ระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้า

EV Tracking Network and Display System

นายเหมวิจักษณ์ บุญมาก รหัส 51362213

นายวิชิต ธรรมพิทักษ์ รหัส 51364965

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 25 ธ.ค. 2556
เลขทะเบียน 16176829
เลขเรียกหนังสือ ม.
มหาวิทยาลัยนเรศวร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2554

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน	ระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้า
ผู้ดำเนินโครงงาน	นายเหมวิจักษณ์ บุญมาก รหัส 51362213
	นายวิชิต ธรรมพิทักษ์ รหัส 51364965
ที่ปรึกษาโครงงาน	อาจารย์เศรษฐา ตั้งค้าวานิช
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2554

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

.....ประธานกรรมการ

(อาจารย์ภาณุพงศ์ สอนคม)

.....กรรมการ

(ดร.พงศ์พันธ์ กิจสนาโยธิน)

.....กรรมการ

(ดร.สุรเดช จิตประไพกุลศาล)

.....กรรมการ

(อาจารย์เศรษฐา ตั้งค้าวานิช)

หัวข้อโครงงาน	ระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้า		
ผู้ดำเนินโครงงาน	นายเหมวิจักษณ์	บุญมาก	รหัส 51362213
	นายวิชิต	ธรรมพิทักษ์	รหัส 51364965
อาจารย์ที่ปรึกษา	อาจารย์เศรษฐา	ตั้งค้าวานิช	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2554		

.....

บทคัดย่อ

โครงงานนี้ถูกพัฒนาขึ้นเป็นระบบเครือข่ายแบบวงแหวนโดยนำเอาเทคโนโลยีการเชื่อมต่อแบบไร้สาย (Wireless) เข้ามาประยุกต์ใช้กับโครงงาน เพื่อการติดตามตำแหน่งของรถไฟฟ้าโดยรับข้อมูลตำแหน่งของรถไฟฟ้าจากโครงงานการติดตามรถไฟฟ้าโดยระบบบอกตำแหน่งพิกัดโลกและการสื่อสารแบบไร้สายเข้าสู่ไมโครคอนโทรลเลอร์แล้วนำข้อมูลตำแหน่งของรถไฟฟ้าไปแสดงผล ซึ่งจะมีการแสดงผล 2 แบบ คือ ป้ายแสดงผลตามจุดรอรถต่าง ๆ โดยใช้หลอด LED แสดงบริเวณตำแหน่งของรถไฟฟ้าบนแผนที่ และแสดงผลบนหน้าเว็บโดยการแสดงภาพกราฟฟิกได้เกือบทันทีทันใด การทำงานหลักแบ่งได้เป็น 2 ส่วน ได้แก่ ส่วนแรกคือ การรับส่งข้อมูลเกี่ยวกับตำแหน่งของรถไฟฟ้า ส่วนที่สองคือ การแสดงผลตำแหน่งของรถไฟฟ้าผ่านทางป้ายแสดงผลและหน้าเว็บ โครงงานนี้เป็นการนำเทคโนโลยีการเชื่อมต่อไร้สายมาประยุกต์ใช้ด้านการคมนาคมขนส่ง เพื่อให้ผู้ดูแลระบบและผู้ใช้บริการมีความสะดวกสบายยิ่งขึ้น

Project title	EV Tracking Network and Display System		
Name	Mr.Hemwijak	Bunmark	ID. 51362213
	Mr.Wichit	Thampitak	ID. 51364965
Project advitor	Mr.Settha	Tangkawanit	
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic Year	2554		

Abstract

EV tracking network and display system is the project that developed for ring network by brings wireless technology to apply with this project for get latitude and longitude from EV tracking via GPS and wireless communication project to microcontroller. Moreover this project able to displays electric vehicle's latitude and longitude in two ways such as displays on Naresuan university map by plug-in LED into each station that electric vehicle arrived in the map. In addition, this project can display vehicle's latitude and longitude on website by virtual real time graphic user interface. Core concepts of this project include two parts. First is receiving electric vehicle's latitude and longitude and second part is monitoring electric vehicle's latitude and longitude on the map and website. In summary, this project is created for electric vehicle user will get more convenience in Naresuan university transportation service.

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องด้วยความกรุณาเป็นอย่างยิ่งจากอาจารย์ที่ปรึกษา อ.เศรษฐา ตั้งค้าวานิช ที่คอยให้คำปรึกษาและช่วยแนะนำการแก้ปัญหาต่าง ๆ ที่พบในโครงงาน และให้ความกรุณาในการตรวจทานปริญญานิพนธ์ และขอขอบพระคุณ คณะกรรมการทั้ง 3 ท่าน อ.ภานุพงศ์ สอนคม ดร.พงศ์พันธ์ กิจสนาโยธิน และ ดร.สุรเดช จิตประไพกุลศาล ที่ช่วยแนะนำสิ่งที่ควรปรับปรุงโครงงาน

ขอขอบคุณคณาจารย์ทุกท่านที่ประสิทธ์ประสาทวิชาความรู้ให้กับคณะผู้จัดทำโครงงาน

ขอขอบคุณคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวรที่ให้แนวคิดและอนุเคราะห์ทุนในการวิจัยโครงงาน

ขอขอบคุณภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ช่วยอนุเคราะห์สถานที่ในการจัดทำโครงงาน

และสุดท้ายนี้ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยช่วยเหลือและคอยให้การสนับสนุนในทุกด้านจนทำให้โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี

นายเหมวิจักษณ์ บุญมาก

นายวิชิต ธรรมพิทักษ์

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ก

บทคัดย่อภาษาอังกฤษ..... ข

กิตติกรรมประกาศ..... ค

สารบัญ..... ง

สารบัญตาราง..... ฉ

สารบัญรูป..... ช

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ..... 1

1.2 วัตถุประสงค์..... 1

1.3 ประโยชน์ที่คาดว่าจะได้รับ..... 1

1.4 ขอบเขตของโครงงาน..... 2

1.5 ขั้นตอนการดำเนินงาน..... 2

1.6 แผนการดำเนินงาน..... 3

1.7 งบประมาณของโครงงาน..... 3

บทที่ 2 หลักการและทฤษฎี

2.1 รูปแบบการเชื่อมโยงเครือข่าย..... 4

2.2 .NET Framework..... 8

2.3 ASP.NET 9

2.4 การทำงานของอุปกรณ์ต่างๆ..... 10

2.5 หลักการทำงานของวงจร..... 12

บทที่ 3 วิธีการดำเนินโครงงาน

3.1 การออกแบบรูปแบบการของเชื่อมต่อของระบบติดตาม และแสดงผลของรถไฟฟ้า..... 15

3.2 การออกแบบรูปแบบของข้อมูล (Data package)..... 17

3.3 ส่วนประกอบทางฝั่งป้ายแสดงผล..... 18

3.4 ส่วนประกอบทางฝั่งแม่ข่าย..... 21

บทที่ 4 ผลการทดลอง

4.1 ส่วนป้ายแสดงผล..... 31

4.2 ส่วนการแสดงผลฝั่งแม่ข่าย..... 33

4.3 การทดลองวัดความคลาดเคลื่อนของจุดที่แสดงบนแผนที่..... 40

บทที่ 5 บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการทดลองของโครงงาน..... 41

5.2 วิเคราะห์ข้อดีและข้อเสียของโครงงาน..... 42

5.3 ปัญหาที่พบและแนวทางการแก้ไขปัญหา..... 42

5.4 ข้อเสนอแนะ..... 43

เอกสารอ้างอิง..... 44

ภาคผนวก ก..... 45

ภาคผนวก ข..... 47

ภาคผนวก ค..... 49

ภาคผนวก ง..... 54

ภาคผนวก จ..... 60

ประวัติผู้เขียนโครงงาน..... 62

สารบัญตาราง

ตารางที่ หน้า

1.1 ตารางแสดงระยะเวลาแผนการดำเนินงาน..... 3

4.1 แสดงค่าวัดความคลาดเคลื่อนของจุดที่แสดงบนแผนที่..... 40

5.1 แสดงปัญหาที่พบและแนวทางการแก้ไขปัญหา..... 42

สารบัญรูป

รูปที่ หน้า

2.1 โครงสร้างการเชื่อมต่อแบบบัส..... 4

2.2 โครงสร้างการเชื่อมต่อแบบวงแหวน..... 6

2.3 โครงสร้างการเชื่อมต่อแบบดาว..... 7

2.4 โครงสร้างการเชื่อมต่อแบบ Hybrid..... 8

2.5 แสดงหน้าที่ของพอร์ทต่าง ๆ..... 11

2.6 แสดงลักษณะของเร็กกูเลเตอร์แบบอนุกรม..... 13

2.7 แสดงลักษณะของเร็กกูเลเตอร์แบบขนาน..... 13

2.8 ภาพของไอซีเร็กกูเลเตอร์แบบสามขา..... 14

2.9 ตำแหน่งขาสัญญาณและวงจรภายในของไอซีปรับแรงดัน MAX232..... 14

3.1 ระบบโดยรวมของระบบเครือข่ายและแสดงผลสำหรับการติดตามรถไฟฟ้า..... 15

3.2 ป้ายแสดงผล..... 16

3.3 รูปแบบข้อมูลที่ใช้ในการสื่อสารในเครือข่าย..... 17

3.4 โครงสร้างของระบบฝั่งป้ายแสดงผล..... 18

3.5 ระบบการทำงานฝั่งป้ายแสดงผล..... 19

3.6 ผังงานการควบคุมการติดดับหลอดไฟของระบบแสดงผล..... 20

3.7 โครงสร้างการทำงานทางฝั่งแม่ข่าย..... 21

3.8 โครงสร้างหลักของโปรแกรมรับข้อมูลจากไมโครคอนโทรลเลอร์..... 21

3.9 โครงสร้างของหน้าเว็บแสดงตำแหน่งรถไฟฟ้าแบบ View Map..... 23

3.10 แผนจาก Google Map 24

3.11 แผนที่สำหรับวาดตำแหน่งรถไฟฟ้า..... 25

3.12 โครงสร้างของหน้าเว็บแสดงตำแหน่งรถไฟฟ้าแบบ Text Log..... 26

3.13 โครงสร้างของหน้าหลักของส่วน Summary..... 27

3.14 โครงสร้างของหน้าแสดงพิกัดของส่วน Summary..... 28

3.15 ผังการทำงานของโปรแกรมรับข้อมูลทางพอร์ตอนุกรม..... 29

3.16 ผังงานการทำงานของระบบแสดงผลทางหน้าเว็บ..... 30
4.1 แสดงการต่อไฟเลี้ยงให้กับวงจรด้วยไฟฟ้า 220 โวลต์..... 31
4.2 ป้ายแสดงผลเริ่มต้นเมื่อยังไม่มีการแสดงผล..... 32
4.3 แสดงสถานะไฟเมื่อรถอยู่ที่ป้ายก่อนหน้า 2 ป้ายขึ้นไป..... 32
4.4 แสดงสถานะเมื่อรถเข้าสู่ป้ายก่อนหน้า 1 ป้าย..... 32
4.5 แสดงสถานะเมื่อรถเข้าสู่ป้ายของตนเอง..... 32
4.6 แสดงผลของโปรแกรมเมื่อได้รับข้อมูล..... 33
4.7 แสดงผลของโปรแกรมเมื่อกรอกคอมพอร์ตผิด..... 33
4.8 แสดงผลภายหลังไฟล์เมื่อไฟล์ถูกลบ..... 34
4.9 Output ส่วน View Map เมื่อได้รับข้อมูล Input 1..... 35
4.10 Output ส่วน View Map เมื่อได้รับข้อมูล Input 1..... 35
4.11 Output ส่วน View Map เมื่อได้รับข้อมูล Input 2..... 36
4.12 Output ส่วน View Map เมื่อได้รับข้อมูล Input 2..... 36
4.13 Output ส่วน View Map เมื่อได้รับข้อมูล Input 3..... 37
4.14 Output ส่วน View Map เมื่อได้รับข้อมูล Input 3..... 37
4.15 Output ส่วน View Map เมื่อได้รับข้อมูล Input 4..... 38
4.16 Output ส่วน View Map เมื่อได้รับข้อมูล Input 4..... 38
4.17 หน้าหลักของส่วน Summary..... 39
4.18 หน้าแสดงพิกัดรถไฟฟ้าของส่วน Summary..... 39
ก1 แสดงโค้ดส่วนโปรแกรมหลัก..... 45
ก2 แสดงโค้ดส่วนการทำให้หลอดไฟสว่าง..... 46
ก3 แสดงโค้ดส่วนของการส่งข้อมูล..... 46
ข1 แสดงโค้ดจาวาสคริปต์ตั้งเวลาเรียกฟังก์ชัน..... 47
ข2 แสดงโค้ดแปลงพิกัดโลกเป็นตำแหน่งพิกเซล..... 47
ข3 แสดงโค้ดสำหรับวาดตำแหน่งรถไฟฟ้า..... 48

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันทางมหาวิทยาลัยนเรศวรมีการให้บริการรถไฟฟ้าภายในมหาวิทยาลัย ซึ่งรถไฟฟ้าจะมีเส้นทางให้บริการ 2 สาย โดยทั้ง 2 สายจะผ่านทุกคณะเพื่อให้ผู้ใช้บริการสามารถเดินทางไปที่ต่างๆได้สะดวกขึ้น แต่ยังพบปัญหาคือเวลาที่รถไฟฟ้ามาถึงจุดบริการนั้นมีความไม่แน่นอน ทำให้ผู้ใช้บริการเสียเวลาในการรอ

โครงงานนี้จัดทำระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้าได้นำเทคโนโลยีการเชื่อมต่อไร้สาย (Wireless) เข้ามาประยุกต์ใช้ เพื่อใช้รับข้อมูลเกี่ยวกับตำแหน่งของรถไฟฟ้าเข้าสู่บอร์ดไมโครคอนโทลเลอร์แล้วนำข้อมูลไปแสดงผลที่ป้ายแสดงผล โดยป้ายแสดงผลจะใช้ภาพแผนที่ของมหาวิทยาลัยนเรศวรที่แทนตำแหน่งของรถไฟฟ้าด้วยหลอดไฟ (LED) และจะมีการส่งข้อมูลต่อกันไปเรื่อยๆ เพื่อแสดงตำแหน่งล่าสุดของรถไฟฟ้าและข้อมูลจะถูกส่งเข้าสู่เซิร์ฟเวอร์และนำไปแสดงผลออกหน้าเว็บ และเพื่อช่วยในการตรวจสอบตำแหน่งของรถไฟฟ้าได้สะดวกและดียิ่งขึ้น เมื่อผู้ใช้ทราบตำแหน่งของรถไฟฟ้าแล้วก็จะสามารถประมาณเวลาที่รถไฟฟ้าจะมาถึงได้สะดวกยิ่งขึ้น

การพัฒนาระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้า เป็นการมุ่งเน้นในการเพิ่มความสะดวกให้กับผู้ใช้บริการรถไฟฟ้า

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อสร้างตัวต้นแบบของระบบแสดงผลระบุตำแหน่งของรถไฟฟ้าภายในมหาวิทยาลัยนเรศวร

1.2.2 เพื่ออำนวยความสะดวกให้กับผู้ใช้บริการรถไฟฟ้าภายในมหาวิทยาลัยนเรศวรให้สามารถทราบตำแหน่งของรถไฟฟ้าได้ตลอดเวลา

1.3 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน

1.3.1 สามารถสร้างตัวต้นแบบของระบบแสดงผลระบุตำแหน่งของรถไฟฟ้าได้ตามที่คาดหวัง

1.3.2 สามารถอำนวยความสะดวกให้กับผู้ใช้บริการรถไฟฟ้าภายในมหาวิทยาลัยนเรศวร

1.4 ขอบเขตของโครงงาน

1.4.1 ระบบนี้สามารถรับค่าข้อมูล รหัสประจำรถไฟฟ้า ละติจูด และลองติจูด ได้จากระบบบอกตำแหน่งพิกัดโลกของรถไฟฟ้าได้

1.4.2 ระบบนี้สามารถทำงานในที่โล่งแจ้งและในสภาวะอากาศที่ปลอดโปร่งได้เท่านั้น

1.4.3 ระบบนี้สามารถแสดงข้อมูลของรถไฟฟ้าผ่านป้ายแสดงผลขนาดใหญ่ได้

1.4.4 ระบบนี้สามารถรับส่งข้อมูลแบบไร้สายไปยังเสารับสัญญาณที่อยู่ใกล้ได้อย่างปกติในระยะไม่เกิน 200 เมตร

1.4.5 ระบบนี้สามารถส่งข้อมูลแบบอนุกรมไร้สายจากเสาไปยังเสาถัดไป จนถึงเครื่องคอมพิวเตอร์แม่ข่ายได้

1.4.6 ระบบนี้สามารถจัดเก็บข้อมูลรถไฟฟ้าลงฐานข้อมูลที่เครื่องแม่ข่ายได้

1.4.7 ระบบนี้สามารถติดตามผลการเดินรถไฟฟ้าผ่านเครื่องแม่ข่ายในรูปแบบ GUI ได้

1.4.8 ระบบนี้เป็นระบบต้นแบบที่จำลองการทำงานเฉพาะป้ายแดงจำนวน 3 ป้ายแสดงผลและเครื่องแม่ข่าย 1 เครื่อง

1.5 ขั้นตอนการดำเนินงาน

1.5.1 สอบถามความต้องการ (Requirement) ของผู้ใช้บริการรถไฟฟ้า

1.5.2 ออกแบบระบบรับส่งข้อมูลแบบไร้สายด้วยบอร์ดไมโครคอนโทลเลอร์

1.5.3 ศึกษาข้อมูลการทำงานของอุปกรณ์รับส่งแบบไร้สาย

1.5.4 ออกแบบ Web Server

1.5.5 ศึกษาข้อมูลเกี่ยวกับบอร์ดไมโครคอนโทลเลอร์

1.5.6 ศึกษา Data frame และการทำ Web Server (ASP.NET)

1.5.7 ทดสอบและบันทึกผลการทำงานของระบบการรับส่งข้อมูลและการแสดงผล

1.5.8 สรุปผลและจัดทำรูปเล่มโครงงาน

1.6 แผนการดำเนินงาน

ตารางที่ 1.1 ตารางแสดงระยะเวลาแผนการดำเนินงาน

กิจกรรม	ปี 2554						ปี 2555			
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.6.1 เก็บ Requirement ของผู้ใช้บริการรถไฟฟ้า	↔									
1.6.2 ออกแบบระบบรับส่งข้อมูลด้วยบอร์ดไมโครคอนโทลเลอร์แบบไร้สาย		←	—	—	—	—	—	→		
1.6.3 ศึกษาข้อมูลการทำงานของอุปกรณ์รับส่งแบบไร้สาย		←	—	—	—	—	—	→		
1.6.4 ออกแบบ Web Server		←	—	—	—	—	—	→		
1.6.5 ศึกษาข้อมูลเกี่ยวกับบอร์ดไมโครคอนโทลเลอร์		←	—	—	—	—	—	→		
1.6.6 ศึกษา Data frame และการทำ Web Server (ASP.NET)			←	—	—	—	—	→		
1.6.7 ทดสอบและบันทึกผลการทำงานของระบบการรับส่งข้อมูลและการแสดงผล								↔	→	
1.6.8 สรุปผลและจัดทำรูปเล่มโครงงาน								←	—	→

1.7 งบประมาณ

1.7.1 ค่าวัสดุ (อุปกรณ์อิเล็กทรอนิกส์ และ อื่นๆ ที่เกี่ยวข้อง) เป็นเงิน 1,000 บาท

1.7.2 ค่าทำเอกสาร เป็นเงิน 1,000 บาท

รวมทั้งสิ้นเป็นเงิน 2,000 บาท

(สองพันบาทถ้วน)

หมายเหตุ ถัวเฉลี่ยทุกรายการ

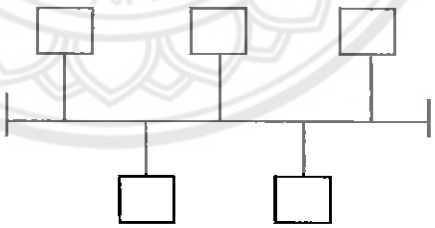
บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎี หลักการ และความรู้เบื้องต้นที่กลุ่มผู้จัดทำโครงงานได้ทำการศึกษาค้นคว้าและคาดว่ามีความเกี่ยวข้องกับโครงงานนี้ โดยจะกล่าวถึงระบบการทำงานของไมโครคอนโทรลเลอร์โดยใช้การสื่อสารไร้สาย ซึ่งการสื่อสารนี้จะรับส่งกันเองโดยไม่พึ่งสัญญาณจากที่อื่น ๆ และคาดหวังว่าจะใช้สัญญาณอินเตอร์เน็ตจริงในการรับส่งข้อมูลในรุ่นต่อไป

2.1 รูปแบบการเชื่อมโยงเครือข่าย

โทโปโลยี คือ ลักษณะทางกายภาพของระบบเครือข่าย ซึ่งหมายถึงลักษณะของการเชื่อมโยงสายสื่อสารเข้ากับอุปกรณ์อิเล็กทรอนิกส์และเครื่องคอมพิวเตอร์ภายในเครือข่ายด้วยกัน โทโปโลยีของข่ายงานบริเวณเฉพาะที่ (LAN) แต่ละแบบจะมีความเหมาะสมในการใช้งานที่แตกต่างกันออกไป การนำไปใช้จึงจำเป็นต้องทำการศึกษาลักษณะและคุณสมบัติข้อดีและข้อเสียของโทโปโลยีแต่ละแบบ เพื่อนำไปใช้ในการออกแบบเครือข่ายให้เหมาะสมกับการใช้งาน ซึ่งรูปแบบของโทโปโลยีหลัก ๆ ก็จะมีดังต่อไปนี้ [1]

2.1.1 โทโปโลยีแบบบัส



แบบบัส

รูปที่ 2.1 โครงสร้างการเชื่อมต่อแบบบัส

ที่มา: http://u51132792063.blogspot.com/2011/01/5-5-25534-a1.html

เป็นรูปแบบที่ใช้เชื่อมต่อคอมพิวเตอร์เข้าด้วยกันโดยผ่านสายสัญญาณแกนหลัก ที่เรียกว่า BUS หรือ แบ็คโบน (Backbone) โดยใช้เป็นทางเดินข้อมูลของทุกเครื่องภายในระบบเครือข่าย และจะมีสายแยกย่อยออกไปในแต่ละจุด เพื่อเชื่อมต่อกับคอมพิวเตอร์เครื่องอื่น ๆ ซึ่งเรียกว่าโหนด (Node) ข้อมูลจากโหนดผู้ส่งจะถูกส่งเข้าสู่สายบัสในรูปของแพ็กเกจ ปลายทั้ง 2 ด้านของบัสจะมีเทอร์มิเนเตอร์ (Terminator) ทำหน้าที่ลบล้างสัญญาณที่ส่งมาถึง เพื่อป้องกันไม่ให้สัญญาณข้อมูลนั้นสะท้อนกลับเข้ามายังบัสอีก และเพื่อป้องกันการชนกันของข้อมูลอื่น ๆ ที่เดินทางอยู่บนบัสในขณะนั้น

สัญญาณข้อมูลจากโหนดผู้ส่งเมื่อเข้าสู่บัส ข้อมูลจะไหลผ่านไปยังปลายทั้ง 2 ด้านของบัส ซึ่งทุก ๆ โหนดภายในเครือข่ายแบบ BUS นั้นสามารถรับรู้สัญญาณข้อมูลได้ แต่จะมีเพียงโหนดปลายทางเพียงโหนดเดียวเท่านั้นที่จะรับข้อมูลนั้นไปได้

ข้อดี

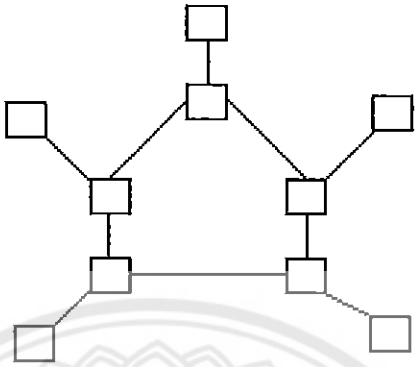
- ไม่ต้องเสียค่าใช้จ่ายในการวางสายสัญญาณมากนัก สามารถขยายระบบได้ง่าย เสียค่าใช้จ่ายน้อย ได้รับความนิยมใช้กันมากที่สุดมาตั้งแต่อดีตจนถึงปัจจุบัน เหตุผลอย่างหนึ่งก็คือสามารถติดตั้งระบบ ดูแลรักษา และติดตั้งอุปกรณ์เพิ่มเติมได้ง่าย

ข้อเสีย

- หากมีสัญญาณขาดที่ตำแหน่งใดตำแหน่งหนึ่งก็จะทำให้เครื่องบางเครื่อง หรือทั้งหมดในระบบไม่สามารถใช้งานได้ตามไปด้วย

- การตรวจหาโหนดเสียทำได้ยาก เนื่องจากขณะใดขณะหนึ่งจะมีคอมพิวเตอร์เพียงเครื่องเดียวเท่านั้นที่สามารถส่งข้อความออกมาบนสายสัญญาณ ดังนั้นถ้ามีเครื่องคอมพิวเตอร์จำนวนมากๆ อาจทำให้เกิดการคับคั่งของเน็ตเวิร์ค ซึ่งจะทำให้ระบบช้าลงได้

2.1.2 โทโปโลยีแบบวงแหวน



แบบวงแหวน

รูปที่ 2.2 โครงสร้างการเชื่อมต่อแบบวงแหวน

ที่มา: http://u51132792063.blogspot.com/2011/01/5-5-25534-a1.html

เป็นรูปแบบที่เครื่องคอมพิวเตอร์ทุกเครื่องในระบบเครือข่ายทั้งเครื่องที่เป็นผู้ให้บริการ (Server) และ เครื่องที่เป็นผู้ขอใช้บริการ(Client) ทุกเครื่องถูกเชื่อมต่อกันเป็นวงกลม ข้อมูลข่าวสารที่ส่งระหว่างกันจะไหลวนอยู่ในเครือข่ายไปในทิศทางเดียวกัน ในแต่ละโหนดหรือแต่ละเครื่องจะมีรีพีตเตอร์ (Repeater) ประจำแต่ละเครื่อง 1 ตัว ซึ่งจะทำหน้าที่เพิ่มเติมข้อมูลที่จำเป็นต่อการติดต่อสื่อสารเข้าในส่วนหัวของแพ็กเกจที่ส่ง และตรวจสอบข้อมูลจากส่วนหัวของ Packet ที่ส่งมาถึง ว่าเป็นข้อมูลของตนหรือไม่ แต่ถ้าไม่ใช่ก็จะปล่อยข้อมูลนั้นไปยัง Repeater ของเครื่องถัดไป

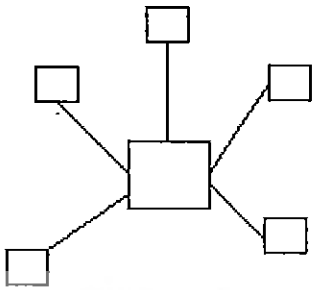
ข้อดี

- ผู้ส่งสามารถส่งข้อมูลไปยังผู้รับได้หลาย ๆ เครื่องพร้อม ๆ กัน
- ไม่มีการชนกันของสัญญาณข้อมูลที่ส่งออกไป
- คอมพิวเตอร์ทุกเครื่องในเน็ตเวิร์กมีโอกาสที่จะส่งข้อมูลได้อย่างทัดเทียมกัน

ข้อเสีย

- ถ้ามีเครื่องใดเครื่องหนึ่งในเครือข่ายเสียหายข้อมูลจะไม่สามารถส่งผ่านไปยังเครื่อง ต่อ ๆ ไปได้ และจะทำให้เครือข่ายทั้งเครือข่ายหยุดชะงักได้
- ขณะที่ข้อมูลถูกส่งผ่านแต่ละเครื่องเวลาส่วนหนึ่งจะสูญเสียไปกับการที่ทุก ๆ Repeater จะต้องทำการตรวจสอบตำแหน่งปลายทางของข้อมูลนั้น ๆ ทุกข้อมูล

2.1.3 โทโปโลยีแบบดาว



แบบดาว

รูปที่ 2.3 โครงสร้างการเชื่อมต่อแบบดาว

ที่มา: http://u51132792063.blogspot.com/2011/01/5-5-25534-a1.html

เป็นรูปแบบที่ เครื่องคอมพิวเตอร์ทุกเครื่องที่เชื่อมต่อเข้าด้วยกันในเครือข่าย จะต้องเชื่อมต่อกับอุปกรณ์ตัวกลางตัวหนึ่งที่เรียกว่า ฮับ (HUB) หรือเครื่อง ๆ หนึ่ง ซึ่งทำหน้าที่เป็นศูนย์กลางของการเชื่อมต่อสายสัญญาณที่มาจากเครื่องต่าง ๆ ในเครือข่าย เมื่อมีเครื่องที่ต้องการส่งข้อมูลไปยังเครื่องอื่น ๆ ที่ต้องการในเครือข่าย เครื่องนั้นก็จะต้องส่งข้อมูลมายัง HUB หรือเครื่องศูนย์กลางก่อน แล้ว HUB ก็จะทำหน้าที่กระจายข้อมูลนั้นไปในเครือข่ายต่อไป

ข้อดี

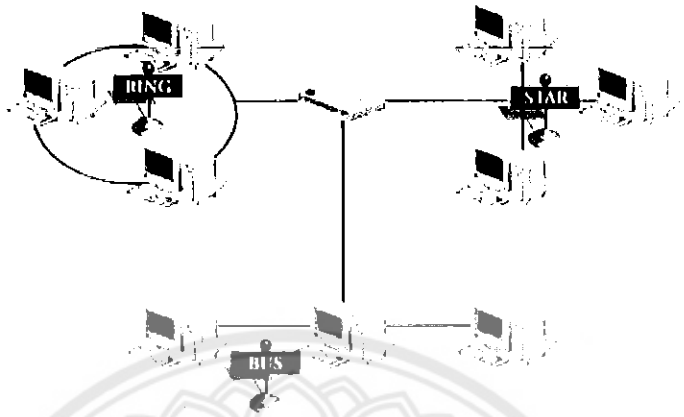
- การติดตั้งเครือข่ายและการดูแลรักษาทำได้ง่ายหากมีเครื่องใดเกิดความเสียหายก็สามารถตรวจสอบได้ง่าย

ข้อเสีย

- เสียค่าใช้จ่ายมากทั้งในด้านของเครื่องที่จะใช้เป็นเครื่องศูนย์กลางหรือตัว HUB เอง และค่าใช้จ่ายในการติดตั้งสายเคเบิลในเครื่องอื่น ๆ ทุกเครื่อง

- การขยายระบบให้ใหญ่ขึ้นทำได้ยาก เพราะการขยายแต่ละครั้งจะต้องเกี่ยวเนื่องกับเครื่องอื่น ๆ ทั้งระบบ

2.1.4 โทโปโลยีแบบ Hybrid



รูปที่ 2.4 โครงสร้างการเชื่อมต่อแบบ Hybrid

ที่มา: http://blog.cstc.ac.th/node/126.

เป็นรูปแบบใหม่ที่เกิดจากการผสมผสานกันของโทโปโลยีแบบ STAR , BUS , RING เข้าด้วยกัน เพื่อเป็นการลดข้อเสียของรูปแบบที่กล่าวมาและเพิ่มข้อดีขึ้นมา มักจะนำมาใช้กับระบบ WAN (Wide Area Network) มาก ซึ่งการเชื่อมต่อกันของแต่ละรูปแบบนั้นต้องใช้ตัวเชื่อมสัญญาณเข้ามาเป็นตัวเชื่อมตัวนั้นก็คือ Router เป็นตัวเชื่อมการติดต่อกัน [1]

2.2 .NET Framework

จากการที่ Microsoft ต้องการที่จะสร้างระบบภาษาสักอย่างที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่างและอุปกรณ์ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด Microsoft ได้คิดค้นระบบระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสภาวะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบ .NET นี้ได้ .NET Framework นั้นมีส่วนประกอบ ภายในแบ่งออกเป็น 3 ชั้นใหญ่ ๆ คือ [2]

2.2.1 Programming Language

เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสภาวะที่เป็น .NET ได้โดย ภาษาหลัก ๆ ที่จะใช้ในการพัฒนาบน .NET นี้ 3 ภาษา

- C# เป็น ภาษา ใหม่ที่ Microsoft พัฒนามาจาก C++ กับ JAVA เป็นหลัก
- VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชั่น 6.0
- JScript.net เป็นภาษาที่พัฒนามาจาก Jscript ซึ่งเป็น JavaScript ในเวอร์ชั่นของ Microsoft

2.2.2 Base Classes Library

Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อย ๆ ที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเครื่องอำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่าง ๆ ส่วนใหญ่จะอยู่ในรูปแบบไฟล์ include แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ component ต่าง ๆ นั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่เป็นตัวเดียวกันได้หมด

2.2.3 Common Language Runtime (CLR)

นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆกัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกันทั้งหมด ซึ่งเรียกภาษาที่ว่านี้ว่า Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสภาวะแวดล้อมการทำงานเช่นใด หลังจากนั้นก็จะคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรมต่าง ๆได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง [2]

2.3 ASP.NET

เอเอสพีดอตเน็ต (อังกฤษ: ASP.NET) คือเทคโนโลยีสำหรับพัฒนาเว็บไซต์ เว็บแอปพลิเคชัน และเว็บเซอร์วิส ซึ่งเป็นส่วนหนึ่งของดอตเน็ตเฟรมเวิร์ก พัฒนาโดยไมโครซอฟท์

ASP.NET เป็นรุ่นถัดจาก Active Server Pages (ASP) แม้ว่า ASP.NET นั้นจะใช้ชื่อเดิมจาก ASP แต่ทั้งสองเทคโนโลยีนั้นแตกต่างกันอย่างสิ้นเชิง โดยไมโครซอฟท์นั้นได้สร้าง ASP.NET ขึ้นมาใหม่หมดบนฐานจาก Common Language Runtime (CLR) ซึ่งทำให้ผู้พัฒนาสามารถเลือกใช้ภาษาใดก็ได้ที่รองรับโดยดอตเน็ตเฟรมเวิร์กเช่น C# และ VB.NET เป็นต้น ปัจจุบันรุ่นล่าสุดคือ ASP.NET 2.0 ซึ่งรวมอยู่ใน .NET Framework 2.0. และ .NET Framework 3.0.

ASP.NET 1.0 ได้ออกมาในเดือนกุมภาพันธ์ ปีพ.ศ. 2545 (ค.ศ. 2002) พร้อมกับ Visual Studio .NET 2002 ในเดือนเมษายน พ.ศ. 2546 ASP.NET 1.1 นั้นได้ออกมาพร้อมกับ Visual Studio .NET 2003 และในวันที่ 7 พฤศจิกายน พ.ศ. 2548 ASP.NET 2.0 ได้ออกมาพร้อมกับ Visual Studio 2005 และ SQL Server 2005. [3]

2.3.1 ภาษา C#

ภาษาซีชาร์ป (C# Programming Language) เป็นภาษาโปรแกรมเชิงวัตถุทำงานบนดอตเน็ตเฟรมเวิร์ก พัฒนาโดยบริษัทไมโครซอฟท์และมี Anders Hejlsberg เป็นหัวหน้าโครงการ โดยมีรากฐานมาจากภาษาซีพลัสพลัสและภาษาอื่น ๆ (โดยเฉพาะภาษาเดลไฟและจาวา) โดยปัจจุบันภาษาซีชาร์ปเป็นภาษามาตรฐานรองรับโดย ECMA และ ISO

2.3.2 รูปแบบไฟล์ ASPX

ASPX เป็นชื่อรูปแบบไฟล์ของหน้าแบบฟอร์ม ASP.NET โดยทั่วไปแล้วในไฟล์จะมีรหัสแบบ HTML หรือ XHTML ซึ่งใช้กำกับรูปแบบฟอร์ม หรือเนื้อหาในหน้าเว็บ และในส่วนของโค้ดนั้นอาจจะอยู่ในหน้าเดียวกันในแท็คหรือบล็อก <% -- รหัสที่ใช้ -- %> (โดยในกรณีนี้จะคล้ายกับเทคโนโลยีที่ใช้พัฒนาเว็บ อย่าง PHP และ JSP) หรือแยกอยู่ในหน้าโค้ดออกมาต่างหาก (Code behind) ASP.NET รองรับการเขียนโค้ดในบรรทัดเดียวกันทั้งหมดในไฟล์ ASPX แต่วิธีนี้นั้นเป็นวิธีที่ไม่แนะนำ [3]

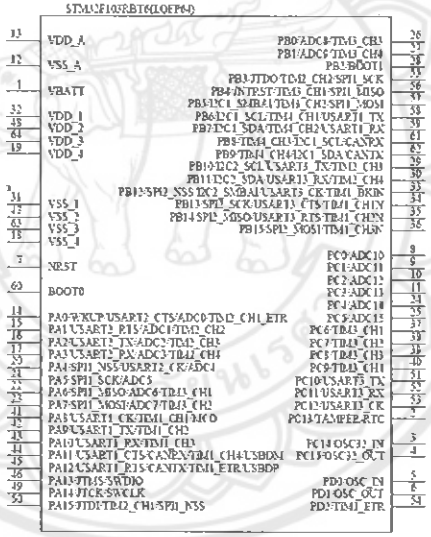
2.4 การทำงานของอุปกรณ์ต่างๆ

2.4.1 บอร์ดไมโครคอนโทรเลอร์ STM23F103/128

ไมโครคอนโทรลเลอร์ชนิดนี้เป็นไมโครคอนโทรลเลอร์ตระกูล ARM โดยใช้ 1 สัญญาณนาฬิกาต่อ 1.25 คำสั่งในการทำงาน (1DMIPS/MHz) ซึ่งเป็นไมโครคอนโทรลเลอร์ที่มีความสามารถและประสิทธิภาพสูงซึ่งในตระกูลเดียวกันจะมีหลาย ๆ รุ่น และแต่ละรุ่นก็จะแบ่งเป็นรุ่นย่อย ๆ อีกหลายตัว สำหรับการทำงานของโครงงานนี้จะเลือกใช้ไมโครคอนโทรลเลอร์รุ่น STAMP STM32F103

คุณสมบัติที่สำคัญของ STAMP STM32F103

- ใช้พลังงานต่ำ ประสิทธิภาพสูง และเป็นไมโครคอนโทรลเลอร์ 32 Bit
- มีความเร็วในการทำงาน 90 MIPS ต่อสัญญาณนาฬิกา 72 MHz
- มี SRAM ขนาด 20 Kbytes
- มีหน่วยความจำแบบ Flash ขนาด 128 Kbytes
- มีโมดูลแปลงสัญญาณอนาล็อกเป็นดิจิตอล (ADC) ขนาด 12 บิต จำนวน 16 ช่อง
- มีวงจร RTC(Real Time Clock) พร้อม XTAL ค่า 32.768KHz และ Battery Backup
- มี 48 Bit GPIO พร้อม 5V-Tolerant สำหรับประยุกต์ต่าง ๆ โดยอิสระ
- การสื่อสารข้อมูลอนุกรมมีทั้งแบบ UART (Universal Asynchronous Receiver Transmitters) หรือ RS232, CAN และแบบ I^2C เป็นต้น [4]



รูปที่ 2.5 แสดงหน้าที่ของพอร์ทต่าง ๆ

ที่มา: http://mcu.cz/news.php?extend.1529.3

หลักการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์

การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์จะต่างจากการเขียนโปรแกรมใช้งานกับคอมพิวเตอร์ เพราะว่าการเขียนโปรแกรมบนไมโครคอนโทรลเลอร์นั้นจะต้องคำนึงถึงส่วนต่าง ๆ ของตัวไมโครคอนโทรลเลอร์เป็นหลัก เนื่องจากคุณสมบัติของไมโครคอนโทรลเลอร์นั้นจะด้อยกว่าคอมพิวเตอร์ทั้งในด้านความเร็ว หน่วยความจำ และแม้กระทั่งจำนวนพอร์ตต่าง ๆ ที่มีให้ใช้อย่างจำกัด ดังนั้นหลักการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์จึงมีหลักดังต่อไปนี้

1. ทำความเข้าใจเกี่ยวกับส่วนที่ต้องการใช้งานของไมโครคอนโทรลเลอร์ (โมดูล) เช่น ข้อจำกัดของพอร์ตต่าง ๆ หน้าที่ของพอร์ตต่าง ๆ ที่สามารถทำได้ เป็นต้น

2. ศึกษาและทำความเข้าใจเกี่ยวกับการตั้งค่าของแต่ละพอร์ตที่ต้องการเรียกใช้งาน เนื่องจากเราจะต้องเขียนโปรแกรมตั้งค่าพอร์ตและเรียกใช้พอร์ต ซึ่งก่อนที่จะเรียกใช้ได้เราต้องตั้งค่าให้ถูกต้องและเหมาะสมเสียก่อน เพื่อให้ไมโครคอนโทรลเลอร์ทำงานได้ตามที่ต้องการได้

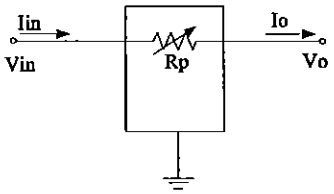
3. พอร์ตหลาย ๆ พอร์ตจะมีการสร้างไว้เฉพาะเจาะจงไว้ด้วย ดังนั้นเราจึงต้องรู้ว่าพอร์ตใดสามารถทำอะไรได้หรือไม่ได้

4. เมื่อได้ศึกษาและทำความเข้าใจเกี่ยวกับพอร์ตต่าง ๆ แล้ว จากนั้นก็จะเป็นการเขียนโปรแกรมเพื่อควบคุมการทำงานของไมโครคอนโทรลเลอร์ ซึ่งจะขึ้นอยู่กับพื้นฐานและความสามารถของแต่ละบุคคล รวมถึงพื้นฐานทางด้านอิเล็กทรอนิกส์ ซึ่งพื้นฐานทางอิเล็กทรอนิกส์นี้จะช่วยให้ชิ้นงานนั้นไม่มีข้อผิดพลาด ซึ่งจะทำให้การเขียนโปรแกรมมีความแม่นยำมากขึ้นเช่นกัน และทำให้งานที่ได้มานั้นมีความถูกต้องและตรงตามความต้องการได้มากที่สุด [4]

2.5 หลักการทำงานของวงจร

2.5.1 เร็กกูเลเตอร์แบบอนุกรม (Series Regulator)

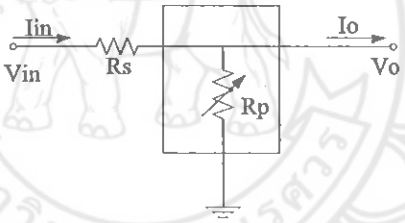
เร็กกูเลเตอร์แบบอนุกรมมีหลักการทำงานคือ มีการจ่ายแรงดันเข้าที่ฝั่งอินพุตแล้วภายในจะมีตัวต้านทานที่สามารถปรับค่าได้เองโดยอัตโนมัติ จึงทำให้เกิดแรงดันตกคร่อมที่ตัวต้านทานอยู่ค่าหนึ่ง ซึ่งจะทำแรงดันตกคร่อมที่เหลือออกไปทางฝั่งเอาต์พุตซึ่งเป็นแรงดันที่ต้องการจะใช้กับวงจรอื่นๆ ต่อไป ดังรูปที่ 2.6



รูปที่ 2.6 แสดงลักษณะของเร็กกูเลเตอร์แบบอนุกรม

2.5.2 เร็กกูเลเตอร์แบบขนาน (Shunt Regulator)

เร็กกูเลเตอร์แบบขนานจะใช้หลักการเดียวกับวงจรแบ่งแรงดันคือ มีตัวต้านทานอยู่ 2 ตัวมาต่อกันแบบอนุกรมซึ่งเมื่อมีการเชื่อมต่ออินพุตแล้วจะทำให้แรงดันตกคร่อมระหว่างตัวต้านทาน 2 ตัว มีค่าไม่เท่ากัน (กรณีที่ค่าของตัวต้านทานทั้ง 2 ตัวนั้นไม่เท่ากัน) ซึ่งจะมีตัวที่อยู่ใกล้ฝั่งเอาต์พุตเป็นตัวต้านทานที่สามารถปรับค่าได้อัตโนมัติ จึงทำให้เราได้ค่าความต่างศักย์ที่ต้องการ แต่เร็กกูเลเตอร์แบบนี้จะดีไม่เท่ากับแบบอนุกรม เนื่องจากในวงจรไฟฟ้าจะมองว่าตัวต้านทานที่สองนั้นต่อขนานกับวงจรจริงๆ ที่เราใช้ จึงทำให้มีกระแสที่น้อยลง ดังรูปที่ 2.7



รูปที่ 2.7 แสดงลักษณะของเร็กกูเลเตอร์แบบขนาน

2.5.3 ไอซีเร็กกูเลเตอร์สามขาชนิดจ่ายแรงดันคงที่

ไอซีเร็กกูเลเตอร์สามขาที่ได้ใช้ในโครงงานนี้เป็นไอซีชนิดอนุกรม โดยในโครงงานนี้จะใช้ไอซีเรกกูเลเตอร์ 2 แบบคือ 7805 และ LM1117 โดย 7805 มีขาอยู่ 3 ขา โดยขาแรกคือไฟอินพุต ขาที่สองเป็นกราวด์ และขาที่สามเป็นเอาต์พุต ส่วน LM1117 จะใช้ขาแรกเป็นกราวด์ ขาที่สองเป็นเอาต์พุต ขาที่สามเป็นอินพุต ซึ่งไอซีทั้งสองชนิดจะจ่ายแรงดันที่ค่อนข้างคงที่

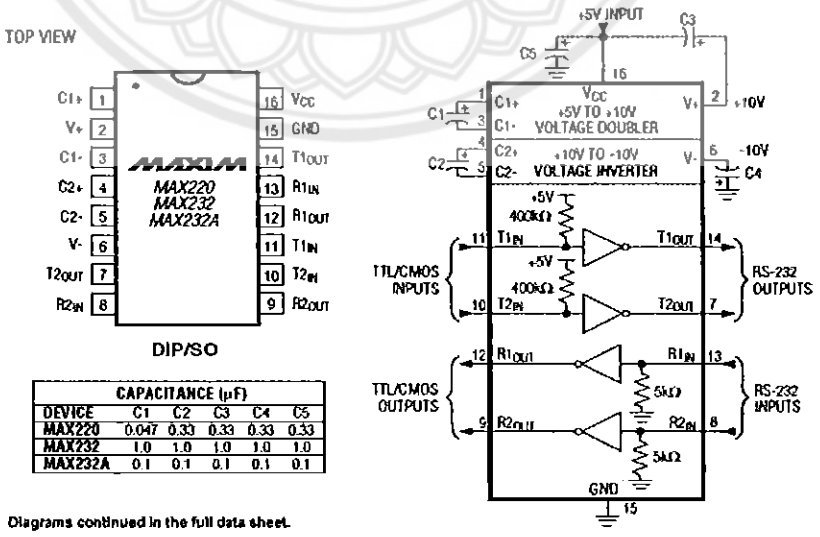
และไอซีเรกูเลเตอร์แบบสามขานี้ใช้งานง่ายไม่ต้องต่ออุปกรณ์ภายนอกเพิ่มเติมมาก และในการต่อวงจรที่ดีเราควรต่อตัวเก็บประจุอิเล็กทรอไลต์ขนาด 10 ไมโครฟารัด 1 ตัวเพื่อป้องกันการกระเพื่อมของกระแสที่มีความถี่สูง ซึ่งเป็นสาเหตุที่ทำให้วงจรขาดเสถียรภาพ และที่เอาต์พุตก็มีกระแสที่ค่อนข้างเรียบอยู่พอสมควร แต่เราก็ควรต่อตัวเก็บประจุที่มีค่าประมาณ 100 ไมโครฟารัดขึ้นไปเพื่อช่วยให้กระแสมีความเรียบมากยิ่งขึ้น และป้องกันการกระชากของไฟได้ด้วย ดังรูปที่ 2.8

รูปที่ 2.8 ภาพของไอซีเร็กกูเลเตอร์แบบสามขา

2.5.4 ไอซีปรับระดับแรงดัน MAX232

ไอซีปรับระดับแรงดัน MAX232 เป็นไอซีที่มีขา 16 ขาที่ใช้ทำหน้าที่ในการรับส่งข้อมูลแบบอนุกรม (serial) ระหว่างอุปกรณ์ที่มีสัญญาณแบบ TTL/CMOS กับพอร์ท RS232 โดยไอซีปรับระดับแรงดัน MAX232 มีช่องทางในการรับข้อมูลแบบอนุกรมอยู่ 2 ช่องทาง และมีช่องทางในการส่งข้อมูลแบบอนุกรมอยู่ 2 ช่องทาง ซึ่งภายในไอซีจะมีวงจรทวีแรงดัน (voltage double) และวงจรอินเวิร์ทแรงดัน (voltage inverter) โดยที่วงจรทวีแรงดันจะทำหน้าที่ในการเพิ่มความต่างศักย์จาก 5 โวลท์เป็น 10 โวลท์ และวงจรอินเวิร์ทแรงดันก็จะทำหน้าที่ในการกลับสัญญาณที่มีความต่างศักย์จาก 0 โวลท์เป็น -10 โวลท์ ซึ่งสัญญาณ TTL/CMOS นี้จะใช้ความต่างศักย์ของสัญญาณที่ 0 กับ 5 โวลท์ ส่วนสัญญาณแบบ RS232 จะใช้ความต่างศักย์ของสัญญาณที่ -10 กับ 10 โวลท์

ดังนั้นไอซีปรับระดับแรงดัน MAX232 ก็คือตัวกลางในการเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์กับไมโครคอนโทรลเลอร์นั่นเอง ดังรูปที่ 2.9

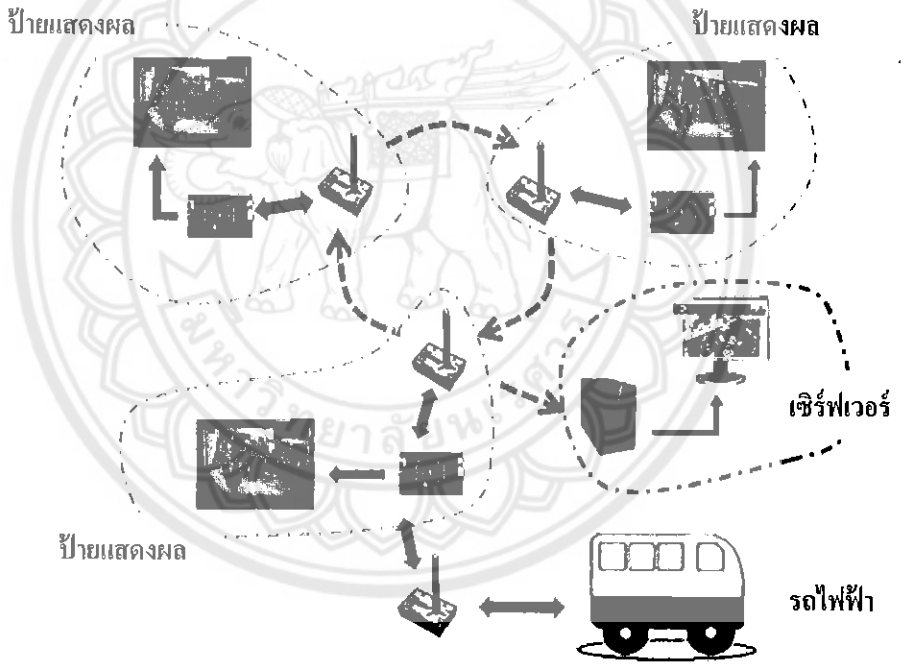


รูปที่ 2.9 ตำแหน่งขาสัญญาณและวงจรภายในของไอซีปรับแรงดัน MAX232

ที่มา: http://www.maxim-ic.com/images/qv/1798.gif

บทที่ 3
วิธีดำเนินโครงงาน

ในบทนี้จะเป็นการกล่าวถึงการออกแบบการทำงานของระบบการเชื่อมต่อเครือข่ายไร้สายของแต่ละสถานีรอรถไฟฟ้า รวมไปถึงการแสดงผลที่คอมพิวเตอร์ โดยรูปแบบของเครือข่ายจะมีการรับสัญญาณจากสถานีก่อนหน้า และส่งต่อกันไปเรื่อย ๆ จนถึงเครื่องแม่ข่าย แล้วเครื่องแม่ข่ายก็จะสามารถแสดงผลผ่านทางหน้าจอของคอมพิวเตอร์และสามารถเข้าดูผ่านทางหน้าเว็บแอปพลิเคชันซึ่งข้อมูลที่ใช้ในการรับและส่งต่อจะได้จากระบบการติดตามรถไฟฟ้าโดยระบบบอกตำแหน่งพิกัดโลกและสื่อสารแบบไร้สายซึ่งเป็นโครงงานร่วม ดังรูปที่ 3.1

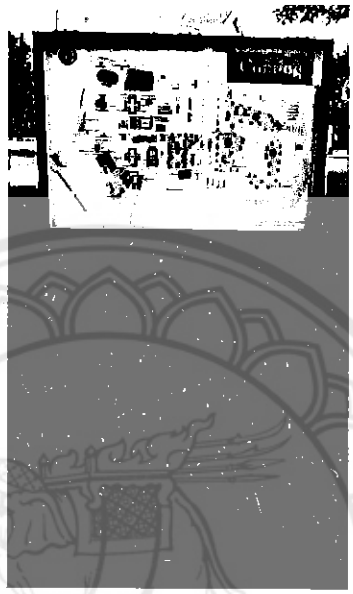


รูปที่ 3.1 แสดงระบบโดยรวมของระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้า

3.1 การออกแบบรูปแบบการของเชื่อมต่อของระบบติดตามและแสดงผลของรถไฟฟ้า

ในการออกแบบระบบและการเชื่อมต่อของเครือข่ายสำหรับติดตามและแสดงผลของรถไฟฟ้านี้เป็นระบบเครือข่ายแบบวงแหวนที่ประกอบไปด้วยสามส่วนหลัก ๆ คือ สถานีรอรถไฟฟ้า โรงจอดรถไฟฟ้า และเครื่องคอมพิวเตอร์แม่ข่าย ดังรูปที่ 3.1

สถานีรอรถไฟฟ้า หมายถึงจุดรอรถไฟฟ้าแต่ละจุดที่ตั้งอยู่ภายในมหาวิทยาลัยนเรศวรซึ่งแต่ละจุดรอรถไฟฟ้าจะมีป้ายแสดงผล เพื่อรายงานผลตำแหน่งของรถไฟฟ้าว่ารถไฟฟ้าแต่ละคันอยู่จุดใดบ้างภายในมหาวิทยาลัย



รูปที่ 3.2 ป้ายแสดงผล

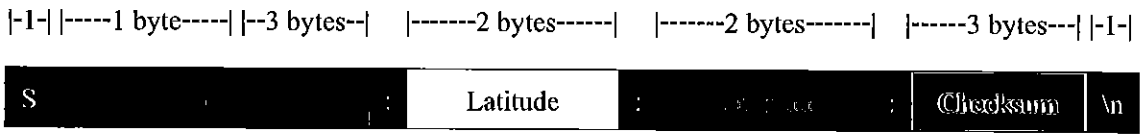
โรงจอดรถไฟฟ้า คือจุดพักรถไฟฟ้าที่อยู่บริเวณหอในของมหาวิทยาลัยนเรศวร และจุดนี้จะทำหน้าที่คล้าย ๆ กับสถานีรอรถไฟฟ้ารวมทั้งจะทำหน้าที่ส่งข้อมูลให้กับคอมพิวเตอร์ผ่านพอร์ตอนุกรม

เครื่องคอมพิวเตอร์แม่ข่าย คือคอมพิวเตอร์ที่ทำหน้าที่ติดต่อกับสถานีจอดรถไฟฟ้าผ่านทางโปรแกรมที่ใช้ในการรับค่า รหัสประจำรถไฟฟ้า ละติจูด และค่าลองจิจูด เพื่อใช้รายงานผลบนหน้าเว็บและเก็บข้อมูลเข้าสู่ฐานข้อมูล

สำหรับการออกแบบระบบนี้ สิ่งที่จำเป็นอีกอย่างคือความทนทานต่อสภาพแวดล้อม ซึ่งความทนทานต่อสภาพแวดล้อมนี้จะขึ้นอยู่กับการห่อหุ้มตัวชิ้นงาน เช่น ต้องกันน้ำได้ เป็นต้น และต้องสามารถบำรุงรักษาได้ เช่น เปลี่ยนหลอดไฟได้ง่ายเมื่อหลอดไฟขาด เป็นต้น

3.2 การออกแบบรูปแบบของข้อมูล (Data package)

ข้อมูลที่ใช้เป็นผลสัมฤทธิ์จากโครงงานการติดตามรถไฟฟ้าโดยระบบบอกตำแหน่งพิกัดโลกและสื่อสารแบบไร้สาย โดยที่ข้อมูลจะมีขนาด 16 ตัวอักษร โดยแบ่งเป็นดังนี้



รูปที่ 3.3 รูปแบบข้อมูลที่ใช้ในการสื่อสารในเครือข่าย

โดยขนาดของข้อมูลจะแบ่งกันดังนี้

S มีขนาด 1 byte ใช้เป็นส่วนเริ่มต้นของข้อมูลนั้น ๆ

Station_Label มีขนาด 1 byte ใช้ในการแสดงผลที่ป้ายแสดงผล

Car_ID มีขนาด 3 bytes ใช้ในการระบุว่าเป็นรถคันใด

Latitude มีขนาด 2 bytes ใช้ในการเก็บค่าละติจูด

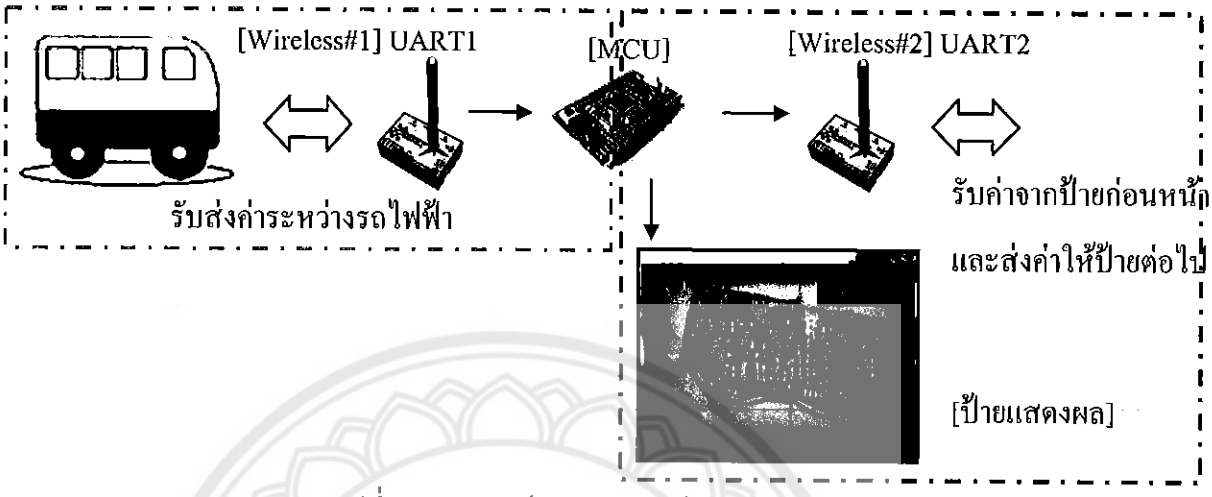
Longitude มีขนาด 2 bytes ใช้ในการเก็บค่าลองจิจูด

Checksum มีขนาด 3 bytes ใช้ในการตรวจสอบความถูกต้องของข้อมูล

: มีขนาด 1 byte ใช้ในการคั่นระหว่างข้อมูล

\n มีขนาด 1 byte ใช้เป็นส่วนสุดท้ายของข้อมูลนั้น ๆ

3.3 ส่วนประกอบทางฝั่งป้ายแสดงผล



รูปที่ 3.4 โครงสร้างของระบบป้ายแสดงผล

อุปกรณ์ภายในกรอบสีแดงคือโครงงานระบบเครือข่ายและติดตามสำหรับการแสดงผลติดตามรถไฟฟ้า และกรอบสีเขียวคือโครงงานการติดตามรถไฟฟ้าโดยระบบบอกตำแหน่งพิกัดโลกและสื่อสารแบบไร้สายซึ่งเป็นโครงงานร่วมและมีการใช้ข้อมูลร่วมกันบนไมโครคอนโทรลเลอร์ในตัวเดียวกัน

อุปกรณ์ฝั่งป้ายแสดงผล ซึ่งจะมีส่วนประกอบดัง รูปที่ 3.4 ในกรอบสีแดง

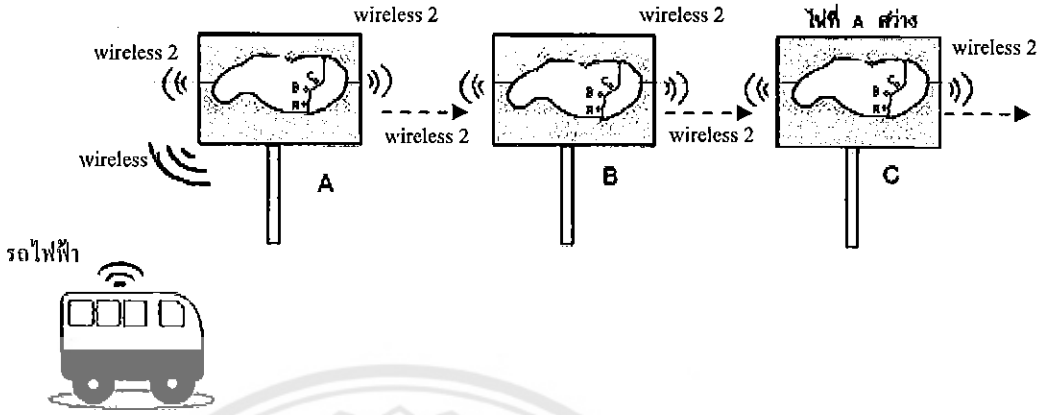
ไมโครคอนโทรลเลอร์ ทำหน้าที่ประมวลผลข้อมูลที่ได้รับจากโครงงานร่วม และนำข้อมูลนั้นไปแสดงผลบนป้ายแสดงผล

อุปกรณ์สื่อสารแบบไร้สาย โดยป้ายแสดงผลจะมีอยู่ 2 ตัว โดยหนึ่งตัวจะทำหน้าที่รับสัญญาณจากรถไฟฟ้าซึ่งเป็นส่วนงานของอีกโครงงานที่ทำร่วมกัน และอีกหนึ่งตัวจะทำหน้าที่รับสัญญาณจากจุดรอรถไฟฟ้าก่อนหน้าและส่งข้อมูลต่อให้จุดรอรถไฟฟ้าถัดไป

ป้ายแสดงผล ทำหน้าที่แสดงตำแหน่งของรถไฟฟ้าว่าอยู่บริเวณจุดรอรถไฟฟ้าใดภายในมหาวิทยาลัยประกอบด้วย 3 ส่วน

1. ไฟแสดงตำแหน่งใช้หลอดไฟแทนตำแหน่งของจุดรอรถไฟฟ้าสายสีแดงทุกจุดภายในมหาวิทยาลัย
2. ไฟแสดงสถานะคำว่า “Coming” ซึ่งหมายถึงว่ารถไฟฟ้ากำลังจะมาถึงป้ายที่ผู้ใช้รออยู่
3. แผนที่ภายในมหาวิทยาลัยนเรศวรใช้แสดงเส้นทางการเดินรถไฟฟ้าสายสีแดง

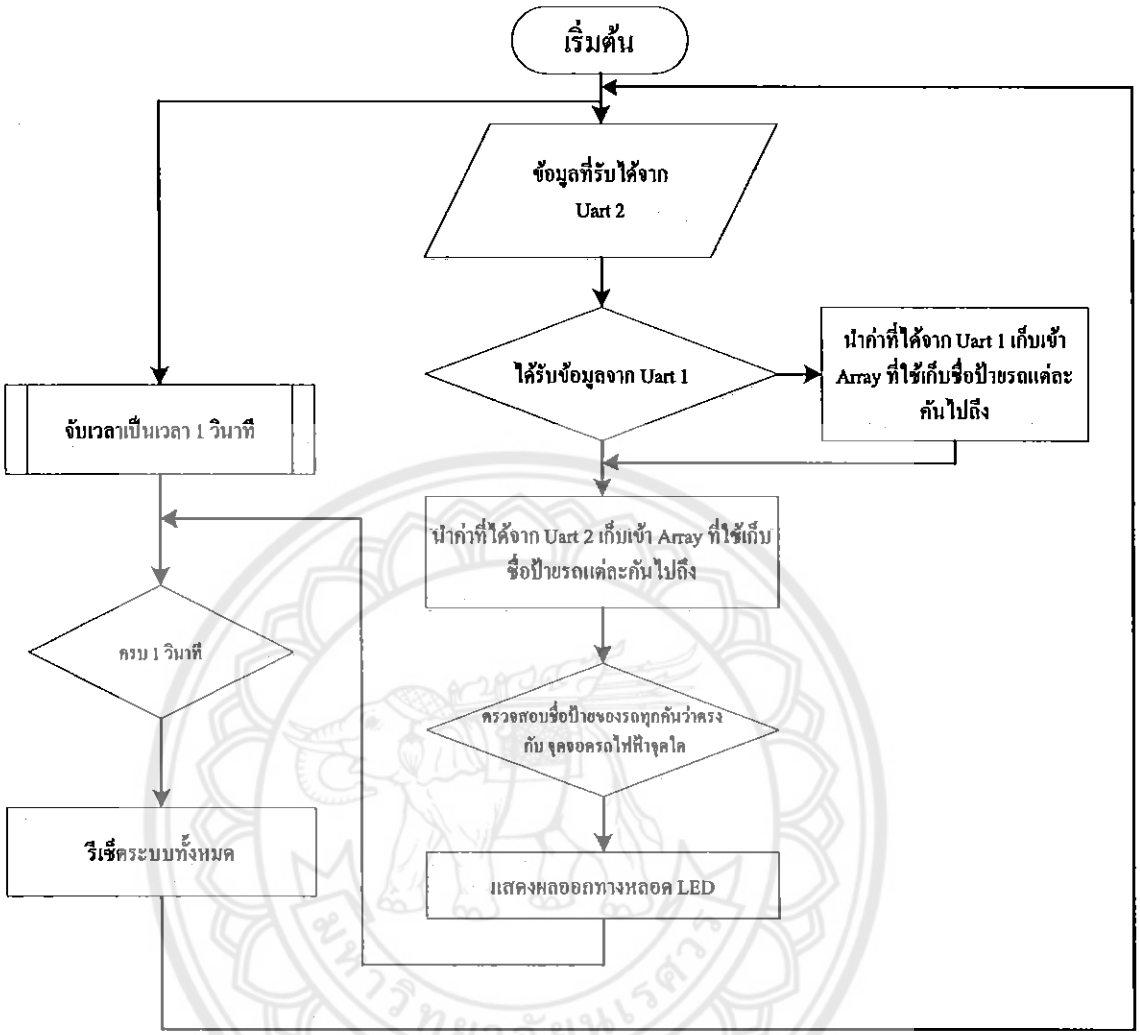
หลักการทำงานของระบบโดยรวมเป็นดังต่อไปนี้



รูปที่ 3.5 แสดงระบบการทำงานของป้ายแสดงผล

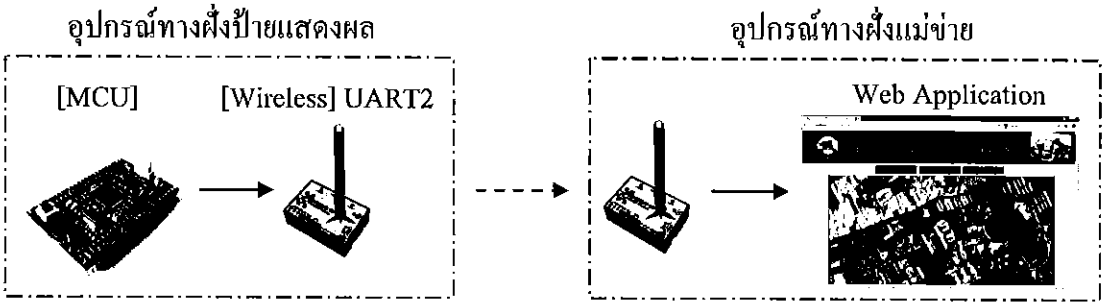
เมื่อไมโครคอนโทรลเลอร์ที่จุดรอรถไฟฟ้า ได้รับข้อมูลมาครบแล้วไมโครคอนโทรลเลอร์จะนำข้อมูลที่ได้รับมาประมวลผลและแสดงผลด้วยหลอดไฟที่อยู่บนป้ายแสดงผลนั้นกระพริบ อย่างเช่น เมื่อรถไฟฟ้าอยู่ใกล้กับจุด A แล้วจะทำให้ป้ายแสดงผลที่จุดรอรถ A, B และ C แสดงผลด้วยการให้ไฟสว่างที่จุด A เหมือนกันทั้งหมดและเมื่อรถไฟฟ้าเคลื่อนไปถึงจุด B แล้วป้ายแสดงผลที่จุดรอรถ A, B และ C ก็จะมีหลอดไฟที่แทนจุดรอรถของจุด B สว่างขึ้นมาเช่นกัน โดยที่ป้ายแสดงผลตามจุดรอรถ A, B และ C มีรูปแบบการเชื่อมต่อกันแบบวงแหวนเส้นทางเดียว แต่การสื่อสารระหว่างป้ายแสดงผลกับรถไฟฟ้าจะเป็นงานของอีกโครงงานที่ทำร่วมกัน

การแสดงผลของป้ายบอกตำแหน่งจะใช้การตรวจสอบค่าจากข้อมูลในตำแหน่งที่ 2 ของข้อมูลในรูปที่ 3.3 ซึ่งข้อมูลในตำแหน่งที่ 2 นี้จะใช้สัญลักษณ์แทนป้ายชื่อของแต่ละจุดรอรถซึ่งแต่ละจุดรอรถจะมีสัญลักษณ์ที่ไม่ซ้ำกัน ถ้าข้อมูลในตำแหน่งนี้เป็นสัญลักษณ์ที่ตรงกับหลอดไฟหลอดใด หลอดไฟนั้นก็จะสว่างขึ้นมาเพื่อบอกตำแหน่งของรถไฟฟ้าว่าอยู่ใกล้กับจุดรอรถไฟฟ้าใด โดยเป็นไปตามผังงาน (Flowchart) ดังนี้



รูปที่ 3.6 ผังงานการควบคุมการติดดับหลอดไฟของระบบแสดงผล

3.4 ส่วนประกอบทางฝั่งแม่ข่าย



รูปที่ 3.7 โครงสร้างการทำงานทางฝั่งแม่ข่าย

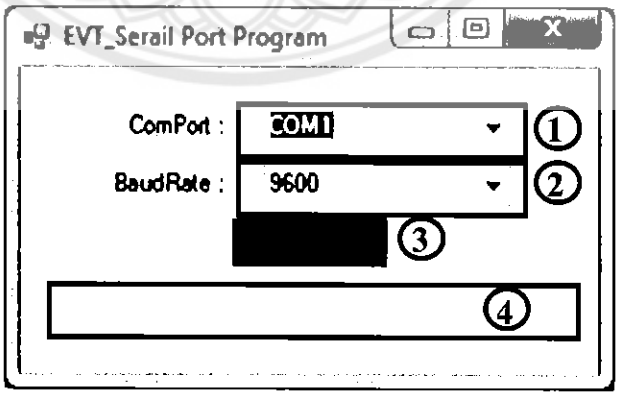
หลักการทำงานของฝั่งแม่ข่าย ซึ่งจะมีส่วนประกอบดัง รูปที่ 3.7

ไมโครคอนโทรลเลอร์ ทำหน้าที่ประมวลผล ข้อมูลที่ได้รับจากสัญญาณไร้สาย และนำค่าที่ได้รับส่งไปยังฝั่งแม่ข่าย

อุปกรณ์รับสัญญาณแบบไร้สาย ทำหน้าที่รับสัญญาณจากไมโครคอนโทรลเลอร์เข้าสู่เครื่องแม่ข่าย

การแสดงผลของฝั่งแม่ข่าย แบ่งการทำงานเป็น 2 ส่วน โดยส่วนที่ 1 คือส่วนรับข้อมูลจากไมโครคอนโทรลเลอร์เขียนลงไฟล์ ส่วนที่ 2 คือส่วนอ่านข้อมูลจากไฟล์แล้วนำค่ามาทำการคำนวณแล้วแสดงค่าในเว็บแอปพลิเคชัน

3.4.1 โปรแกรมรับข้อมูลจากไมโครคอนโทรลเลอร์



รูปที่ 3.8 โครงสร้างหลักของโปรแกรมรับข้อมูลจากไมโครคอนโทรลเลอร์

ส่วนประกอบของโปรแกรม รูปที่ 3.8

ส่วนที่ 1 ช่อง ComboBox สำหรับเลือก Com Port

ส่วนที่ 2 ช่อง ComboBox สำหรับเลือก BaudRate

ส่วนที่ 3 ปุ่ม Connect สำหรับกดเพื่อเริ่มต้นให้โปรแกรมทำงาน

ส่วนที่ 4 Label สำหรับแสดงค่าที่รับมาจากไมโครคอนโทลเลอร์

หลักการทำงานโปรแกรม รูปที่ 3.7

การทำงานของโปรแกรมคือการรับข้อมูลจากไมโครคอนโทรลเลอร์โปรแกรมจะทำการเก็บข้อมูลไว้ก่อนเพื่อรอให้เว็บแอปพลิเคชัน นำค่าไปวาดและลบไฟล์ก่อนจึงจะเขียนค่าลงไฟล์โดยจะใช้ FileSystemWatcher ตรวจสอบว่าไฟล์ถูกลบไปแล้วหรือยัง ถ้าไฟล์ถูกลบแล้วคือเว็บแอปพลิเคชันได้นำค่าไปวาดขึ้นหน้าเว็บแล้วอีกโปรแกรมจึงจะเขียนข้อมูลลงในไฟล์ ซึ่งการใช้ FileSystemWatcher เป็นการใช้เพื่อป้องกันการผิดพลาดที่เกิดจากการเข้าใช้ไฟล์พร้อมกันของ 2 โปรแกรม

3.4.2 ออกแบบระบบแสดงผลทาง Web Application

สามารถแสดงตำแหน่งรถไฟฟ้าทางเว็บแอปพลิเคชันแบบเวลาจริง (Real Time) โดยจำลองให้แสดงได้มากสุด 2 คัน ซึ่งจะรับข้อมูลมาอัพเดทการแสดงผลได้เพียงทีละหนึ่งคัน ส่วนคันที่ไม่มีการส่งข้อมูลมาอัพเดทก็จะแสดงตำแหน่งเดิม และจะรอรับข้อมูลใหม่เรื่อย ๆ ทุกรอบ โดยแบ่งการแสดงผลออกเป็น 3 ส่วน คือ ส่วนแรก View Map เป็นการแสดงผลโดยวาดพิกัดรถไฟฟ้าลงบนแผนที่ ส่วนที่สอง Text Log เป็นการแสดงผลพิกัดรถไฟฟ้าเป็นตัวเลข ส่วนที่สาม เป็นการแสดงผลโดยนำค่าจากฐานข้อมูลมาวาดพิกัดรถไฟฟ้า

3.4.2.1 ส่วน View Map

ในหน้านี้จะเป็นส่วนของการแสดงตำแหน่งของรถไฟฟ้าบนแผนที่ของมหาวิทยาลัยนเรศวรซึ่งจะเห็นความเคลื่อนไหวของรถไฟฟ้าแบบเวลาจริง ดังในรูปที่ 3.9

รูปที่ 3.9 โครงสร้างของหน้าเว็บแสดงตำแหน่งรถไฟฟ้าแบบ View Map

ส่วนประกอบของหน้าเว็บ View Map รูปที่ 3.9

ส่วนที่ 1 ส่วนของ Button สำหรับเชื่อมต่อไปหน้า Text Log และ Summary

ส่วนที่ 2 ส่วนของ Image สำหรับแสดงแผนที่มหาวิทยาลัยนเรศวร

ส่วนที่ 3 ส่วนแสดงตำแหน่งรถไฟฟ้า

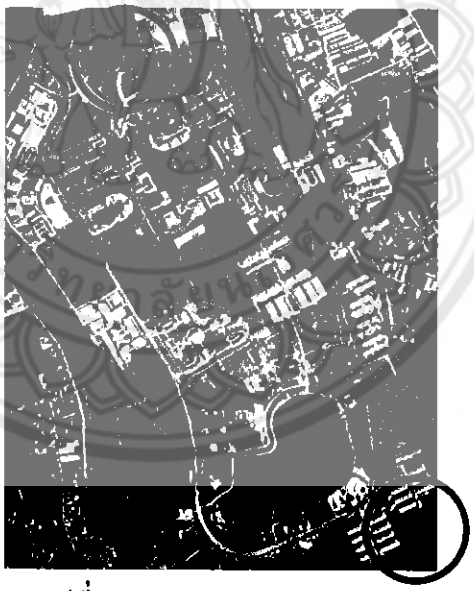
หลักการทำงานหน้าเว็บ View Map รูปที่ 3.9

ใช้ JavaScript ตั้งเวลาให้ฟังก์ชันทำงานโดยจะทำงานวนซ้ำไปเรื่อย ๆ ทุก 1 วินาที เพื่อแสดงพิกัดของรถไฟฟ้าทางหน้าเว็บได้แบบเวลาจริง เมื่อฟังก์ชันทำงานจะอ่านข้อมูลรหัสรถไฟฟ้า ละติจูดและลองจิจูดจากไฟล์นำค่ามาคำนวณให้ตรงกับตำแหน่งพิกเซลใน Bitmap โดยใช้สมการที่ 3.1 และ สมการที่ 3.2

$$m = \frac{\Delta y}{\Delta x} \qquad (3.1)$$

$$y = mx + c \qquad (3.2)$$

การคำนวณจากพิกัดโลกเป็นตำแหน่งพิกเซล คือนำค่าตำแหน่งพิกเซลกับค่าพิกัดโลกบนแผนที่ 2 จุด มาคำนวณเพื่อหาค่า mX, mY, cX, cY มาเป็นค่าในการแปลงพิกัดโลกมาเป็นพิกเซลเพื่อวาดลงบนแผนที่ในหน้าเว็บ ค่าพิกัดโลกที่นำมาคำนวณจะใช้เฉพาะหลักหน่วยและทศนิยมโดย ละติจูดเทียบกับค่า y – yMax เนื่องจากจุดเริ่มของพิกเซลอยู่ด้านบนแต่จุดเริ่มของละติจูดอยู่ด้านล่าง และ ลองจิจูดเทียบกับค่า x มีการหมุนแผนที่ไปทางซ้ายจากแนวตั้งเป็นแนวนอน เพื่อให้สามารถดูแผนที่ได้ง่ายขึ้น เมื่อทำการหมุนแผนที่แล้วจะต้องทำการสลับค่า x และ y และจุดเริ่มต้นของค่าพิกเซลที่คำนวณได้จะอยู่ที่มุมขวาล่างเพื่อให้แปรผันตรงกับพิกัดโลก แต่การวาดลงบนแผนที่จุดเริ่มต้นอยู่ที่มุมซ้ายบน จึงต้องเอาค่า xMax และ yMax มาลบค่าที่คำนวณได้ก่อนที่จะวาดลงบนหน้าเว็บ



รูปที่ 3.10 แผนจาก Google Map

ค่าพิกัดโลก จุดที่ 1 ลองจิจูด 0.9678 เทียบกับ x = 668 และ ละติจูด 0.2947 เทียบกับ y = 193

ค่าพิกัดโลก จุดที่ 2 ลองจิจูด 0.3939 เทียบกับ x = 66 และ ละติจูด 1.0414 เทียบกับ y = 1000

$$mX = \frac{668 - 66}{0.9678 - 0.3938} = 1048.96323 \quad (3.3)$$

$$mY = \frac{193 - 1000}{0.2947 - 1.0414} = 1080.75532 \quad (3.4)$$

$$cX = 668 - 0.9678 * 1048.96323 = -347.18662 \quad (3.5)$$

$$cY = 193 - 0.2947 * 1080.75532 = -125.49859 \quad (3.6)$$

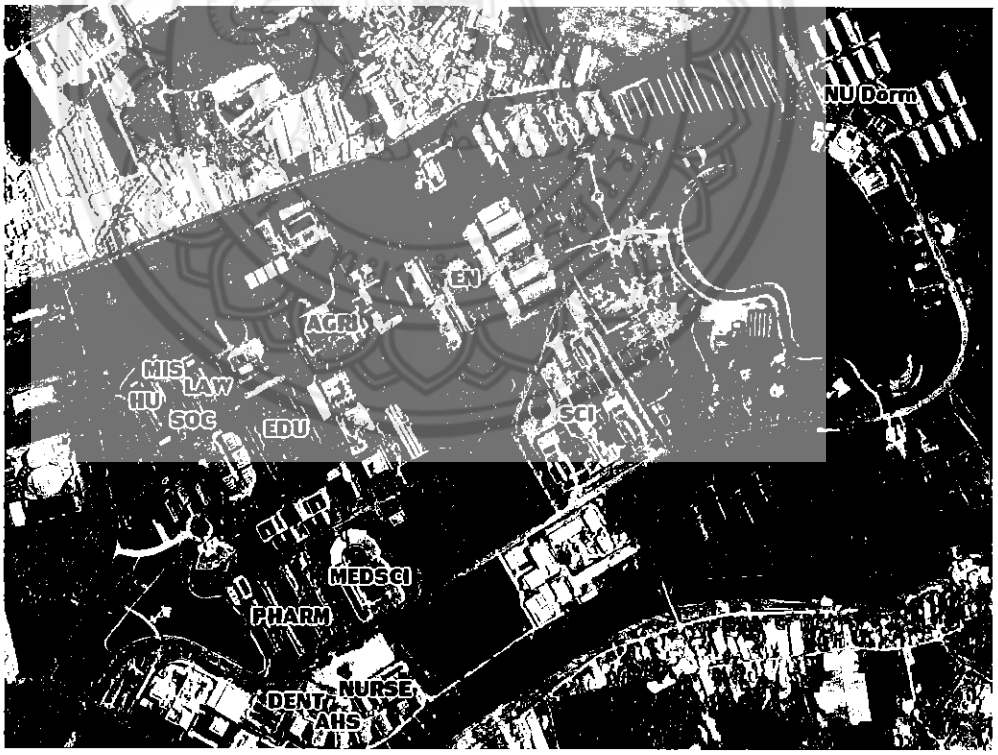
เมื่อได้ค่า mX, mY, cX, cY

$$x = yMax - (mY * Latitude + mY) \quad (3.7)$$

$$y = xMax - (mX * Longitude + mX) \quad (3.8)$$

ค่า x ที่นำไปวาดบนแผนที่หาจากสมการที่ 3.7

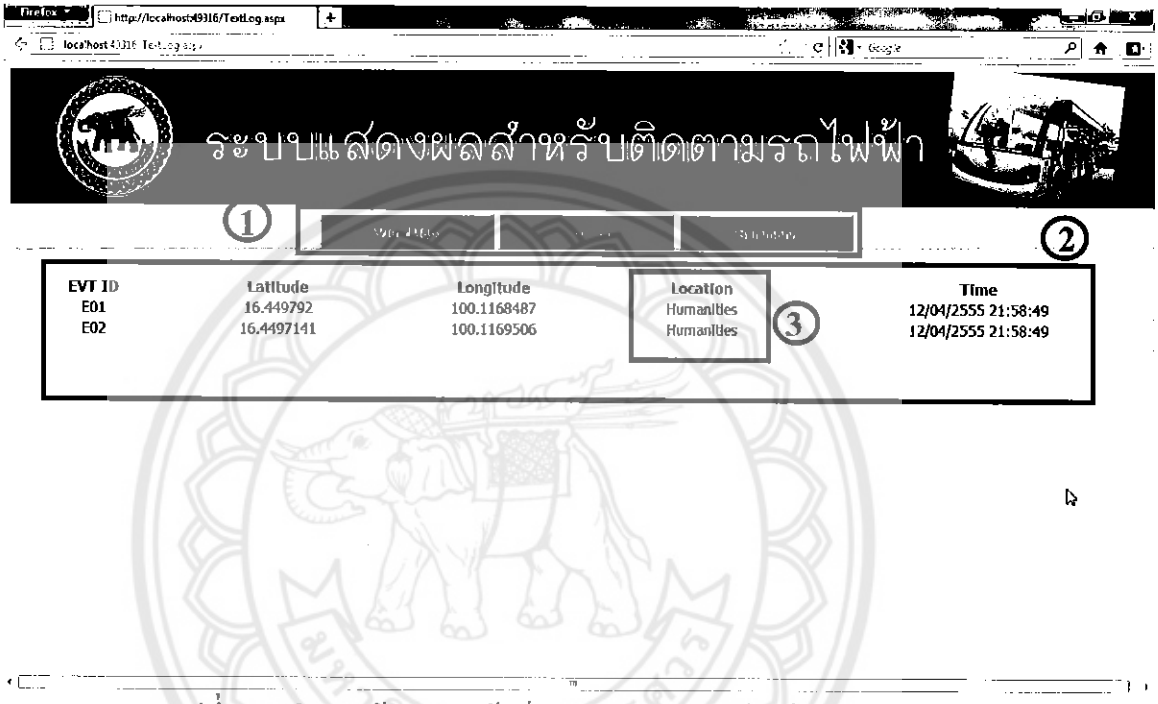
ค่า y ที่นำไปวาดบนแผนที่หาจากสมการที่ 3.8



รูปที่ 3.11 แผนที่สำหรับวาดตำแหน่งรถไฟฟ้า

3.4.2.2 ส่วน Text Log

ในหน้านี้จะเป็นส่วนของการแสดงผลพิกัดรถไฟฟ้าเป็นตัวเลข ซึ่งประกอบด้วยค่าละติจูด และค่าลองจิจูดที่รถไฟฟ้าอยู่ ณ ขณะนั้น นอกจากนั้นยังแสดงชื่อสถานที่ที่รถไฟฟ้าอยู่ใกล้และวันที่ เวลา ณ ปัจจุบันด้วย ดังรูปที่ 3.10



รูปที่ 3.12 โครงสร้างของหน้าเว็บแสดงตำแหน่งรถไฟฟ้าแบบ Text Log

ส่วนประกอบของหน้าเว็บ **Text Log** รูปที่ 3.12

ส่วนที่ 1 ส่วนของ Button สำหรับเชื่อมต่อไปหน้า View Map และ Summary

ส่วนที่ 2 ส่วนแสดงตำแหน่งรถไฟฟ้า

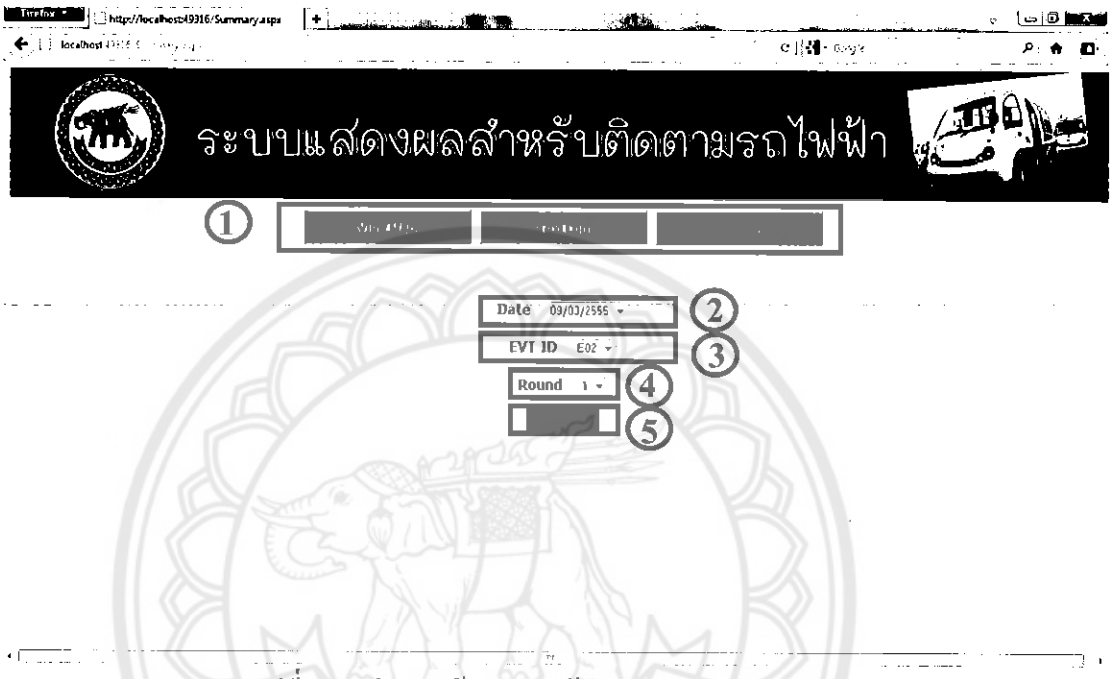
ส่วนที่ 3 ส่วนแสดงสถานที่ที่ใกล้รถไฟฟ้า

หลักการทำงานหน้าเว็บ View Map รูปที่ 3.12

ใช้ JavaScript ตั้งเวลาให้ฟังก์ชันทำงาน โดยจะทำงานวนซ้ำไปเรื่อย ๆ ในเวลาทุก ๆ 0.5 วินาที เพื่อแสดงพิกัดของรถไฟฟ้าทางหน้าเว็บได้แบบเวลาจริง เมื่อฟังก์ชันทำงานจะอ่านข้อมูลรหัสรถไฟฟ้าละติจูดและลองจิจูดจากไฟล์นำค่ามาหาสถานที่ โดยกำหนดขอบเขตละติจูดและลองจิจูดของแต่สถานที่ แล้วนำละติจูดและลองจิจูดที่อ่านจากไฟล์มาตรวจสอบว่าอยู่ในขอบเขตของสถานที่ไหน แล้วทำการแสดงขึ้นหน้าเว็บ

3.4.2.3 ส่วน Summary

ในหน้านี้เป็นส่วนของการแสดงผลข้อมูลรอบการเดินทางของรถไฟฟ้าแต่ละคัน ซึ่งจะสามารถเลือกรถมาแสดงได้ ดังรูปที่ 3.11



รูปที่ 3.13 โครงสร้างของหน้าหลักของส่วน Summary

ส่วนประกอบของหน้าหลักของส่วน Summary รูปที่ 3.11

ส่วนที่ 1 ส่วนของ Button สำหรับเชื่อมต่อไปหน้า View Map และ Text Log

ส่วนที่ 2 ส่วนของ DropDownList สำหรับเลือกวันที่

ส่วนที่ 3 ส่วนของ DropDownList สำหรับเลือกรถไฟฟ้า

ส่วนที่ 4 ส่วนของ DropDownList สำหรับเลือกรอบ

ส่วนที่ 5 ปุ่ม Run สำหรับเชื่อมต่อ (Link) ไปหน้าวาดพิกัดรถไฟฟ้า

หลักการทำงานหน้าหลักของส่วน Summary รูปที่ 3.13

เมื่อเลือกรถไฟฟ้า โปรแกรมจะดึงข้อมูลจากฐานข้อมูลมาตรวจสอบรอบและจำนวนข้อมูลในแต่ละรอบแล้ว นำค่ารอบการเดินทางของรถไฟฟ้าคันที่เลือกไปใส่ใน DropDownList สำหรับเลือกรอบ เมื่อเริ่มต้นโปรแกรมจะเชื่อมพร้อมส่งค่า รหัสรถ รอบที่ และจำนวนข้อมูลในรอบนั้นไปที่หน้าวาดแผนที่ ดังรูปที่ 3.14

รูปที่ 3.14 โครงสร้างของหน้าแสดงพิกัดของส่วน Summary

ส่วนประกอบของหน้าแสดงพิกัดส่วน Summary รูปที่ 3.14

ส่วนที่ 1 ส่วนของ Button สำหรับเชื่อมต่อกับไปหน้าหลักส่วน Summary

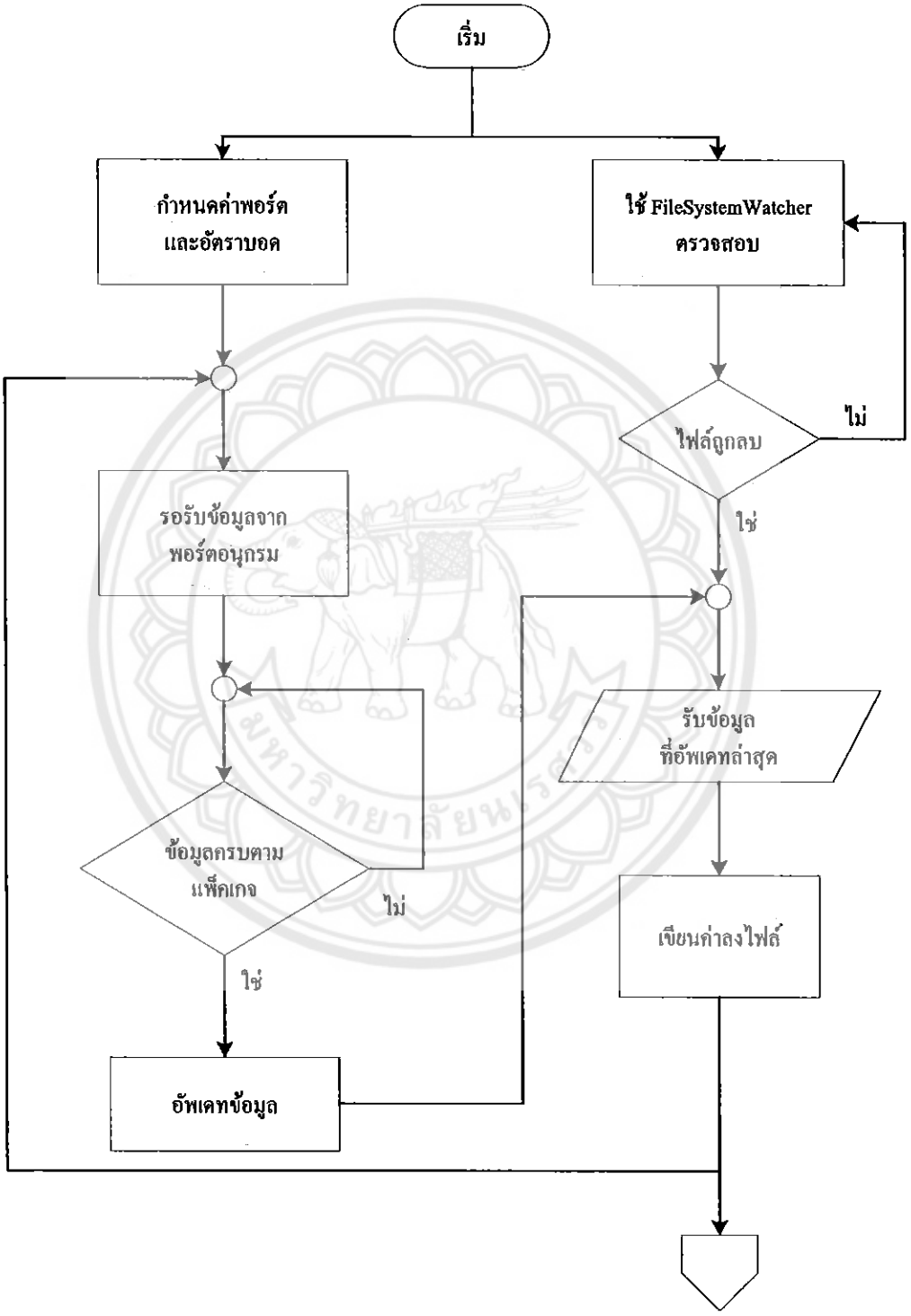
ส่วนที่ 2 ส่วนของ Image สำหรับแสดงแผนที่มหาวิทยาลัยนเรศวร

ส่วนที่ 3 ส่วนแสดงตำแหน่งรถไฟฟ้า

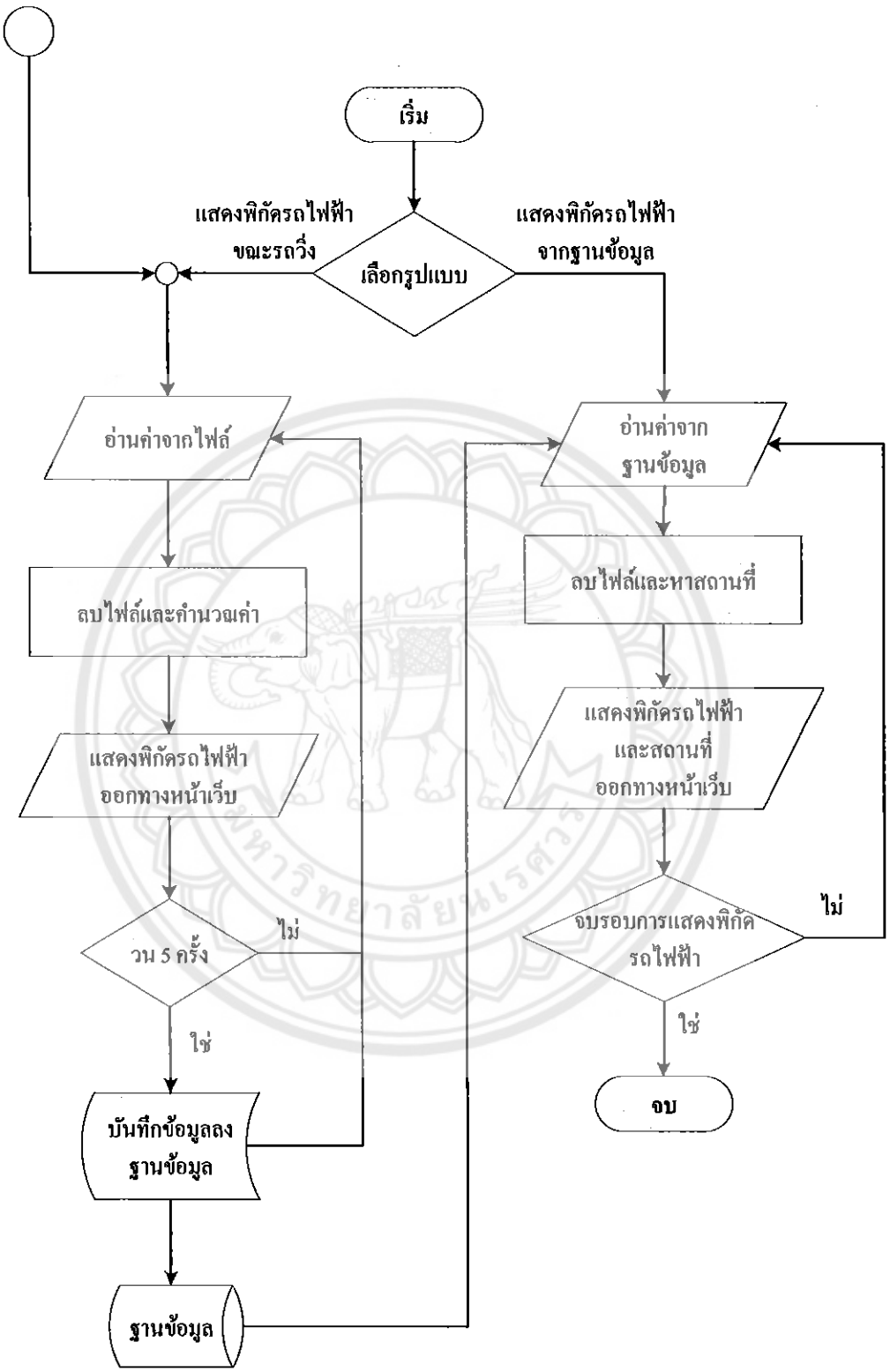
หลักการทำงานของหน้าแสดงพิกัดส่วน Summary รูปที่ 3.14

ใช้ JavaScript ตั้งเวลาให้ฟังก์ชันทำงานและหลักการวาดแผนที่เหมือนหน้า View Map แตกต่างกันตรงค่าที่นำมาใช้ในการวาดพิกัดรถไฟฟ้าจะดึงข้อมูลจากฐานข้อมูล โดยจะรับค่ารหัสรถไฟฟ้า ลำดับเริ่มต้นของรอบที่เลือก และจำนวนข้อมูลของรอบที่เลือก จากหน้าหลักมาใช้เป็นเงื่อนไขในการดึงข้อมูล แล้วนำข้อมูลมาเก็บไว้ในอาเรย์ เมื่อมีการเรียกฟังก์ชันโปรแกรมจะนำค่าในอาเรย์มาใช้วาดพิกัดรถไฟฟ้าและเพิ่มอินเด็กซ์ ของอาเรย์ เมื่อมีการเรียกใช้ฟังก์ชันโปรแกรมจะนำค่าตำแหน่งถัดไปในอาเรย์มาวาดพิกัดรถไฟฟ้า

3.4.3 แผนผังงานแสดงการทำงานของฝั่งแม่ข่าย



รูปที่ 3.15 แสดงผังการทำงานของโปรแกรมรับข้อมูลทางพอร์ตอนุกรม



รูปที่ 3.16 แสดงผังงานการทำงานของระบบแสดงผลทางหน้าเว็บ

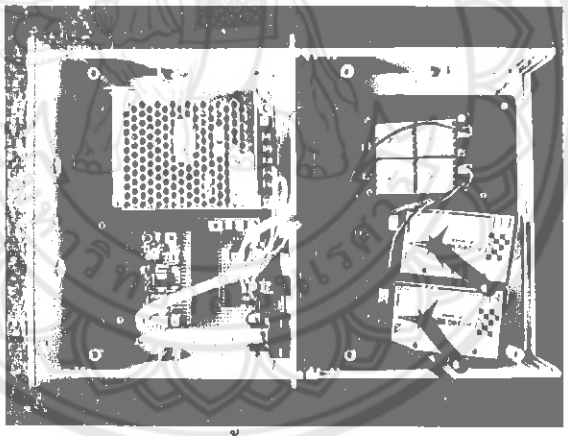
บทที่ 4

ผลการทดลอง

ในบทนี้กล่าวถึงการทดสอบและการวิเคราะห์ผลการทำงานของโปรแกรมที่พัฒนาเพื่อรับข้อมูลทางพอร์ตอนุกรมและโปรแกรมแสดงผลทางเว็บเซิร์ฟเวอร์

4.1 ส่วนป้ายแสดงผล

จ่ายไฟให้กับวงจรทั้งสาม แล้วแต่ละป้ายจะรอการรับค่าข้อมูลจาก เมื่อได้รับข้อมูลทั้งหมดแล้ว ไมโครคอนโทรลเลอร์จะทำการแสดงผลว่ารถไฟฟ้าอยู่บริเวณป้ายใด โดยหลอดไฟจะกระพริบอยู่ที่ป้ายนั้น ๆ และอักษรไฟคำว่า “Coming” จะสว่างขึ้นเมื่อรถไฟฟ้าอยู่ใกล้กับจุดรอรถไฟฟ้าก่อนหน้าของจุดที่สังเกตเห็น และในขณะเดียวกันไมโครคอนโทรลเลอร์จะทำการส่งข้อมูลต่อให้กับป้ายที่อยู่ต่อ ๆ ไปจนถึงเครื่องคอมพิวเตอร์แม่ข่าย เพื่อนำค่าไปแสดงผลต่อทางหน้าเว็บ

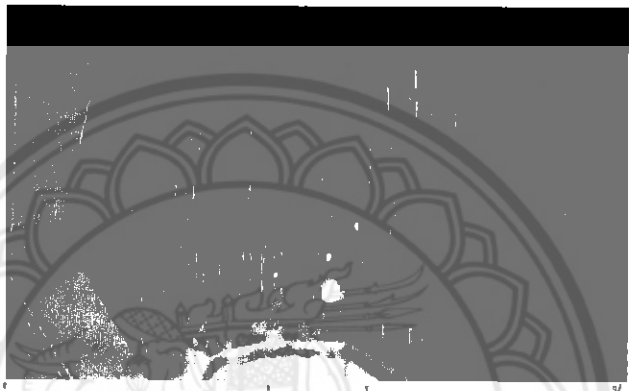


รูปที่ 4.1 แสดงการต่อไฟเลี้ยงให้กับวงจรด้วยไฟฟ้า 220 โวลต์

4.1.1 ลักษณะการติดดับของหลอดไฟขณะทดลอง

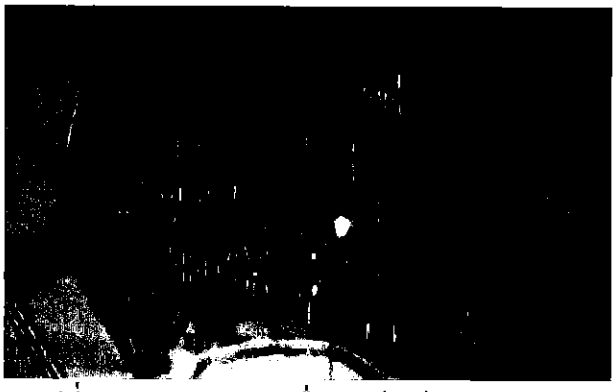
เมื่อรถไฟฟ้าวิ่งเข้าสู่ป้ายใด แล้วที่ป้ายแสดงผลจะแสดงสถานะด้วยไฟกระพริบที่หลอดไฟที่ตำแหน่งนั้น ๆ ซึ่งหมายความว่ารถไฟฟ้าอยู่ที่บริเวณนั้นของมหาวิทยาลัยนั่นเอง และถ้ารถไฟฟ้าและป้ายที่ผู้ใช้อยู่ห่างจากกันออกไปมากกว่า 2 ป้าย ป้ายแสดงผลก็จะแสดงผลดังรูปที่ 4.2 แต่ถ้าหากรถไฟฟ้าและป้ายที่ผู้ใช้อยู่ห่างกัน 1 ป้าย ป้ายแสดงผลก็จะแสดงผลดังรูปที่ 4.3 และเมื่อรถไฟฟ้ามาถึงป้ายที่ผู้ใช้อยู่ป้ายแสดงผลก็จะแสดงผลรูปที่ 4.4

รูปที่ 4.2 ป้ายแสดงผลเริ่มต้นเมื่อยังไม่มีการแสดงผล



รูปที่ 4.3 แสดงสถานะไฟเมื่อรถอยู่ที่ป้ายก่อนหน้า 2 ป้ายขึ้นไป

รูปที่ 4.4 แสดงสถานะเมื่อรถเข้าสู่ป้ายก่อนหน้า 1 ป้าย



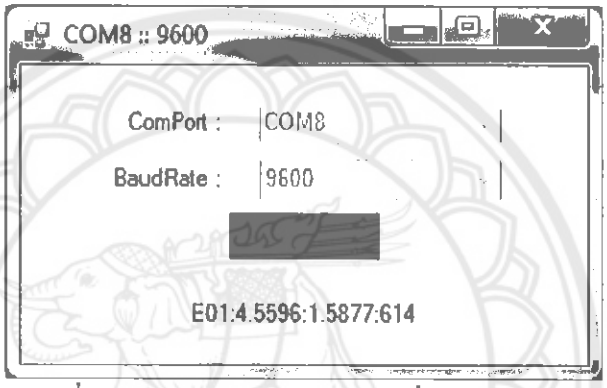
รูปที่ 4.5 แสดงสถานะเมื่อรถเข้าสู่ป้ายของตนเอง

4.2 ส่วนการแสดงผลฝั่งแม่ข่าย

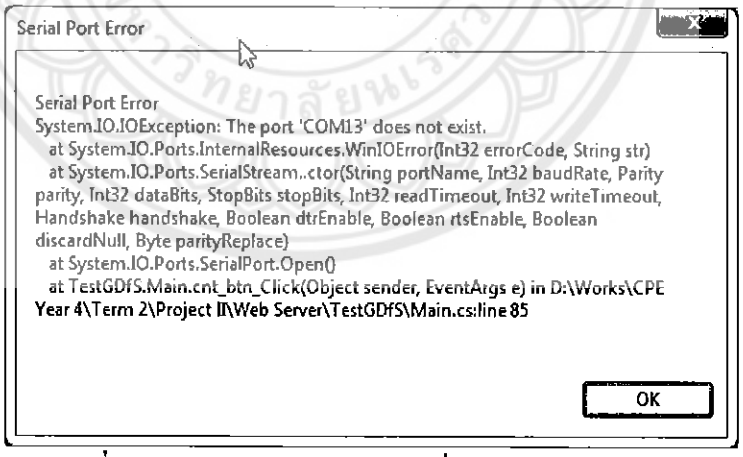
เปิดโปรแกรมรับข้อมูลจากไมโครคอนโทลเลอร์เพื่อรับข้อมูลพิกัดรถไฟฟ้า จากนั้นเปิดโปรแกรมเว็บแอปพลิเคชันเพื่ออ่านข้อมูลจากไฟล์แล้วนำข้อมูลมาแสดงขึ้นหน้าเว็บ

4.2.1 ส่วนโปรแกรมรับข้อมูลจากไมโครคอนโทรลเลอร์

ทำการเลือกคอมพอร์ตให้ตรงกับช่องพอร์ตที่เราเสียบสายพอร์ตอนุกรม และทำการเลือก baud rate ให้ตรงกับ baud rate ของไมโครคอนโทรลเลอร์ หลังจากนั้นกดปุ่ม Connect เช่นในรูปที่ 4.6 ในกรณีที่เลือกคอมพอร์ตไม่ถูกต้องจะปรากฏกล่องข้อความ (Message Box) ขึ้นมาแสดงข้อผิดพลาดดังรูปที่ 4.7



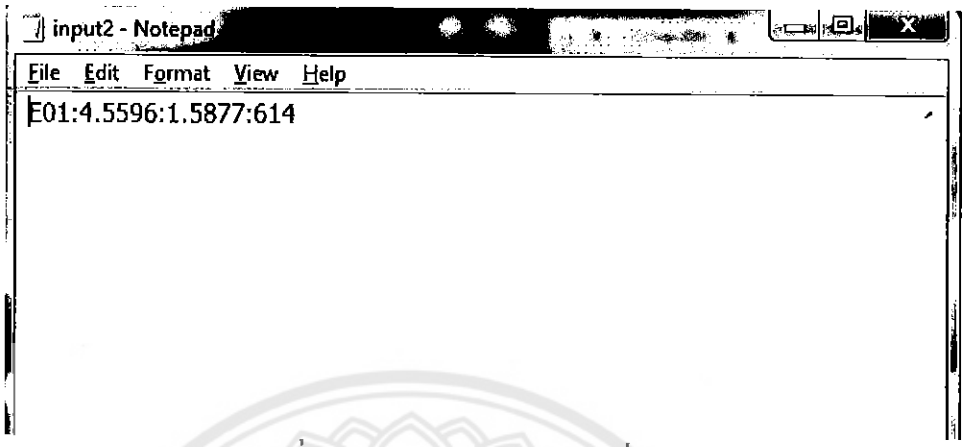
รูปที่ 4.6 แสดงผลของโปรแกรมเมื่อได้รับข้อมูล



รูปที่ 4.7 แสดงผลของโปรแกรมเมื่อกรอกคอมพอร์ตผิด

เมื่อได้รับค่าจากไมโครคอนโทรลเลอร์ผ่านทางเครื่องส่งสัญญาณไร้สายจะโปรแกรมจะทำการคำนวณ Checksum เพื่อตรวจสอบว่าข้อมูลที่ได้รับมาถูกต้องหรือไม่ ถ้าข้อมูลถูกต้องก็จะแสดงข้อมูลที่ได้รับมาดังรูปที่ 4.7 จากนั้นโปรแกรมจะทำการปรับค่าตัวแปรเพื่อรอเขียนลงไฟล์

เมื่อทางฝั่งเว็บแอปพลิเคชัน นำค่าพิกัดรถไฟฟ้าไปวาดแล้วทำการลบไฟล์ ก็จะนำค่าที่ได้รับจากไมโครคอนโทรลเลอร์เขียนลงในไฟล์ และจะได้ผลออกมาดังรูปที่ 4.8



รูปที่ 4.8 แสดงผลภายหลังไฟล์เมื่อไฟล์ถูกลบ

4.2.2 ส่วนการแสดงผลทาง Web Application

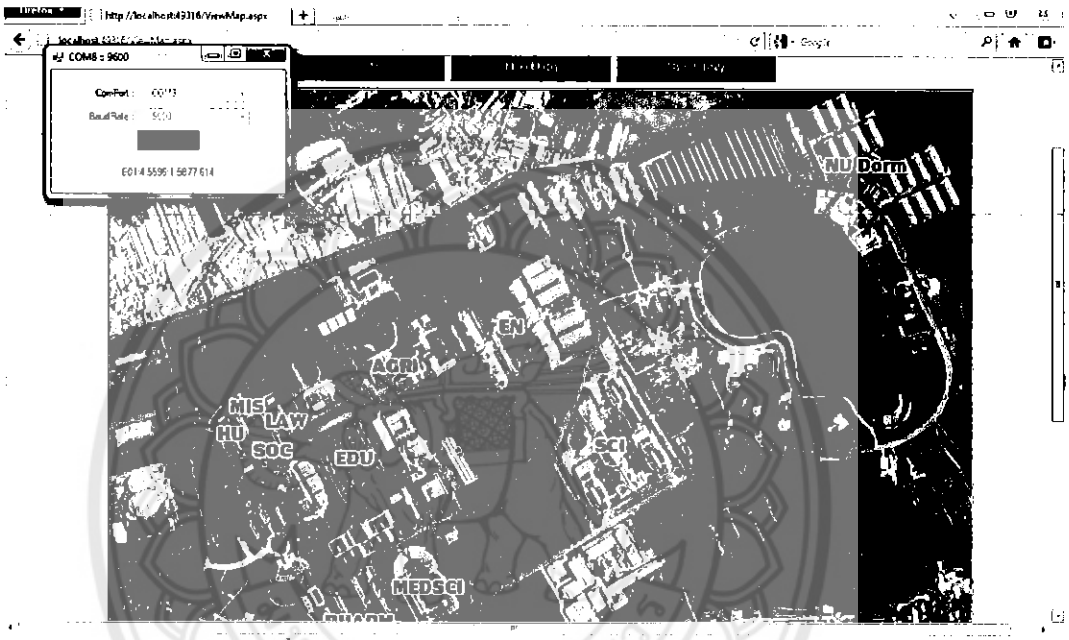
4.2.2.1 อ่านค่าจากไฟล์

เมื่อ JavaScript ทำการตั้งเวลาให้ฟังก์ชันทำงานทุก 0.5 วินาที เพื่อให้ได้ข้อมูลที่อัพเดทตลอดเวลาและนำมาแสดงพิกัดรถไฟฟ้าแบบเวลาจริง และใช้ฟังก์ชันที่ทำการอ่านข้อมูลพิกัดตำแหน่งของรถไฟฟ้าจากไฟล์ที่ถูกเขียนและนำค่ามาแสดงตำแหน่งของรถไฟฟ้าขึ้นบนเว็บไซต์ ดังรูปที่ 4.9 - 4.16

ผลการทดลองการแสดงผล

Input 1 คือ E01:4.5596:1.5877:614

โปรแกรมจะรับข้อมูล Input 1 มาคำนวณแล้ววาดจุดลงบนแผนที่ในหน้า View Map ผลที่ได้คือ จุดสีน้ำเงิน แสดงดังรูปที่ 4.9 และจะทำการแบ่งข้อมูลออกเป็นส่วน ๆ แสดงตัวเลขค่าละติจูด และลองจิจูดในหน้า Text Log พร้อมยังคำนวณหาสถานที่ที่ใกล้มาแสดงด้วยดังรูปที่ 4.10



รูปที่ 4.9 Output ส่วน View Map เมื่อได้รับข้อมูล Input 1

EVT ID	Latitude	Longitude	Location	Time
E01	16.445596	100.115877	Sdence	01/04/2555 23:50:05
:	:	:	:	:
:	:	:	:	:

รูปที่ 4.10 Output ส่วน Text Log เมื่อได้รับข้อมูล Input 1

Input 2 คือ E01:4.5822:1.6354:62

โปรแกรมจะรับข้อมูล Input 2 มาคำนวณแล้ววาดจุดลงบนแผนที่ในหน้า View Map ผลที่ได้คือ จุดสีน้ำเงิน แสดงดังรูปที่ 4.11 และจะทำการแบ่งข้อมูลออกเป็นส่วนๆ แสดงตัวเลขค่าละติจูด และลองจิจูดในหน้า Text Log พร้อมยังคำนวณหาสถานที่ที่ใกล้มาแสดงด้วยดังรูปที่ 4.12

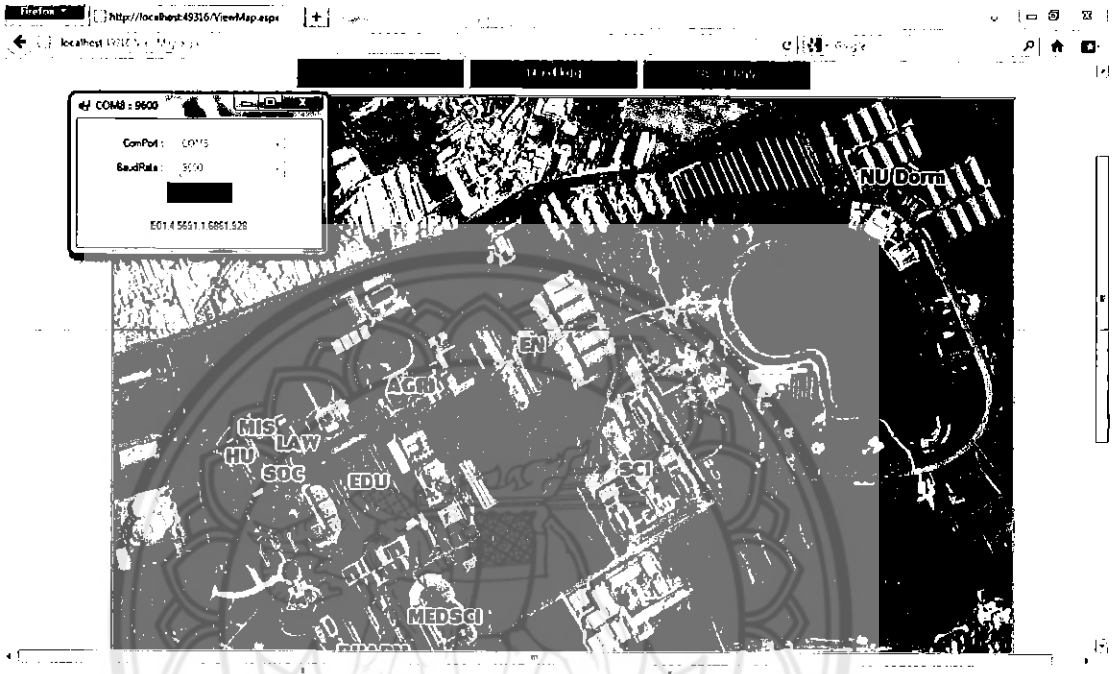
รูปที่ 4.11 Output ส่วน View Map เมื่อได้รับข้อมูล Input 2

EVT ID	Latitude	Longitude	Location	Time
E01	16.445822	100.116354	Science	01/04/2555 23:51:09
:	:	:	:	:
:	:	:	:	:

รูปที่ 4.12 Output ส่วน Text Log เมื่อได้รับข้อมูล Input 2

Input 3 คือ E01:4.5691:1.6861:928

โปรแกรมจะรับข้อมูล Input 3 มาคำนวณแล้ววาดจุดลงบนแผนที่ในหน้า View Map ผลที่ได้คือ จุดสีน้ำเงิน แสดงดังรูปที่ 4.13 และจะทำการแบ่งข้อมูลออกเป็นส่วนๆ แสดงตัวเลขค่าละติจูด และลองจิจูดในหน้า Text Log พร้อมยังคำนวณหาสถานที่ที่ใกล้มาแสดงด้วย ดังรูปที่ 4.14



รูปที่ 4.13 Output ส่วน View Map เมื่อได้รับข้อมูล Input 3

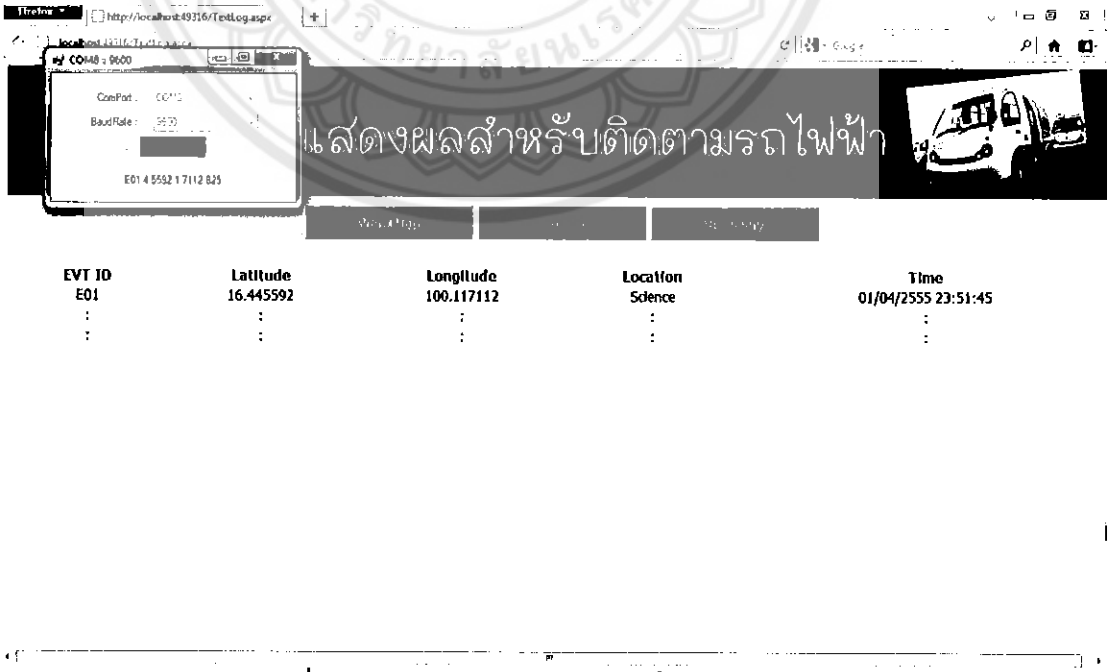
EVT ID	Latitude	Longitude	Location	Time
E01	16.445691	100.116861	Science	01/04/2555 23:51:27
:	:	:	:	:
:	:	:	:	:

รูปที่ 4.14 Output ส่วน Text Log เมื่อได้รับข้อมูล Input 3

Input 4 คือ E01:4.5592:1.7112:825

โปรแกรมจะรับข้อมูล Input 4 มาคำนวณแล้ววาดจุดลงบนแผนที่ในหน้า View Map ผลที่ได้คือ จุดสีน้ำเงิน แสดงดังรูปที่ 4.15 และจะทำการแบ่งข้อมูลออกเป็นส่วนๆ แสดงตัวเลขค่าละติจูดและลองจิจูดในหน้า Text Log พร้อมยังคำนวณหาสถานที่ที่ใกล้มาแสดงด้วย ดังรูปที่ 4.16

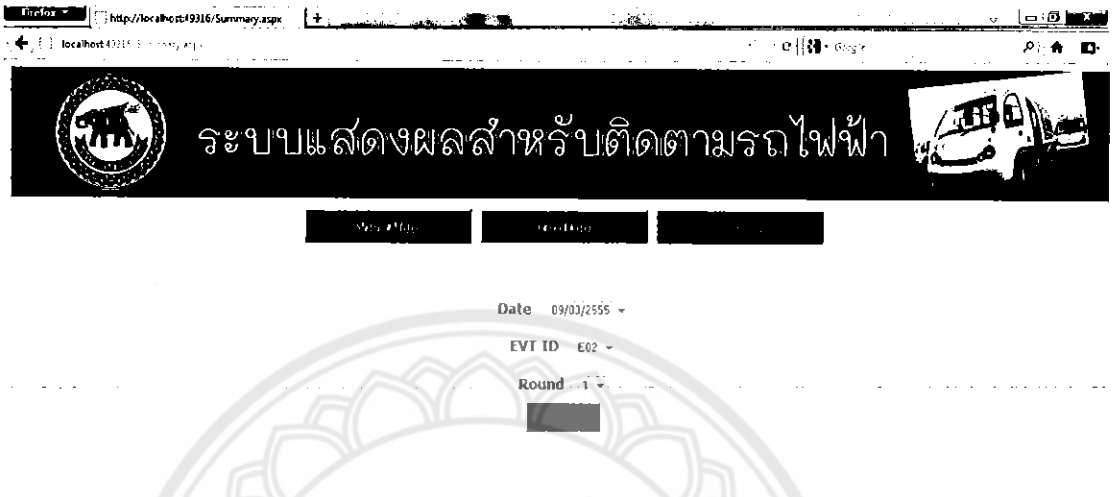
รูปที่ 4.15 Output ส่วน View Map เมื่อได้รับข้อมูล Input 4



รูปที่ 4.16 Output ส่วน Text Log เมื่อได้รับข้อมูล Input 4

4.2.2.2 อ่านค่าจากฐานข้อมูล

เลือกรถไฟฟ้าและรอบที่ต้องตรวจสอบ แล้วกดปุ่ม Run ในรูปที่ 4.17 หน้าเว็บจะเชื่อมต่อไปยังหน้าแสดงพิกัดรถไฟฟ้าดังรูปที่ 4.18



รูปที่ 4.17 หน้าหลักของส่วน Summary

รูปที่ 4.18 หน้าแสดงพิกัดรถไฟฟ้าของส่วน Summary

4.3 การทดลองวัดความคลาดเคลื่อนของจุดที่แสดงบนแผนที่บนหน้าเว็บ

ตารางที่ 4.1 แสดงค่าวัดความคลาดเคลื่อนของจุดที่แสดงบนแผนที่

จุดที่	ละติจูด	ลองจิจูด	ค่าความคลาดเคลื่อน (มม.)	สถานที่
1	1644.3315	10011.9657	2	จุดออกรถไฟฟ้า
2	1644.5007	10011.9321	0	โภชนาการ 2
3	1644.6065	10011.8867	0	คณะวิศวกรรมศาสตร์
4	1644.7246	10011.8421	0	คณะวิศวกรรมศาสตร์
5	1644.8501	10011.7718	0	สถานบริการเทคโนโลยีสารสนเทศและการสื่อสาร
6	1644.9849	10011.6664	0	คณะมนุษย์ศาสตร์
7	1644.9807	10011.5762	1	อาคารอเนกประสงค์
8	1644.9868	10011.493	0	ลานสมเด็จ
9	1645.0322	10011.4064	0	สามแยกหน้าโรงพยาบาล
10	1645.0403	10011.3773	2	สามแยกหน้าโรงพยาบาล
11	1645.0112	10011.3653	2	โรงพยาบาล
12	1644.8813	10011.4017	4	โรงพยาบาล
13	1644.818	10011.3784	3	คณะทันตะแพทยศาสตร์
14	1644.6278	10011.4938	1	อาคารเอกาทศรถ
15	1644.6741	10011.5864	0	อาคารเฉลิมพระเกียจติ
16	1644.5668	10011.5315	2	อาคารเอกาทศรถ
17	1644.544	10011.5741	0	คณะวิทยาศาสตร์
18	1644.5281	10011.7976	1	คณะวิศวกรรมศาสตร์
19	1644.5656	10011.8898	1	คณะวิศวกรรมศาสตร์
20	1644.1921	10011.9175	5	หอพักนิสิต

ผลที่ได้จากการทดลองคือ ค่าความคาดเคลื่อนเฉลี่ยเท่ากับ 1.2 มิลลิเมตร

บทที่ 5

สรุปผลการดำเนินงานโครงงาน

ระบบเครือข่ายและแสดงผลสำหรับติดตามรถไฟฟ้าถูกพัฒนาขึ้นเพื่อนำเทคโนโลยีการเชื่อมต่อไร้สายเข้ามาประยุกต์ใช้ เพื่อใช้รับข้อมูลเกี่ยวกับตำแหน่งของรถไฟฟ้าภายในมหาวิทยาลัยเข้าสู่บอร์ดไมโครคอนโทลเลอร์แล้วนำข้อมูลตำแหน่งของรถไฟฟ้าไปแสดงผลเพื่อช่วยเพิ่มความสะดวกให้กับผู้ใช้บริการรถไฟฟ้า

หลักการพื้นฐานที่เกี่ยวข้องกับรูปแบบการเชื่อมโยงเครือข่ายเพื่อใช้ในการรับ-ส่งข้อมูลด้วยไมโครคอนโทรลเลอร์ STM23F103-STAMP/128 และการเขียนเว็บแอพพลิเคชั่นเพื่อแสดงผลด้วย ASP.NET และ .NET Framework

ขั้นตอนการทำงานจะเริ่มจากรับข้อมูลเมื่อรถไฟฟ้าเดินทางมาถึงจุดรับสัญญาณต่าง ๆ เมื่อสถานีใด ๆ รับสัญญาณได้แล้วก็จะประมวลผลข้อมูลไปแสดงผลที่ป้ายแสดงผล พร้อมกับส่งข้อมูลต่อ ไปยังสถานีรับสัญญาณถัดไปเพื่อใช้แสดงผลและจะเป็นเช่นนี้ไปเรื่อย ๆ จนข้อมูลส่งกลับมาถึงตัวเอง และข้อมูลก็จะส่งไปถึงเครื่องแม่ข่ายด้วย เมื่อเครื่องแม่ข่ายได้รับข้อมูลแล้ว จะประมวลผลข้อมูลให้แสดงผลผ่านทางเว็บแอปพลิเคชัน

ผลทดลองการรับส่งข้อมูลต้องอยู่ในพื้นที่โล่งและค่อนข้างปลอดโปร่งเพื่อให้สามารถรับส่งข้อมูลระหว่างรถกับสถานีได้ในระยะไกลที่สุดและข้อมูลครบถ้วนถูกต้อง เพื่อนำข้อมูลไปประมวลผลนำไปแสดงผลที่ป้ายแสดงผลและบนเว็บแอพพลิเคชั่น และสุดท้ายจะเป็นการทดลองวัดความคลาดเคลื่อนของจุดที่แสดงบนแผนที่ ซึ่งได้ทำการสรุปไว้ดังนี้

5.1 สรุปผลการทดลองของโครงงาน

5.2 สรุปปัญหาและอุปสรรคของโครงงาน

5.3 ข้อเสนอแนะและแนวทางการพัฒนา

5.1 สรุปผลการทดลองของโครงงาน

จากการทดลองทั้งหมดเป็นการวัดประสิทธิภาพของการรับส่งข้อมูลตำแหน่งของรถไฟฟ้าเข้าสู่บอร์ดไมโครคอนโทรลเลอร์ ซึ่งสามารถรับข้อมูลได้ และสามารถส่งค่าไปจนถึงปลายทางได้อย่างปกติ

การแสดงผลที่ป้ายแสดงผลสามารถแสดงผลได้อย่างถูกต้อง แต่การแสดงผลหน้าเว็บที่รับข้อมูลทางพอร์ตอนุกรมและนำข้อมูลมาประมวลผลเพื่อแสดงทางหน้าเว็บยังมีความคาดเคลื่อนในการแสดงผลตำแหน่งของรถไฟฟ้าบนแผนที่เล็กน้อย ซึ่งผลจากการทดลองวัดค่าความคาดเคลื่อนเฉลี่ยได้ 1.2 มิลลิเมตร

5.2 วิเคราะห์ข้อดีและข้อเสียของโครงงาน

5.2.1 ข้อดี

5.2.1.1 ประหยัดค่าใช้จ่ายในการติดตั้ง โดยที่ค่าใช้งบประมาณต่อสถานีประมาณ 8,000 บาท

5.2.1.2 ง่ายต่อการเพิ่มสถานีเข้าสู่ระบบ

5.2.1.3 เนื่องจากมีการติดต่อกันแบบไร้สาย จึงทำให้ง่ายต่อการติดตั้งป้ายแสดงผล

5.2.2 ข้อเสีย

5.2.2.1 เนื่องจากเครือข่ายเป็นแบบวงแหวน ดังนั้นระบบจะทำงานได้ก็ต่อเมื่อทุกสถานีมีการทำงานอย่างปกติ เพราะถ้าหากมีสถานีใดสถานีหนึ่งเกิดรับส่งข้อมูลไม่ได้ แล้วจะทำให้สถานีต่อมาไม่สามารถทำงานได้

5.2.2.2 เนื่องจากอุปกรณ์รับส่งแบบไร้สายไม่สามารถจัดการการชนกันของข้อมูลได้จึงทำให้ได้ข้อมูลที่ไม่ถูกต้องถ้าหากมีการส่งข้อมูลพร้อม ๆ กัน

5.2.2.3 ยากต่อการตรวจสอบว่าสถานีใดเสีย

5.3 ปัญหาที่พบและแนวทางการแก้ไขปัญหา

ตารางที่ 5.1 แสดงปัญหาที่พบและแนวทางการแก้ไขปัญหา

ปัญหาที่พบ	แนวทางการแก้ไขปัญหา
1. ไม่สามารถส่งสัญญาณแบบไร้สายผ่านสิ่งกีดขวางที่มีขนาดใหญ่ได้ เช่น บริเวณที่มีต้นไม้หนาทึบ หรืออาคารใหญ่	การใช้โหนดรับส่งสัญญาณกั้นกลางระหว่างสถานี 2 สถานี
2. ความเร็วในการประมวลผลระหว่างเครื่องแม่ข่ายกับบอร์ดไมโครคอนโทรลเลอร์ที่ไม่เท่ากัน จึงทำให้เครื่องแม่ข่ายได้รับข้อมูลไม่ถูกต้อง	ใช้การหน่วงการทำงานที่เครื่องแม่ข่าย เพื่อให้มีการรับข้อมูลได้ถูกต้อง
3. ไมโครคอนโทรลเลอร์ไม่มีการตอบสนอง เนื่องจากข้อมูลที่ได้รับมีการชนกัน	1. กดปุ่มรีเซตที่บอร์ดไมโครคอนโทรลเลอร์ 2. สร้างฟังก์ชัน Watchdog เพื่อทำการรีเซตไมโครคอนโทรลเลอร์แบบอัตโนมัติ

5.4 ข้อเสนอแนะ

จากปัญหาที่พบในโครงงานนี้ ทางผู้จัดทำขอเสนอแก้ไขปัญหา ดังนี้

5.3.1 กล่องที่บรรจุไมโครคอนโทรลเลอร์ควรมีระบบระบายอากาศ เพื่อช่วยระบายความร้อน

5.3.2 ในกรณีที่มีสิ่งกีดขวางระหว่างสถานีรับส่งสัญญาณหรือสถานีรับส่งสัญญาณอยู่ห่างกันเกินกว่า 200 เมตร ควรมีสถานีรับส่งสัญญาณคั่นกลางระหว่างสถานีทั้ง 2 สถานี

5.3.3 การปรับปรุงกระบวนการทำงานทางด้านซอฟต์แวร์ ช่วยทำให้ระบบการทำงานภายในไมโครคอนโทรลเลอร์มีความเร็วเพิ่มขึ้น

5.3.4 ใช้ zigbee เพื่อเพิ่มประสิทธิภาพในการรับส่งข้อมูลให้มีความถูกต้องแม่นยำยิ่งขึ้น

5.3.5 ควรปรับปรุงเครือข่ายเป็นแบบตาข่าย (Mesh) เพื่อตัดปัญหาที่เกิดเหตุการณ์ที่สถานีใดสถานีหนึ่งไม่สามารถส่งข้อมูลได้ จะทำให้สถานีที่อยู่ต่อ ๆ ไปและเครื่องแม่ข่ายไม่สามารถแสดงผลไม่ได้เนื่องจากเครือข่ายที่ใช้เป็นแบบวงแหวน

เอกสารอ้างอิง

[1] โครงสร้างเครือข่ายคอมพิวเตอร์. สืบค้นเมื่อ สิงหาคม 2554, จาก http://blog.cstc.ac.th/node/126.

[2] .NET Framework คืออะไร. สืบค้นเมื่อ สิงหาคม 2554, จาก http://www.sapaan.net/forum/programming-community/net-framework-xiidaa-aooeaoaadcoaeon -acaoadaaa1iaadeioiaeoaaoeo/.

[3] ASP. NET. สืบค้นเมื่อ สิงหาคม 2554, จาก http://th.wikipedia.org/wiki/เอเอสพีดอตเน็ต.

[4] คู่มือการใช้งานบอร์ดไมโครคอนโทรลเลอร์รุ่น ET-ARM STAMP STM32 (Cortex-M3). สืบค้นเมื่อ สิงหาคม 2554, จาก http://www.ett.co.th/product/ARM/manET-STAMP-STM32.pdf.

[5] STM32F103RBT6 - Performance line, ARM-based 32-bit MCU with Flash, USB, CAN, seven 16-bit timers, two ADCs and nine communication interfaces – STMicroelectronics. Retrieved August , 2011, from http://pdf1.alldatasheet.com/datasheet-pdf/view/201599/STMICROELECTRONICS/STM32F103RBT6.html.

ภาคผนวก ก

โค้ดภายในไมโครคอนโทรลเลอร์

```
int main()
{
  while(1){
    //copy key is recieved from USART_2 to RxDataBuffer
    RxDataBuff = GetKeyUart2();

    //check key is recieved is not null
    if(RxDataBuff != 0){
      GPIO_SetBits(GPIOB, GPIO_Pin_14);
    }

    //when data package is carry Station, CarID, Lat, Long
    if((nData2 == 0 && RxDataBuff == 'S') || ((1 <= nData2) && ( nData2 < 15))){
      GPIO_SetBits(GPIOB, GPIO_Pin_13);
      RxUart2[nData2] = RxDataBuff;
      SysTick_CounterCmd(SysTick_Counter_Disable);
      nData2 = nData2 + 1;
      GPIO_ResetBits(GPIOB, GPIO_Pin_13);
    }

    if(nData2 >= 15){   //when recieve the car-package complete
      GPIO_SetBits(GPIOB, GPIO_Pin_15);
      CarLabel2 = ((short)RxUart2[2] + (short)RxUart2[3] + (short)RxUart2[4]) - 166;
      StatusCar[CarLabel2] = RxUart2[1];
      nData2 = 0;

      DisplayLED();

      if(RxUart2[1] != thisStation){
        //DelayuS(500000);
        sendDataUART2();
      }
      else{
        RxUart2[1] = 0;
      }
      GPIO_ResetBits(GPIOB, GPIO_Pin_15);
      GPIO_ResetBits(GPIOB, GPIO_Pin_14);
    }
  }
}
```

รูป ก1 แสดงโค้ดส่วนโปรแกรมหลัก

โค้ด รูป ก1 คือส่วนของการทำงานหลักของโปรแกรม คือการรอรับค่าข้อมูลจากสถานีแล้วเพื่อนำค่าข้อมูลที่ได้ไปประมวลผลและอัพเดทตำแหน่งที่รถแต่ละคันถึงเพื่อการนำข้อมูลที่ได้ไปแสดงผลผ่านทางป้ายแสดง และส่งข้อมูลต่อให้กับเสารับสัญญาณเสาต่อไป เพื่อให้ป้ายแสดงผลทั้งหมดมีการแสดงผลแบบเดียวกัน

```
void DisplayLED(){
  // A = 8, B = 9, C = 10
  for(int n = 0; n < 25; n = n+1){
    if(StatusCar[n] == 'A'){
      GPIO_SetBits(GPIOB, GPIO_Pin_8);
    }
    if(StatusCar[n] == 'B'){
      GPIO_SetBits(GPIOB, GPIO_Pin_9);
    }
    if(StatusCar[n] == 'C'){
      GPIO_SetBits(GPIOB,
GPIO_Pin_10);
    }
  }
}
```

รูป ก2 แสดงโค้ดส่วนการทำให้หลอดไฟสว่าง

โค้ด รูป ก2 คือการสั่งให้หลอดไฟสว่างขึ้นโดยในโค้ดจะเทียบข้อมูลป้ายที่รถแต่ละคันไปถึงว่าถึงสถานีใด แล้วก็จะสั่งให้หลอดไฟที่แทนตำแหน่งของสถานีนั้นสว่างขึ้นมา

```
void sendDataUART2(){
  u8 i;
  for(i = 0; i < 16; i = i+1){
    SendCharUart2(RxUart2[i]);
  }
}
```

รูป ก3 แสดงโค้ดส่วนของการส่งข้อมูล

โค้ด รูป ก3 คือการส่งข้อมูลที่ถูกเก็บเรียงกันในอาร์เรย์โดยการส่งจะเป็นการส่งข้อมูลออกไปทีละตัวจนครบ

ภาคพนวก ข

โค้ดการทำงานของฝั่งแม่ข่าย

```
<script type="text/javascript">

            // Set timeout to call page method.
            setInterval(nextPicture, 500);

            // Call page method to get data and call function to  show image.
            function nextPicture() {

                PageMethods.readTextFile(OnMyMethodComplete);
            }

            // Read data and call image by send data to plot page.
            function OnMyMethodComplete(result, userContext, methodName) {

                document.getElementById("Image1").src = "plotMap.aspx?" + result;

            }

</script>
```

รูป ข1 แสดงโค้ดจาวาสคริปต์ตั้งเวลาเรียกฟังก์ชัน

โค้ด รูป ข1 เป็นจาวาสคริปต์ที่ใช้สำหรับตั้งเวลาในการเรียกฟังก์ชันทุกๆ ครึ่งวินาที โดยจะเรียกฟังก์ชันอ่านข้อมูลจากไฟล์แล้วนำข้อมูลมา

```
            mX = 1048.853579;
            mY = 1080.78427;
            ccX = -340.111959;
            ccY = -125.550356;

            cid = Car_ID;
            lat = Car_Lat - 4;
            lng = Car_Long - 1;

             // Find x and y from latitude and longitude
            inputX = 1044 - (int)Math.Round(mY * lat + ccY, 0);
            inputY = 782 - (int)Math.Round(mX * lng + ccX, 0);
```

รูป ข2 แสดงโค้ดแปลงพิกัดโลกเป็นตำแหน่งพิกเซล

โค้ด รูป ข2 ใช้สำหรับคำนวณแปลงข้อมูลพิกัดโลกไปเป็นตำแหน่งพิกเซลเพื่อใช้ในการแสดงตำแหน่งรถไฟฟ้าขึ้นหน้าเว็ปแอปพลิเคชัน

```
 // Create bitmap for Map and Plot
Bitmap bmp = new Bitmap(Properties.Resources.MapNUGG2);
int height = bmp.Height;
int width = bmp.Width;
Graphics g = Graphics.FromImage(bmp);
g.TextRenderingHint = TextRenderingHint.AntiAlias;
//g.Clear(Color.Orange);
g.DrawRectangle(Pens.White, 1, 1, width - 3, height - 3);
g.DrawRectangle(Pens.Gray, 2, 2, width - 3, height - 3);
g.DrawRectangle(Pens.Black, 0, 0, width, height);
//-----------------------------------------------------------//

 // Plot electric vehicle position on Map.
 g.DrawPolygon(pen1, poly);
g.DrawString(cid1, new Font("Tahoma", 15, FontStyle.Bold),
    Brushes.Red, inputX - 22, inputY + 10);
g.FillPolygon(blueBrush, poly);


Response.ContentType = "image/gif";
bmp.Save(Response.OutputStream, ImageFormat.Gif);
g.Dispose();
bmp.Dispose();
Response.End();
```

รูป ข3 แสดงโค้ดสำหรับวาดตำแหน่งรถไฟฟ้า

โค้ด รูป ข3 ใช้สำหรับแสดงแผนที่และตำแหน่งรถไฟฟ้า โดยจะทำให้หน้าเว็บเป็นข้อมูลรูปภาพ เพื่อให้หน้าหลักใช้ดึงรูปแผนที่ที่วาดตำแหน่งแล้วไปแสดงขึ้นหน้าเว็บ

ภาคผนวก ค

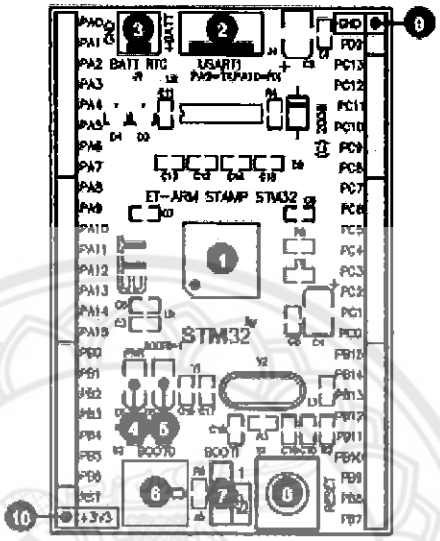
รายละเอียดของไมโครคอนโทรลเลอร์ ARM STM32F103 Cortex-M3

คุณสมบัติของบอร์ด รุ่น ET-ARM STAMP STM32

1. ใช้ MCU 32Bit ตระกูล ARM Cortex-M3 ขนาด 64Pin(LQFP) ของ "STmicroelectronics"
2. ใช้ Crystal 8.00 MHz + Phase-Locked Loop (PLL) Run ความถี่ 72 MHz ประมวลผลด้วยความเร็ว 1.25 DMIPS/MHz เทียบเท่ากับความเร็วการประมวลผลเท่ากับ 90MIPS
3. มีวงจร RTC(Real Time Clock) พร้อม XTAL ค่า 32.768KHz และ Battery Backup
4. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต USART1 (RS232)
5. มีวงจรสื่อสาร RS232 โดยใช้ขั้วต่อแบบ 4-PIN มาตรฐาน ETT จำนวน 1 ช่อง
6. มี 48 Bit GPIO พร้อม 5V-Tolerant สำหรับประยุกต์ต่างๆโดยอิสระ
7. ใช้กับแหล่งจ่ายไฟขนาด +3.3V
8. ขั้วต่อใช้งานวางตัวบน Pin Header ระยะห่าง 2.54mm(100mil) ขนาด 50 Pin (ด้านละ 25Pin) ระยะห่าง 1.5 นิ้ว(1500mil/38.1mm)ง่ายต่อการนำไปต่อประยุกต์ใช้งาน และ ขยายวงจร I/O สามารถใช้กับ Project Board และ PCB

Specification Peripheral	ET-ARM STAMP STM32F103/128	ET-ARM STAMP STM32F103/512
MCU	STM32F103RBT6	STM32F103RET6
Flash	128K	512K
RAM	20K	64K
SPI	2	3 (I2S x 2)
I2C	2	2
USART	3	5
USB	1	1
CAN	1	1
SDIO	-	1
ADC 12 Bit	**16**	**16**
DAC 12 Bit	**-**	**2**

โครงสร้างบอร์ด ET-ARM STAMP STM32



รูปแสดง ตำแหน่งของอุปกรณ์ต่าง ๆในบอร์ด ET-ARM STAMP STM32

- หมายเลข 1 คือ MCU ประจำบอร์ด
- หมายเลข 2 คือ ขั้วต่อ USART1(RS232) สำหรับใช้งาน และ Download
- หมายเลข 3 คือ ขั้วต่อ Battery Backup-RTC ขนาด +3V
- หมายเลข 4 คือ LED Power
- หมายเลข 5 คือ LED BOOT0 (BOOT0=1)
- หมายเลข 6 คือ สวิตช์ BOOT0 ใช้ร่วมกับ Jumper J1 สำหรับเลือกโหมดการทำงานของบอร์ด แบบ Download จาก USART1 หรือ Run ปรกติ โดย ON เพื่อ Download และ OFF เพื่อ Run
- หมายเลข 7 คือ Jumper(BOOT1) ใช้ร่วมกับสวิตช์ BOOT0 สำหรับเลือกโหมดการทำงานของบอร์ดแบบ Download จาก USART1 หรือ Run ปรกติ ซึ่งต้องกำหนดไว้ที่ ISP เสมอ
- หมายเลข 8 คือ สวิตช์ RESET สำหรับ RESET การทำงานของ MCU
- หมายเลข 9 คือ จุดต่อ GND
- หมายเลข 10 คือ ไฟเลี้ยงวงจร +3.3V

รายละเอียดของไอซีหมายเลข STM32F103RBT6

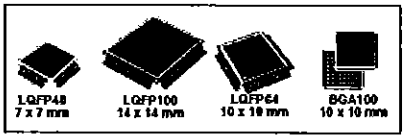
STM32F103x6
STM32F103x8 STM32F103xB

Performance line, ARM-based 32-bit MCU with Flash, USB, CAN, seven 16-bit timers, two ADCs and nine communication interfaces

Preliminary Data

Features

- Core: ARM 32-bit Cortex™-M3 CPU
  - 72 MHz, 90 DMIPS with 1.25 DMIPS/MHz
  - Single-cycle multiplication and hardware division
  - Nested interrupt controller with 43 maskable interrupt channels
  - Interrupt processing (down to 6 CPU cycles) with tail chaining
- Memories
  - 32-to-128 Kbytes of Flash memory
  - 6-to-20 Kbytes of SRAM
- Clock, reset and supply management
  - 2.0 to 3.6 V application supply and I/Os
  - POR, PDR, and programmable voltage detector (PVD)
  - 4-to-16 MHz quartz oscillator
  - Internal 8 MHz factory-trimmed RC
  - Internal 32 kHz RC
  - PLL for CPU clock
  - Dedicated 32 kHz oscillator for RTC with calibration
- Low power
  - Sleep, Stop and Standby modes
  - V_{BAT} supply for RTC and backup registers
- 2 x 12-bit, 1 µs A/D converters (16-channel)
  - Conversion range: 0 to 3.6 V
  - Dual-sample and hold capability
  - Synchronizable with advanced control timer
  - Temperature sensor
- DMA
  - 7-channel DMA controller
  - Peripherals supported: timers, ADC, SPIs, I^2Cs and USARTs



- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
- Up to 80 fast I/O ports
  - 32/49/80 5 V-tolerant I/Os
  - All mappable on 16 external interrupt vectors
  - Atomic read/modify/write operations
- Up to 7 timers
  - Up to three 16-bit timers, each with up to 4 IC/OC/PWM or pulse counter
  - 16-bit, 6-channel advanced control timer:
    - up to 6 channels for PWM output
    - Dead time generation and emergency stop
  - 2 x 16-bit watchdog timers (Independent and Window)
  - SysTick timer: a 24-bit downcounter
- Up to 9 communication interfaces
  - Up to 2 x I^2C interfaces (SMBus/PMBus)
  - Up to 3 USARTs (ISO 7816 interface, LIN, IrDA capability, modem control)
  - Up to 2 SPIs (18 Mbit/s)
  - CAN interface (2.0B Active)
  - USB 2.0 full speed interface

Table 1. Device summary

Reference	Root part number
STM32F103x6	STM32F103C6, STM32F103R6
STM32F103x8	STM32F103C8, STM32F103R8 STM32F103V8
STM32F103xB	STM32F103RB STM32F103VB

This is preliminary information on a new product now in development or undergoing evaluation. Details are subject to change without notice.

www.st.com

Figure 3. STM32F103xx performance line LQFP64 pinout

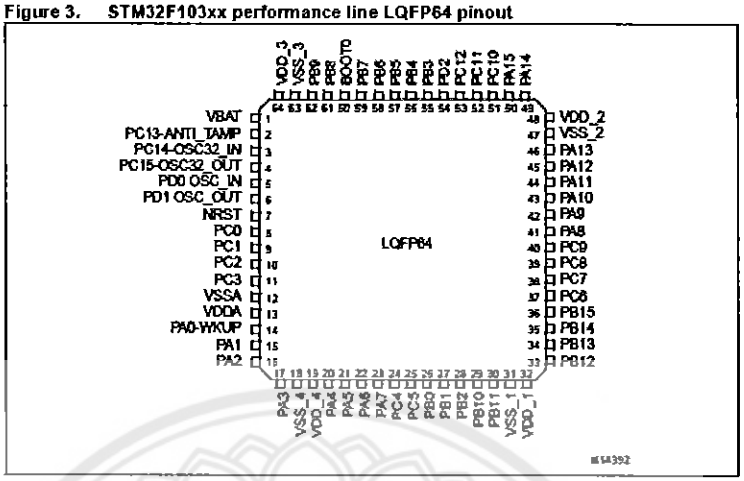


Figure 4. STM32F103xx performance line LQFP48 pinout

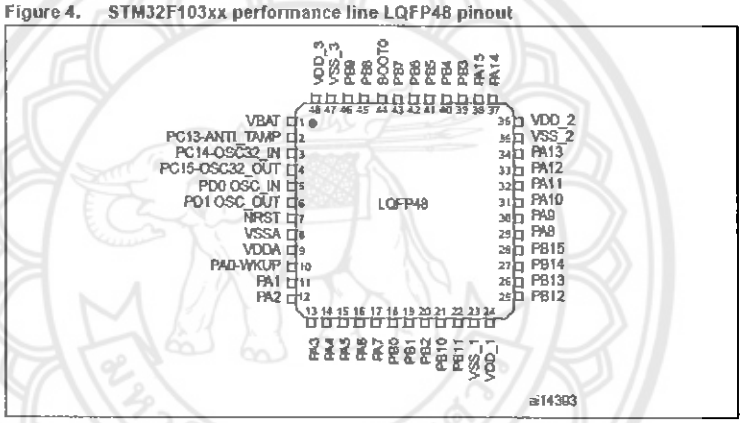
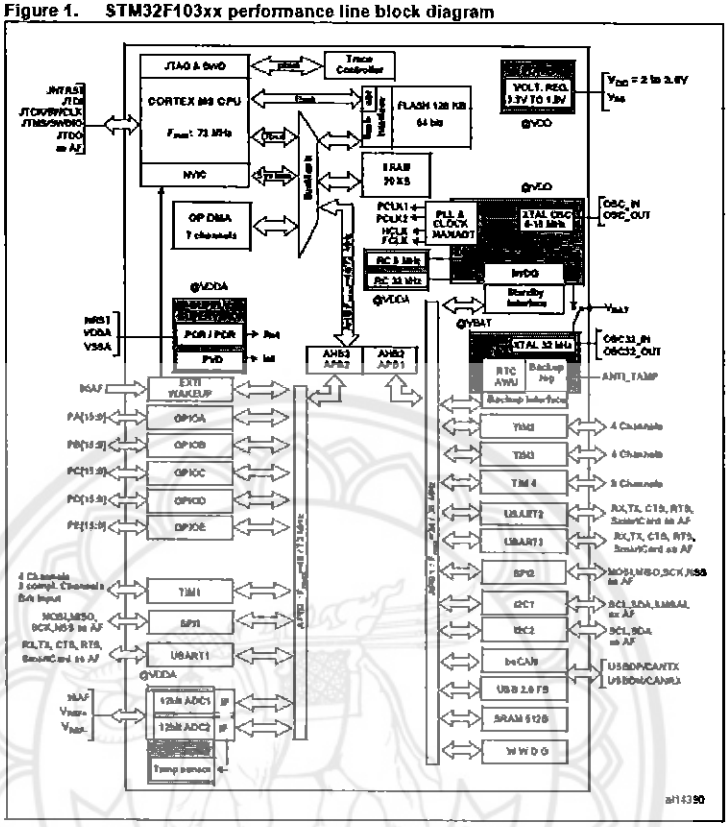


Figure 1. STM32F103xx performance line block diagram



1. T_A = -40 °C to +105 °C (junction temperature up to 125 °C).
2. AF = alternate function on I/O port pin.

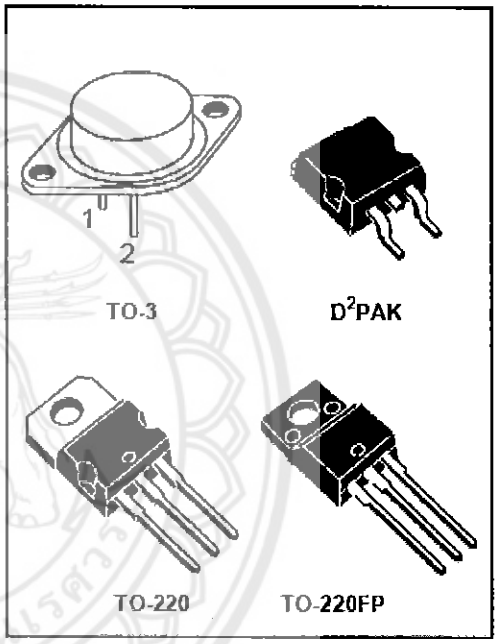
ภาคผนวก ง

รายละเอียดไอซี L7805

L7800 SERIES

POSITIVE VOLTAGE REGULATORS

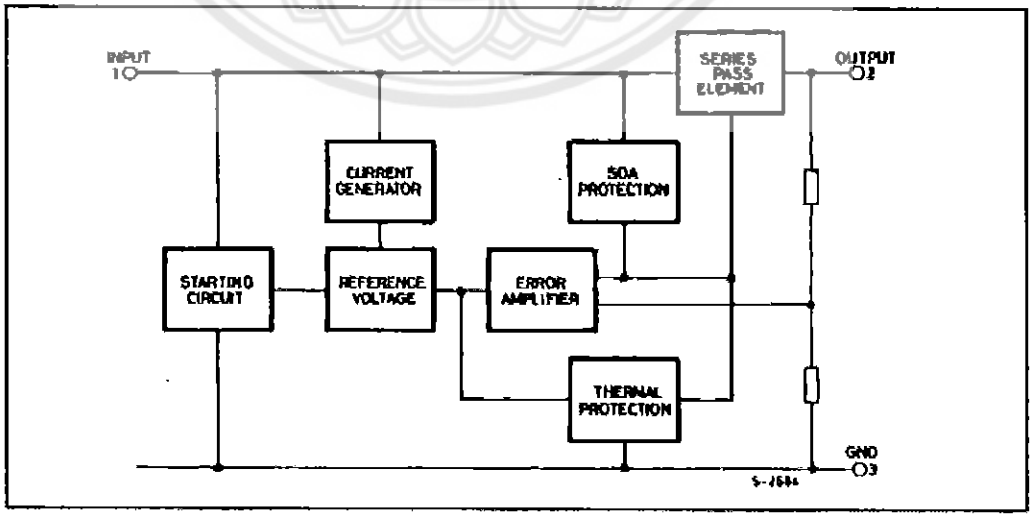
- OUTPUT CURRENT UP TO 1.5 A
- OUTPUT VOLTAGES OF 5; 5.2; 6; 8; 8.5; 9; 12; 15; 18; 24V
- THERMAL OVERLOAD PROTECTION
- SHORT CIRCUIT PROTECTION
- OUTPUT TRANSITION SOA PROTECTION



DESCRIPTION

The L7800 series of three-terminal positive regulators is available in TO-220 TO-220FP TO-3 and D^2PAK packages and several fixed output voltages, making it useful in a wide range of applications.These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.

BLOCK DIAGRAM



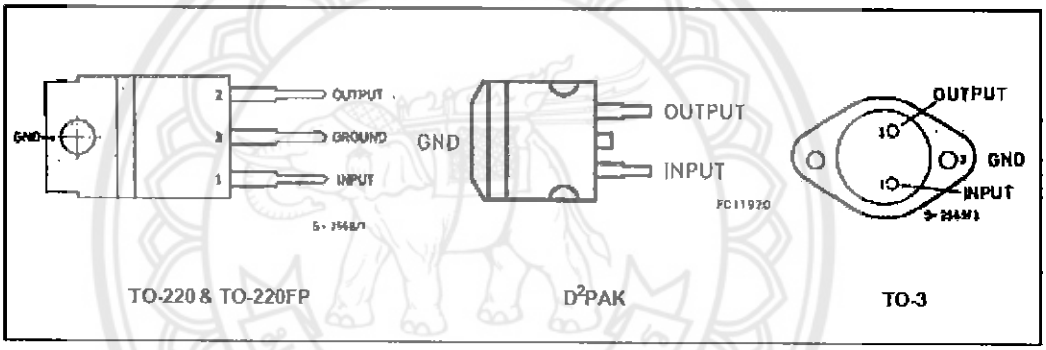
ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_I	DC Input Voltage (for V_O = 5 to 18V) (for V_O = 20, 24V)	35 40	V V
I_O	Output Current	Internally limited	
P_{tot}	Power Dissipation	Internally limited	
T_{op}	Operating Junction Temperature Range (for L7800) (for L7800C)	-55 to 150 0 to 150	°C °C
T_{stg}	Storage Temperature Range	-65 to 150	°C

THERMAL DATA

Symbol	Parameter		D^2PAK	TO-220	TO-220FP	TO-3	Unit
$R_{thj\text{-}case}$	Thermal Resistance Junction-case	Max	3	3	5	4	°C/W
$R_{thj\text{-}amb}$	Thermal Resistance Junction-ambient	Max	62.5	50	60	35	°C/W

CONNECTION DIAGRAM AND ORDERING NUMBERS (top view)

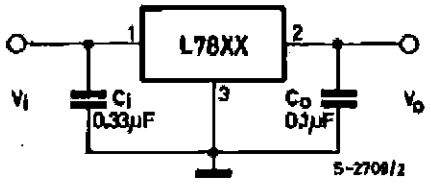


TO-220 & TO-220FP | D^2PAK | TO-3

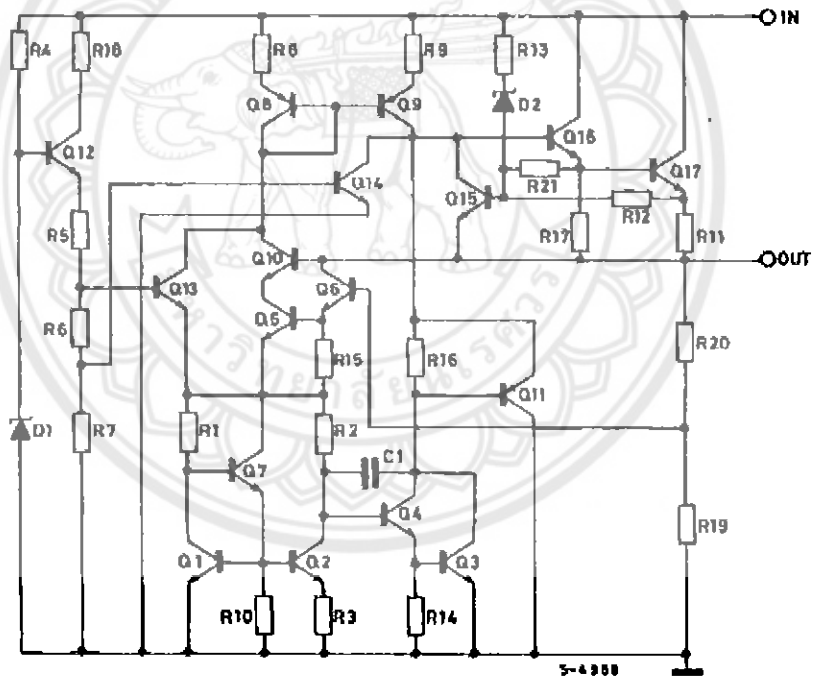
Type	TO-220	D^2PAK (*)	TO-220FP	TO-3	Output Voltage
L7805				L7805T	5V
L7805C	L7805CV	L7805CD2T	L7805CP	L7805CT	5V
L7852C	L7852CV	L7852CD2T	L7852CP	L7852CT	5.2V
L7806				L7806T	6V
L7806C	L7806CV	L7806CD2T	L7806CP	L7806CT	6V
L7808				L7808T	8V
L7808C	L7808CV	L7808CD2T	L7808CP	L7808CT	8V
L7885C	L7885CV	L7885CD2T	L7885CP	L7885CT	8.5V
L7809C	L7809CV	L7809CD2T	L7809CP	L7809CT	9V
L7812				L7812T	12V
L7812C	L7812CV	L7812CD2T	L7812CP	L7812CT	12V
L7815				L7815T	15V
L7815C	L7815CV	L7815CD2T	L7815CP	L7815CT	15V
L7818				L7818T	18V
L7818C	L7818CV	L7818CD2T	L7818CP	L7818CT	18V
L7820				L7820T	20V
L7820C	L7820CV	L7820CD2T	L7820CP	L7820CT	20V
L7824				L7824T	24V
L7824C	L7824CV	L7824CD2T	L7824CP	L7824CT	24V

(*) AVAILABLE IN TAPE AND REEL WITH "-TR" SUFFIX

APPLICATION CIRCUIT



SCHEMATIC DIAGRAM



TEST CIRCUITS

Figure 1 : DC Parameter

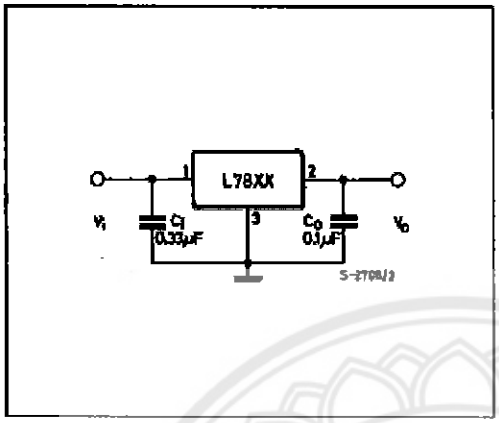


Figure 2 : Load Regulation.

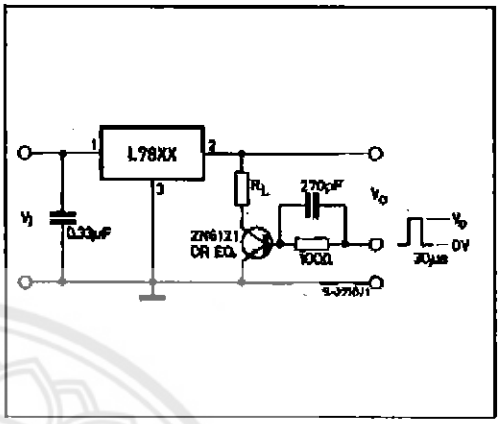
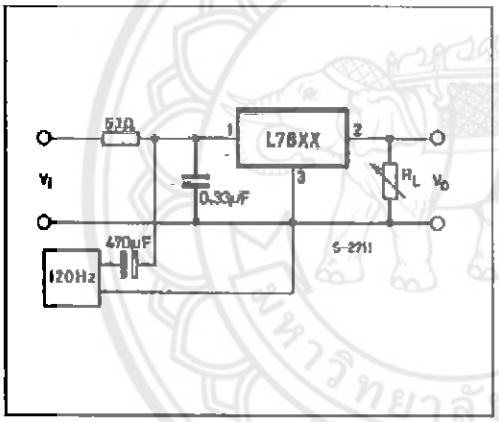


Figure 3 : Ripple Rejection.



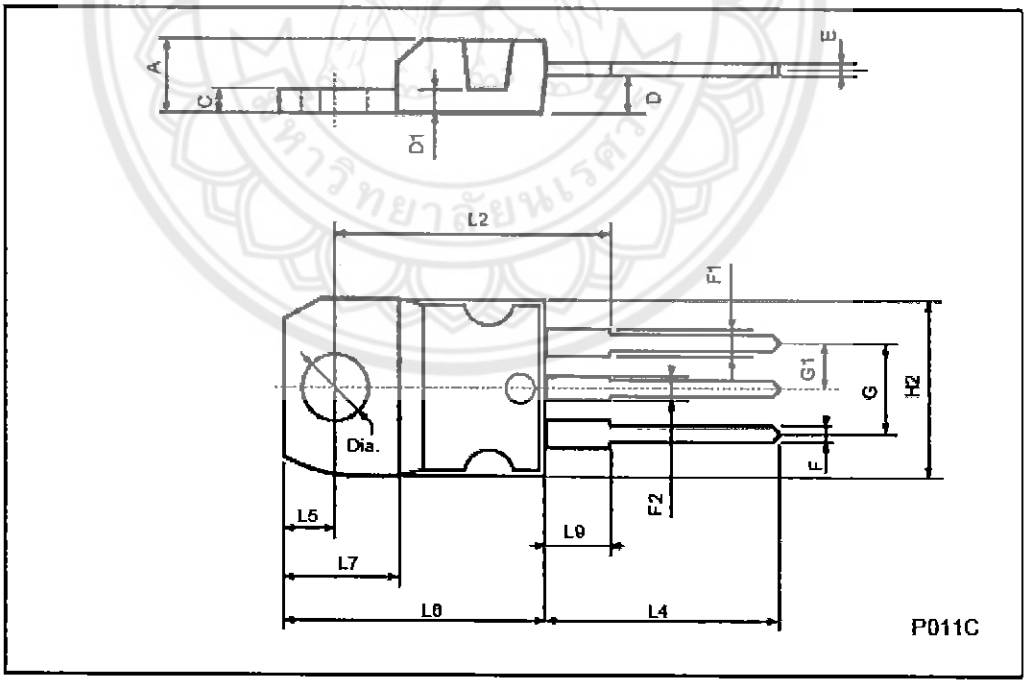
ELECTRICAL CHARACTERISTICS FOR L7805 (refer to the test circuits, T_j = -55 to 150 °C, V_i = 10V, I_o = 500 mA, C_i = 0.33 µF, C_o = 0.1 µF unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_o	Output Voltage	T_j = 25 °C	4.8	5	5.2	V
V_o	Output Voltage	I_o = 5 mA to 1 A $P_o \leq$ 15 W V_i = 8 to 20 V	4.65	5	5.35	V
ΔV_o*	Line Regulation	V_i = 7 to 25 V T_j = 25 °C V_i = 8 to 12 V T_j = 25 °C		3 1	50 25	mV mV
ΔV_o*	Load Regulation	I_o = 5 to 1500 mA T_j = 25 °C I_o = 250 to 750 mA T_j = 25 °C			100 25	mV mV
I_d	Quiescent Current	T_j = 25 °C			6	mA
ΔI_d	Quiescent Current Change	I_o = 5 to 1000 mA			0.5	mA
ΔI_d	Quiescent Current Change	V_i = 8 to 25 V			0.8	mA
$\frac{\Delta V_o}{\Delta T}$	Output Voltage Drift	I_o = 5 mA		0.6		mV/°C
eN	Output Noise Voltage	B = 10Hz to 100KHz T_j = 25 °C			40	µV/V_o
SVR	Supply Voltage Rejection	V_i = 8 to 18 V f = 120Hz	68			dB
V_d	Dropout Voltage	I_o = 1 A T_j = 25 °C		2	2.5	V
R_o	Output Resistance	f = 1 KHz		17		mΩ
I_{sc}	Short Circuit Current	V_i = 35 V T_j = 25 °C		0.75	1.2	A
I_{scp}	Short Circuit Peak Current	T_j = 25 °C	1.3	2.2	3.3	A

* Load and line regulation are specified at constant junction temperature. Changes in V_o due to heating effects must be taken into account separately. Pulse testing with low duty cycle is used.

TO-220 MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A	4.40		4.60	0.173		0.181
C	1.23		1.32	0.048		0.051
D	2.40		2.72	0.094		0.107
D1		1.27			0.050	
E	0.49		0.70	0.019		0.027
F	0.61		0.88	0.024		0.034
F1	1.14		1.70	0.044		0.067
F2	1.14		1.70	0.044		0.067
G	4.95		5.15	0.194		0.203
G1	2.4		2.7	0.094		0.106
H2	10.0		10.40	0.393		0.409
L2		16.4			0.645	
L4	13.0		14.0	0.511		0.551
L5	2.65		2.95	0.104		0.116
L6	15.25		15.75	0.600		0.620
L7	6.2		6.6	0.244		0.260
L9	3.5		3.93	0.137		0.154
DIA.	3.75		3.85	0.147		0.151



ภาคผนวก จ

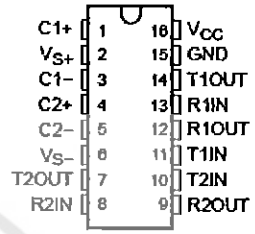
รายละเอียดไอซี MAX232

MAX232, MAX232I
DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L – FEBRUARY 1989 – REVISED MARCH 2004

- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Operates From a Single 5-V Power Supply With 1.0-μF Charge-Pump Capacitors
- Operates Up To 120 kbit/s
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- ESD Protection Exceeds JESD 22
  – 2000-V Human-Body Model (A114-A)
- Upgrade With Improved ESD (15-kV HBM) and 0.1-μF Charge-Pump Capacitors is Available With the MAX202
- Applications
  – TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube of 25	MAX232N	MAX232N
	SOIC (D)	Tube of 40	MAX232D	MAX232
		Reel of 2500	MAX232DR	
	SOIC (DW)	Tube of 40	MAX232DW	MAX232
		Reel of 2000	MAX232DWR	
	SOP (NS)	Reel of 2000	MAX232NSR	MAX232
–40°C to 85°C	PDIP (N)	Tube of 25	MAX232IN	MAX232IN
	SOIC (D)	Tube of 40	MAX232ID	MAX232I
		Reel of 2500	MAX232IDR	
	SOIC (DW)	Tube of 40	MAX232IDW	MAX232I
		Reel of 2000	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC is a trademark of Texas Instruments.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 2004, Texas Instruments Incorporated

Function Tables

EACH DRIVER

INPUT TIN	OUTPUT TOUT
L	H
H	L

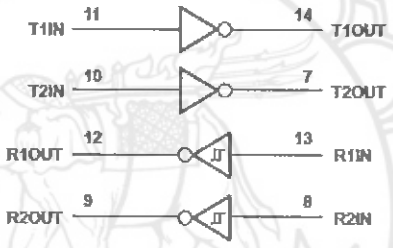
H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)



ประวัติผู้เขียนโครงงาน

ชื่อ นายเหมวิจักษณ์ บุญมาก

ภูมิลำเนา 139 หมู่ 6 ต.นางั่ว อ.เมือง จ.เพชรบูรณ์ 67000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนเพชรพิทยาคม จังหวัดเพชรบูรณ์
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: **pai.cpenu@gmail.com**

ชื่อ นายวิชิต ธรรมพิทักษ์

ภูมิลำเนา 22/2 หมู่ 3 ต.บ้านยาง อ.วัดโบสถ์ จ.พิษณุโลก 65160

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนเตรียมอุดมศึกษาภาคเหนือ จังหวัดพิษณุโลก
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: **wichit_a_007@hotmail.com**