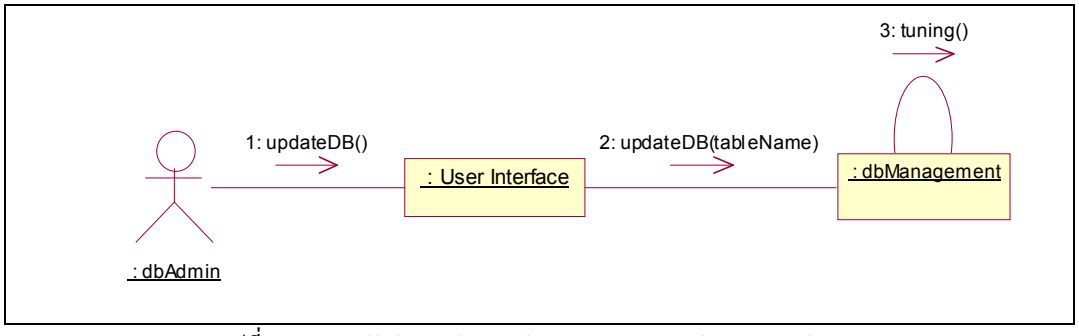
รูปที่ ค-32: Sequence Diagramของ View Road Detail Tab



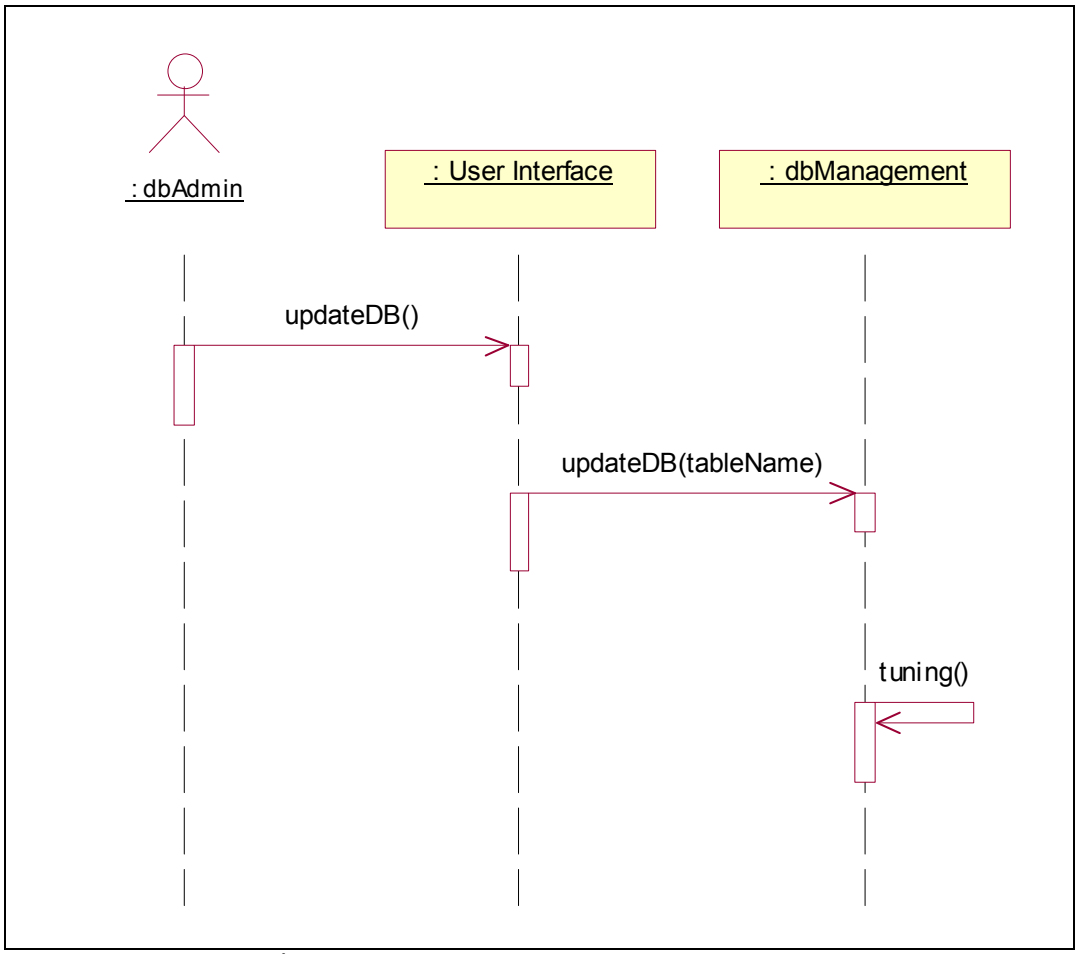
รูปที่ ค-33: Collaboration Diagramของ Use Database Management



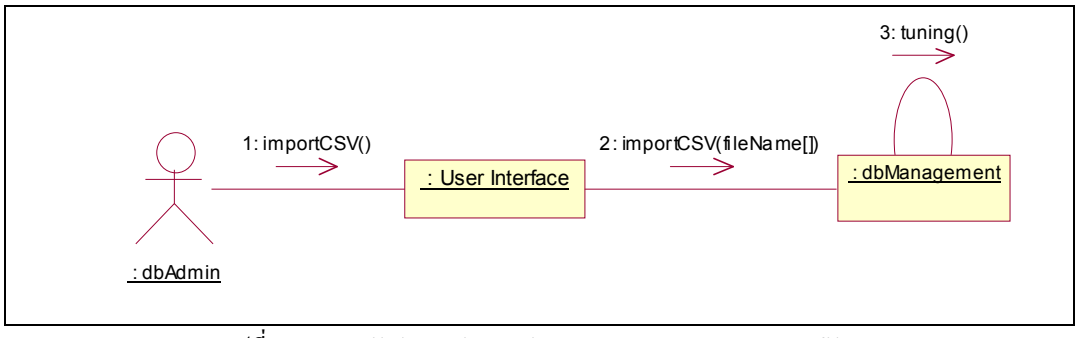
รูปที่ ค-34: Sequence Diagramของ Use Database Management



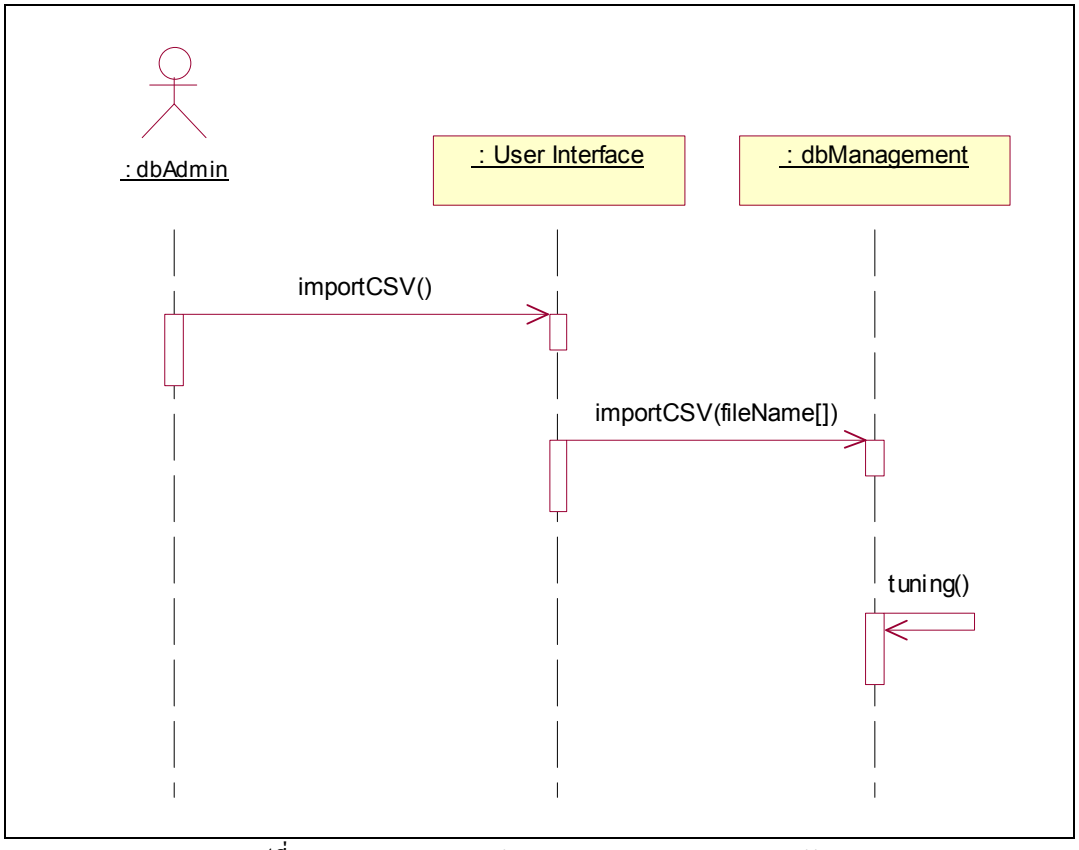
รูปที่ ค-35: Collaboration Diagramของ Update Database



รูปที่ ค-36: Sequence Diagramของ Update Database



รูปที่ ค-37: Collaboration Diagramของ Import CSV file



รูปที่ ค-38: Sequence Diagramของ Import CSV File

ภาคผนวก ง

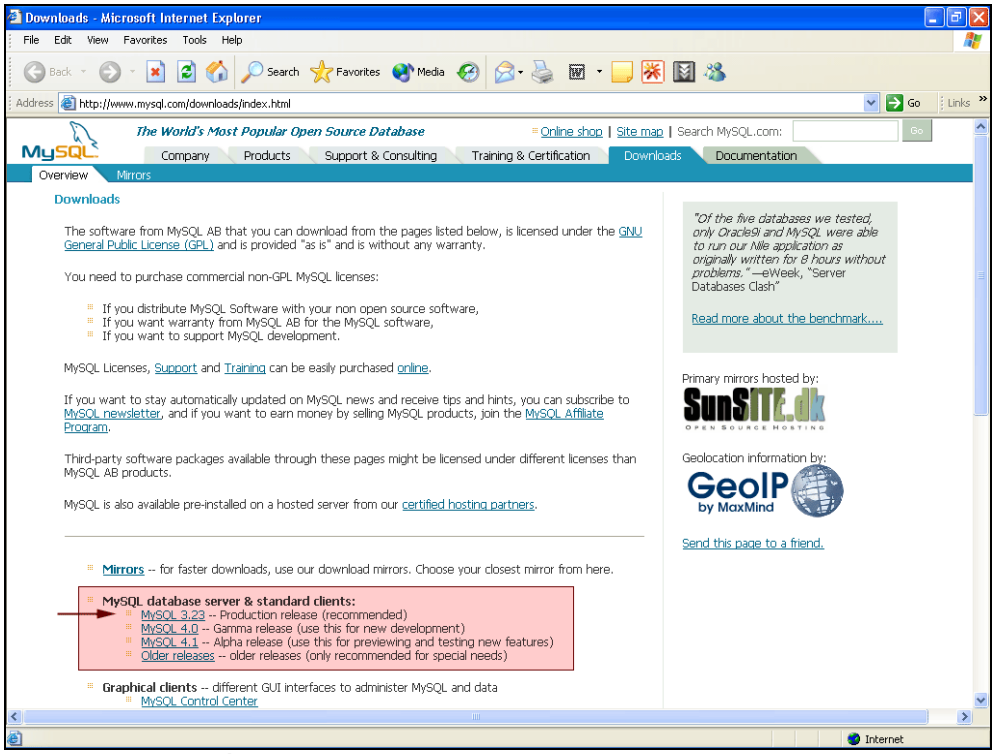
การติดตั้งโปรแกรมจัดการฐานข้อมูล MySQL

ในการติดตั้งระบบนั้น จำเป็นต้องมีการติดตั้งเครื่องมือที่ใช้ร่วมกับตัวระบบด้วย ซึ่งวิธีการติดตั้งเครื่องมือที่ใช้ในบทนี้ จะมีเนื้อหาดังต่อไปนี้

- การติดตั้ง MySQL ในเครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการ Windows
- การติดตั้ง MySQL ในเครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการ Linux

วิธีการติดตั้ง MySQL ใน MS Windows

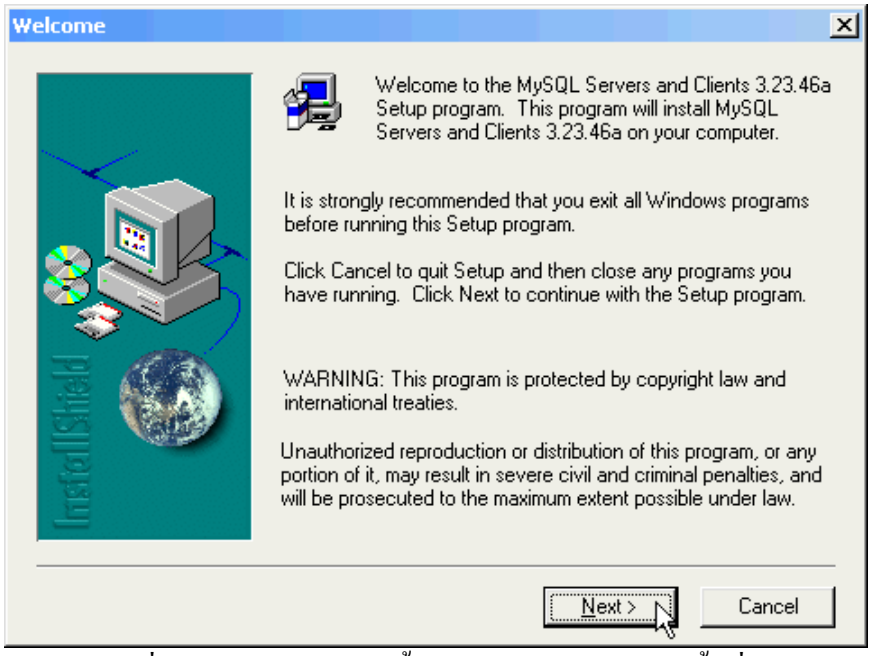
สิ่งที่ต้องเตรียมก่อนการติดตั้ง ทำการ Download Program ที่จำเป็นต้องใช้นั่นคือ MySQL (Windows binary) มาจาก http://www.mysql.com/downloads/index.html/ ซึ่งมีชื่อไฟล์เป็น mysql-3.23.xx-win.zip ดังรูปภาพทางด้านล่างนี้



รูปที่ ง-1: รูปแสดงการ Download MySQL for Windows

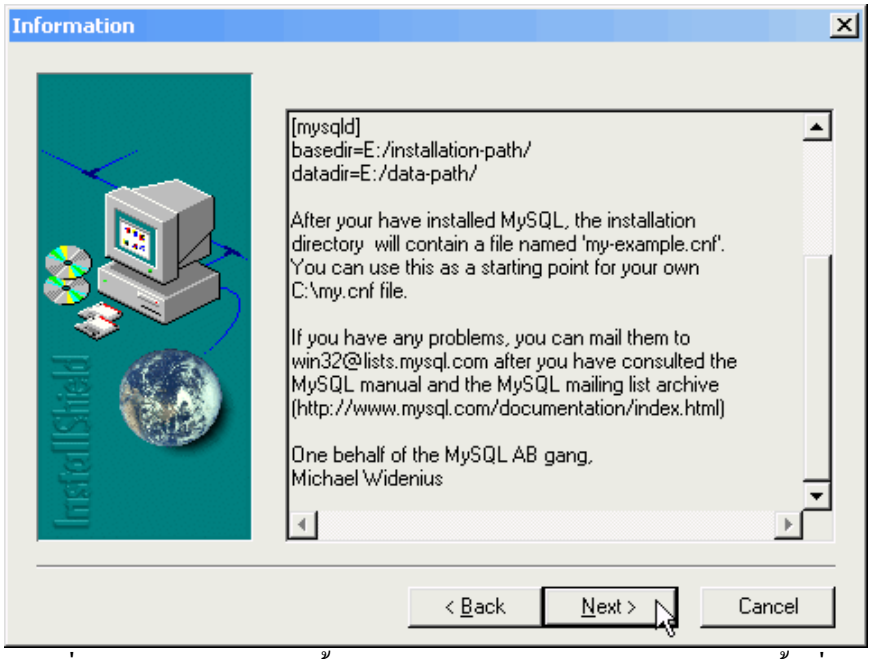
วิธีการติดตั้ง MySQL

1. หลังจากดาวน์โหลดมาแล้วก็ให้แตกไฟล์ออก แล้วก็รันไฟล์ชื่อ Setup.exe ก็จะปรากฏหน้าต่างดังรูป ให้คลิกที่ปุ่ม Next



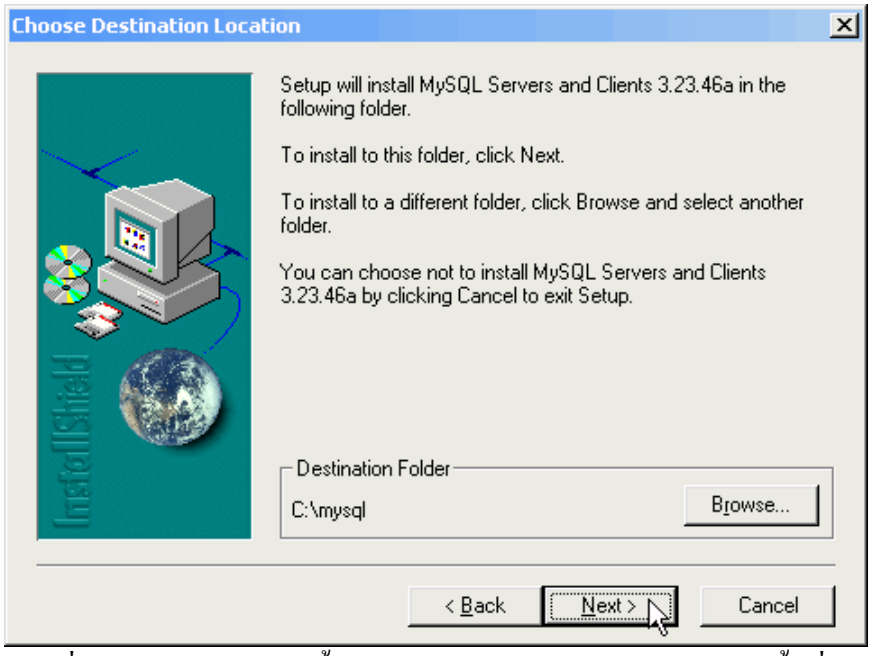
รูปที่ ง-2: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 1

2. หลังจากอ่านข้อความเสร็จแล้วให้คลิกที่ปุ่ม Next



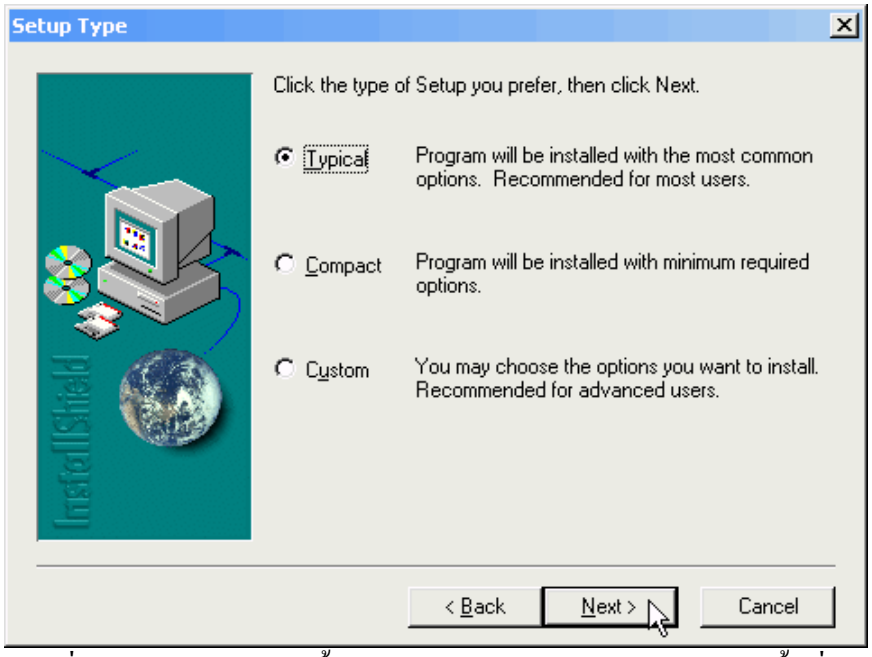
รูปที่ ง-3: รูปแสดงการติดตั้ง MySQL บน Windows บน Windows ขั้นที่ 2

3. เลือกโฟลเดอร์ที่จะทำการติดตั้งโปรแกรม MySQL จากนั้นให้คลิกที่ปุ่ม Next ดังรูป



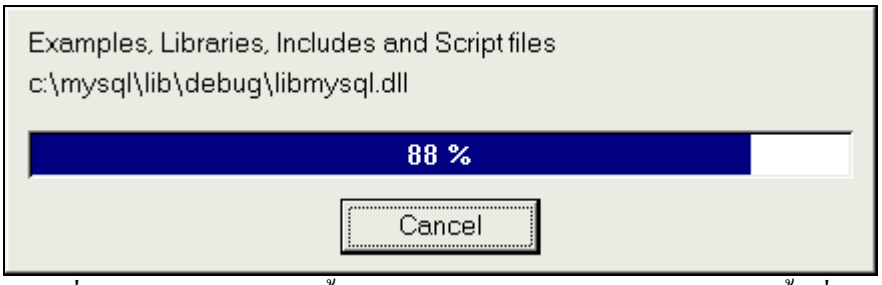
รูปที่ ง-4: รูปแสดงการติดตั้ง MySQL บน Windows บน Windows ขั้นที่ 3

4. เลือกรูปแบบที่จะทำการติดตั้ง การติดตั้งโดยทั่วไปให้เลือกที่ Typical แล้วคลิกที่ปุ่ม Next



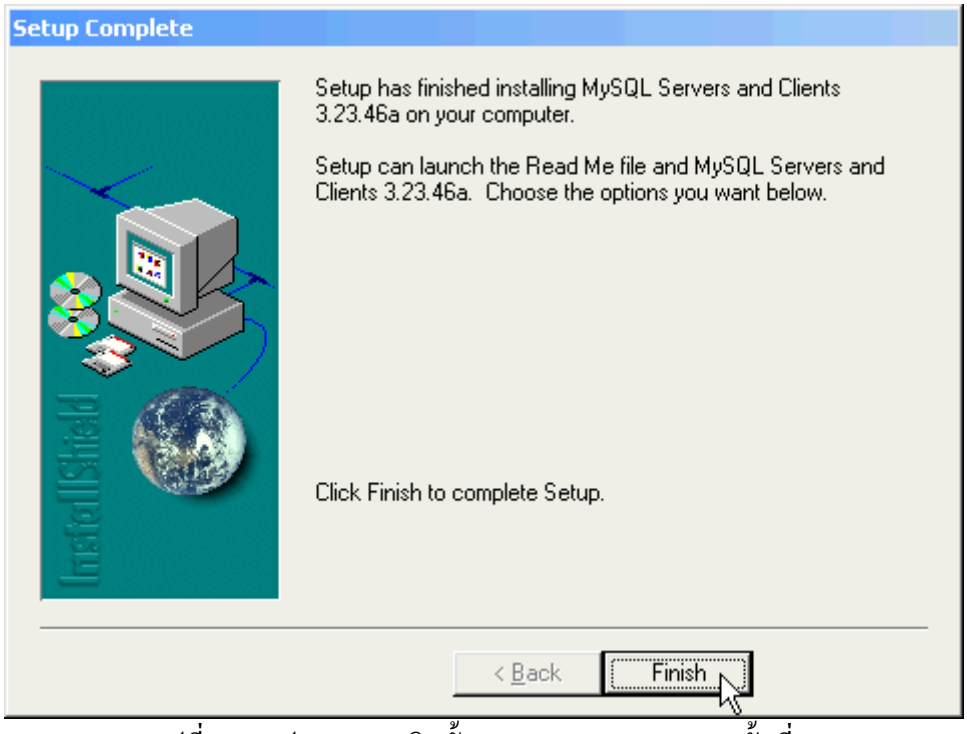
รูปที่ ง-5: รูปแสดงการติดตั้ง MySQL บน Windows บน Windows ขั้นที่ 4

5. จากนั้นโปรแกรม MySQL จะทำการติดตั้งตัวเองลงไปในเครื่อง ดังรูป



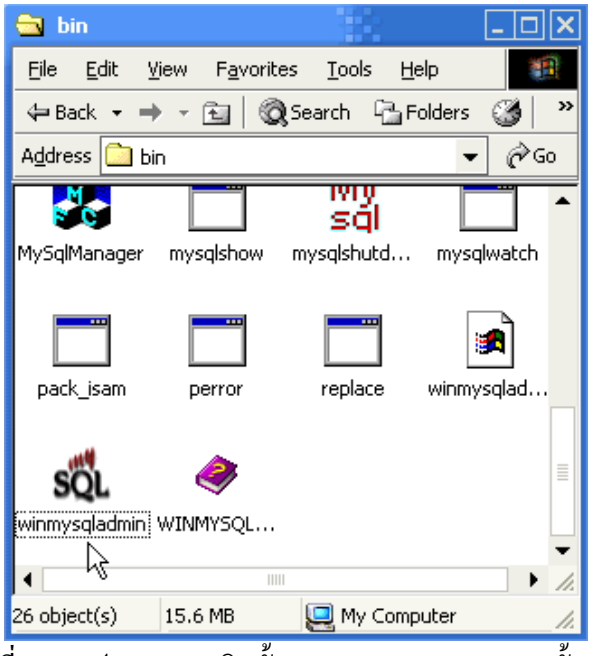
รูปที่ ง-6: รูปแสดงการติดตั้ง MySQL บน Windows บน Windows ขั้นที่ 5

6. เมื่อเสร็จสิ้นการติดตั้งแล้วก็จะมีข้อความแจ้งให้คุณทราบดังรูป ให้คลิกที่ปุ่ม Finish



รูปที่ ง-7: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 6

7. ขั้นตอนต่อไปให้รันโปรแกรม winmysqladmin.exe ซึ่งจะอยู่ใน path เช่น c:\mysql\bin\



รูปที่ ง-8: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 7

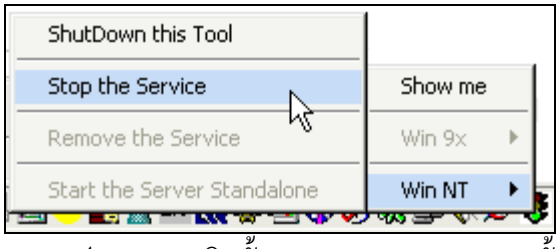
8. หลังจากที่รันโปรแกรม winmysqladmin.exe เป็นครั้งแรก ก็จะมีหน้าต่างดังรูป ให้คุณใส่ username และ password ที่จะใช้สำหรับ access MySQL แล้วคลิกที่ปุ่ม OK

รูปที่ ง-9: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 8

9. ถ้าสังเกตที่ Taskbar จะเห็นว่ามี icon เล็กๆ เป็นรูปสัญญาณไฟปรากฏอยู่ ซึ่งก็คือ icon ของโปรแกรม winmysqladmin นั่นเอง ถ้าเป็นสัญญาณไฟเขียวก็แสดงว่า MySQL ทำงานอยู่ ถ้าสัญญาณไฟเป็นสีแดงก็แสดงว่า MySQL หยุดทำงาน โดยเราสามารถที่จะใช้ winmysqladmin ในการเปิดปิดโปรแกรมโปรแกรม MySQL ได้

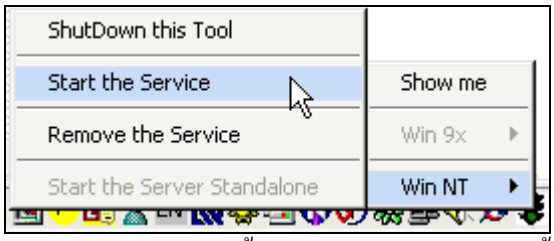
รูปที่ ง-10: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 9

10. ถ้าต้องการให้ MySQL หยุดทำงานก็ให้คลิกขวาที่สัญญาณไฟเลือก Win NT -> Stop the Service (ถ้าเป็น Win95/98/Me ก็ให้เลือก Win9x แทน) ดังรูป



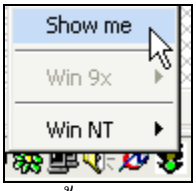
รูปที่ ง-11: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 10

11. ถ้าต้องการให้ MySQL ทำงานก็ให้คลิกขวาที่สัญญาณไฟเลือก Win NT -> Start the Service (ถ้าเป็น Win95/98/Me ก็ให้เลือก Win9x แทน) ดังรูป



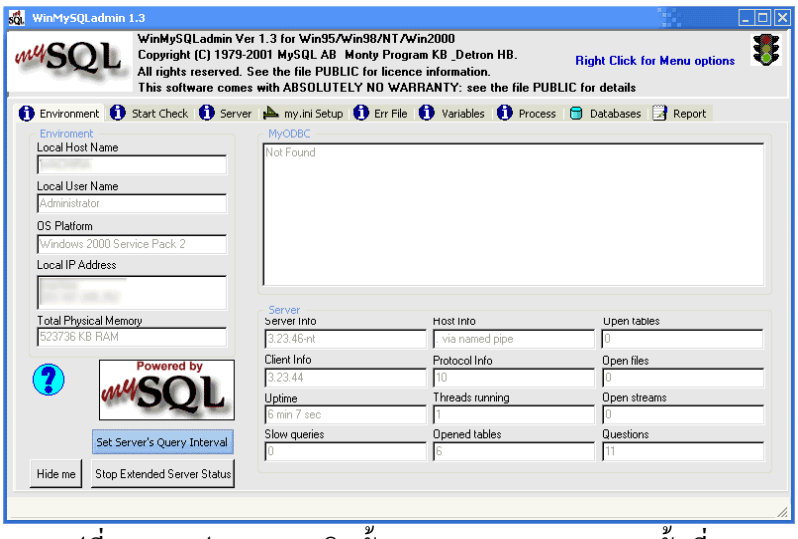
รูปที่ ง-12: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 11

12.ถ้าต้องการเปิดหน้าต่างของโปรแกรม mysqladmin ให้คลิกขวาที่สัญญาณไฟแล้วเลือก Show me ดังรูป



รูปที่ ง-13: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 12

แล้วหน้าต่างของโปรแกรมจะแสดงขึ้นมาดังรูป

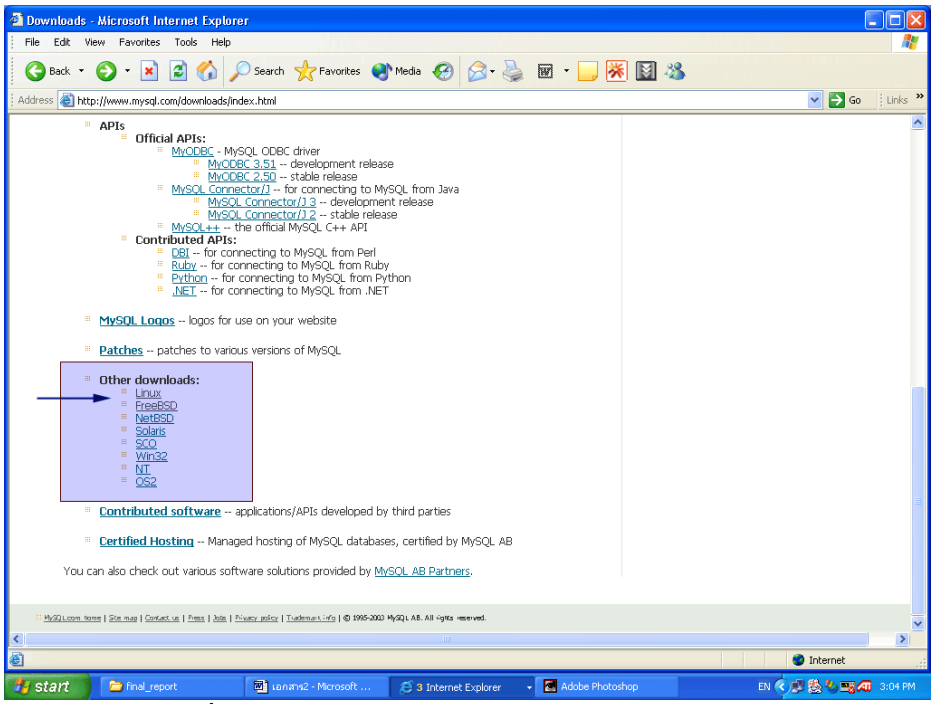


รูปที่ ง-14: รูปแสดงการติดตั้ง MySQL บน Windows ขั้นที่ 13

วิธีการติดตั้ง MySQL บน Linux

สิ่งที่ต้องเตรียมก่อนการติดตั้ง

ทำการ Download MySQL (For Unix) มาจาก http://www.mysql.com/download.html ดังรูปภาพนี้



รูปที่ ง-15: รูปแสดงการ Download MySQL for Linux

วิธีการติดตั้ง MySQL

1. เมื่อ Download มาแล้ว เราจะได้ชื่อไฟล์ mysql-3.22.16a-gamma.tar.gz ให้เก็บไฟล์นี้เอาไว้ในระบบ LINUX ของเราจากนั้น login เป็น root และทำการ unzip ไฟล์ออกมาโดยใช้ tar

   [root@linix] # tar xfvz mysql-3.22.16a-gamma.tar.gz

2. เราจะได้ไดเร็กทอรี่ mysql-3.22.16a-gamma ขึ้นมา คราวนี้ก็ทำการติดตั้งดังนี้

   [root@linix]# cd mysql-3.22.16a-gamma

   [root@linix mysql-3.22.16a-gamma]# ./configure - -prefix=/usr/local/mysql

   จากข้างต้น เราได้ติดตั้ง MySQL ลงใน /usr/local/mysql ในตอนนี้โปรแกรมก็จะทำการกำหนดค่าต่าง ๆ ไปเรื่อย ๆ สักพักก็จะปรากฏข้อความลักษณะนี้ออกมา

MySQL has a Web site at http://www.tcx.se/ which carries details on the latest release, upcoming features, and other information to make your work or play with MySQL more productive. There you can also find information about mailing lists for MySQL discussion.
Remember to check the platform specific part in the reference manual for hints about installing on your platfrom. See the Docs directory.
Thank you for choosing MySQL!

สิ้นข้อความก็จะคืนกลับมาที่ shell ของระบบให้ จากนั้นเราก็ทำการ compile และติดตั้งดังนี้

[root@linix mysql-3.22.16a-gamma]# make

[root@linix mysql-3.22.16a-gamma]# make install

3. จากนั้นพิมพ์คำสั่งต่อไปนี้ลงไปที่ shell เพื่อเพิ่ม mySQL Library เข้าไปใน ldconfig

   # echo "/usr/local/mysql/lib/mysql" >> /etc/ld.so.conf

   # ldconfig –v | grep libmysqlclient

4. เพิ่มคำสั่งรัน mySQL เข้าไปใน /etc/rc.d/rc.local เพื่อให้ mySQL เริ่มทำงานเมื่อ Linux บูตระบบ

   # echo “/usr/local/mysql/bin/safe_mysqld > /dev/null &” >> /etc/rc.d/rc.local

5. ทำการ initial ฐานข้อมูล

   # sh scripts/mysql_install_db.sh

6. รันโปรแกรม mySQL Server แบบหลังฉากโดยเรียก safe_mysql

   # /usr/local/mysql/bin/safe_mysqld > /dev/null &

ตอนนี้ เราก็สามารถทดสอบ mySQL Server ของเราได้แล้ว ซึ่งถ้าจะให้ดี เราควรจะปรับแต่งให้สามารถเรียก MySQL ได้จากทุก ๆ ตำแหน่งที่เราอยู่ภายในระบบ ให้แก้ไขไฟล์ /etc/profile โดยเพิ่มเข้าไปอีกหนึ่งบรรทัด คือ

PATH="$PATH:/usr/X11R6/bin"

PATH="$PATH:/usr/local/mysql/bin"

เป็นการระบุว่า MySQLจะสามารถเรียกใช้งานได้ถ้าเราอยู่ในตำแหน่งใดของระบบ คราวนี้ก็ reboot ระบบอีกครั้งเพื่อให้แน่ใจว่า MySQL ได้ติดตั้งสมบูรณ์แล้ว

ภาคผนวก จ

ความรู้ทั่วไปในการใช้ MySQL

การ Login เข้าสู่ MySQL

การเข้าสู่ MySQL สามารถทำได้หลายทางทั้งทาง Client โดยการใช้ Telnet หรือ Console โดย Login เป็น user ต่างๆ แต่ก่อนอื่นจะต้องทำการกำหนด password ให้กับ mysqladmin เพื่อความปลอดภัยของระบบ โดย mysqladmin เท่านั้นที่จะเป็นผู้สามารถเข้าสู่ MySQL ได้

- การกำหนด Password ให้กับ mysqladmin มีรูปแบบคำสั่งมีดังนี้

  #/usr/local/bin/mysqladmin -u root password secret

  -u ตามด้วยชื่อ user หมายถึงการกำหนดชื่อ user

  password ตามด้วย password หมายถึงการกำหนด password ให้กับ user ดังกล่าวข้างต้น

- จากนั้นก็มา Login เข้า MySQL โดยมีรูปแบบคำสั่งมีดังนี้

  #/usr/local/bin/mysql -u root password

  Enter password: ******** *(ให้ใส่ password ที่กำหนดลงไป)*

เมื่อ Login สำเร็จ หน้าจอจะแสดงเครื่องหมาย mysql> โดยมีข้อความดังนี้

```
#/usr/local/bin/mysql -u root password
Enter password: ********
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 459 to server version: 3.22.20a-log
Type 'help' for help.
mysql>
```

- การออกจาก MySQL มีรูปแบบคำสั่งดังนี้

  mysql> QUIT

นอกจากนี้เรายังขอความช่วยเหลือจาก MySQL ได้ด้วยคำสั่ง help ดังในตัวอย่างด้านล่าง

```
mysql> help
MySQL commands:
help    (\h)    Display this text
?       (\h)    Synonym for `help'
clear   (\c)    Clear command
connect (\r)    Reconnect to the server. Optional arguments are db and host
edit    (\e)    Edit command with $EDITOR
exit    (\)     Exit mysql. Same as quit
go      (\g)    Send command to mysql server
print   (\p)    print current command
quit    (\q)    Quit mysql
rehash  (\#)    Rebuild completion hash
status  (\s)    Get status information from the server
use     (\u)    Use another database. Takes database name as argument
Connection id: 6  (Can be used with mysqladmin kill)
mysql>
```

รายละเอียดในแต่ละส่วนของ Help มีดังนี้

1. Column แรก เป็นคำสั่ง
2. Column ที่สองเป็น short key เราสามารถพิมพ์ short key ได้เลยซึ่งมีผลเท่ากับคำสั่งเช่น พิมพ์ \q มีค่าเท่ากับ quit
3. Column ที่สามเป็นคำอธิบายว่าคำสั่งนี้ทำอะไร

ตัวอย่างชนิดของข้อมูลใน MySQL

ตัวอย่างชนิดของข้อมูลในฐานข้อมูล MySQL (Data Type of MySQL) ได้แก่

ตารางที่ จ.1: ตารางแสดงชนิดของข้อมูลใน MySQL

Data Type	คำอธิบาย	ตัวอย่าง
CHAR (M)	จะเอาไว้เก็บข้อมูลที่เป็น String (สายอักษร) โดยที่ขนาดของการเก็บมีความคงที่	**Firstname CHAR (25);**
VARCHAR (M)	ข้อมูลที่เป็น String (สายอักษร) โดยที่ขนาดของการเก็บขึ้นกับความยาวที่เก็บ	**Firstname CHAR (25);**
INT (M) Unsigned	INT เก็บค่าจำนวนเต็ม มีค่าตั้งแต่ -2147483648 ถึง +2147483647 แต่ถ้าใส่ Unsigned จะมีค่าได้ตั้งแต่ 0 ถึง 4294967295	**Light-year INT;** **electron INT unsigned;**
FLOAT [(M,D)]	ใช้เก็บเลขที่มีค่าเป็นทศนิยม ในตัวอย่าง เลข 4 และ 2 หมายความว่า ตัวแปรนี้เก็บค่าได้ 4 ตัว และเป็นเลขทศนิยม 2 ตำแหน่ง ตัวอย่างค่าที่เก็บลงในรูปแบบ Float ที่กำหนด • 42.35 เก็บค่าได้ถูกต้อง • 324.56 เก็บค่าเป็น 324.5 • 2.2 เก็บค่าได้ถูกต้อง • 34.524 ปัดเป็น 34.52	**Rainfall FLOAT (4,2);**
DATE	เก็บข้อมูลในรูป "YYYY-MM-DD"	**Today DATE;**
TEXT/BLOB	เก็บข้อมูลตั้งแต่ 255-65535 ตัวอักษร ข้อแตกต่างระหว่าง TEXT กับ BLOB คือ BLOB จะถือ cases sensitivity	**Comment BLOB;**
SET	เป็นกลุ่มของข้อมูลที่ยอมให้เลือกได้ 1 ค่าหรือหลายๆ ค่า สามารถกำหนดได้ถึง 64 ค่า เราสามารถเลือกได้ เป็น "", "SUT", "MIT" หรือ "MIT, KMITNB";	**University SET ("SUT", "MIT", "AIT", "KMITNB");**

Operator ที่มีอยู่ใน MySQL

ตัว Operator ใน MySQL สามารถแยกได้ 2 ประเภทดังนี้

1.Relational operator

2.Bulletin operator

Relational operator ได้แก่

ตารางที่ จ.2: ตารางแสดง Relational operator

Operator	**ความหมาย**
=	เท่ากับ
>	มากกว่า
<	น้อยกว่า
>=	มากกว่า หรือเท่ากับ
<=	น้อยกว่า หรือเท่ากับ
<>	ไม่เท่ากับ

Bulletin operator ได้แก่

ตารางที่ จ.3: ตารางแสดง Bulletin operator

Operator	**ความหมาย**
AND	และ
OR	หรือ
NOT	เป็นเท็จ

Operator นอกจากจะเป็นตัวช่วยในการสืบค้นข้อมูลแล้ว ยังนำมาช่วยในการแก้ไขและลบข้อมูลได้อีกด้วย การนำตัว Operator ไปช่วยในงานจัดการข้อมูลต่างๆ ให้มีประสิทธิภาพนั้นผู้เรียนจำเป็นต้องเข้าใจเกี่ยวกับเรื่องตรรก หรือ Logic ให้ดีก่อน ถ้าเข้าใจดีแล้วก็จะใช้ได้อย่างมีประสิทธิภาพ

สรุปคำสั่ง MySOL ที่ใช้บ่อย

ตารางที่ จ.4: ตารางแสดงคำสั่ง MySOL ที่ใช้บ่อย

COMMAND	DESCRIPTION	EXAMPLE
CREATE	CREATE database and table	CREATE DATABASES; CREATE TABLES nametable (fied1 VARCHAR (25), fied2 VARCHAR (25), fied3 INT, fied1 VARCHAR (25), ID INT NOT AUTO_INCREMENT, PRIMARY KEY (ID));
SHOW	DISPLAY database and table	SHOW DATABASES; SHOW TABLES; SHOW COLUMNS FROM tables; DESC tablename;
USE		USE database;
SELECT		SELECT * FROM (table); SELECT * FROM (table) where (fied=data) or (fied2=data2); SELECT * FROM (table) where (fied=data) and (fied2=data2);
UPDATE	EDIT database	UPDATE table SET fied='datanew' WHERE fied='dataold';
DELETE	DELETE record	DELETE FROM table WHERE (fied=data);
ALTER	RENAME table	ALTER TABLE tableold RENAME tablenew; ALTER TABLE table ADD fiednew DATATYPE;
INSERT	INSERT record	INSERT INTO table VALUES ('fied1','fied2','fied3')

DROP	DROP database DROP table	DROP DATABASE datatest; DROP TABLE tablename;
ORDER BY	SORT data	SELECT fieldname FROM tablename ORDER BY fieldname;

ภาคผนวก ฉ

ตัวอย่างไฟล์โปรเจค KCN (.KCN File)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE kcnSystemFile [
<!ELEMENT kcnSystemFile (current, history) >
<!ELEMENT current (data_c, incomming_c)>
<!ELEMENT data_c (road_c*)>
<!ELEMENT road_c EMPTY>
<!ELEMENT incomming_c (order_c*)>
<!ELEMENT order_c EMPTY>
<!ELEMENT history (turn*)>
<!ELEMENT turn (data_h, incomming_h)>
<!ATTLIST turn id ID #REQUIRED  action CDATA "0">
<!ELEMENT data_h (road_h)*>
<!ELEMENT incomming_h (order_h)*>
<!ELEMENT road_h EMPTY >
<!ELEMENT order_h EMPTY >
<!ATTLIST road_c  id ID #REQUIRED  rel_id CDATA "0"  car_out CDATA "0"  car_remain CDATA "0"
light_order CDATA "0"  light_type CDATA "0"  light_state CDATA "0"  light_length CDATA "0"
light_remain CDATA "0"  tf_flow CDATA "0"  tf_ratio CDATA "0">
<!ATTLIST order_c id ID #REQUIRED  rel_id CDATA "0"  distance CDATA "0"  carvolume CDATA "0">
<!ATTLIST road_h   id ID #REQUIRED  rel_id CDATA "0" car_out CDATA "0"  car_remain CDATA "0"
light_order CDATA "0"  light_type CDATA "0"  light_state CDATA "0"  light_length CDATA "0"
light_remain CDATA "0"  tf_flow CDATA "0"  tf_ratio CDATA "0">
<!ATTLIST order_h  id ID #REQUIRED  rel_id CDATA "0"  distance CDATA "0"  carvolume CDATA
"0">]>
<kcnSystemFile>
        <current>
                <data_c>
```

<road_c id="road_c1" rel_id="1" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="30" tf_ratio="0"></road_c>

<road_c id="road_c2" rel_id="2" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c3" rel_id="3" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c4" rel_id="4" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c5" rel_id="5" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c6" rel_id="6" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c7" rel_id="7" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c8" rel_id="8" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c9" rel_id="9" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c10" rel_id="10" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c11" rel_id="11" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="30" tf_ratio="0"></road_c>

<road_c id="road_c12" rel_id="12" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c13" rel_id="13" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c14" rel_id="14" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c15" rel_id="15" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c16" rel_id="16" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c17" rel_id="17" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c18" rel_id="18" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c19" rel_id="19" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c20" rel_id="20" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c21" rel_id="21" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c22" rel_id="22" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c23" rel_id="23" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c24" rel_id="24" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c25" rel_id="25" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c26" rel_id="26" car_out ="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c27" rel_id="27" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c28" rel_id="28" car_out ="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c29" rel_id="29" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c30" rel_id="30" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c31" rel_id="31" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c32" rel_id="32" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c33" rel_id="33" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c34" rel_id="34" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c35" rel_id="35" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c36" rel_id="36" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c37" rel_id="37" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c38" rel_id="38" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c39" rel_id="39" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c40" rel_id="40" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c41" rel_id="41" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c42" rel_id="42" car_out ="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c43" rel_id="43" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c44" rel_id="44" car_out ="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c45" rel_id="45" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="30" tf_ratio="0"></road_c>

<road_c id ="road_c46" rel_id="46" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c47" rel_id="47" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c48" rel_id="48" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c49" rel_id="49" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c50" rel_id="50" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c51" rel_id="51" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c52" rel_id="52" car_out ="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c53" rel_id="53" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c54" rel_id="54" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c55" rel_id="55" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c56" rel_id="56" car_out ="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c57" rel_id="57" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c58" rel_id="58" car_out ="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c59" rel_id="59" car_out ="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id ="road_c60" rel_id="60" car_out ="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c61" rel_id="61" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c62" rel_id="62" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c63" rel_id="63" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c64" rel_id="64" car_out="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c65" rel_id="65" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c66" rel_id="66" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c67" rel_id="67" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c68" rel_id="68" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c69" rel_id="69" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c70" rel_id="70" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c71" rel_id="71" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c72" rel_id="72" car_out="0" car_remain="2" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c73" rel_id="73" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c74" rel_id="74" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c75" rel_id="75" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c76" rel_id="76" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c77" rel_id="77" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c78" rel_id="78" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c79" rel_id="79" car_out="0" car_remain="2" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c80" rel_id="80" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c81" rel_id="81" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c82" rel_id="82" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c83" rel_id="83" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_c>

<road_c id="road_c84" rel_id="84" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_c>

```
                </data_c>
                <incomming_c>
                        <order_c id="order_c1" rel_id="3" distance="80" carvolume="1"></order_c>
                        <order_c id="order_c2" rel_id="14" distance="40" carvolume="1"></order_c>
                </incomming_c>
        </current>
        <history>
                <turn id="turn1" action="play">
                        <data_h>
                        <road_h id ="road_h1_1" rel_id="1" car_out ="1" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_2" rel_id="2" car_out ="0" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_3" rel_id="3" car_out ="0" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_4" rel_id="4" car_out ="0" car_remain="0"
light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_5" rel_id="5" car_out ="0" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_6" rel_id="6" car_out ="0" car_remain="0"
light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_7" rel_id="7" car_out ="0" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_8" rel_id="8" car_out ="0" car_remain="0"
light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h id ="road_h1_9" rel_id="9" car_out ="0" car_remain="0"
light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0"
tf_ratio="0"></road_h>
```

<road_h id="road_h1_10" rel_id="10" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_11" rel_id="11" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_12" rel_id="12" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_13" rel_id="13" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_14" rel_id="14" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_15" rel_id="15" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_16" rel_id="16" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_17" rel_id="17" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_18" rel_id="18" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_19" rel_id="19" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_20" rel_id="20" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_21" rel_id="21" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_22" rel_id="22" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_23" rel_id="23" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_24" rel_id="24" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_25" rel_id="25" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_26" rel_id="26" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_27" rel_id="27" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_28" rel_id="28" car_out="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h1_29" rel_id="29" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_30" rel_id="30" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_31" rel_id="31" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_32" rel_id="32" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_33" rel_id="33" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_34" rel_id="34" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_35" rel_id="35" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_36" rel_id="36" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_37" rel_id="37" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_38" rel_id="38" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_39" rel_id="39" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_40" rel_id="40" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_41" rel_id="41" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_42" rel_id="42" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_43" rel_id="43" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_44" rel_id="44" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_45" rel_id="45" car_out="1" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_46" rel_id="46" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_47" rel_id="47" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_48" rel_id="48" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_49" rel_id="49" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_50" rel_id="50" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_51" rel_id="51" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_52" rel_id="52" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_53" rel_id="53" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_54" rel_id="54" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_55" rel_id="55" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_56" rel_id="56" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_57" rel_id="57" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_58" rel_id="58" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_59" rel_id="59" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_60" rel_id="60" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_61" rel_id="61" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_62" rel_id="62" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_63" rel_id="63" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_64" rel_id="64" car_out="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h1_65" rel_id="65" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_66" rel_id="66" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_67" rel_id="67" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_68" rel_id="68" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_69" rel_id="69" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_70" rel_id="70" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_71" rel_id="71" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_72" rel_id="72" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h1_73" rel_id="73" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_74" rel_id="74" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_75" rel_id="75" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_76" rel_id="76" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_77" rel_id="77" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_78" rel_id="78" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_79" rel_id="79" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h1_80" rel_id="80" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_81" rel_id="81" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_82" rel_id="82" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_83" rel_id="83" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h1_84" rel_id="84" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="59" tf_flow="0" tf_ratio="0"></road_h>

</data_h>

<incomming_h>

<order_h id="order_h1_1" rel_id="3" distance="40" carvolume="1"></order_h>

</incomming_h>

</turn>

<turn id="turn2" action="stop">

<data_h>

<road_h id="road_h2_1" rel_id="1" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="60" tf_ratio="0"></road_h>

<road_h id="road_h2_2" rel_id="2" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_3" rel_id="3" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_4" rel_id="4" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_5" rel_id="5" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_6" rel_id="6" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_7" rel_id="7" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_8" rel_id="8" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_9" rel_id="9" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_10" rel_id="10" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_11" rel_id="11" car_out="1" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_12" rel_id="12" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_13" rel_id="13" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_14" rel_id="14" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_15" rel_id="15" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_16" rel_id="16" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_17" rel_id="17" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_18" rel_id="18" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_19" rel_id="19" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_20" rel_id="20" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_21" rel_id="21" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_22" rel_id="22" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_23" rel_id="23" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_24" rel_id="24" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_25" rel_id="25" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_26" rel_id="26" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_27" rel_id="27" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_28" rel_id="28" car_out="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_29" rel_id="29" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_30" rel_id="30" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_31" rel_id="31" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_32" rel_id="32" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_33" rel_id="33" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_34" rel_id="34" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_35" rel_id="35" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_36" rel_id="36" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_37" rel_id="37" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_38" rel_id="38" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_39" rel_id="39" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_40" rel_id="40" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_41" rel_id="41" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_42" rel_id="42" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_43" rel_id="43" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_44" rel_id="44" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_45" rel_id="45" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="60" tf_ratio="0"></road_h>

<road_h id="road_h2_46" rel_id="46" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_47" rel_id="47" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_48" rel_id="48" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_49" rel_id="49" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_50" rel_id="50" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_51" rel_id="51" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_52" rel_id="52" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_53" rel_id="53" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_54" rel_id="54" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_55" rel_id="55" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_56" rel_id="56" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_57" rel_id="57" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_58" rel_id="58" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_59" rel_id="59" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_60" rel_id="60" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_61" rel_id="61" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_62" rel_id="62" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_63" rel_id="63" car_out="0" car_remain="0" light_order="2" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_64" rel_id="64" car_out="0" car_remain="1" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_65" rel_id="65" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_66" rel_id="66" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h2_67" rel_id="67" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_68" rel_id="68" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h2_69" rel_id="69" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_70" rel_id="70" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_71" rel_id="71" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_72" rel_id="72" car_out="0" car_remain="2" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h2_73" rel_id="73" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_74" rel_id="74" car_out="0" car_remain="1" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="1"></road_h>

<road_h id="road_h2_75" rel_id="75" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_76" rel_id="76" car_out="0" car_remain="0" light_order="3" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_77" rel_id="77" car_out="0" car_remain="0" light_order="4" light_type="single" light_state="red" light_length="60" light_remain="0" tf_flow="0" tf_ratio="0"></road_h>

<road_h id="road_h2_78" rel_id="78" car_out="0" car_remain="0" light_order="1" light_type="single" light_state="green" light_length="60" light_remain="58" tf_flow="0" tf_ratio="0"></road_h>

```
                        <road_h  id ="road_h2_79"  rel_id="79" car_out ="0"  car_remain="2"
light_order="4"  light_type="single"  light_state="red"  light_length="60"  light_remain="0"  tf_flow="0"
tf_ratio="1"></road_h>
                        <road_h  id ="road_h2_80"  rel_id="80" car_out ="0"  car_remain="0"
light_order="1"  light_type="single"  light_state="green"  light_length="60"  light_remain="58"  tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h  id ="road_h2_81"  rel_id="81" car_out ="0"  car_remain="0"
light_order="4"  light_type="single"  light_state="red"  light_length="60"  light_remain="0"  tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h  id ="road_h2_82"  rel_id="82" car_out ="0"  car_remain="0"
light_order="1"  light_type="single"  light_state="green"  light_length="60"  light_remain="58"  tf_flow="0"
tf_ratio="0"></road_h>
                        <road_h  id ="road_h2_83"  rel_id="83" car_out ="0"  car_remain="1"
light_order="4"  light_type="single"  light_state="red"  light_length="60"  light_remain="0"  tf_flow="0"
tf_ratio="1"></road_h>
                        <road_h  id ="road_h2_84"  rel_id="84" car_out ="0"  car_remain="0"
light_order="1"  light_type="single"  light_state="green"  light_length="60"  light_remain="58"  tf_flow="0"
tf_ratio="0"></road_h>
                        </data_h>
                        <incomming_h>
                                <order_h  id="order_h2_1"  rel_id="3"  distance="80"
carvolume="1"></order_h>
                                <order_h  id="order_h2_2"  rel_id="14"  distance="40"
carvolume="1"></order_h>
                        </incomming_h>
                </turn>
        </history>
</kcnSystemFile>
```

บรรณานุกรม

[1] Horton, Thamus R.,**"Traffic Control Theroy and Instrumentation"** ,Plenum Press,New York,1965.

[2] Drew, Donald R.,**"Traffic Flow Theory and Control"** ,McGraw-Hill,New York.

[3] Law, Averill M. and W. David Kelton,**"Simulation Modeling and Analysis"** ,McGraw-Hill ,New York ,1991.

[4] Traffic, **"Traffic in Towns"** ,Her Majesty' Stationery Office,1963.

[5] ขยัน จันทรสถาพร, **"XML Extensible Markup Language"**,บริษัท เอ.อาร์.อินฟอร์เมชัน แอนด์ พับลิเคชัน จำกัด,พ.ศ. 2544.

[6] วีระศักดิ์ ซึงถาวร, **"JAVA Programming Volume1"**, บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน), พ.ศ. 2543.

[7] สงกรานต์ ทองสว่าง. **"MySQL ระบบฐานข้อมูลสำหรับอินเตอร์เน็ต"**, บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน), พ.ศ. 2543.

[8] www.njcmr.org\mtids\deet-eng\IE\Simulation\Chapter1%20-%20Final.DOC

[9] http://www.thaixml.com/

[10] http://www.thaiwbi.com/course/mysql/