



ระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถยนต์ประหยัดพลังงาน
DATA LOGGER AND ONLINE MONITORING SYSTEM FOR ECO CAR



นายอภิพงศ์ ปิงยศ รหัส 50362979

นายชวัน ประทีปศิริ รหัส 50364997

นายชาติชาย ไชยพานิช รหัส 50365024

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 19/๗.ค. 2555
เลขทะเบียน..... 157๖6๙๑๖
เลขเรียกหนังสือ..... นร.
มหาวิทยาลัยนเรศวร ๑25๘ ๘

2553
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2553



ใบรับรองปริญญาโท

ชื่อหัวข้อโครงการ ระบบจัดเก็บข้อมูลและเฝ้าสังเกตออนไลน์สำหรับรถยนต์
ประหยัดพลังงาน

ผู้ดำเนินโครงการ นายอภิพงศ์ ปิงยศ รหัส 50362979
นายชวณ ประทีปศิริ รหัส 50364997
นายชาติชาย ไชยพานิช รหัส 50365024

ที่ปรึกษาโครงการ อ. เสรมฐา ตั้งคำวานิช

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2553

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธนบุรี อนุมัติให้ปริญญาโทฉบับนี้ เป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

..... เสรมฐา ตั้งคำวานิช ที่ปรึกษาโครงการ
(อ. เสรมฐา ตั้งคำวานิช)

..... สุรเดช จิตประไพกุลศาล กรรมการ
(ดร. สุรเดช จิตประไพกุลศาล)

..... พงศ์พันธ์ กิจสนาโชชิน กรรมการ
(ดร. พงศ์พันธ์ กิจสนาโชชิน)

ชื่อหัวข้อโครงการ ระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถยนต์
ประหยัคพลังงาน

ผู้ดำเนินโครงการ นายอภิพงษ์ ปิงยศ รหัส 50362979
นายชวณ ประทีปศิริ รหัส 50364997
นายชาติชาย ไชยพานิช รหัส 50365024

ที่ปรึกษาโครงการ อ. เสริมฐา ตั้งคำวานิช

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2553

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอโครงการเกี่ยวกับการสร้างระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถยนต์ประหยัคพลังงาน จากสถานการณ์การใช้น้ำมันเชื้อเพลิงในรถยนต์ในปัจจุบัน ที่มากจนอาจเกิดภาวะขาดแคลนในอนาคต ทำให้ต้องคิดหาพลังงานทดแทนมาใช้ คณะผู้จัดทำโครงการจึงได้พัฒนาโครงการนี้ขึ้นมา เพื่อสนับสนุนการทำงานของรถประหยัคพลังงานต้นแบบ โดยระบบที่พัฒนาขึ้นจะเป็นพื้นฐานสำคัญสำหรับการพัฒนาประสิทธิภาพของรถประหยัคพลังงานให้ดีขึ้นในอนาคต ซึ่งระบบสามารถจัดเก็บแรงดันไฟฟ้า และความเร็วรถโดยใช้ไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีประสิทธิภาพสูงมาควบคุมและประมวลผลการทำงานของระบบทั้งหมด ในส่วนของตัวจัดเก็บข้อมูลนั้น ได้ใช้ SD card เป็นตัวเก็บข้อมูล พร้อมทั้งยังมีระบบรับส่งข้อมูลทางไกลที่สามารถส่งข้อมูลไร้สายมาแสดงผลยังคอมพิวเตอร์ในขณะที่รถกำลังวิ่งได้อีกด้วย

Project title Data Logger and Online Monitoring System for Eco Car

Name Mr. Apipong Pingyod ID. 50362979
Mr. Chuan Prateepkeeree ID. 50364997
Mr. Chatchai Chaipanit ID. 50365024

Project advisor Mr. Settha Thangkawanit

Major Computer Engineering

Department Electrical and Computer Engineering

Academic year 2010

Abstract

In this senior project, an online data collection and monitoring system was built for an ecology (ECO) car. At the current fuel consumption rate, we will soon face with the fuel shortage situation. Therefore, there have been a lot of researches on alternative energy. To support the development of the ECO car, in this project, we built a system capable of gathering voltage and speed of the ECO car. At the heart of our system, a high performance AVR microcontroller was used to control and process all the required tasks. In addition, our system can store the data in a secure digital (SD) card and transmit data wirelessly to the monitoring computer while the vehicle is in motion.

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงไปได้ด้วยความกรุณาอย่างยิ่งจาก อาจารย์เศรษฐา ตั้งคำวานิช ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการ ที่ได้ชี้แนะแนวทางการทำโครงการจนสำเร็จลุล่วง คณะผู้ดำเนินโครงการขอกราบขอบพระคุณท่านเป็นอย่างสูง และขอระลึกถึงความกรุณาของท่านไว้ตลอดไป

ขอขอบคุณ ดร. สุรเดช จิตประไพกุลศาล และ ดร. พงศ์พันธ์ กิจสนาโยธิน ที่กรุณาสละเวลามาเป็นกรรมการสอบโครงการ พร้อมให้คำแนะนำที่เป็นประโยชน์

ขอขอบคุณคุณอาจารย์ทุกท่านที่เป็นผู้ประสิทธิ์ประสาทวิชาความรู้จนประสบความสำเร็จ

ขอขอบคุณภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ให้ยืมอุปกรณ์และเครื่องมือวัดมาใช้งาน จนทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณคณะวิศวกรรมศาสตร์ที่ได้สนับสนุนค่าใช้จ่ายส่วนหนึ่งในการทำโครงการนี้

นอกจากนี้ยังต้องขอขอบคุณวิทยาลัยพลังงานทดแทน ที่ได้จุติประกายความคิดให้แก่คณะผู้จัดทำในการทำโครงการนี้ รวมทั้งยังเป็นທີ່ปรึกษาที่ติดต่อการทำโครงการนี้อีกด้วย

เหนือสิ่งอื่นใด คณะผู้ดำเนินโครงการขอกราบขอบพระคุณบุพการีของคณะผู้จัดทำ ที่มอบความรักความเมตตา สติปัญญา รวมทั้งเป็นผู้ให้ทุกสิ่งทุกอย่างตั้งแต่วัยเยาว์จวบจนถึงปัจจุบัน คอยเป็นกำลังใจให้ฟันฝ่าอุปสรรคมาจนถึงทุกวันนี้

นายอภิพงศ์ ปิงยศ

นายชวณ ประทีปศิริ

นายชาติชาย ไชยพานิช

สารบัญ

หน้า

ใบรับรองปริญญาโท.....	ก
บทคัดย่อ	ข
ABSTRACT	ก
กิตติกรรมประกาศ	ง
สารบัญ.....	จ
สารบัญตาราง.....	ช
สารบัญรูป.....	ซ
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของ โครงการงาน	1
1.2 วัตถุประสงค์ของโครงการงาน	1
1.3 ขอบเขตของโครงการงาน	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ขั้นตอนและแผนการดำเนินงาน	3
1.6 ประโยชน์ที่คาดว่าจะได้รับจากโครงการงาน	3
1.7 งบประมาณ.....	4
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	5
2.1 การส่งข้อมูลแบบ ANALOG และ DIGITAL	5
2.2 การแปลงสัญญาณที่เกี่ยวข้องกับการสื่อสาร (MODULATION/DEMODULATION)	9
2.3 ไมโครคอนโทรลเลอร์ AVR	12
2.4 ไอซี MAX232 ,L232	16
2.5 วงจรแปลงไฟและจ่ายไฟ.....	17
2.6 การสื่อสารข้อมูลแบบอนุกรม	18
2.7 การใช้งานพอร์ตอนุกรม RS232 [7].....	21
2.8 วงจรแบ่งแรงดันไฟฟ้า [8]	24
2.9 ระบบจับเก็บข้อมูล (DATA LOGGER) [9].....	27

บทที่ 3 ออกแบบ และพัฒนาระบบ.....	32
3.1 การออกแบบส่วนอินพุต.....	33
3.2 ส่วนประมวลผล	34
3.3 การออกแบบส่วนเอาต์พุต	39
3.4 การออกแบบส่วนติดต่อกับผู้ใช้.....	44
บทที่ 4 ผลการทดสอบ.....	46
4.1 ทดสอบการอ่านข้อมูลเข้า SERIAL PORT	46
4.2 การทดสอบการอ่านค่าความเร็วจากคีย์บอร์ด	49
4.3 การทดสอบการบันทึกข้อมูลลงสู่ SD CARD	50
4.4 การทดสอบการรับส่งข้อมูลไร้สาย.....	54
4.5 การทดสอบส่วนติดต่อกับผู้ใช้ หรือ GUI	57
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	58
5.1 สรุปผลการดำเนิน โครงการ.....	58
5.2 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ	60
5.3 แนวทางในการพัฒนาต่อไป.....	63
เอกสารอ้างอิง	64
ภาคผนวก ก รายละเอียดของไมโครคอนโทรลเลอร์ ATMEGA32	65
ภาคผนวก ข รายละเอียดของ SD CARD MODULE.....	72
ภาคผนวก ค ลายวงจรต่างๆ ที่ใช้ในโครงการ.....	76
ประวัติผู้ดำเนินโครงการ	81

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ขาของคอนกรีตเดือร่อนกรรม DB9.....	22
ตารางที่ 4.1 ผลการทดสอบความคลาดเคลื่อนของข้อมูล.....	46
ตารางที่ 5.1 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ด้านฮาร์ดแวร์.....	60
ตารางที่ 5.2 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ด้านซอฟต์แวร์.....	61
ตารางที่ 5.3 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ในด้านอื่นๆ.....	62



สารบัญรูป

หน้า

รูปที่ 2.1 การแปลงสัญญาณแบบต่างๆ.....	6
รูปที่ 2.2 Analog - to - Digital Encoding	7
รูปที่ 2.3 Analog Signal to PAM Signal	8
รูปที่ 2.4 Quantized PAM Signal	8
รูปที่ 2.5 Quantizing Using Sign and Magnitude.....	9
รูปที่ 2.6 PCM.....	9
รูปที่ 2.7 From Analog to PCM	9
รูปที่ 2.8 การ Modulate ของสัญญาณแบบ FM.....	12
รูปที่ 2.9 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ ATmega32.....	14
รูปที่ 2.10 ขาพอร์ต ATmega32 ตัวถังแบบ PDIP	15
รูปที่ 2.11 ไอซี MAX232	17
รูปที่ 2.12 วงจรแปลงไฟจาก AC/DC 9-12V เป็น DC 5 V	17
รูปที่ 2.13 แสดงการส่งข้อมูลแบบขนานและแบบอนุกรม	18
รูปที่ 2.14 แสดงการส่งข้อมูลแบบอนุกรมด้วยความเร็ว 9600 บิตต่อวินาที.....	19
รูปที่ 2.16 การเชื่อมต่อระหว่างคอมพิวเตอร์และไมโครคอนโทรลเลอร์ผ่านพอร์ต RS232.....	21
รูปที่ 2.17 พอร์ตอนุกรม DB9 ทั้งตัวผู้และตัวเมีย	21
รูปที่ 2.18 DB9 ตัวผู้ เมื่อนมองจากด้านหลัง.....	22
รูปที่ 2.19 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9.....	23
รูปที่ 2.20 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL.....	24
รูปที่ 2.21 วงจรแบ่งแรงดันที่ไม่มีโหลดและวงจรแบ่งแรงดันที่มีโหลด.....	25
รูปที่ 2.22 วงจรการแบ่งแรงดันที่มีโหลด.....	25
รูปที่ 2.23 ระบบจัดเก็บข้อมูล (Data Logger).....	27

รูปที่ 2.24 โครงสร้างของระบบจัดเก็บข้อมูล (Data Logger).....	28
รูปที่ 2.25 อุปกรณ์ที่สามารถเก็บข้อมูลได้.....	29
รูปที่ 2.26 ความละเอียด A/D ของ Data Logger.....	30
รูปที่ 2.27 ความถี่ในการเก็บข้อมูล.....	30
รูปที่ 3.1 แผนผังระบบจัดเก็บข้อมูลสำหรับรถประชิดพลังงาน.....	32
รูปที่ 3.2 กระบวนการวัดแรงดันไฟฟ้า.....	33
รูปที่ 3.3 วงจรแบ่งแรงดันไฟฟ้า.....	34
รูปที่ 3.4 ไอซีแปลงไฟและจ่ายไฟทั้ง 3 เบอร์.....	35
รูปที่ 3.5 วงจรการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆ.....	36
รูปที่ 3.6 บอร์ดไมโครคอนโทรลเลอร์ ATmega32.....	37
รูปที่ 3.7 Flowchart การทำงานของโปรแกรม.....	38
รูปที่ 3.8 แสดงอุปกรณ์เชื่อมต่อหน่วยความจำ SD Card (SD Module).....	39
รูปที่ 3.9 หน่วยความจำ SD Card.....	40
รูปที่ 3.10 ขาของหน่วยความจำ SD Card.....	40
รูปที่ 3.11 ไฟล์ที่เขียนลงใน SD Card.....	41
รูปที่ 3.12 ข้อมูลที่เขียนลงใน SD Card.....	41
รูปที่ 3.13 อุปกรณ์และขั้นตอนการทำงานของ Futaba Transfer Module.....	42
รูปที่ 3.14 Futaba Transfer Module.....	42
รูปที่ 3.15 อุปกรณ์และการทำงานของ Futaba Receiver Module.....	43
รูปที่ 3.16 Futaba Receiver Module.....	43
รูปที่ 3.17 โปรแกรมติดต่อกับผู้ใช้.....	44
รูปที่ 4.1 แสดงการเชื่อมต่อวงจรต่างๆ เข้าด้วยกัน.....	47
รูปที่ 4.2 การตั้งค่า HyperTerminal.....	47
รูปที่ 4.3 แสดงค่าที่อ่านได้จากไมโครคอนโทรลเลอร์.....	48
รูปที่ 4.4 แบบจำลองการวัดความเร็วดีซีมอเตอร์.....	49

รูปที่ 4.5 แสดงค่าเมื่อมอเตอร์ไม่ได้หมุน.....	49
รูปที่ 4.6 แสดงค่ามอเตอร์ที่กำลังหมุน.....	50
รูปที่ 4.7 แสดงสถานะการเชื่อมต่อของ SD Card กับ SD card module.....	51
รูปที่ 4.8 แสดงข้อมูลภายใน SD Card.....	52
รูปที่ 4.9 ขนาดของไฟล์ที่เวลา 1 นาที.....	53
รูปที่ 4.6 ผลการเปรียบเทียบระหว่าง RS232 และการรับส่งทางไกล.....	55
รูปที่ 4.7 แสดงผลการทดสอบกรณีที่มีปริมาณข้อมูลมากและน้อย.....	56
รูปที่ 4.8 ผลการทดสอบส่วนติดต่อกับผู้ใช้.....	57



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ในภาวะโลกปัจจุบันทรัพยากรของโลกมีจำนวนจำกัดและกำลังลดลงทุกวัน ตัวอย่างเช่น น้ำมัน ที่มีอัตราการใช้พุ่งสูงขึ้นในทุกๆปี แต่กลับมีอัตราการผลิตน้ำมันน้อยลง นอกจากนี้ทรัพยากรด้านอื่น ๆ เช่น ก๊าซธรรมชาติ แร่ธาตุ ถ่านหิน ก็มีอัตราการใช้ที่สูงขึ้นในทุกๆปีเช่นเดียวกัน ซึ่งในที่สุดแหล่งพลังงานต่าง ๆ ในโลกก็จะเหลือน้อยลง จนทำให้เกิดปัญหาความขาดแคลนตามมา

ในอดีตรถยนต์ต้องใช้น้ำมันในการขับเคลื่อนทำให้สิ้นเปลืองพลังงานน้ำมันเป็นจำนวนมากและยังก่อให้เกิดมลภาวะเป็นพิษทางอากาศอีกด้วย แต่ในปัจจุบันได้มีการคิดค้นรถประหยัดพลังงาน (ECO Car) ขึ้นหลายชนิด เช่น รถพลังงานไฟฟ้า รถพลังงานแสงอาทิตย์ รถที่ใช้พลังงานจากเซลล์เชื้อเพลิง (Fuel Cells) เป็นต้น ซึ่งรถประหยัดพลังงานนั้นล้วนแต่แปรรูปพลังงานออกมาเป็นพลังงานไฟฟ้าในการขับเคลื่อน ทำให้ไม่ต้องใช้น้ำมัน อีกทั้งยังไม่มีมลภาวะเกิดขึ้น หากแต่รถประหยัดพลังงานนั้นยังไม่มีระบบที่สามารถจัดเก็บข้อมูลของรถขณะทำงานได้ เช่น แรงดันไฟฟ้า กระแสไฟฟ้า และความเร็วของตัวรถ เป็นต้น

จากปัญหาที่กล่าวมา ผู้จัดทำจึงได้ทำการสร้างระบบจัดเก็บข้อมูลขณะที่รถทำงาน เพื่อให้สามารถนำข้อมูลของรถไปพัฒนาประสิทธิภาพของตัวรถต่อไปได้ และยังทำให้เกิดการใช้พลังงานที่ผลิตได้อย่างเต็มประสิทธิภาพอีกด้วย

1.2 วัตถุประสงค์ของโครงการ

- 1) เพื่อสร้างอุปกรณ์สำหรับบันทึกข้อมูลและสามารถติดตามผลสถานภาพระยะไกล สำหรับรถยนต์ประหยัดพลังงานได้

1.3 ขอบเขตของโครงการงาน

1) สร้างระบบจัดเก็บข้อมูล ซึ่งสามารถเก็บข้อมูลได้ดังต่อไปนี้

- แรงดันไฟฟ้า
- ความเร็วล้อ

โดยแสดงผล โดยผ่านทั้งระบบไร้สายและบันทึกข้อมูลลง SD card ได้

2) สร้างโปรแกรมควบคุมระบบไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATmega32

โดยใช้ภาษา C ในการพัฒนา

3) สร้างส่วนเชื่อมต่อระหว่างระบบกับผู้ใช้งาน โดยใช้ Visual Basic

4) สามารถนำอุปกรณ์นี้ไปใช้ได้กับระบบจำลองของรถประหยัคพลังงานเท่านั้น

1.4 ขั้นตอนการดำเนินโครงการงาน

1) ศึกษาและรวบรวมข้อมูลของวงจรต่างๆที่จำเป็น

- วงจร Microcontroller เบอร์ ATmega32
- ตัวจัดเก็บข้อมูลแบบ SD card (Data Logger)
- ตัวรับส่งข้อมูลแบบไร้สาย (Futaba)
- วงจรแบ่งแรงดันไฟฟ้า
- วงจรแปลงไฟและจ่ายไฟ

2) สร้างวงจร

- วงจร Microcontroller เบอร์ ATmega32
- ตัวจัดเก็บข้อมูลแบบ SD card (Data Logger)
- ตัวรับส่งข้อมูลแบบไร้สาย (Futaba)
- วงจรแบ่งแรงดันไฟฟ้า
- วงจรแปลงไฟและจ่ายไฟ

3) สร้างโปรแกรมควบคุมระบบการทำงาน

4) ทดสอบการทำงานของระบบ

- ทดสอบทีละส่วน
- ทดสอบกับแบบจำลองของระบบลดประหยัดพลังงาน

5) แก้ไขปรับปรุงการทำงานของระบบให้มีประสิทธิภาพ

6) สรุปผลการทำงานของระบบและจัดทำรูปเล่มรายงาน

1.5 ขั้นตอนและแผนการดำเนินงาน

รายละเอียด	ปี 2553							ปี 2554		
	มี.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.5.1 ศึกษาและรวบรวมข้อมูลของวงจรต่างๆ										
1.5.2 สร้างวงจร										
1.5.3 สร้างโปรแกรมควบคุมระบบการทำงาน										
1.5.4 ทดสอบการทำงานของระบบ										
1.5.5 แก้ไขปรับปรุงการทำงานของระบบให้มีประสิทธิภาพ										
1.5.6 สรุปผลการทำงานของระบบและจัดทำรูปเล่มรายงาน										

1.6 ประโยชน์ที่คาดว่าจะได้รับจากโครงการ

1. ได้อุปกรณ์สำหรับบันทึกข้อมูลและสามารถติดตามผลสถานภาพระยะไกลสำหรับรถยนต์ประหยัดพลังงานได้

1.7 งบประมาณ

- ค่าวัสดุอุปกรณ์อิเล็กทรอนิกส์ 2,000 บาท
 - ค่าจัดทำรูปเล่มรายงาน 1,000 บาท
- รวมเป็นเงินทั้งสิ้น 3,000 บาท

ทั้งนี้ขอด้วงเฉลี่ยทุกรายการ



บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

การเก็บข้อมูลรถประหยัคพลังงานโดยใช้ตัวเก็บข้อมูล (Data Logger) นั้น จำเป็นจะต้องมีความรู้หลายๆ ส่วนประกอบกัน ไม่ว่าจะเป็นส่วนของไมโครคอนโทรลเลอร์ (Microcontroller) วงจรแบ่งแรงดันไฟฟ้า (Voltage Dividers) การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล (Analog To Digital Converting) การสื่อสารข้อมูลแบบอนุกรม (Serial) ความรู้เกี่ยวกับตัวเก็บข้อมูล (Data Logger) รวมถึงการส่งข้อมูลไร้สายระบบ FM อีกด้วย

การเก็บข้อมูลของรถประหยัคพลังงาน เพื่อนำข้อมูลมาวิเคราะห์ จะต้องมีความรู้และทราบหลักการที่เกี่ยวข้องเพื่อนำมาใช้ประกอบในการสร้างชุดควบคุม ซึ่งหลักการของอุปกรณ์ติดตั้ง และทฤษฎีที่สำคัญทั้งหมด ได้อธิบายไว้ในบทนี้แล้ว

2.1 การส่งข้อมูลแบบ Analog และ Digital [1]

ขณะที่ส่งผ่านข้อมูลไปในสื่อที่จะเป็นรูปแบบของสัญญาณแบ่งออกเป็น 2 ชนิด คือ Analog และ Digital ก่อนที่จะส่งต้องมีการแปลงสัญญาณให้อยู่ในรูปของ 0 และ 1 หรือ Binary Form (ASCII) ซึ่งแบ่งออกได้ดังนี้

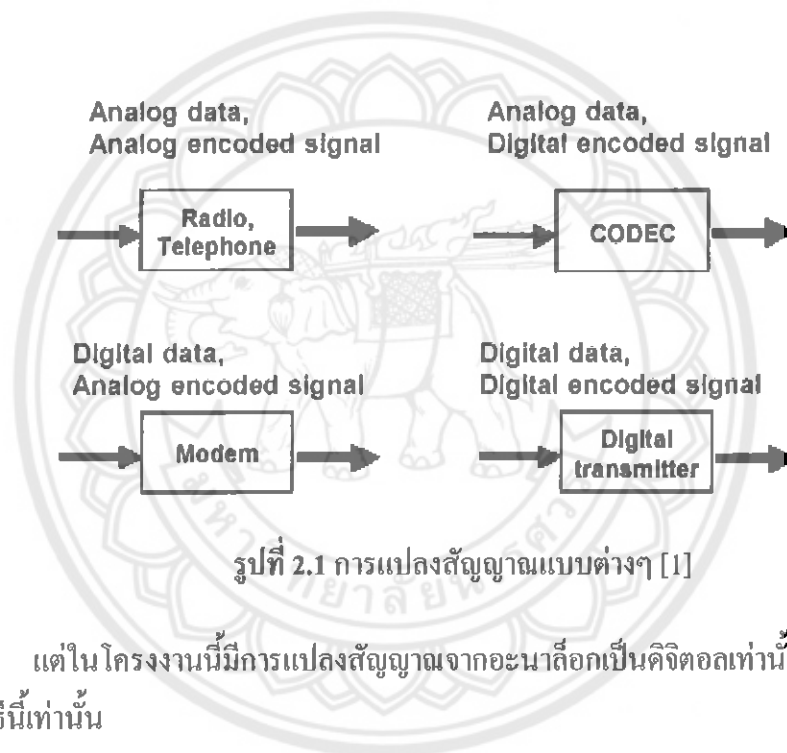
- การแปลงสัญญาณเป็น Digital เป็น Digital เรียกว่า Digital to Digital หรือ encoding digital data into a digital signal
- การแปลงสัญญาณ Analog ให้อยู่ในรูปของสัญญาณ Digital เพื่อคุณภาพสัญญาณที่ดีขึ้น เรียกว่า analog to digital หรือ encoding analog data into a digital signal
- การแปลงสัญญาณ digital เป็น analog เพื่อส่งข้อมูลลงสายโทรศัพท์ เช่น Modem เรียกว่า digital to analog หรือ encoding digital data into a analog signal
- การแปลงสัญญาณ analog ให้อยู่ในรูปของสัญญาณ analog เพราะความถี่ไม่เหมาะสมที่จะส่งไปหรือความถี่ต่ำไปเช่น สัญญาณวิทยุ(เรียกว่า analog to analog หรือ encoding analog data into a analog data)

สรุปการแปลงสัญญาณมี 4 กระบวนการ คือ

1. digital data , digital signals
2. analog data , digital signals
3. digital data , analog signals
4. analog data, analog signals

Modulation ใช้ในการแปลงสัญญาณ analog

Encoding ใช้ในการแปลงสัญญาณ digital

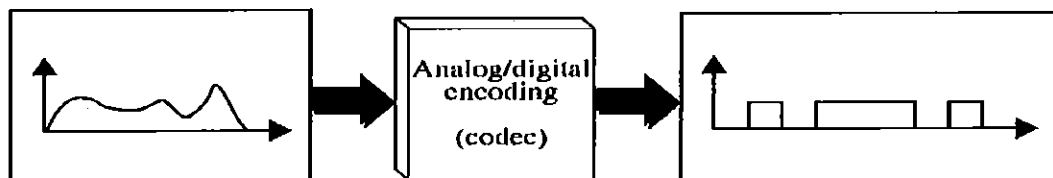


รูปที่ 2.1 การแปลงสัญญาณแบบต่างๆ [1]

แต่ในโครงการนี้มีการแปลงสัญญาณจากอะนาล็อกเป็นดิจิทัลเท่านั้น จึงขอกล่าวถึงเพียงวิธีนี้เท่านั้น

Analog to Digital

ในบางครั้งเราต้องแปลงสัญญาณ Analog เป็น Digital เช่น เสียงคนที่ต้องส่งในระยะทางไกลเราจะต้องแปลงเสียงเป็น Digital เสียก่อน เรียกว่า analog - to - digital เป็นการแปลงสัญญาณอนาล็อกไปเป็นสัญญาณดิจิทัล โดยผ่านวิธีการ Codec (Coder -Decoder)



รูปที่ 2.2 Analog - to - Digital Encoding [1]

ส่วนใหญ่จะใช้วิธี PCM ในการแปลงสัญญาณวิธี โดยทำการสุ่มช่วงเวลาใดเวลาหนึ่งทำให้ไม่ต่อเนื่องกัน โดยให้ขนาดของ Amplitude เท่าเดิม มีการ Quantizing ทำให้สัญญาณเป็น digital โดยสมบรูณ์คือการใส่ Voltage เข้าไปในแต่ละ levels ของ pulses ซึ่งขั้นตอนมีดังนี้

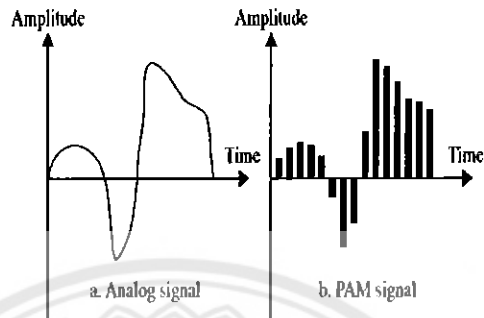
Sampling rate

เป็นปริมาณที่มักจะถูกใช้กับ Analog to Digital Conversion ซึ่ง Sampling rate เป็นจำนวนของเวลาต่อวินาทีที่สัญญาณ Analog ถูกเปลี่ยนเป็น Digital Code สำหรับคุณสมบัติของ Analog to Digital Conversion ค่าที่น้อยที่สุดของ Sampling rate จะมีค่าเป็นสองเท่าของความถี่ Analog ที่สูงที่สุด ซึ่ง Sampling rate ที่น้อยที่สุดนี้ มักจะเรียกเป็น Nyquist sampling rate

ถ้า Sampling frequency น้อยกว่าสองเท่าของความถี่ Analog ที่สูงที่สุดจะเกิด Aliasing error ขึ้น ซึ่งจะทำให้เกิดความผิดเพี้ยนที่ความถี่สูงของสัญญาณ เนื่องจากตัว Spectral ของสัญญาณที่ถูก Sampling ซึ่ง Aliasing error เป็นปรากฏการณ์ที่มีสาเหตุมาจากส่วนประกอบของความถี่ที่เกิดขึ้นจากสัญญาณ Sampling จะถูกสอดแทรกจากความถี่ของสัญญาณที่จะถูก Sampling เราสามารถหลีกเลี่ยง Aliasing error ได้โดยการใช้ Sampling frequency ที่น้อยกว่า 2 เท่าของค่าสูงสุดของ Analog Frequency

Pulse Amplitude Modulation (PAM)

ขั้นแรกของการ PCM จะต้องทำ PAM ก่อน คือการเอาข้อมูลที่เป็น analog มาทำการ Sampling

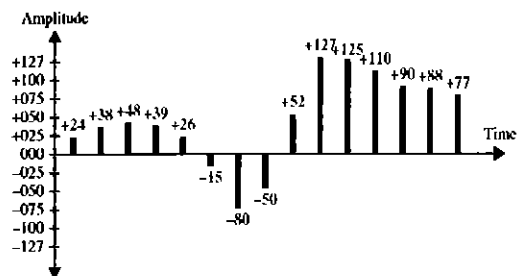


รูปที่ 2.3 Analog Signal to PAM Signal [1]

Pulse Code Modulation (PCM)

PCM จะแก้ไข pulses ที่สร้างโดย PAM ให้เป็น digital signal ที่สมบูรณ์ ซึ่งการทำให้เป็น digital โดยสมบูรณ์มี 4 ขั้นตอนคือ

- PAM
- ไล่ Voltage (Quantization)
- เข้ารหัสเป็น Binary (Binary Encoding)
- เปลี่ยน Binary เป็น digital แล้วส่งออกไป (Digital - to - digital Encoding)

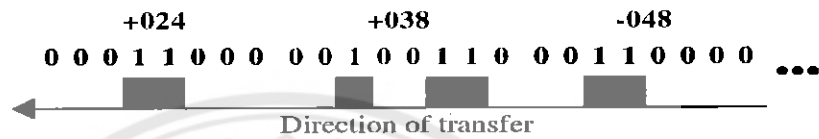


รูปที่ 2.4 Quantized PAM Signal [1]

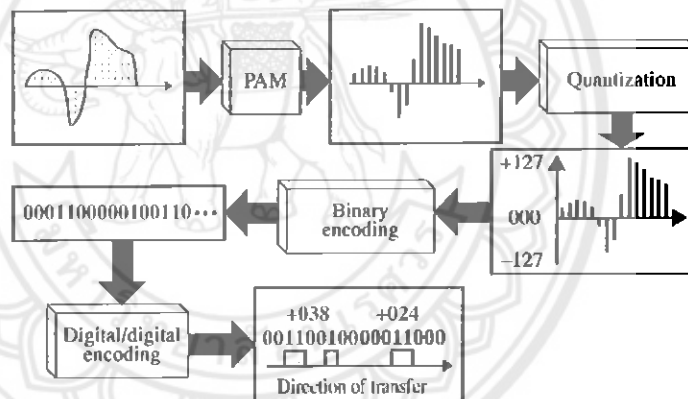
+024	00011000	-015	10001111	+125	01111011
+038	00100110	-080	11010000	+110	01101110
+048	00110000	-050	10110010	+090	01011010
+039	00100111	+052	00110110	+088	01011000
+026	00011010	+127	01111111	+077	01001101

Sign bit
+ is 0 - is 1

รูปที่ 2.5 Quantizing Using Sign and Magnitude [1]



รูปที่ 2.6 PCM [1]



รูปที่ 2.7 From Analog to PCM [1]

2.2 การแปลงสัญญาณที่เกี่ยวข้องกับการสื่อสาร (Modulation/Demodulation) [2]

เมื่อต้องการจะส่งสัญญาณเสียงหรือข้อมูลผ่านช่องทางการสื่อสาร จำเป็นอย่างยิ่งที่จะต้องอาศัยพลังงานไฟฟ้าช่วยพาสัญญาณเหล่านั้นเคลื่อนย้ายจากจุดหนึ่งไปยังอีกจุดหนึ่ง ขบวนการหรือขั้นตอนในการเพิ่มพลังงานไฟฟ้าดังกล่าวเราเรียกว่า “การมอดูเลต” (Modulation) หรือการกล้ำสัญญาณที่เป็นพลังงานไฟฟ้าซึ่งมีความถี่สูงและคงที่ รวมทั้งมีขนาด (Amplitude) สูงด้วยนั้น เราเรียกว่า “สัญญาณคลื่นพาหะ” (Signal Carrier) ดังนั้น จึงสรุปได้ว่าการแปลงสัญญาณหรือ

การมอดูเลท คือ เทคนิคการพาสัญญาณที่ต้องการส่งเดินทางไปยังจุดหมายปลายทางผ่านตัวกลาง การสื่อสาร สัญญาณที่เป็นตัวพาสัญญาณข้อมูล ไปเรียกว่า คลื่นพาหะ (Carrier wave) อุปกรณ์ สำหรับมอดูเลตสัญญาณ (Modulator) จะสร้างสัญญาณคลื่นพาหะ และรวมเข้ากับสัญญาณข้อมูล เพื่อให้สัญญาณมีความแรงพอที่จะส่งผ่านสื่อกลาง ไปยังอีกจุดหนึ่งที่อยู่ไกลออกไปได้ และเมื่อถึง ปลายทางก็จะมีอุปกรณ์ซึ่งทำหน้าที่แยกสัญญาณคลื่นพาหะออกให้เหลือเพียงสัญญาณข้อมูล เราเรียกวิธีการแยกสัญญาณนี้ว่า “การดีมอดูเลต” (Demodulation) ในการแปลงสัญญาณ Analog เพื่อส่งผ่านช่องสื่อสาร Analog นั้นมี 3 วิธีคือ

1. Amplitude Modulation (AM)
2. Frequency Modulation (FM)
3. Phase Modulation (PM)

แต่ในการทำโครงการนี้นั้นจะใช้เพียง Frequency Modulation ดังนั้นจึงขอกว่าเพียงแต่ Frequency Modulation เท่านั้น

Frequency Modulation (FM)

การส่งสัญญาณระบบเอฟเอ็มมีความทนทานต่อการรบกวนจากภายนอกได้ดีกว่าระบบ เอเอ็ม เช่น ในระหว่างที่เกิดพายุฝน สัญญาณแบบเอฟเอ็มจะถูกรบกวนน้อยมาก หรือไม่ถูกรบกวนเลย ดังนั้นเมื่อนำมาใช้ส่งข้อมูลอิเล็กทรอนิกส์ก็ทำให้โอกาสที่จะเกิดข้อผิดพลาดลดลง ไปด้วย วิธีการตรวจสอบและแก้ไขข้อมูลจึงไม่จำเป็นต้องมีความซับซ้อนมากนักเมื่อเทียบกับการ ใช้สัญญาณเอฟเอ็ม สมการของ FM wave คือ

$$S_{FM}(t) = A_0 \cos(2\pi f_0 t + 2\pi k_f \int e_m(t) dt) \quad (2.1)$$

ซึ่งเป็น nonlinear function ของ modulating wave $m(t)$ ดังนั้น Frequency Modulation จึงเป็น Nonlinear process โดยปกติแล้วจะไม่เหมือน Amplitude Modulation ดังนั้น Spectrum ของ FM wave จะไม่สัมพันธ์อย่างง่าย ๆ กับ Modulating wave ดังนั้นในการเริ่มต้นศึกษา Spectral properties ของ FM wave เราจะเริ่มต้นด้วย Single - tone modulation ก่อน Single-tone modulation พิจารณา Sinusoidal modulation wave กำหนดได้ว่า

$$m(t) = A_m \cos(\omega t + \theta) \quad (2.2)$$

ค่า Instantaneous frequency ของ resulting FM wave เท่ากับ

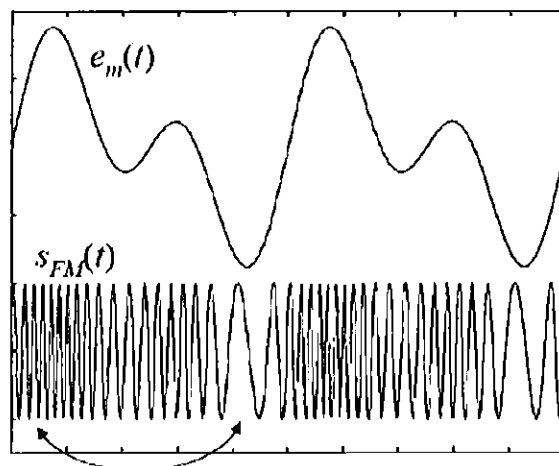
$$f_i(t) = f_c + K_f A_m \cos(\omega_m t + \theta) \quad (2.3)$$

$$= f_c + \Delta f \cos(\omega_m t + \theta) \quad (2.4)$$

เมื่อ

$$\Delta f = K_f A_m \quad (2.5)$$

ปริมาณที่เรียกว่า "Frequency deviation" จะแสดงถึงความห่างมากที่สุดของ Instantaneous frequency ของ FM wave จาก Carrier frequency (f_c) ค่า fundamental characteristic ของ FM wave คือ ค่า Frequency deviation(จะเป็นสัดส่วนกับค่า Amplitude ของ Modulating wave และเป็นอิสระ จาก Modulation frequency



รูปที่ 2.8 การ Modulate ของสัญญาณแบบ FM [2]

2.3 ไมโครคอนโทรลเลอร์ AVR [3]

ไมโครคอนโทรลเลอร์ AVR เป็นหนึ่งในไมโครคอนโทรลเลอร์ที่ได้รับความนิยมมากในต่างประเทศ และกำลังได้รับความนิยมเพิ่มขึ้นในประเทศไทย ด้วยความสามารถบวกกับราคาที่ไม่สูงนัก ผสมกับเครื่องมือในการพัฒนาโปรแกรมที่เรียกว่า AVR Studio ที่สามารถทำงานร่วมกับ WinAVR (C Compiler) ได้เป็นอย่างดี โดยที่ WinAVR และ AVR Studio เป็นซอฟต์แวร์ฟรี สามารถดาวน์โหลดมาใช้งานสำหรับการพัฒนาโปรแกรมใช้งานไมโครคอนโทรลเลอร์ AVR ด้วยภาษา C

ไมโครคอนโทรลเลอร์ AVR เป็นหนึ่งในไมโครคอนโทรลเลอร์ที่ผลิตโดยบริษัท ATMEL ซึ่งเป็นผู้นำทางด้านไมโครคอนโทรลเลอร์ตระกูล MCS-51 ไมโครคอนโทรลเลอร์ AVR จัดเป็นไมโครคอนโทรลเลอร์ตระกูลใหม่ที่มีสถาปัตยกรรมแบบ RISC คือหนึ่งคำสั่งทำงานใช้สัญญาณนาฬิกาเพียง 1 ลูก เป็นไมโครคอนโทรลเลอร์ที่มีประสิทธิภาพและความสามารถสูง แบ่งออกเป็นหลายอนุกรม ในแต่ละอนุกรมยังแบ่งออกเป็นหลายเบอร์ เพื่อรองรับความต้องการที่แตกต่างกันของผู้ใช้งาน ในขณะที่ยังคงประสิทธิภาพที่เท่ากัน

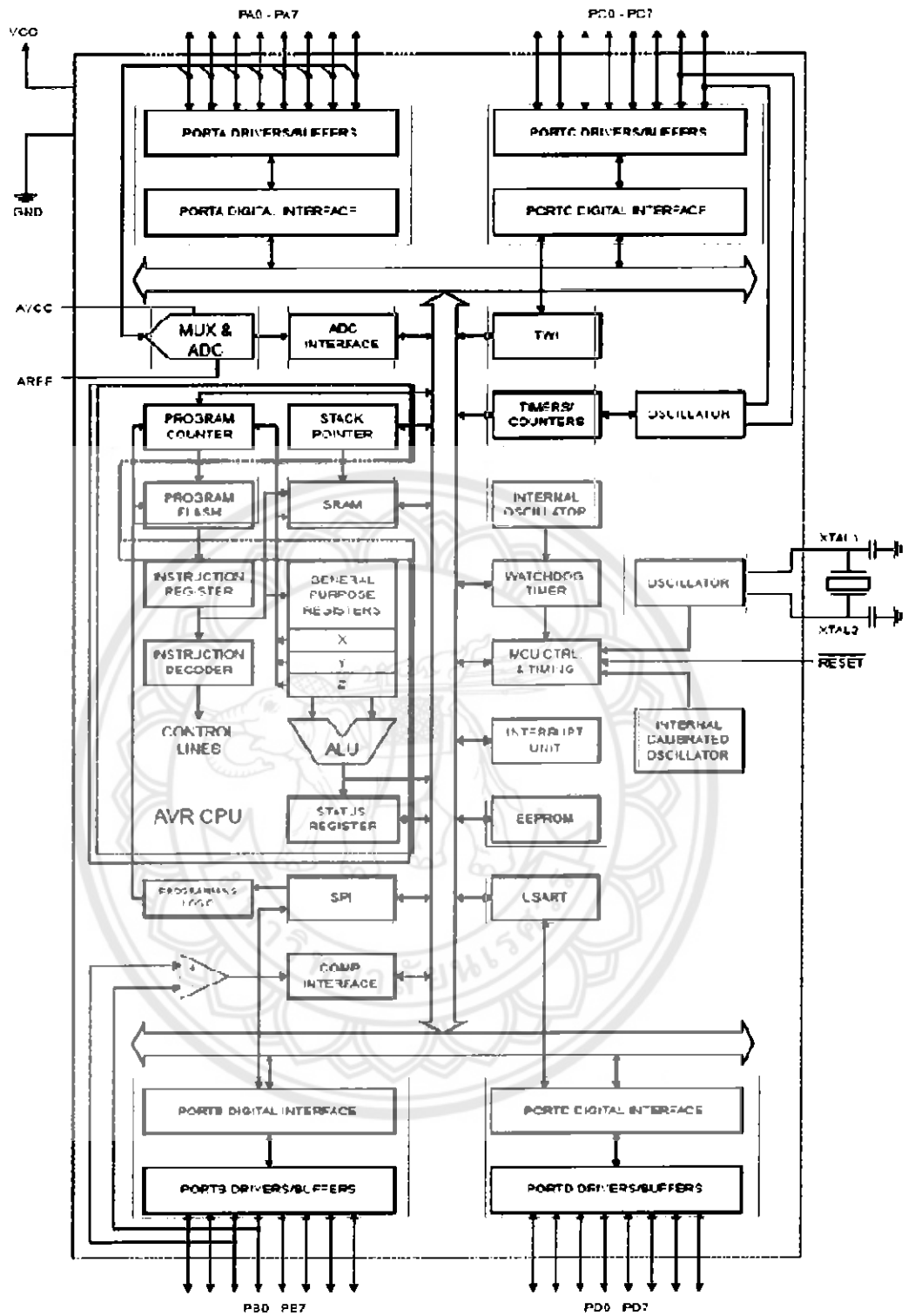
ในโครงการนี้ได้เลือกใช้ไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega32 ซึ่งมีคุณสมบัติที่สำคัญดังนี้ [4]

- สถาปัตยกรรมภายในแบบ Advanced RISC (Reduce Instruction Set Computer)
- มีคำสั่งควบคุมการทำงาน 131 คำสั่ง โดยมีความเร็วในการประมวลผล 1 คำสั่งต่อ 1 สัญญาณนาฬิกา (และมากถึง 16 MIPS เมื่อใช้ความถี่ 16 MHz)
- รีจิสเตอร์ใช้งานทั่วไปขนาด 8 bit จำนวน 32 ตัว
- หน่วยความจำ ROM แบบ Flash ขนาด 32 K Bytes ซึ่งเขียน/ลบได้ 10,000 ครั้ง

- หน่วยความจำ EEPROM ขนาด 1024 Bytes ซึ่งเขียน/ลบได้ 100,000 ครั้ง
- มีหน่วยความจำ SRAM ขนาด 2 K Bytes
- ไทเมอร์/คาน์เตอร์ทั้งแบบ 8 บิต และ 16 บิต พร้อมปริสเกลเลอร์
- มีระบบตรวจสอบความผิดปกติในการทำงานของซอฟต์แวร์(Watchdog Timer with Separate On-Chip Oscillator)
- โมดูลสร้างสัญญาณ PWM (Pulse Width Modulation) จำนวน 4 แชนแนล
- โมดูลแปลงสัญญาณอะนาล็อกเป็นดิจิทัล (ADC) ขนาด 10 บิต จำนวน 8 แชนแนล
- ใช้แรงดันไฟเลี้ยง 4.5 – 5.5 V
- ความเร็วในการทำงาน 0 – 16 MHz
- พอร์ตอินพุตเอาต์พุต จำนวน 40 ขา

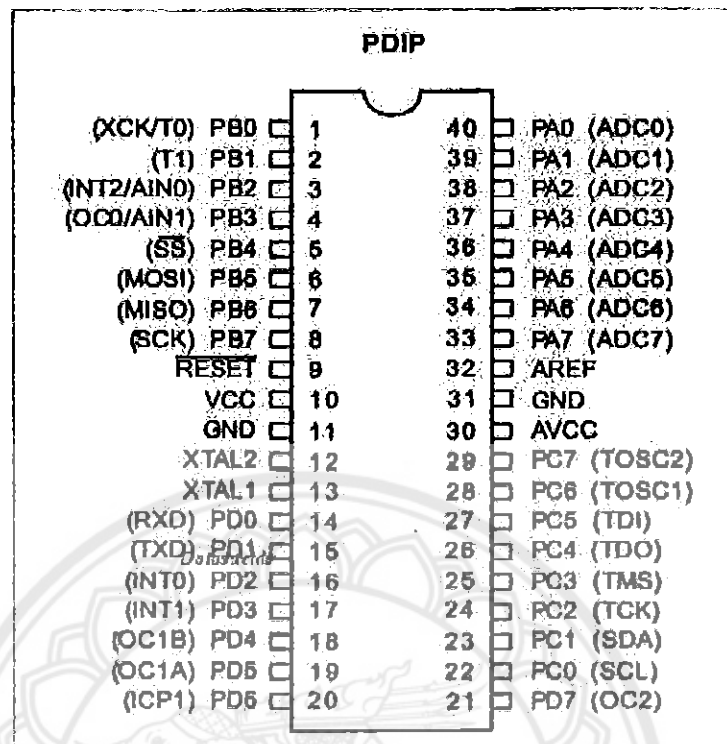
รายละเอียดและคุณสมบัติภายในของไมโครคอนโทรลเลอร์ ATmega32 แสดง
ดังไดอะแกรมดังรูปที่ 2.9





รูปที่ 2.9 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ ATmega32 [4]

ในส่วนของขาพอร์ตอินพุตเอาต์พุตของไมโครคอนโทรลเลอร์ ATmega32 จะมีจำนวน 40 ขา แบ่งเป็น พอร์ตอินพุตเอาต์พุตอิสระจำนวน 32 ขา ประกอบไปด้วย PA,PB,PC,PD ขนาด 8 บิต และขาพอร์ตที่เกี่ยวข้องกับสัญญาณอนาล็อกจำนวน 2 ขา คือ AREF และ AVCC รายละเอียดขาพอร์ตแสดงดังรูปที่ 2.10



รูปที่ 2.10 ขาพอร์ต ATmega32 ตัวถังแบบ PDIP [4]

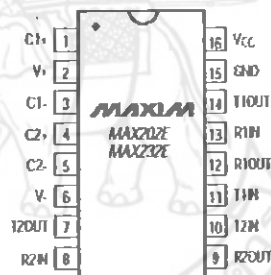
รายละเอียดในแต่ละขาพอร์ตมีดังนี้

- VCC ขาแรงดันไฟตรง
- GND ขากราวด์
- Port A (PA0-PA7) เป็นขาพอร์ตอินพุตเอาต์พุตดิจิทัล กำหนดการพูลอัพภายในขาพอร์ตได้ (Internal pull-up register) และสามารถกำหนดใช้งานเป็นพอร์ตอินพุตสัญญาณอะนาล็อก (A/D Converter) ได้
- Port B (PB0-PB7) เป็นขาพอร์ตอินพุตเอาต์พุตดิจิทัล กำหนดการพูลอัพภายในขาพอร์ตได้ และเป็นขาพอร์ตหน้าที่พิเศษอีกด้วย เช่น ขาสำหรับการโปรแกรมชิพ ขาป้องกันสัญญาณนาฬิกาภายนอก เป็นต้น
- Port C (PC0-PC7) นอกจากจะเป็นขาพอร์ตอินพุตเอาต์พุตดิจิทัลที่กำหนดการพูลอัพภายในขาพอร์ตได้แล้ว ยังเป็นขาพอร์ตหน้าที่พิเศษ เช่น ขาเชื่อมต่อกับคีย์บอร์ดและโปรแกรมด้วยการเชื่อมต่อแบบ JTAG เป็นต้น
- Port D (PD0-PD7) เป็นขาพอร์ตอินพุตเอาต์พุตดิจิทัล กำหนดการพูลอัพภายในขาพอร์ตได้ และขาพอร์ตทำหน้าที่พิเศษ เช่น ขาเชื่อมต่อพอร์ตอนุกรม ขาอินเตอร์รัพเนื่องจากสัญญาณภายนอก เป็นต้น

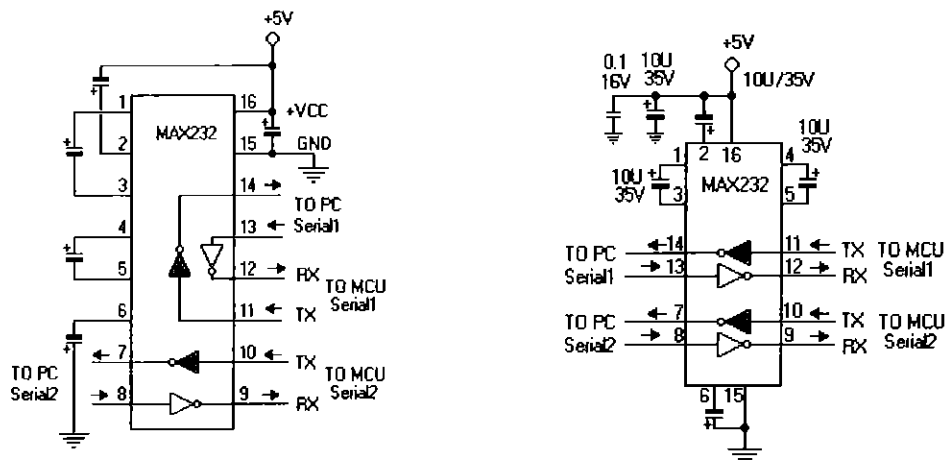
- **RESET** ขารี่เซ็ควงจร
- **XTAL1** ขาค่อคริสตอลออสซิลเลเตอร์ ช่องที่ 1 ด้านอินพุต
- **XTAL2** ขาค่อคริสตอลออสซิลเลเตอร์ ช่องที่ 2 ด้านเอาต์พุต
- **AVCC** ขาแรงดันสำหรับพอร์ต A และ โมดูลแปลงสัญญาณอะนาล็อก เป็นดิจิทัล
- **AREF** ขาแรงดันอะนาล็อกอ้างอิง สำหรับ โมดูลแปลงสัญญาณอะนาล็อกเป็นดิจิทัล

2.4 ไอซี MAX232 ,L232 [5]

ไอซี MAX232, L232 เป็นไอซีที่แปลงระดับสัญญาณจากระดับ TTL ไปเป็นระดับของ RS-232 และในทำนองเดียวกันก็รับระดับสัญญาณจาก RS-232 เพื่อแปลงเป็นระดับสัญญาณระดับ TTL ให้กับไมโครคอนโทรลเลอร์ได้ รายละเอียดของไอซีแสดงดังรูปที่ 2.11



(ก) แสดงตำแหน่งขาของไอซี MAX232 [5]

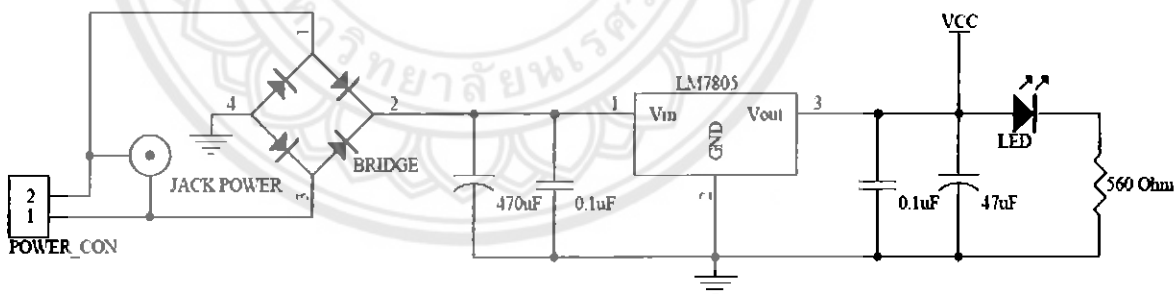


(ข) รูปวงจรภายใน (ค) การต่อใช้งาน MAX232 กับ ไมโครคอนโทรลเลอร์

รูปที่ 2.11 ไอซี MAX232 [5]

2.5 วงจรแปลงไฟและจ่ายไฟ [11]

วงจรแปลงไฟและจ่ายไฟนั้นจำเป็นอย่างยิ่งสำหรับไมโครคอนโทรลเลอร์ เพราะไมโครคอนโทรลเลอร์นั้น ใช้ไฟ DC 5 V เท่านั้น แต่แหล่งจ่ายนั้นมีแรงดันที่มากกว่านี้ ซึ่งเราสามารถต่อวงจรได้ดังรูปที่ 2.12 ซึ่งจะเห็นได้ว่าใช้ IC Regulator เบอร์ LM7805 ซึ่งทำหน้าที่ Regulate ไฟจาก 9-12 V ให้เป็นไฟ DC 5 V สามารถจ่ายกระแสได้สูงสุด 1 A

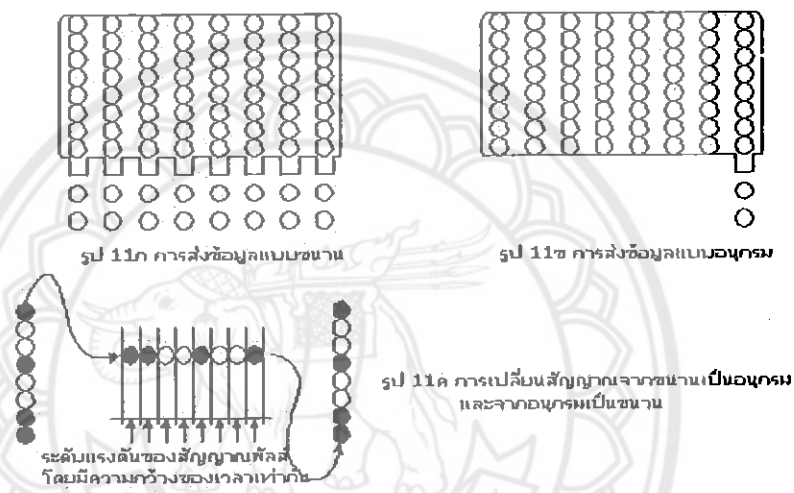


รูปที่ 2.12 วงจรแปลงไฟจาก AC/DC 9-12V เป็น DC 5 V [11]

ตามรูปที่ 2.12 Bridge Diode ทำหน้าที่แปลงไฟให้เป็นไฟบวก ทำให้เราสามารถจ่ายไฟเข้าวงจรนี้เป็นไฟกระแสสลับ หรือไฟกระแสตรงที่ 9 - 12V ได้ โดยไม่จำเป็นต้องคำนึงถึงขั้วไฟฟ้า

2.6 การสื่อสารข้อมูลแบบอนุกรม [6]

ข้อมูลในไมโครคอนโทรลเลอร์ที่เราใช้ศึกษาอยู่นี้ จะเป็นข้อมูลที่มีความยาวขนาด 1 ไบต์ หรือ 8 บิตซึ่งโดยปกติถ้าเราจะให้ส่งข้อมูลพร้อมๆกันไป 8 บิตจะเป็นวิธีการส่งข้อมูลแบบขนาน แสดงได้ดังรูป 11ก จะเป็นการส่งข้อมูลขนาด 8 บิตพร้อมกันไปยังอุปกรณ์ภายนอก และจะต้องมีจำนวนของสายสัญญาณจำนวน 8 เส้น เพื่อให้พอดีกับจำนวนของบิตที่ต้องการจะส่ง การส่งข้อมูลแบบขนานจึงทำให้มีการส่งข้อมูลที่มีความรวดเร็ว แต่ถ้าหากมีการสื่อสารข้อมูลในระยะไกล ก็จะต้องใช้จำนวนของสาย และระยะทางของสายมากขึ้นจึงทำให้มีการสิ้นเปลืองค่าใช้จ่ายสูง



รูปที่ 2.13 แสดงการส่งข้อมูลแบบขนานและแบบอนุกรม [6]

ดังนั้นการสื่อสารข้อมูลแบบอนุกรมจึงถูกนำมาใช้ ในการสื่อสาร โดยจะใช้สายเพียงเส้นเดียวในการส่งข้อมูล หรือรับข้อมูล (คำว่าเส้นเดียวหมายความว่าสายส่ง (Tx/D) 1 เส้น สายรับ (Rx/D) 1 เส้น และสายกราวด์ร่วม(Ground) 1 เส้น) นำมาใช้สื่อสารข้อมูลกับอุปกรณ์ภายนอกในระยะทางที่ไกล ดังในรูป 1ข ถ้าหากต้องการส่งข้อมูลขนาด 8 บิต ก็จะทำการส่งข้อมูลออกไปทีละบิตเป็นลำดับไป จนกว่าจะครบจำนวนทั้ง 8 บิต ดังในรูป 1ค จะแสดงการเปลี่ยนข้อมูลแบบขนานให้เป็นแบบอนุกรม ข้อมูลจะถูกส่งไปตามสายสัญญาณทีละบิตตามจังหวะเวลาที่กำหนดเป็นความกว้างของพัลส์ โดยจังหวะเวลาที่กล่าวนี้จะต้องมีมาตรฐาน ของฝ่ายส่ง และฝ่ายรับด้วย ในการรับสัญญาณที่ส่งมาทีละบิต จะทำการตรวจสอบระดับแรงดันของสัญญาณที่เข้ามาเพื่อแปลงเป็นลอจิก "1" หรือ "0" เมื่อรับข้อมูลเข้ามาครบใน 1 ไบต์ที่กำหนดไว้ ก็จะถูกเปลี่ยนให้อยู่ในรูปแบบของข้อมูลแบบขนานเหมือนเดิม

1) จังหวะเวลาของการสื่อสารข้อมูลอนุกรม

ในการสื่อสารข้อมูลแบบอนุกรม เพื่อรับหรือส่งข้อมูล จะเป็นลักษณะของกลุ่มข้อมูล ดังนั้นอัตราความเร็วจะต้องมีค่าเท่ากันระหว่างการรับและการส่งโดยทั่วไปเราจะระบุความเร็วของจำนวนบิตในการรับและส่งข้อมูล เป็นจำนวนของบิตที่จะส่งใน 1 วินาที โดยเรียกความเร็วในการส่งข้อมูลว่า อัตราบอด (Baud Rate) ซึ่งมีหน่วยเป็นบิตต่อวินาที เช่น 300, 1,200, 2,400, 4,800 และ 9,600 บิตต่อวินาที ถ้าหากมีการส่งข้อมูลด้วยความเร็ว 9600 บิตต่อวินาที จะใช้เวลาในการรับส่งข้อมูลหนึ่งบิตมีค่าเท่ากับ $1/9600$ หรือ 104.1 ไมโครวินาที และเวลาในการรับส่งข้อมูลทั้ง 8 บิต จะมีค่าเท่ากับ 8×104.1 หรือ 832.8 ไมโครวินาที



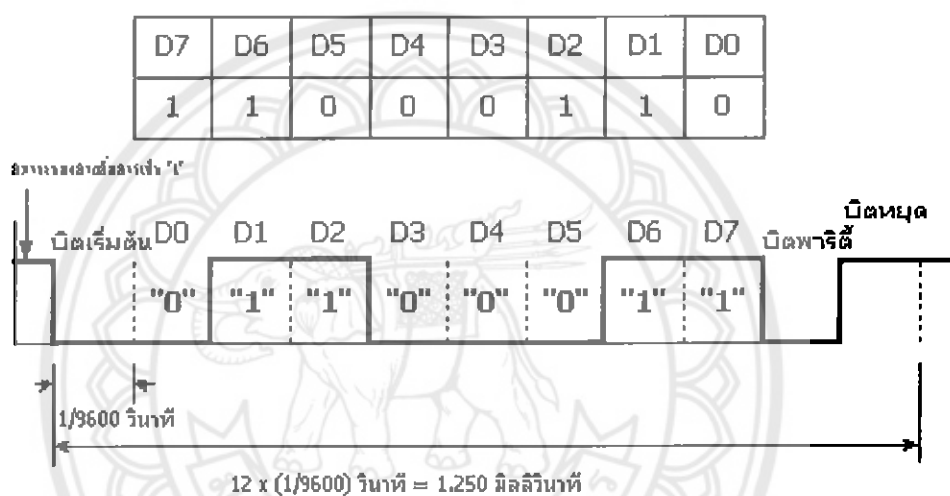
รูปที่ 2.14 แสดงการส่งข้อมูลแบบอนุกรมด้วยความเร็ว 9600 บิตต่อวินาที [6]

2) รูปแบบของการสื่อสารข้อมูลอนุกรม

การสื่อสารข้อมูลอนุกรมแบบอะซิงโครนัส เป็นวิธีการรับและส่งข้อมูลโดยไม่ต้องอาศัยสัญญาณนาฬิกาส่งร่วมไปด้วย แต่จะใช้อัตราความเร็วของจำนวนข้อมูลต่อวินาที และจะทำการเพิ่มบิตข้อมูลบางอย่างร่วมไปกับการส่งข้อมูลจริง เพื่อจะได้ทำการตรวจสอบข้อมูลได้อย่างถูกต้องมากยิ่งขึ้นแสดงดังรูปที่ 13 ซึ่งประกอบด้วยกัน 4 ส่วนคือ

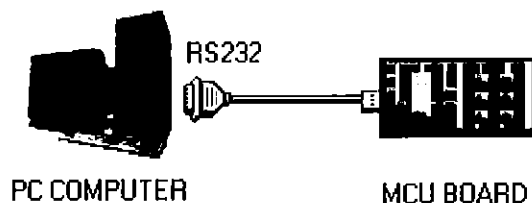
1. บิตเริ่มต้น (Start bit) จะมีขนาด 1 บิต จะเป็นระดับลอจิกตรงกันข้ามกับระดับลอจิกของสถานะสายสื่อสาร ขณะที่ยังไม่มีการส่งข้อมูล
2. บิตข้อมูล (Data bit) จะเริ่มจากบิตที่มีนัยสำคัญต่ำสุดก่อนหรือ บิต LSB ก่อน โดยข้อมูลที่จะส่งอาจจะมีขนาด 5, 6, 7 หรือ 8 บิตก็ได้

3. บิตแสดงสถานะเลขคู่หรือเลขคี่ (Parity bit) มีขนาด 1 บิตโดยบิตนี้จะนำไปต่อเข้ากับบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของข้อมูลที่เป็น "1" โดยเลือกการส่งข้อมูลเป็นแบบพาริตีคู่ หรือ พาริตีคี่ ตัวอย่างเช่น ถ้ากำหนดให้มีการส่งข้อมูลแบบพาริตีคู่ แต่ข้อมูลมีเลข 1 เป็นจำนวนคี่ ก็จะทำให้บิตพาริตีนี้เป็น "1" เพื่อจะได้จำนวนเลข "1" เป็นคู่นั่นเอง ทำนองเดียวกันทางด้านรับเองก็ต้องมีการตรวจสอบจำนวนข้อมูลที่ได้รับเข้ามาเป็น "1" รวมทั้งบิตพาริตี 1 บิต ถ้ามีค่า "1" เป็นจำนวนคู่แสดงว่าข้อมูลที่ได้รับเข้ามาถูกต้อง ส่วน 4 บิตสุดท้ายหรือบิตหยุด (Stop bit) เป็นการระบุถึงขอบเขตของการสิ้นสุดข้อมูล โดยจะทำให้ขาข้อมูลมีสถานะ ลอจิกเป็น "1" ซึ่งอาจมีจำนวนมากกว่าหนึ่งบิตก็ได้ เช่น 1 บิต 1.5 บิต หรือ 2 บิต



รูปที่ 2.15 แสดงการส่งข้อมูลขนาด 8 บิตแบบอนุกรม ด้วยความเร็ว 9600 บิตต่อวินาที [6]

2.7 การใช้งานพอร์ตอนุกรม RS232 [7]



รูปที่ 2.16 การเชื่อมต่อระหว่างคอมพิวเตอร์และไมโครคอนโทรลเลอร์ผ่านพอร์ต RS232 [7]

การสื่อสารแบบอนุกรม นับว่ามีความสำคัญ ต่อการใช้งาน ไมโครคอนโทรลเลอร์มาก เพราะสามารถใช้เป็นพิมพ์และจอภาพของ PC เป็น อินพุต และ เอาต์พุต ในการติดต่อหรือควบคุม ไมโครคอนโทรลเลอร์ ด้วยสัญญาณอย่างน้อยเพียง 3 เส้นเท่านั้น คือ

- สายส่งสัญญาณ TX
- สายรับสัญญาณ RX
- สาย GND

โดยปกติพอร์ตอนุกรม RS-232C จะสามารถต่อสายได้ยาว 50 ฟุตโดยประมาณ ขึ้นอยู่กับชนิดของสายสัญญาณ, ระยะทาง, และปริมาณสัญญาณรบกวน



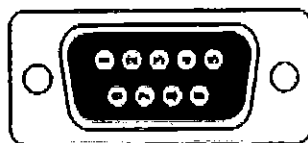
ก. พอร์ตอนุกรม DB9 ตัวผู้

ข. พอร์ตอนุกรม DB9 ตัวเมีย

รูปที่ 2.17 พอร์ตอนุกรม DB9 ทั้งตัวผู้และตัวเมีย [7]

- พอร์ตอนุกรมของ PC จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวผู้ (Male)
- พอร์ตอนุกรม ของอุปกรณ์ภายนอก จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวเมีย (Female)

1) การจัดขาของคอนเน็คเตอร์อนุกรมแบบ DB9 และหน้าที่การใช้งานต่างๆ

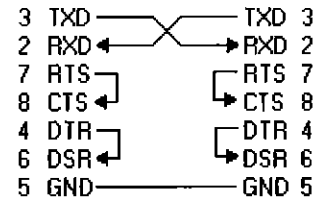
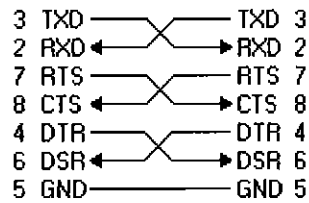


รูปที่ 2.18 DB9 ตัวผู้ เมื่อมองจากด้านหลัง [7]

ตารางที่ 2.1 ขาของคอนเน็คเตอร์อนุกรม DB9

Pin	Description	Type
1	Data Carrier Detect (DCD)	Input
2	Received Data (RXD)	Input
3	Transmitted Data (TXD)	Output
4	Data Terminal Ready (DTR)	Output
5	Signal Ground (GND)	Input
6	Data Set Ready (DSR)	Input
7	Request To Send (RTS)	Output
8	Clear to Send (CTS)	Input
9	Ring Indicator (RI)	Input

2) การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9



ก. การเชื่อมต่ออุปกรณ์ภายนอก

ข. การต่ออุปกรณ์ภายนอก

ผ่าน DB9 แบบ Null modem

ผ่าน DB9 แบบ 3 เส้น

รูปที่ 2.19 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9 [7]

3) การทำงานของขาสัญญาณ DB9

TXD เป็นขาที่ใช้ส่งข้อมูล

RXD เป็นขาที่ใช้รับข้อมูล

DTR แสดงสถานะพอร์ตว่าเปิดใช้งานอยู่

DSR ตรวจสอบว่าพอร์ตที่ติดต่อด้านนั้นเปิดอยู่หรือไม่

- เมื่อเปิดพอร์ตอนุกรม ขา DTR จะ ON เพื่อให้อุปกรณ์ได้รับทราบว่าการติดต่อดำเนินการ

- ในขณะเดียวกันก็จะตรวจสอบขา DSR ว่าอุปกรณ์พร้อมหรือไม่

RTS แสดงสถานะพอร์ตว่าต้องการส่งข้อมูล

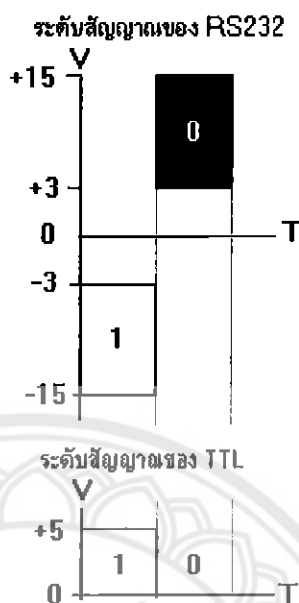
CTS ตรวจสอบว่าพอร์ตที่ติดต่อด้านนั้นต้องการส่งข้อมูลหรือไม่

- เมื่อต้องการส่งข้อมูลขา RTS จะ ON และจะส่งข้อมูลออกที่ขา TXD เมื่อส่งเสร็จก็จะ OFF

- ในขณะเดียวกันก็จะตรวจสอบขา CTS ว่าอุปกรณ์ที่ต้องการที่จะส่งข้อมูลหรือไม่

GND ขา ground

4) ระดับสัญญาณของ RS232



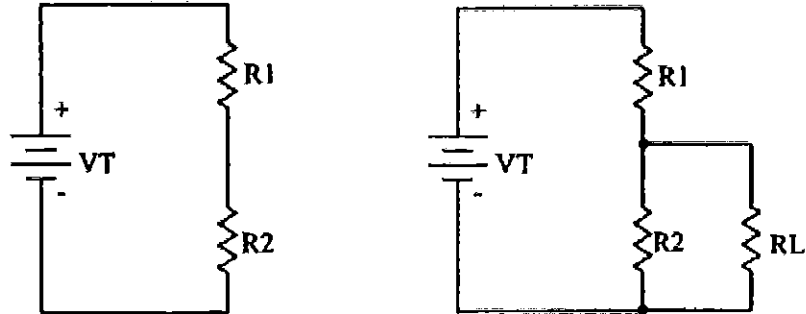
รูปที่ 2.20 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL [7]

สัญญาณรบกวนที่เกิดขึ้นในสายนำสัญญาณมักจะมีแรงดันเป็นบวก เมื่อเทียบกับกราวด์ เพื่อป้องกันสัญญาณรบกวนนี้ จึงออกแบบแรงดัน ของโลจิก "1" เป็นลบ คืออยู่ในช่วง -3V ถึง -15V ส่วนแรงดัน ของโลจิก "0" อยู่ในช่วง +3V ถึง +15V และเหตุที่ ระดับสัญญาณ ของ RS232 อยู่ในช่วง +15V ถึง -15V ก็เพื่อให้การต่อสายสัญญาณไปได้ไกลขึ้นนั่นเอง ดังนั้นจึงจำเป็นต้องมีวงจรเปลี่ยนระดับแรงดันของ RS232 มาเป็นระดับแรงดันของ TTL ด้วย

2.8 วงจรแบ่งแรงดันไฟฟ้า [8]

วงจรแบ่งแรงดันไฟฟ้า เรียกว่า “โวลต์เตจ ดิไวเดอร์” (Voltage Dividers) ใช้หลักการของวงจรไฟฟ้าแบบอนุกรม (Series Circuit) เนื่องจากวงจรอนุกรมมีแรงดันตกคร่อมตัวต้านทานหรือโหลดไม่เท่ากัน แต่เมื่อนำแรงดันที่ตกคร่อม โหลดทุกตัวในวงจรมารวมกันแล้วจะมีค่าเท่ากับแรงดันที่จ่ายให้แก่วงจร จากหลักการนี้ หากเรามีแหล่งจ่ายไฟฟ้ากระแสตรงขนาด 24 V ซึ่งแรงดันดังกล่าวมีค่ามากเกินไป แต่แรงดันที่ต้องการคือ 12 V เราก็สามารถใช้ตัวต้านทานมาแบ่งแรงดันจาก 24 V ให้เหลือเพียง 12 V แล้วนำแรงดันส่วนนี้ไปใช้งานได้เช่นกัน โดยไม่ต้องซื้อแหล่งจ่ายไฟฟ้า 12 V ใหม่ วงจรแบ่งแรงดันไฟฟ้าสามารถแบ่งออกได้ 2 แบบ ได้แก่

- วงจรแบ่งแรงดันที่ไม่มีโหลด (Unloaded Voltage Divider)
- วงจรแบ่งแรงดันที่มีโหลด (Load Voltage Divider)



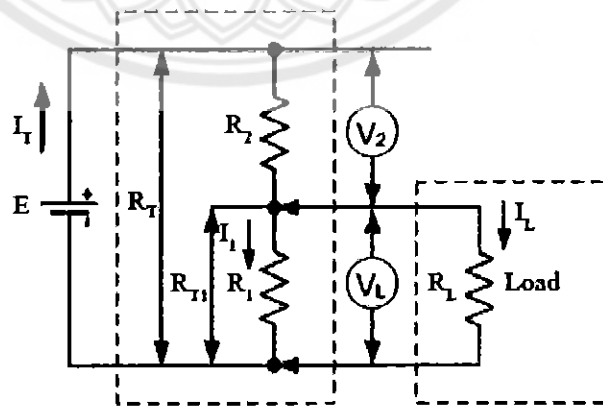
ก) วงจรแบ่งแรงดันที่ไม่มีโหลด ข) วงจรแบ่งแรงดันที่มีโหลด

รูปที่ 2.21 วงจรแบ่งแรงดันที่ไม่มีโหลดและวงจรแบ่งแรงดันที่มีโหลด [8]

ซึ่งในการใช้งานในโครงการนี้นั้น เราไม่ได้ใช้วงจรแบ่งแรงดันที่ไม่มีโหลด ดังนั้นจึงขอกล่าวเฉพาะวงจรแบ่งแรงดันที่มีโหลดเท่านั้น

วงจรแบ่งแรงดันที่มีโหลด

เป็นสภาวะของวงจรที่ต่อโหลดเข้ากับแหล่งจ่ายแรงดันไฟฟ้า ซึ่งจะทำให้มีกระแสไฟฟ้าไหลภายในวงจรเกิดขึ้น โดยกระแสที่ไหลผ่านความต้านทานแต่ละตัวที่ทำหน้าที่แบ่งแรงดันวงจร เรียกว่า กระแสบริดเคอร์ (Breeder Current) ซึ่งในวงจรแบ่งแรงดันนี้ ค่ากระแสบริดเคอร์ควรมีค่าน้อยประมาณ 10 – 20 เปอร์เซ็นต์ของกระแสโหลด มีการต่อวงจร ดังรูป



รูปที่ 2.22 วงจรการแบ่งแรงดันที่มีโหลด [8]

15256923

ร/ร.

22585

2551

เมื่อ	E	=	แหล่งจ่ายแรงดันไฟฟ้าของวงจร
	R_1, R_2	=	ความต้านทานที่ต่ออยู่ภายในวงจร
	R_L	=	ความต้านทานของโหลด
	R_T	=	ความต้านทานรวมทั้งหมดของวงจร
	I_T	=	กระแสที่ไหลในวงจรทั้งหมด
	I_1	=	กระแสที่ไหลผ่านความต้านทาน R_1
	I_L	=	กระแสที่ไหลผ่านโหลด
	V_L	=	แรงดันตกคร่อมโหลด
	V_2	=	แรงดันตกคร่อมความต้านทาน R_2

จากวงจรดังรูปที่ 2.22 จะได้ความสัมพันธ์ของสมการดังนี้

$$R_{T1} = \frac{R_1 \times R_L}{R_1 + R_L} \quad (2.6)$$

ดังนั้น

$$R_T = R_{T1} + R_2 \quad (2.7)$$

และสมการหาค่าแรงดันจะได้

$$V_L = E \frac{R_{T1}}{R_T} \quad (2.8)$$

$$V_2 = E \frac{R_2}{R_T} \quad (2.9)$$

หรือ

$$V_2 = E - V_L \quad (2.10)$$

สมการหาค่ากระแสจะได้

$$I_L = \frac{V_L}{R_L} \quad (2.11)$$

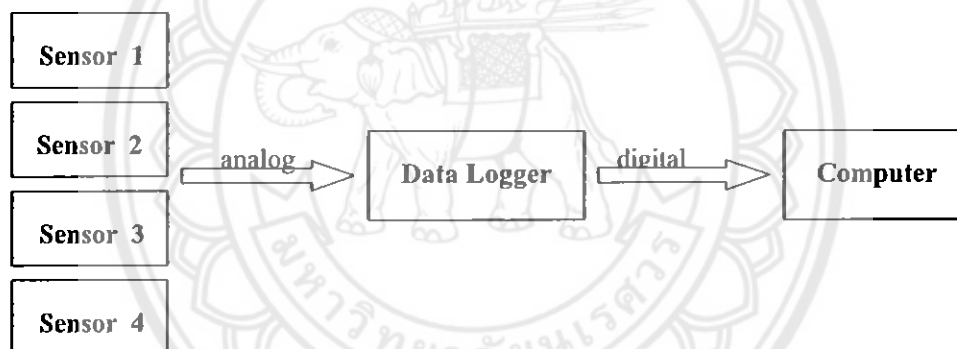
$$I_1 = \frac{V_L}{R_L} \quad (2.12)$$

และ

$$I_T = I_1 + I_2 \quad (2.13)$$

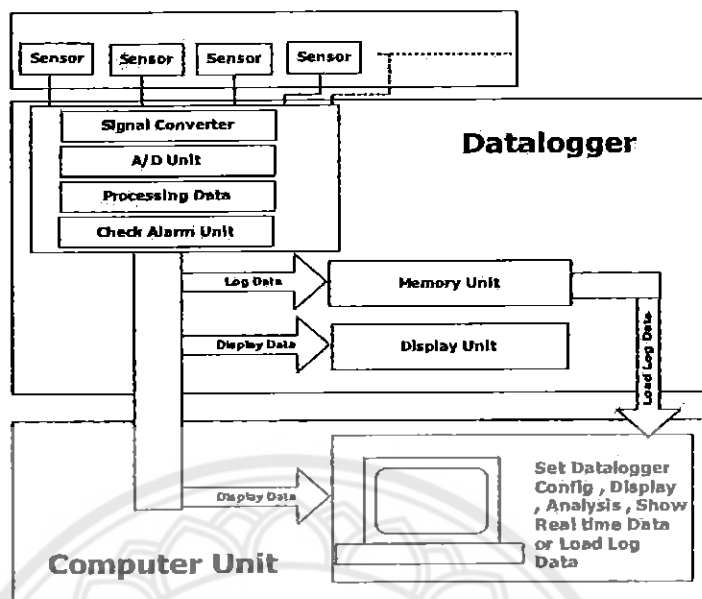
2.9 ระบบจัดเก็บข้อมูล (Data Logger) [9]

Data Logger คือ ระบบที่ใช้ในการจัดการและจัดเก็บข้อมูล องค์ประกอบพื้นฐานของระบบ Data Logger ประกอบไปด้วย Scanner หรือ multiplexer digital-voltmeter และตัวบันทึกข้อมูล ซึ่งรับ input ที่เป็นระบบ analog จาก sensor แล้วทำการเปลี่ยนข้อมูลเป็นระบบ digital และเก็บข้อมูลไว้ในหน่วยความจำเพื่อการนำไปใช้ต่อไป



รูปที่ 2.23 ระบบจัดเก็บข้อมูล (Data Logger) [9]

1) โครงสร้างของระบบ Data Logger



รูปที่ 2.24 โครงสร้างของระบบจัดเก็บข้อมูล (Data Logger) [9]

จากรูป 2.24 สามารถอธิบายการทำงานของระบบได้ว่า Data Logger จะรับค่าที่จะบันทึกจากตัว Sensor นำมาผ่าน Signal Converter เพื่อทำการแปลงสัญญาณที่รับมาให้เป็นสัญญาณ A/D ซึ่ง Data Logger สามารถนำมาใช้ในการแปลงให้เป็นข้อมูล Digital ได้ หลังจากนั้น Data Logger อาจนำข้อมูล Digital นั้นมาประมวลผล หรือนำข้อมูลมาตรวจสอบ เพื่อทำการส่ง Alarm ไปเตือนผู้ใช้งาน ข้อมูล มีค่ามากไปหรือน้อยไปหรือไม่ แล้วจึง นำข้อมูลที่ได้ไปเก็บใน Memory ของ Data Logger หรือนำมาแสดงผลบนหน้าจอ LCD ของ Data Logger หรือบนหน้าจอกอมพิวเตอร์ได้

2) Data logging systems

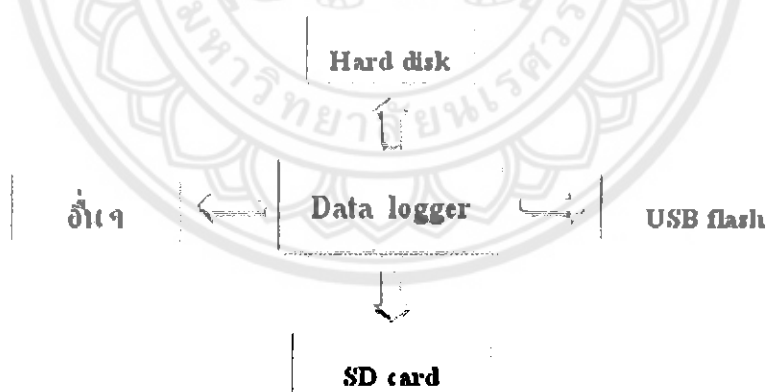
ระบบ Data Logger จะใช้การควบคุมโดย Microprocessor ซึ่งใช้ระบบ bus สองระบบคือ ระบบ Data bus ซึ่งใช้ในการส่งข้อมูล และ ระบบ Address bus ซึ่งใช้ในการอ้างตำแหน่ง โดย software ที่ใช้ในการควบคุมจะถูกเก็บไว้ในหน่วยความจำทั้ง ROM (read only memory) และ RAM (random access memory) ซึ่งชุดคำสั่ง programs หลักจะถูกเก็บไว้ใน ROM ส่วนข้อมูล input ของตัวแปรในแต่ละช่องสัญญาณและชุดคำสั่ง program อื่นๆจะถูกเก็บไว้ใน RAM

ระบบ Scanner จะบรรจุสวิตช์และมีสายเชื่อมต่อกับ integrating digital voltmeter ซึ่งส่วนมากจะมีความเร็วสูง โดยใช้ solid-state switching devices (JFET) ซึ่งการทำงานของ Scanner นั้นจะถูกควบคุมโดยระบบ controller โดยรูปแบบการทำงานของ Scanner แบ่งได้ดังนี้

1. single-channel recording mode
2. single scan of all channel mode
3. continuous scan mode
4. periodic scan mode

3) การเก็บข้อมูล digital

สัญญาณจาก Transducer จะผ่านเข้าไปใน Scanner ต่อไปยัง Integrating Digital Voltmeter โดยใช้ Analog to Digital Converter ซึ่งความเร็วของการทำงานนั้นขึ้นอยู่กับประสิทธิภาพของ Integrating Digital Voltmeter นอกจากนี้ความละเอียดที่ต้องการเป็น output จาก Data Logger นั้นจะเป็นค่าที่แสดงในจอภาพ ซึ่งแสดงเป็นแรงดันของค่า output และการบันทึกข้อมูลจาก Data Logger โดยมากจะพิมพ์ออกทางเครื่องพิมพ์ แต่อย่างไรก็ตามยังสามารถเก็บบันทึกข้อมูล ไว้ในอุปกรณ์อื่นได้อีกด้วย เช่น SD-card Flash memory และ Harddisk เป็นต้น

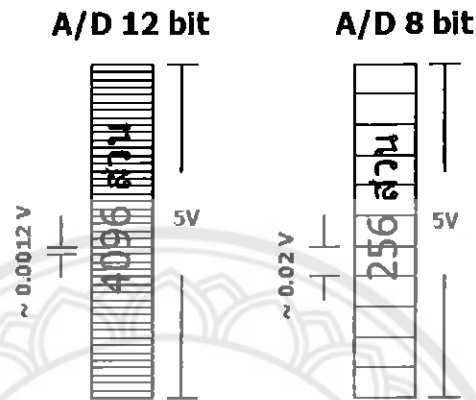


รูปที่ 2.25 อุปกรณ์ที่สามารถเก็บข้อมูลได้

4) คุณสมบัติที่สำคัญของ Data Logger

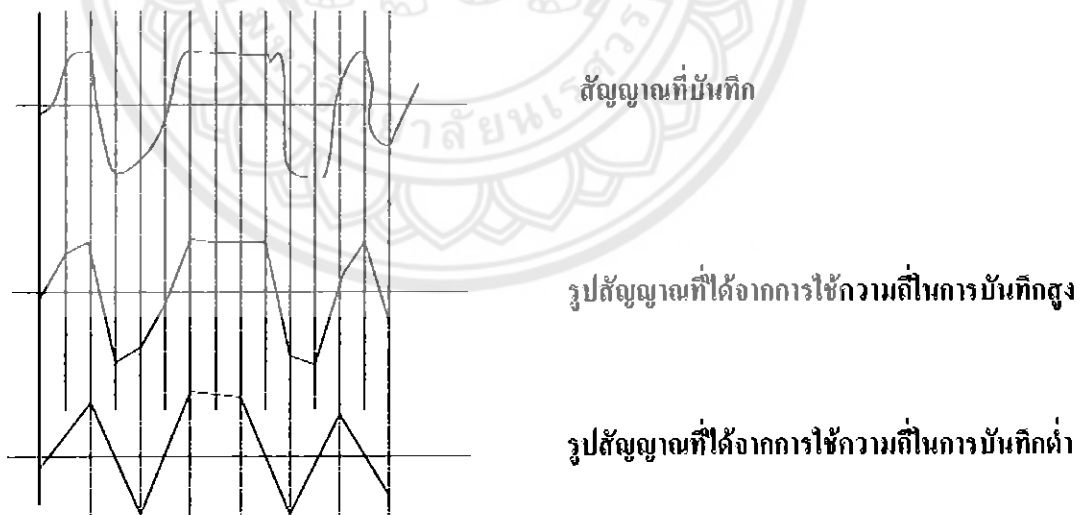
ความละเอียด (Resolution)

ความละเอียดของค่าที่ได้จากการวัดขึ้นอยู่กับ A/D ของ Data Logger ที่เราเลือกใช้ว่ามีจำนวน bit เป็นเท่าไร โดยที่จำนวน bit ยิ่งมากเท่าไร ก็จะได้ค่าที่วัดละเอียดมากขึ้นเท่านั้น



รูปที่ 2.26 ความละเอียด A/D ของ Data Logger [9]

ความถี่ในการเก็บข้อมูล (Sampling Frequency)



รูปที่ 2.27 ความถี่ในการเก็บข้อมูล [9]

ขนาดของ memory ที่ใช้ในการเก็บข้อมูล

พิจารณาจาก

- ช่วงเวลาทั้งหมดที่เราต้องการให้ Data logger เก็บบันทึกข้อมูล (Recording Duration)
- ค่า Sampling Time ที่เราต้องการใช้ในการบันทึกข้อมูล

โดยที่

$$\frac{\text{Recording Duration}}{\text{Sampling Time}} = \text{จำนวน Record ที่สามารถบันทึกได้ของ Memory}$$

จากหลักการและทฤษฎีในส่วนต่างๆ ที่ได้กล่าวมาทั้งหมดในบทนี้ มีความจำเป็นอย่างยิ่งในการทำโครงการนี้ให้สำเร็จลุล่วงได้ ซึ่งทฤษฎีเหล่านี้จะถูกนำไปประกอบการออกแบบระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถยนต์ประหยัดพลังงาน ดังที่จะได้กล่าวถึงในบทถัดไป



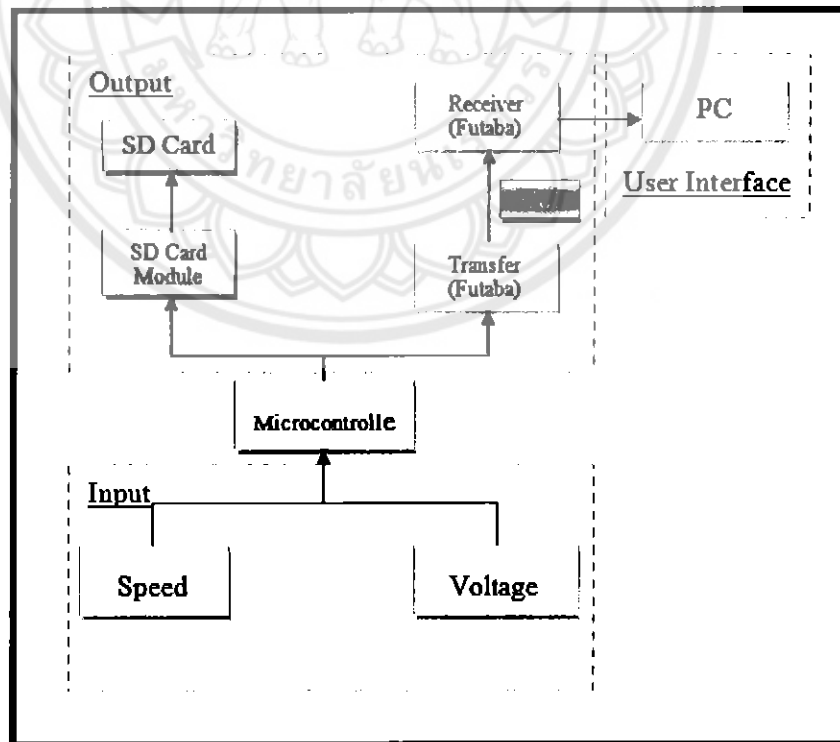
บทที่ 3

ออกแบบ และพัฒนาระบบ

การพัฒนาระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์นั้น ได้มีการออกแบบให้สามารถบันทึกและแสดงผลข้อมูลไว้ได้ 2 ส่วน คือ ส่วนแรกคือบันทึกไฟล์ลงสู่ SD card และส่วนที่สองคือการส่งข้อมูลไร้สาย เพื่อแสดงผลมายังคอมพิวเตอร์ เพื่อรับรู้สถานะต่างๆในขณะนั้น การเก็บข้อมูลทั้งสองส่วนนั้นก็เพื่อเพิ่มความยืดหยุ่นในการใช้งาน และยังเป็นการเพิ่มช่องทางการจัดเก็บข้อมูลอีกด้วย จากบทที่ผ่านมาได้กล่าวถึงทฤษฎีต่างๆ ที่จะนำมาพัฒนาระบบจัดเก็บข้อมูลมาแล้ว ในบทนี้จะกล่าวถึงการออกแบบและพัฒนาระบบต่อไป

การออกแบบระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์

ในการออกแบบระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์นั้นสามารถแบ่งย่อยได้เป็น 4 ส่วน คือ ส่วนอินพุต ส่วนเอาต์พุต ส่วนประมวลผลโดยใช้ไมโครคอนโทรลเลอร์ และส่วนติดต่อกับผู้ใช้แบบออนไลน์ ซึ่งในแต่ละส่วนนั้นมีหน้าที่แตกต่างกันไป โดยที่การออกแบบโดยรวมของระบบแสดงได้ดังรูปที่ 3.1 และจะได้กล่าวถึงการออกแบบส่วนต่างๆต่อไป



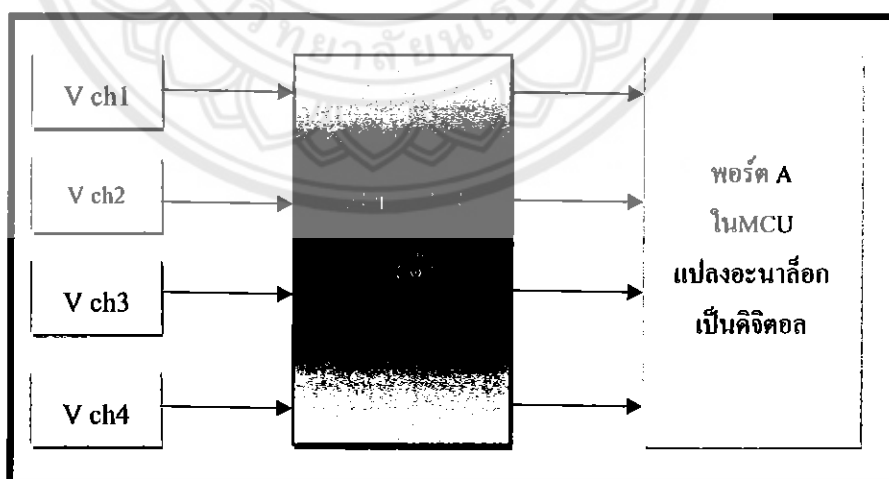
รูปที่ 3.1 แผนผังระบบจัดเก็บข้อมูลสำหรับรถประหยัดพลังงาน

3.1 การออกแบบส่วนอินพุต

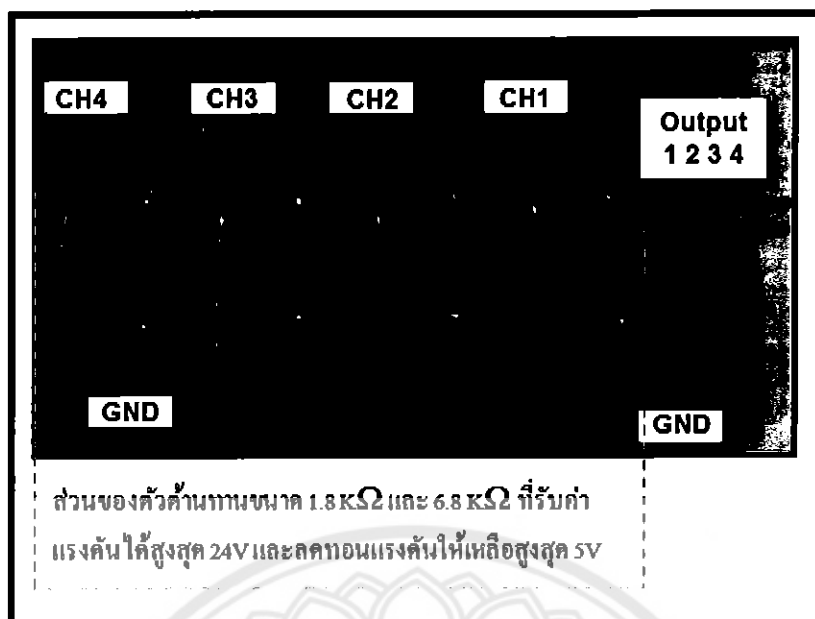
มีหน้าที่ในการรับข้อมูลต่างๆจากภายนอกเข้ามา และส่งต่อไปให้กับไมโครคอนโทรลเลอร์เพื่อทำการประมวลผลต่อไป ซึ่งข้อมูลที่อินพุตเข้ามามี 2 แบบ คือ ค่าแรงดันไฟฟ้าและค่าความเร็ว

1) การวัดค่าแรงดันไฟฟ้า

ในการวัดค่าแรงดันไฟฟ้านั้นจะใช้สายไฟวัดค่ามาจากจุดที่ต้องการ ซึ่งปกติแล้วจุดที่วัดคือ ที่แหล่งกำเนิดไฟฟ้า และที่มอเตอร์ล้อ ซึ่งเป็นส่วนอินพุตและเอาต์พุตในระบบของรถประหยัดพลังงาน แต่อย่างไรก็ตามในโครงการนี้ได้ตั้งค่าให้สามารถวัดได้ทั้งหมด 4 แชนแนล เพื่อเพิ่มความยืดหยุ่นในการใช้งาน ต่อจากนั้นจึงมีการนำแรงดันไฟฟ้าเข้าสู่วงจรแบ่งแรงดันไฟฟ้า (Voltage divider) เพื่อลดทอนแรงดันไฟฟ้าที่เข้ามาให้เหลือแรงดันน้อยกว่าหรือเท่ากับ 5 V (รายละเอียดและการคำนวณวงจรแบ่งแรงดันแสดงไว้แล้วในบทที่ 2 โดยเฉพาะสมการ 2.9) เพื่อให้ไมโครคอนโทรลเลอร์สามารถอ่านค่าได้ และไม่เกิดความเสียหายแก่วงจรไฟฟ้าอื่นๆ ต่อจากนั้นจึงจะนำค่าแรงดันที่ถูกลดทอนแล้วเข้าสู่พอร์ต A ของไมโครคอนโทรลเลอร์ซึ่งเป็นพอร์ตที่ทำหน้าที่แปลงสัญญาณอะนาล็อก ซึ่งก็คือแรงดันไฟฟ้าที่ถูกลดทอนสัญญาณมาแล้ว ไปเป็นสัญญาณดิจิทัล ที่ไมโครคอนโทรลเลอร์สามารถนำไปประมวลผลได้ต่อไป ซึ่งกระบวนการวัดค่าแรงดันไฟฟ้านั้น แสดงดังรูปที่ 3.2 และวงจรแบ่งแรงดันแสดงดังรูปที่ 3.3



รูปที่ 3.2 กระบวนการวัดแรงดันไฟฟ้า



รูปที่ 3.3 วงจรแบ่งแรงดันไฟฟ้า

2) การวัดค่าความเร็วของล้อ

จะใช้เซ็นเซอร์มาช่วยในการนับจำนวนรอบของล้อ โดยใช้ทฤษฎีทางฟิสิกส์ในการคำนวณหาอัตราเร็ว จากสูตรที่ว่าอัตราเร็วมีค่าเท่ากับระยะทางหารด้วยเวลา โดยค่าของระยะทาง จะได้มาจากการคำนวณหาเส้นรอบวงของล้อรถ ส่วนค่าของเวลาจะได้จากไมโครคอนโทรลเลอร์โดยใช้ Timer นับแล้วทำการ Interrupt มาบอกว่าจำนวนรอบของล้อในแต่ละรอบใช้เวลาเท่าใด แล้วจึงนำค่าเวลาที่ได้มาคำนวณในสูตรดังกล่าว

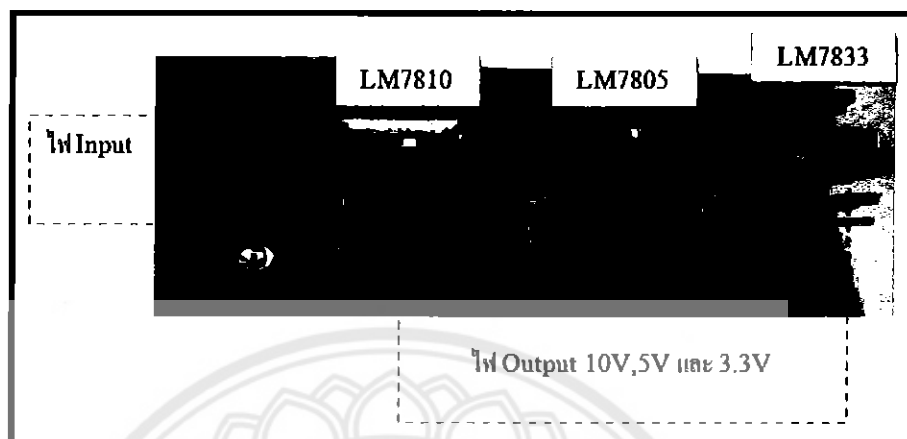
3.2 ส่วนประมวลผล

เป็นส่วนที่ทำหน้าที่เป็นศูนย์กลางในการรับอินพุตมาทำการประมวลผล และทำการส่งข้อมูลออกทางส่วนเอาต์พุต ซึ่งส่วนประมวลผลนี้ประกอบไปด้วยวงจร 3 ส่วน ที่ทำงานอย่างสอดคล้องประสานและสนับสนุนซึ่งกันและกัน อันประกอบไปด้วย

1) วงจรแปลงไฟและจ่ายไฟ

คือวงจรที่ทำหน้าที่แปลงไฟฟ้ากระแสตรงให้ได้ค่าแรงดันที่ต้องการ โดยใช้ไอซี Regulator เบอร์ต่างๆ ซึ่งในวงจรแปลงไฟและจ่ายไฟในโครงงานนี้ได้กำหนดให้สามารถจ่ายไฟฟ้ากระแสตรงได้ 3 ค่าด้วยกัน สำหรับนำไปเลี้ยงวงจรต่างๆ คือ ไอซี LM7805 สำหรับจ่ายไฟ 5 V ให้กับไมโครคอนโทรลเลอร์ ไอซี LM7810 สำหรับจ่ายไฟ 10 V สำหรับจ่ายไฟ

ให้กับวงจรส่งข้อมูลไร้สายระบบ FM และไอซี LM7833 สำหรับจ่ายไฟ 3.3V ให้กับ SD card แสดงได้ดังรูปที่ 3.4



รูปที่ 3.4 ไอซีแปลงไฟและจ่ายไฟทั้ง 3 เบอร์

2) วงจรติดต่อสื่อสารผ่านพอร์ตอนุกรม RS-232

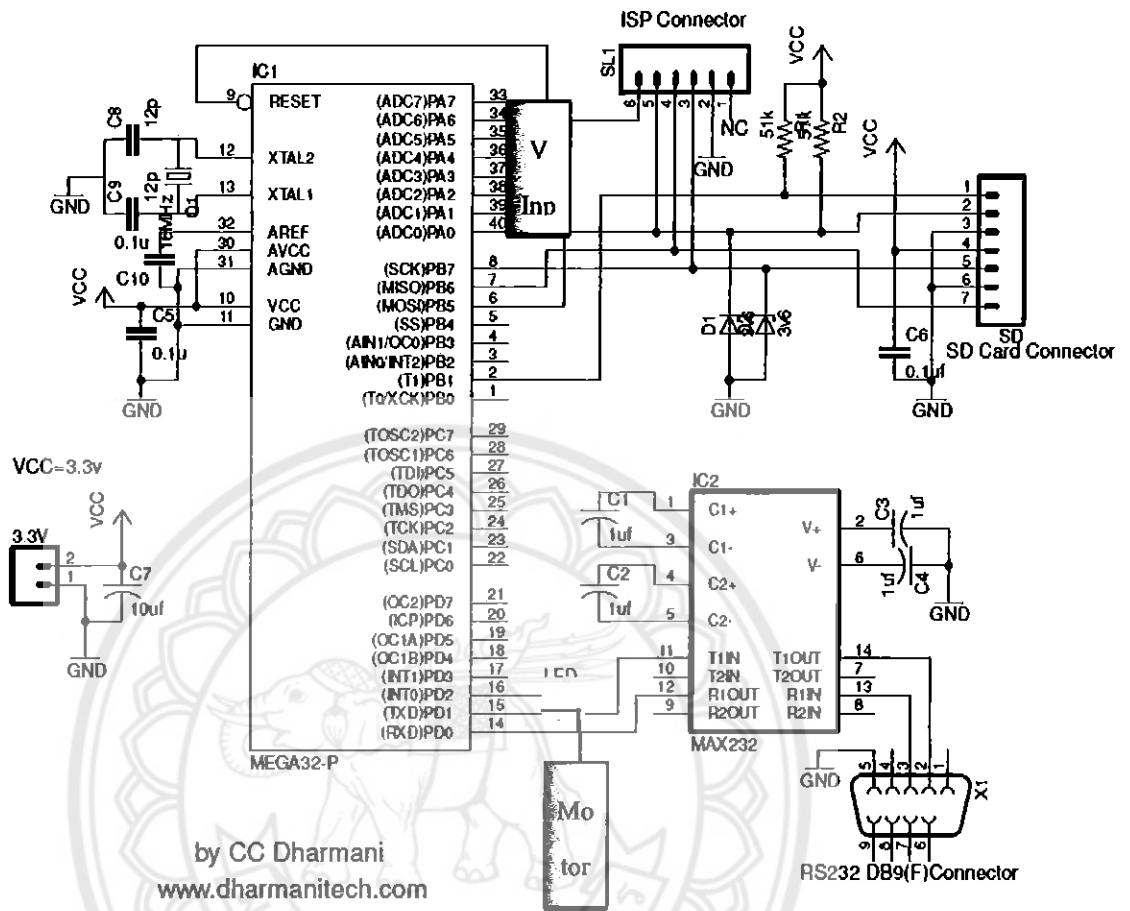
คือวงจรที่ทำหน้าที่สื่อสารระหว่างไมโครคอนโทรลเลอร์และอุปกรณ์เชื่อมต่อต่างๆ ผ่านพอร์ต DB9 ที่เป็น Serial port สามารถเชื่อมต่อกับคอมพิวเตอร์ได้ โดยมี IC MAX232 เป็นตัวแปลงระดับสัญญาณ TTL ของไมโครคอนโทรลเลอร์ให้เป็นสัญญาณ RS-232 โดยมีช่องสัญญาณ 2 ช่อง และยังสามารถแปลงสัญญาณกลับกันได้อีกด้วย

3) วงจรไมโครคอนโทรลเลอร์

คือวงจรที่ทำหน้าที่สั่งการ ประมวลผล และทำงานตามโปรแกรมที่ได้เขียนไว้ โดยใช้ไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATmega32 ผ่านพอร์ต I/O แบบขนาน 8 บิตที่มีอยู่ทั้งหมด 4 พอร์ต โดยแต่ละพอร์ตได้ออกแบบให้ใช้งานในโครงงานนี้ดังนี้

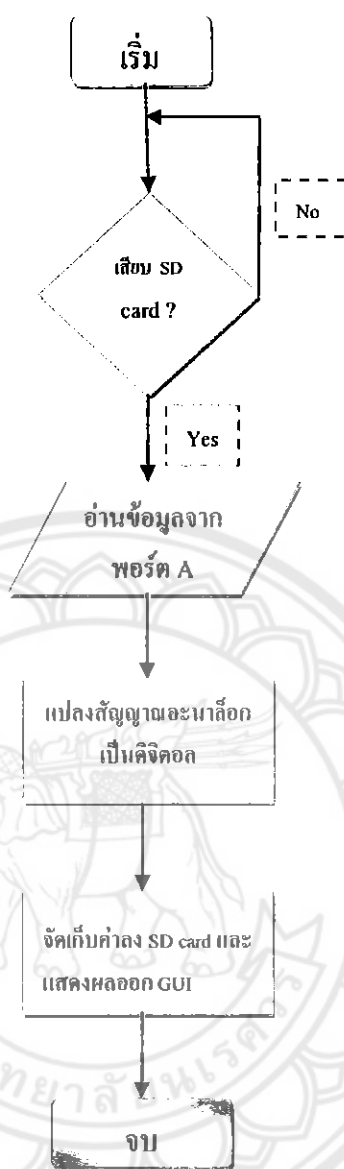
- Port A ได้ออกแบบให้ใช้งานเป็นพอร์ตแปลงสัญญาณอะนาล็อกเป็นสัญญาณดิจิทัล (ADC) โดยสัญญาณอินพุตอะนาล็อกที่เข้ามานั้น มาจากวงจรแบ่งแรงดันไฟฟ้าในหัวข้อที่แล้วนั่นเอง
- Port B ได้ออกแบบให้ใช้งานกับ SD card และ ISP connector สำหรับบันทึก (Burn) โปรแกรมลงสู่ไมโครคอนโทรลเลอร์
- Port C เป็นพอร์ตสำรอง
- Port D ได้ออกแบบให้ใช้งานกับวงจรติดต่อสื่อสารพอร์ตอนุกรม RS-232 และวงจรส่งข้อมูลไร้สายระบบ FM (ที่ขา TX)

การเชื่อมต่อสำหรับพอร์ตต่างๆของไมโครคอนโทรลเลอร์ แสดงไว้ดังรูปที่ 3.5



รูปที่ 3.5 วงจรการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆ [10]

ส่วนของบอร์ดไมโครคอนโทรลเลอร์ ATmega32 ที่ได้ออกแบบไว้ แสดงดังรูปที่ 3.6



รูปที่ 3.7 Flowchart การทำงานของโปรแกรม

ผลการแปลงสัญญาณอะนาล็อกเป็นดิจิทัล คำนวณผลลัพธ์ได้จากสูตรต่อไปนี้

$$ADC = \frac{V_{in} \times 1024}{V_{REF}} \quad (3.1)$$

โดย V_{IN} คือ แรงดันด้านขาอินพุต
 V_{REF} คือ แรงดันอ้างอิงที่ต่อเข้ากับ AREF (ขาที่ 32)
 และค่า 1024 มาจาก ADC ใช้งานที่ 10 บิต

3.3 การออกแบบส่วนเอาต์พุต

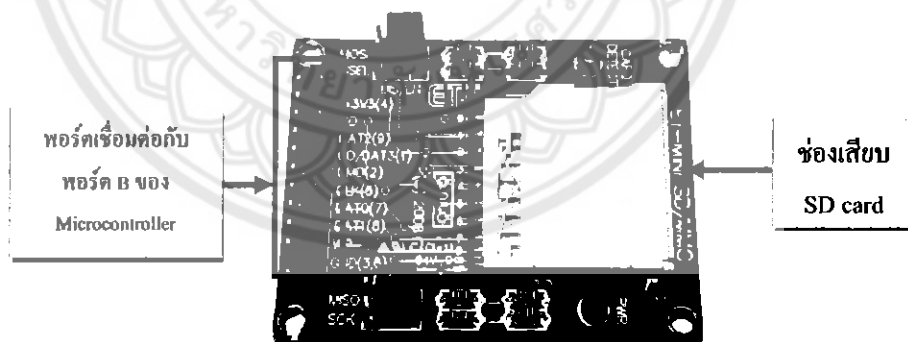
ในส่วนเอาต์พุตนี้จะหมายถึงส่วนของตัวจัดเก็บข้อมูลและระบบติดตามผลแบบออนไลน์ โดยตรง ซึ่งมี 2 ส่วนด้วยกันคือ ตัวจัดเก็บข้อมูลโดยใช้ SD card และ ระบบติดตามผลแบบออนไลน์โดยใช้ระบบรับส่งข้อมูลแบบไร้สาย ซึ่งการทำงานของทั้ง 2 ระบบ มีรายละเอียดดังต่อไปนี้

1) ระบบเก็บข้อมูลโดย SD card (SD card Data logger)

การออกแบบระบบเก็บข้อมูลโดย SD card จะมีการเชื่อมต่ออุปกรณ์ที่ต้องใช้ในการพัฒนาระบบดังต่อไปนี้

3.3.1.1 อุปกรณ์เชื่อมต่อหน่วยความจำ SD Card (SD Module)

คือ ชุดอุปกรณ์สำหรับการเชื่อมต่อกับอุปกรณ์หน่วยความจำ SD Card ของบริษัท ETT ซึ่งในโครงการนี้จะเป็นการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับหน่วยความจำ SD card ภายในชุดอุปกรณ์ประกอบด้วย ช่อง (Socket) สำหรับใส่การ์ดความจำประเภท SD โดยจะมีการจัดขาสัญญาณออกมาที่ Connector Pin เพื่อให้สามารถนำไปต่อใช้งานได้สะดวก นอกจากนี้ยังมีวงจรตรวจสอบสถานะการเสียบการ์ด (Card Detector) และวงจรพูลอัพ (Pull-Up) สัญญาณต่างๆ อีกด้วย รูปของอุปกรณ์แสดงไว้ดังรูปที่ 3.8



รูปที่ 3.8 แสดงอุปกรณ์เชื่อมต่อหน่วยความจำ SD Card (SD Module)

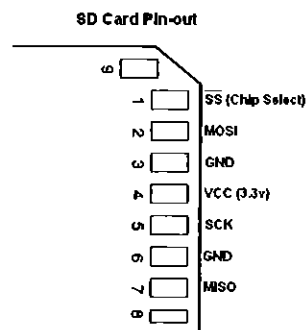
3.3.1.2 หน่วยความจำ SD Card

SD card คือ สื่อจัดเก็บข้อมูลแบบ Nand Flash Memory ที่ถูกพัฒนาขึ้นจากสื่อเก็บข้อมูลแบบ MMC card เพื่อเป็นหน่วยเก็บข้อมูลขนาดกลาง สำหรับกล้องถ่ายภาพดิจิทัล โทรศัพท์มือถือ และ PDA ซึ่งมีคุณสมบัติ คือ ราคาถูก เก็บข้อมูลได้มาก และมีอายุการใช้งานได้นานขึ้น รวมถึงการมีระบบ security ล็อกที่ตัวการ์ด ไม่ให้มีการเขียนทับ โดยขนาดของหน่วยความจำ SD card อยู่ที่ 32(L) x 24(W) x 2.1(H) ใช้แรงดันไฟฟ้ากระแสตรง 3.3 V

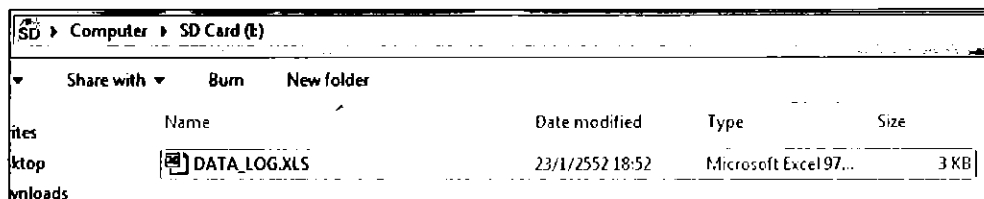
ขั้นตอนการทำงานของระบบนี้นั้น เริ่มแรกจะมีการเขียนโปรแกรมให้ตรวจสอบสถานะว่า SD card นั้น ได้เสียบอยู่หรือไม่ หากไม่มี SD card ระบบนี้จะไม่มีการทำงาน จากนั้นโปรแกรมจะมีการสร้างไฟล์ขึ้นชื่อว่า DATA_LOG.XLS ซึ่งเป็นไฟล์ของ Microsoft Excel เพื่อให้ข้อมูลที่ได้นั้นเป็นตารางที่เป็นระเบียบเรียบร้อย ต่อมาไมโครคอนโทรลเลอร์จึงจะรับข้อมูลเข้ามาเป็นระบบดิจิทัล และเขียนข้อมูลลงสู่ SD card ไปเรื่อยๆ จนกว่าจะจบการทำงาน โดยข้อมูลแต่ละชุดมีระยะเวลาห่างกัน 1 วินาที โดยรูปของ SD card, ขาของ SD card, ไฟล์ที่เขียน และข้อมูลที่ได้ แสดงดังรูปที่ 3.9 ถึง 3.12 ตามลำดับ



รูปที่ 3.9 หน่วยความจำ SD Card



รูปที่ 3.10 ขาของหน่วยความจำ SD Card



รูปที่ 3.11 ไฟล์ที่เขียนลงใน SD Card

	A	B	C	D	E	F	G
1	Time	Ch1	Ch2	Ch3	Ch4	Sp(Km/hr)	
2	0:01	0.:0	0.65	0.45	0.8		
3	0:02	0.9	0.55	0.65	0.55		
4	0:03	0.13	0.5	0.9	0.45		
5	0:04	0.2	0.8	0.45	0.45		
6	0:05	0.55	0.55	0.:0	0.1		
7	0:06	0.9	0.45	0.12	0.3		
8	0:07	0.9	0.65	0.1	0.12		
9	0:08	0.45	0.55	0.12	0.9		
10	0:09	0.75	0.12	0.9	0.2		
11	0:10	0.13	0.5	0.5	0.65		
12	0:11	0.:0	0.55	0.55	0.5		

รูปที่ 3.12 ข้อมูลที่เขียนลงใน SD Card

2) ระบบติดตามผลแบบออนไลน์ (Online monitoring system)

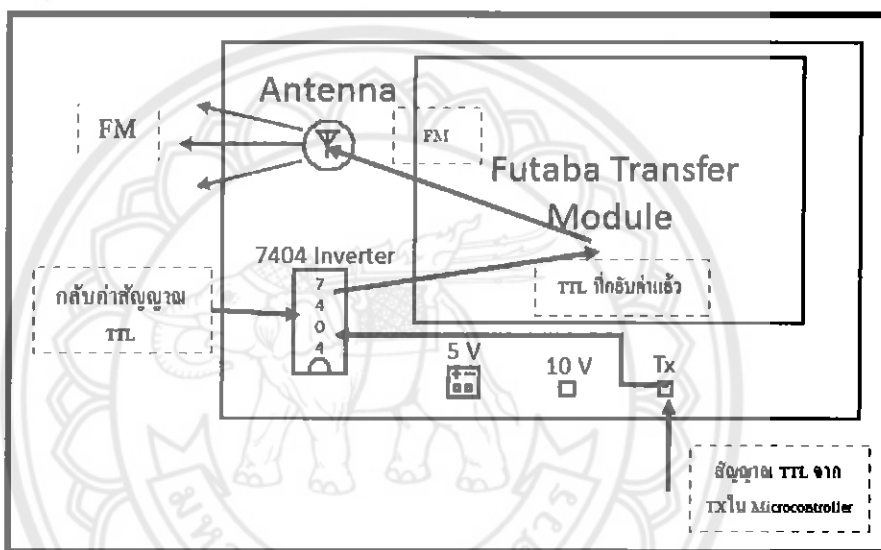
เป็นระบบที่สามารถส่งข้อมูลต่างๆ จากรถประจำทางมายังผู้ติดตามผลผ่านระบบไร้สายได้ทันที (real-time) ในขณะที่รถกำลังแล่นอยู่ได้ ซึ่งในระบบนี้ใช้อุปกรณ์การรับส่งข้อมูลไร้สายด้วยระบบ FM ชื่อ Futaba ซึ่งเป็นอุปกรณ์ที่ใช้อย่างแพร่หลายในวงการรถบังคับ เครื่องบินบังคับ และเรือบังคับ ซึ่งมีขั้นตอนและอุปกรณ์ต่างๆที่ใช้ในการรับส่งข้อมูลไร้สาย ดังต่อไปนี้

3.3.2.1 **Futaba Transfer Module** หรือตัวส่งสัญญาณ ซึ่งมีขั้นตอนการทำงานคือ มีขารับสัญญาณ TX มาจากไมโครคอนโทรลเลอร์ซึ่งเป็นสัญญาณ TTL แล้วเข้าสู่ IC Inverter 7404 ซึ่งเป็น NOT gate สำหรับกลับค่าสัญญาณ TTL จากนั้นจึงส่งสัญญาณเข้าสู่ Futaba Transfer Module เพื่อแปลงสัญญาณ TTL เป็นสัญญาณวิทยุ FM สุดท้ายจึงส่งสัญญาณไปยังเสาอากาศเพื่อกระจายคลื่นความถี่ออกไปเป็นระบบ FM

คุณสมบัติของ Futaba Transfer Module

- รองรับสัญญาณ Tx แบบ TTL (0-5 V)
- ใช้ไฟฟ้ากระแสตรง 5 และ 10 V เลี้ยงอุปกรณ์
- สามารถเปลี่ยนอุปกรณ์ส่งสัญญาณ Futaba ให้เป็นย่านความถี่อื่นๆได้โดยการเปลี่ยนคริสตอล (ในโครงการนี้ใช้ย่านความถี่ 72 MHz)
- มีระยะส่งสัญญาณประมาณ 500 เมตร

รูปของอุปกรณ์และขั้นตอนการทำงานแสดงดังรูปที่ 3.13 ส่วนรูปที่ 3.14 เป็นรูปอุปกรณ์จริง



รูปที่ 3.13 อุปกรณ์และขั้นตอนการทำงานของ Futaba Transfer Module



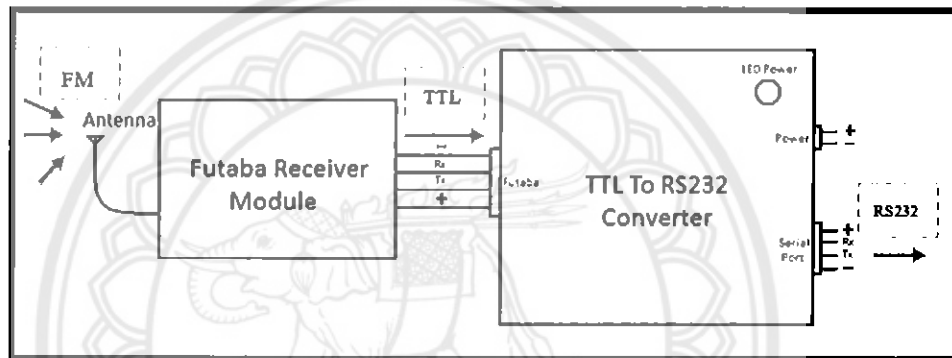
รูปที่ 3.14 Futaba Transfer Module

3.3.2.2 Futaba Receiver Module หรือตัวรับสัญญาณ จะมีอุปกรณ์สำหรับแปลงสัญญาณ FM ที่ได้รับจาก Transfer Module ออกมาเป็นสัญญาณ TTL และยังมี IC RS232 ที่ใช้ในการแปลงสัญญาณ TTL ไปเป็น RS232 จากนั้นจึงส่งสัญญาณ RS232 ออกทางพอร์ต DB9 Serial Port (RS232) ไปยังเครื่องคอมพิวเตอร์ต่อไป

คุณสมบัติของ Futaba Receiver Module

- ใช้ไฟฟ้ากระแสตรง 5 V เลี้ยงอุปกรณ์
- มี DB9 Serial Port (RS232) เพื่อเชื่อมต่อกับคอมพิวเตอร์ได้

รูปของอุปกรณ์และการทำงานแสดงดังรูปที่ 3.15 ส่วนอุปกรณ์จริงคือรูปที่ 3.16



รูปที่ 3.15 อุปกรณ์และการทำงานของ Futaba Receiver Module



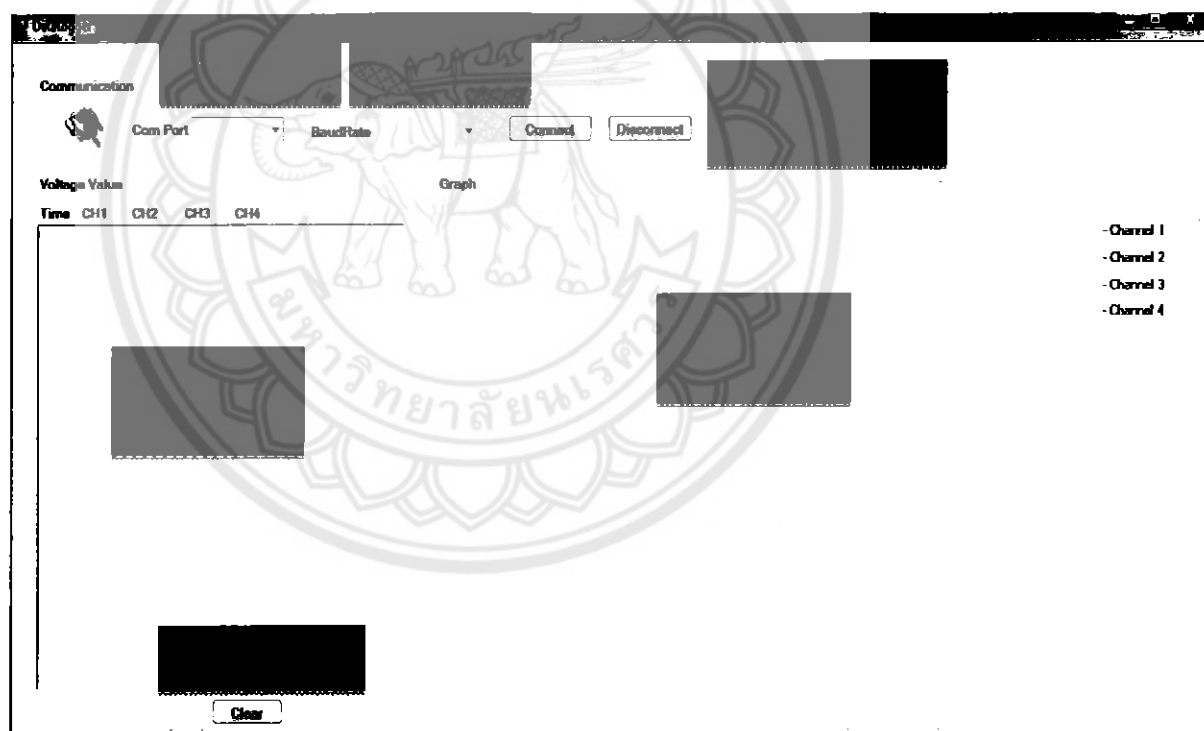
รูปที่ 3.16 Futaba Receiver Module

3.4 การออกแบบส่วนติดต่อกับผู้ใช้

ในส่วนติดต่อกับผู้ใช้นั้นสร้างขึ้นมาเพื่อใช้เชื่อมต่อและแสดงผลบนระบบติดตามผลแบบออนไลน์ โดยได้ออกแบบพัฒนาในภาษา Visual Basic ซึ่งมีคำสั่งให้ใช้งานกับ Serial port ได้ โดยเป็น Serial Port ที่รับสัญญาณ RS232 มาจากตัวรับสัญญาณไร้สาย (Futaba receiver module) หรืออาจจะรับสัญญาณ RS232 มาจากไมโครคอนโทรลเลอร์ก็ได้ โดยที่โปรแกรมนี้ได้กำหนดฟังก์ชันการใช้งานไว้ดังนี้

- สามารถเลือก Serial port ได้
- สามารถเลือกอัตราการรับส่งข้อมูล (Baud rate) ได้
- สามารถติดตามผลข้อมูลออนไลน์ได้ 4 แชนแนล ทั้งข้อมูลตัวเลขและกราฟ
- สามารถต่อและตัดการเชื่อมต่อได้

ซึ่งรูปของโปรแกรมที่ได้ออกแบบไว้แสดงดังรูปที่ 3.17



รูปที่ 3.17 โปรแกรมติดต่อกับผู้ใช้

เมื่อการออกแบบในแต่ละส่วนได้เสร็จสิ้นและนำมาประกอบรวมกันทุกๆส่วนแล้ว
ในขั้นต่อไปจึงเป็นการทดสอบระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถ
ประชิดพลังงาน เพื่อหาข้อดีข้อเสียของระบบและทดสอบว่าระบบสามารถทำงานได้บรรลุตาม
วัตถุประสงค์หรือไม่ ซึ่งการทดสอบทั้งหมดจะได้กล่าวถึงต่อไปในบทที่ 4



บทที่ 4

ผลการทดสอบ

การทดสอบความถูกต้องในการทำงานของระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์นั้นเป็นสิ่งที่สำคัญ เพราะเป็นตัวบ่งชี้ว่าอุปกรณ์ที่ได้สร้างขึ้นมานั้นสามารถใช้งานได้จริงหรือไม่ ซึ่งการทดสอบการทำงานในแต่ละส่วนนั้นมีดังนี้

ส่วนที่ 1 การทดสอบการอ่านข้อมูลเข้าทาง Serial port เพื่อตรวจสอบความถูกต้องของข้อมูลที่อ่านเข้ามาได้

ส่วนที่ 2 การทดสอบการอ่านค่าความเร็วจากคีย์บอร์ด

ส่วนที่ 3 การทดสอบการบันทึกข้อมูลลงสู่ SD card

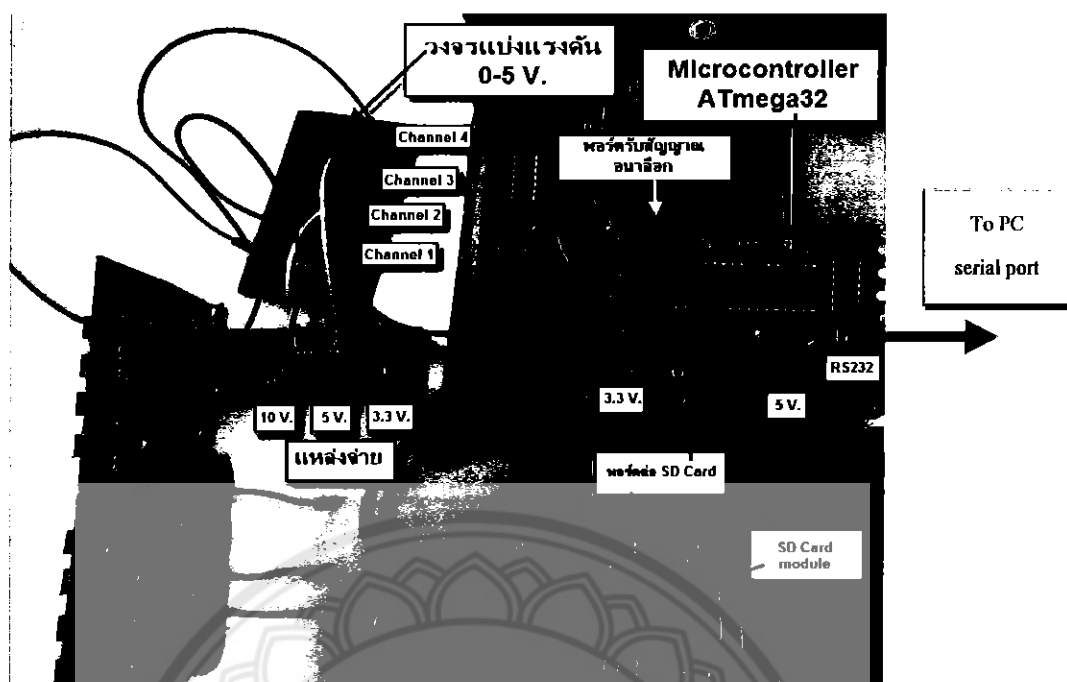
ส่วนที่ 4 การทดสอบการรับส่งข้อมูลไร้สาย

ส่วนที่ 5 การทดสอบส่วนติดต่อกับผู้ใช้ หรือส่วน GUI

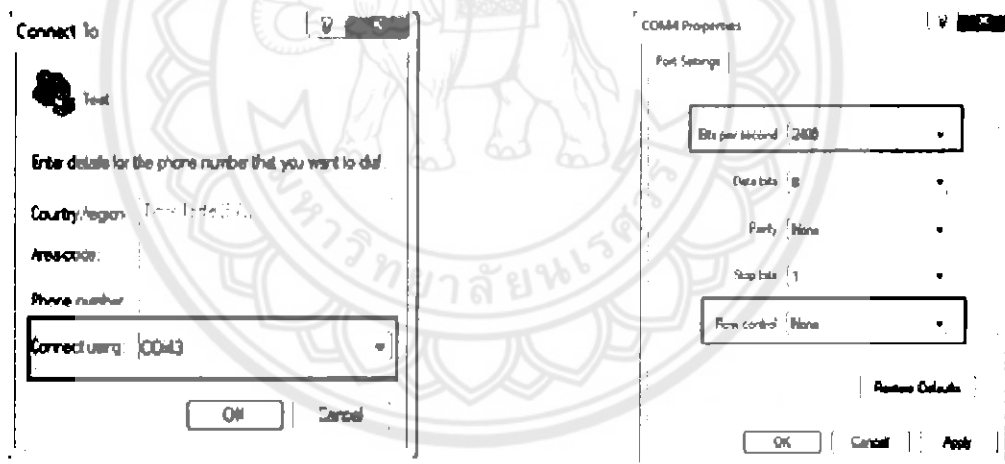
4.1 ทดสอบการอ่านข้อมูลเข้า Serial port

เป็นการทดสอบการทำงานของระบบว่า ระบบสามารถรับส่งข้อมูลเพื่อนำมาแสดงผลผ่านทาง Serial port (RS232) ได้หรือไม่ ซึ่งนับเป็นการทดสอบขั้นแรกของระบบ หากระบบสามารถแสดงผลออกทาง Serial port ได้แล้ว จะทำให้การประยุกต์ใช้ในรูปแบบอื่นๆกระทำได้ง่ายมากยิ่งขึ้น ซึ่งโปรแกรมที่นำมาทดสอบในส่วนนี้คือโปรแกรม Hyper Terminal ที่แต่เดิมมีอยู่แล้วโดยไม่ต้องติดตั้งใหม่ภายใน Windows XP แต่หากใช้ Windows vista หรือ Windows 7 แล้วนั้นจะต้องทำการดาวน์โหลดโปรแกรมนี้มาติดตั้งเอง ซึ่งเป็นฟรี โปรแกรมและสามารถหาแหล่งดาวน์โหลดได้ง่าย ขั้นตอนการทดลองมีดังนี้

ทำการเชื่อมต่อวงจรและสายพอร์ต ดังรูปที่ 4.1 หลังจากนั้นทำการเซตค่าการเชื่อมต่อ HyperTerminal โดยเลือก Serial port ตามที่ได้เชื่อมต่อเข้ามา (ในที่นี้ใช้พอร์ต COM3) จากนั้นเลือก Baud rate ให้มีค่า 2400 และปิดการใช้งาน Flow control โดยต้องเชื่อมต่อ SD card ไว้ตลอด มิเช่นนั้น โปรแกรมจะไม่ทำงาน ดังรูปที่ 4.2



รูปที่ 4.1 แสดงการเชื่อมต่อวงจรต่างๆ เข้าด้วยกัน



รูปที่ 4.2 การตั้งค่า HyperTerminal

หลังจากนั้นสั่งให้ไมโครคอนโทรลเลอร์ทำงาน และรอดูผลการทำงานทาง Hyper Terminal ดังรูปที่ 4.3

Time	CH1	CH2	CH3	CH4
00:02	4.14	4.41	4.10	4.25
00:03	4.14	4.16	4.73	4.87
00:04	4.23	4.13	4.25	4.19
00:05	4.22	4.11	3.13	3.75
00:06	3.60	3.75	3.00	4.74

รูปที่ 4.3 แสดงค่าที่อ่านได้จากไมโครคอนโทรลเลอร์

จากรูปที่ 4.3 จะเห็นได้ว่าระบบสามารถอ่านค่าแรงดันไฟฟ้าและแสดงผลออกมาได้ทั้ง 4 แชนแนล ซึ่งขั้นต่อไปจะเป็นการทดสอบว่าค่าที่ระบบอ่านเข้ามาได้นั้น มีความคลาดเคลื่อนไปจากค่าที่อ่านจากโวลต์มิเตอร์มากน้อยเพียงใด

ตารางที่ 4.1 ผลการทดสอบความคลาดเคลื่อนของข้อมูล

แรงดันจาก โวลต์มิเตอร์ (V)	แรงดันจากระบบ (V)	ค่าความคลาดเคลื่อน (%)
5	3.19	36.20
6	4.18	30.33
7	5.14	37.20
8	5.16	35.5
9	6.16	31.55
10	7.21	27.9
11	8.21	25.36
12	9.19	23.42
ค่าความคลาดเคลื่อนเฉลี่ย		30.93

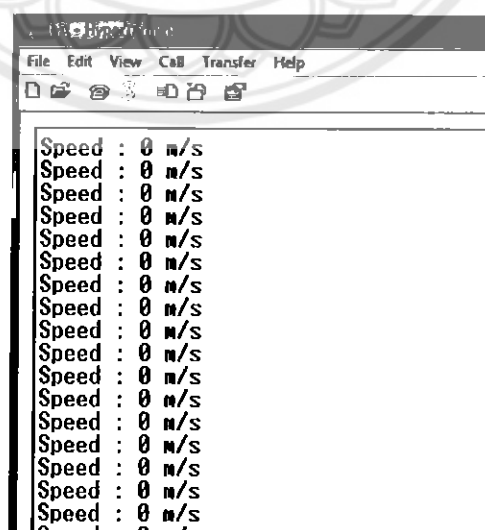
จากตารางที่ 4.1 แสดงให้เห็นว่าค่าที่อ่านเข้ามาได้นั้นยังมีความคลาดเคลื่อนสูงถึง 30.93% ซึ่งมีสาเหตุมาจากวงจรแบ่งแรงดันนั้นค่าของตัวต้านทานที่คำนวณ ได้เป็นค่าที่ไม่มีขายในท้องตลาด จึงต้องใช้ตัวต้านทานที่มีค่าใกล้เคียง นอกจากนี้ความคลาดเคลื่อนยังมีผลมาจากค่าสัมประสิทธิ์ที่ใช้คูณในโปรแกรมเพื่อปรับค่าให้ใกล้เคียง โปรแกรมไม่สามารถคูณด้วยจุดทศนิยมได้เพราะการแสดงผลเป็นตัวแปรประเภท Character จึงทำให้ใช้ค่าจำนวนเต็มได้เพียงอย่างเดียว ส่งผลให้มีความคลาดเคลื่อนเกิดขึ้น

4.2 การทดสอบการอ่านค่าความเร็วจากดีซีมอเตอร์

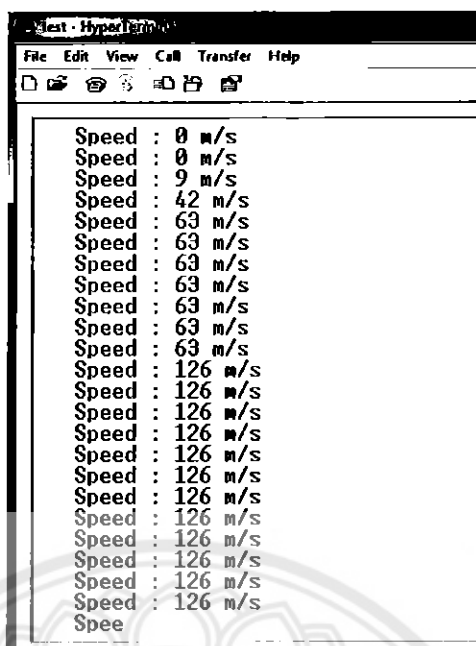
เป็นการทดสอบอ่านค่าความเร็วของล้อจากการจำลองสถานการณ์ขึ้น โดยใช้ดีซีมอเตอร์ดีคิไบพัดแทนล้อรถ ซึ่งมีอัตราส่วนระหว่างเส้นรอบวงของล้อกับเส้นรอบวงของใบพัดประมาณ 27.5 เท่า โดยที่ดีซีมอเตอร์นั้นหากหมุนครบ 1 รอบ จะนับได้ 2 ครั้ง จึงต้องนับทั้งหมด 55 รอบ จึงจะเท่ากับการหมุนของล้อ 1 รอบ นอกจากนี้ยังได้จำลองการเพิ่มลดความเร็วให้กับดีซีมอเตอร์อีกด้วย รูปการทดสอบกับดีซีมอเตอร์แสดงดังรูปที่ 4.4 และได้ผลการทดลองดังรูปที่ 4.5 และ 4.6



รูปที่ 4.4 แบบจำลองการวัดความเร็วดีซีมอเตอร์



รูปที่ 4.5 แสดงค่าเมื่อมอเตอร์ไม่ได้หมุน



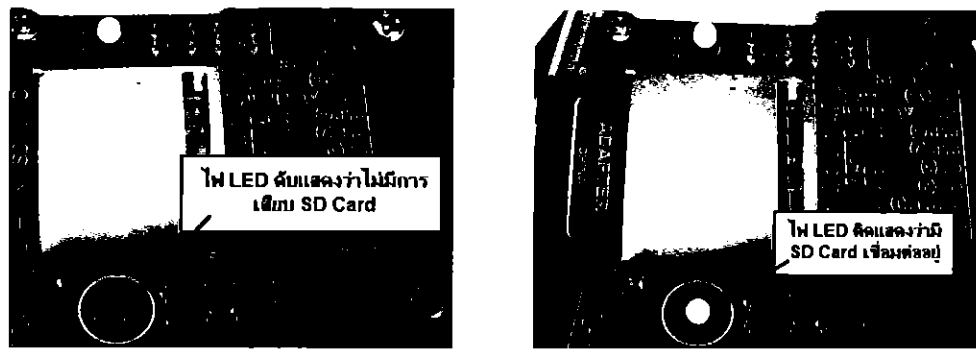
รูปที่ 4.6 แสดงค่ามอเตอร์ที่กำลังหมุน

จากผลการทดลองที่ได้จะเห็นได้ว่าโปรแกรมสามารถอ่านค่าความเร็วมาจากคีย์บอร์ดได้ โดยที่ค่าที่อ่านได้สามารถเปลี่ยนแปลงไปตามค่าที่ผู้ทดสอบปรับแต่ง แต่ที่ค่าความเร็วมีค่าที่สูงมากก็เป็นเพราะว่าไม่มีโหลดในระบบเลย ทำให้มอเตอร์หมุนได้อย่างอิสระ อย่างไรก็ตามค่าความเร็วที่อ่านได้ทำได้เพียงแค่การแสดงผลออกมาเท่านั้น ยังไม่สามารถจัดเก็บลง SD card ได้ เพราะ memory ของไมโครคอนโทรลเลอร์ไม่เพียงพอต่อการโปรแกรมระบบ SD card และการวัดความเร็วได้พร้อมกัน นอกจากนี้แล้วก็ยังคงไม่สามารถทำการรับส่งข้อมูลระยะไกลได้เช่นกัน

4.3 การทดสอบการบันทึกข้อมูลลงสู่ SD card

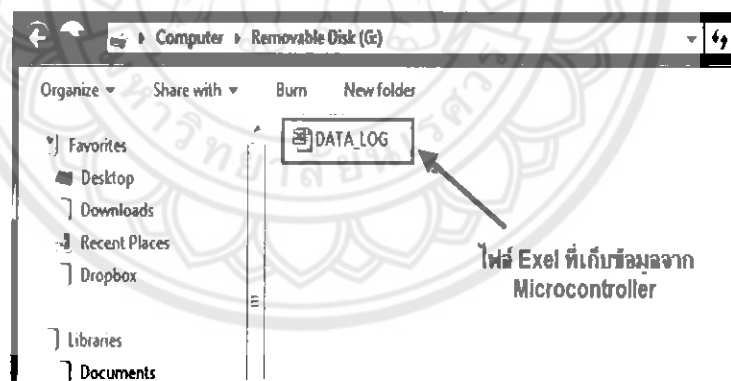
การทดสอบนี้เป็นการทดสอบว่าระบบจัดเก็บข้อมูลสามารถบันทึกข้อมูลลงสู่ SD card ได้หรือไม่ ในการทดสอบนี้ได้โปรแกรมให้ข้อมูลที่อ่านเข้ามาได้นั้น บันทึกลงสู่ไฟล์ Microsoft Excel ชื่อไฟล์ DATA_LOG.XLS ขั้นตอนการทดสอบมีดังต่อไปนี้

เชื่อมต่ออุปกรณ์เข้าด้วยกันเหมือนดังรูปที่ 4.1 แล้วจ่ายไฟเข้าระบบ จากนั้นเสียบการ์ดเข้าที่ช่องเสียบ ซึ่งตัว SD card module จะทำหน้าที่ตรวจสอบสถานะการเสียบการ์ด ดังรูปที่ 4.4



รูปที่ 4.7 แสดงสถานะการเชื่อมต่อของ SD Card กับ SD card module

จากนั้นไมโครคอนโทรลเลอร์จะอ่านสัญญาณอนาล็อกจากภายนอกผ่านวงจรแบ่งแรงดันเข้าสู่พอร์ต PA0-PA3 โดยในที่นี้จะใช้แหล่งจ่าย 5 V. เป็นสัญญาณภายนอกต่อเข้ากับวงจรแบ่งแรงดัน เพื่อแสดงการเปลี่ยนแปลงของสัญญาณภายนอก จะทำการปรับค่าของตัวต้านทานปรับค่าได้ ทำให้ค่าที่ได้มีการเปลี่ยนแปลงเกิดขึ้น จากนั้นข้อมูลจะถูกเขียนลงใน SD Card ดังรูป 4.5



ก) ไฟล์ที่บันทึกลงใน SD card

	A	B	C	D	E	F
1	Time	Ch1	Ch2	Ch3	Ch4	Sp(Km/hr)
2		0:01				
3		0:02				
4		0:03				
5		0:04				
6		0:05				
7		0:06				
8		0:07				
9		0:08				
10		0:09				

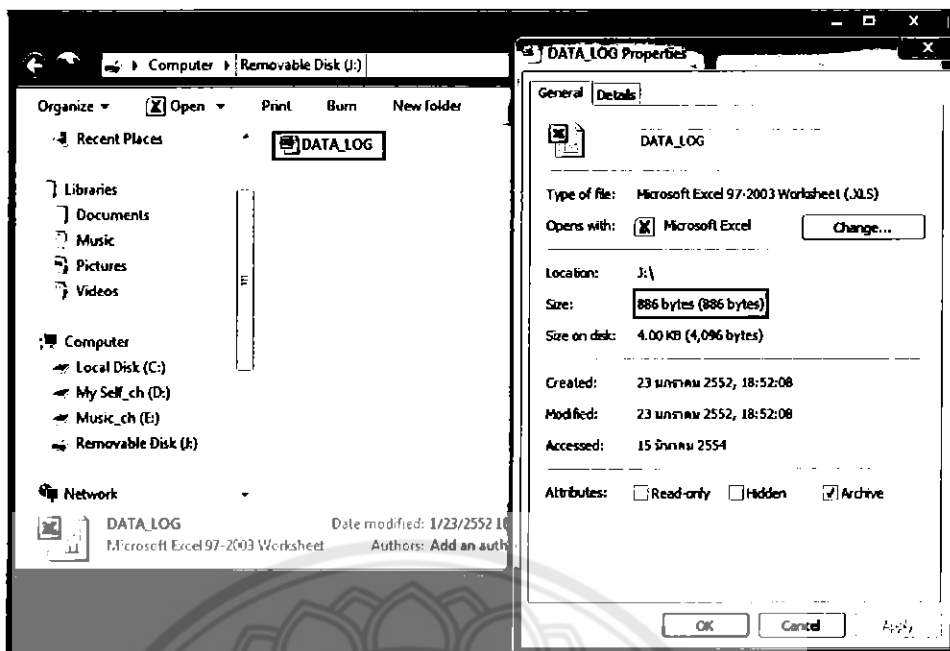
ข) แสดงรูปเมื่อไม่มีการรับข้อมูลเข้ามา

	Time	Ch1	Ch2	Ch3	Ch4	Sp(Km/hr)
180	2:49					4.59
181	2:50					4.59
182	2:51					4.59
183	2:52					4.59
184	2:53					4.59
185	2:54					4.59
186	2:55					4.59
187	2:56					4.59
188	2:57					3.45
189	2:58					3.45
190	2:59					3.76

ค) แสดงการรับข้อมูลและผลการเปลี่ยนแปลงเมื่อปรับค่าตัวด้านทานปรับค่าได้

รูปที่ 4.8 แสดงข้อมูลภายใน SD Card

เมื่อสามารถเขียนข้อมูลลง SD card ได้แล้ว จะทำการทดสอบต่อไปว่าไฟล์ที่เขียนเข้ามานั้นสามารถเขียนได้เต็มความจุ SD card ขนาด 2 GB ที่ระยะเวลาเท่าไร ซึ่งในการทดสอบนี้นั้นจะเขียนข้อมูลเต็มทั้ง 4 แชนแนล ผลการทดสอบเป็นดังต่อไปนี้



รูปที่ 4.9 ขนาดของไฟล์ที่เวลา 1 นาที

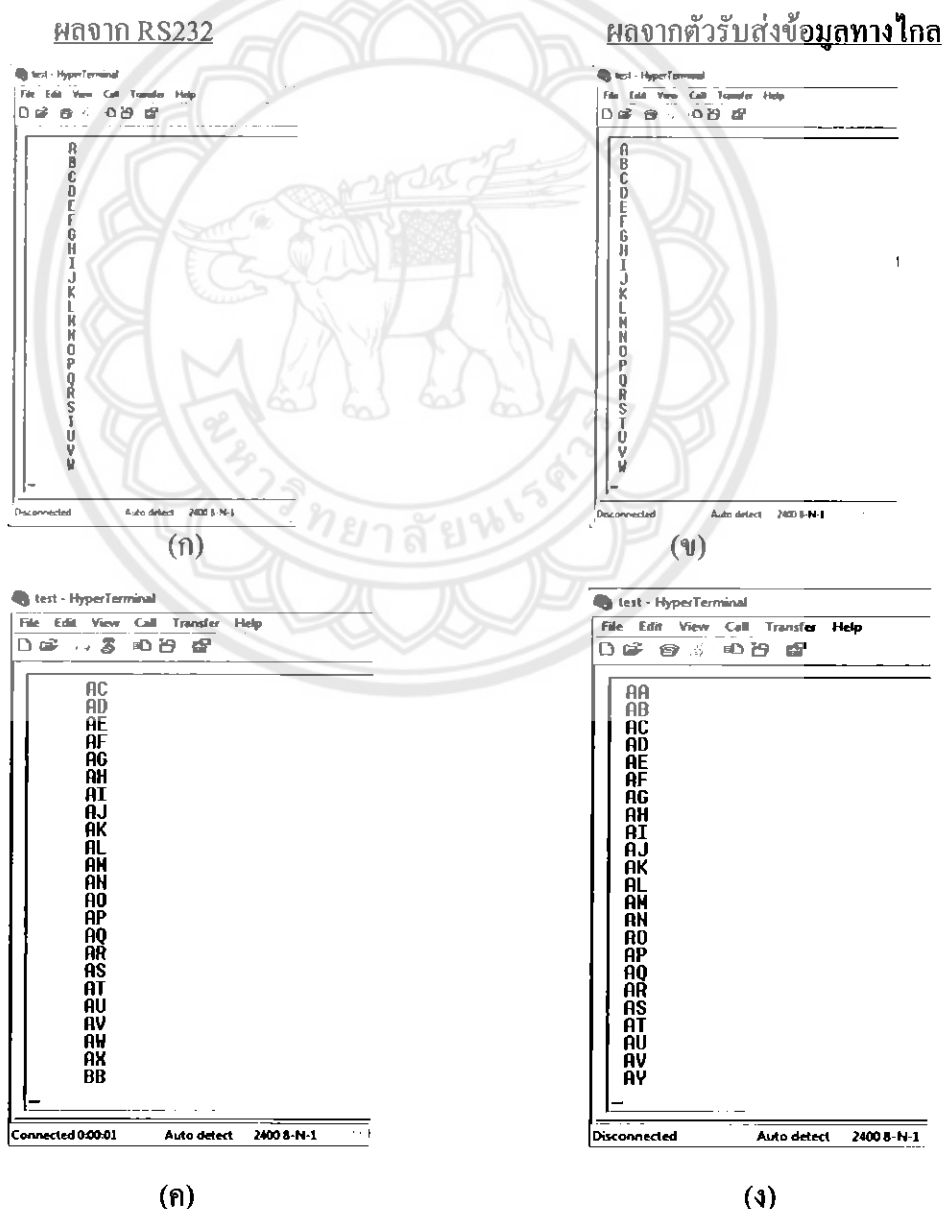
จากรูปที่ 4.9 นั้นที่ระยะเวลา 1 นาทีนั้น ใช้พื้นที่ 886 bytes เทียบเวลา 1 ชั่วโมง จะใช้พื้นที่ 53.16 KB และเทียบเวลา 1 วัน จะใช้พื้นที่ 1.28 MB จากนั้นภายใน 1 ปี จะใช้พื้นที่ 467.2 MB และสุดท้ายที่ความจุ SD card ขนาด 2 GB จะสามารถบันทึกข้อมูลได้เป็นระยะเวลาอย่างน้อย 4 ปี

จากผลการทดสอบที่ได้แสดงให้เห็นว่าระบบสามารถเขียนข้อมูลบันทึกลง SD card และบันทึกเป็นไฟล์ Microsoft Excel ได้อย่างถูกต้อง และเมื่อมีการเขียนไฟล์ครั้งใหม่ ข้อมูลใหม่จะบันทึกลงในไฟล์เดิม แต่ข้อมูลใหม่จะต่อจากข้อมูลเดิม ไม่มีการเขียนทับกัน โดยที่ค่าเวลาจะเริ่มนับจาก 0.00 ใหม่ และหากบันทึกข้อมูลทั้ง 4 แชนแนล ที่ SD card ขนาด 2 GB จะสามารถบันทึกข้อมูลได้ถึง 4 ปี เป็นอย่างน้อย

4.4 การทดสอบการรับส่งข้อมูลไร้สาย

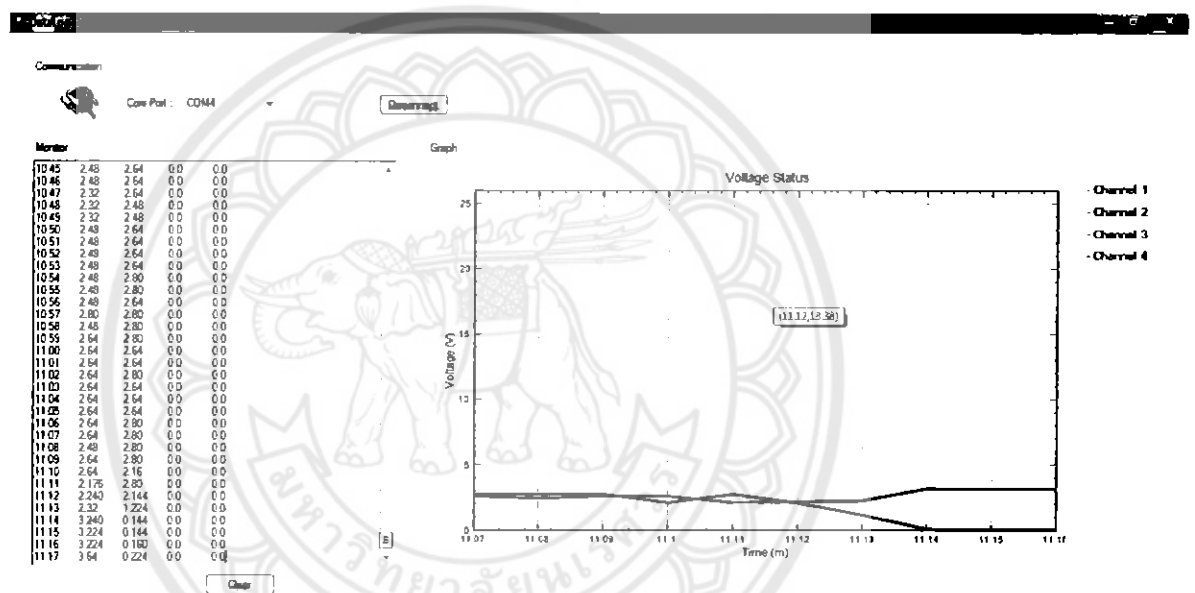
การทดสอบการรับส่งข้อมูลไร้สายนี้มีวัตถุประสงค์เพื่อต้องการตรวจสอบว่าข้อมูลที่รับส่งกันนั้นมีความผิดเพี้ยนคลาดเคลื่อนมากน้อยเพียงใด และระยะเวลาที่ระบบสามารถทำงานได้ค่อนข้างเสถียรอยู่ที่ระยะทางไกลสุดเท่าไร ซึ่งสามารถวัดระยะทางได้จากการวัดระยะทางของหน้าปัดรถมอเตอร์ไซด์ ซึ่งขั้นตอนการทดสอบและผลการทดสอบ มีดังต่อไปนี้

ในขั้นแรกการทดสอบจะทำการทดสอบระหว่างตัวรับส่งข้อมูลระยะไกล เทียบกับการทดสอบจากการส่งข้อมูลผ่าน Serial port (RS232) เข้าคอมพิวเตอร์โดยตรง โดยข้อมูลที่ส่งนั้นมีตั้งแต่หนึ่งไปมาก และระยะห่างระหว่างตัวรับและตัวส่งยังอยู่ในระดับใกล้ ซึ่งมีผลการทดสอบดังต่อไปนี้



4.5 การทดสอบส่วนติดต่อกับผู้ใช้ หรือ GUI

เป็นการทดสอบโปรแกรมติดต่อระหว่างผู้ใช้งานกับคอมพิวเตอร์ เพื่อให้ผู้ใช้งานสามารถตรวจสอบการทำงานของรถประหยัคพลังงานได้ในทันที โดยมีการแสดงผลได้ 4 แชนแนลทั้งข้อมูลตัวเลขและข้อมูลกราฟ ซึ่งการทดสอบในส่วนนี้นั้นจะทำการป้อนข้อมูลอินพุตเป็นแรงดันไฟฟ้าเข้ามา 2 แชนแนล ผ่านตัวต้านทานปรับค่าได้เพื่อให้ค่าที่อ่านเข้ามามีการเปลี่ยนแปลง ซึ่งผลการทดสอบเป็นดังต่อไปนี้



รูปที่ 4.8 ผลการทดสอบส่วนติดต่อกับผู้ใช้

จากผลการทดสอบส่วนติดต่อกับผู้ใช้นั้น จะเห็นได้ว่าข้อมูลที่อ่านเข้ามามีความสัมพันธ์กับผลของกราฟที่ได้ และเมื่อมีการเปลี่ยนแปลงของข้อมูลตัวเลข ข้อมูลกราฟก็มีการเปลี่ยนแปลงด้วย ดังนั้นสามารถสรุปได้ว่าการทดสอบส่วนติดต่อกับผู้ใช้สามารถใช้งานได้อย่างดี สามารถแสดงผลทั้งข้อมูลตัวเลขและข้อมูลกราฟได้อย่างถูกต้อง

จากผลการทดสอบทั้งหมดที่ผ่านมาสามารถสรุปผลโดยรวมได้ว่าระบบสามารถทำงานได้ตามวัตถุประสงค์ คือ สามารถรับค่าที่อ่านเข้ามาและบันทึกลง SD card ได้และสามารถส่งข้อมูลระยะไกลพร้อมแสดงผลได้ แต่ข้อมูลระยะไกลที่ได้ยังไม่เสถียรเท่าใดนัก ต้องมีการพัฒนาต่อไป ซึ่งผลการทดสอบที่สรุปได้ ปัญหาอุปสรรคที่พบ แนวทางแก้ไข ข้อเสนอแนะ และแนวทางในการพัฒนาต่อไปนั้น จะสรุปไว้ในบทถัดไป

บทที่ 5

สรุปผลและข้อเสนอแนะ

ในบทนี้เป็นการสรุปผลการดำเนินโครงการ ปัญหาอุปสรรคพร้อมแนวทางแก้ไขและข้อเสนอแนะ รวมทั้งแนวทางในการพัฒนาต่อของโครงการนี้ต่อไป

5.1 สรุปผลการดำเนินโครงการ

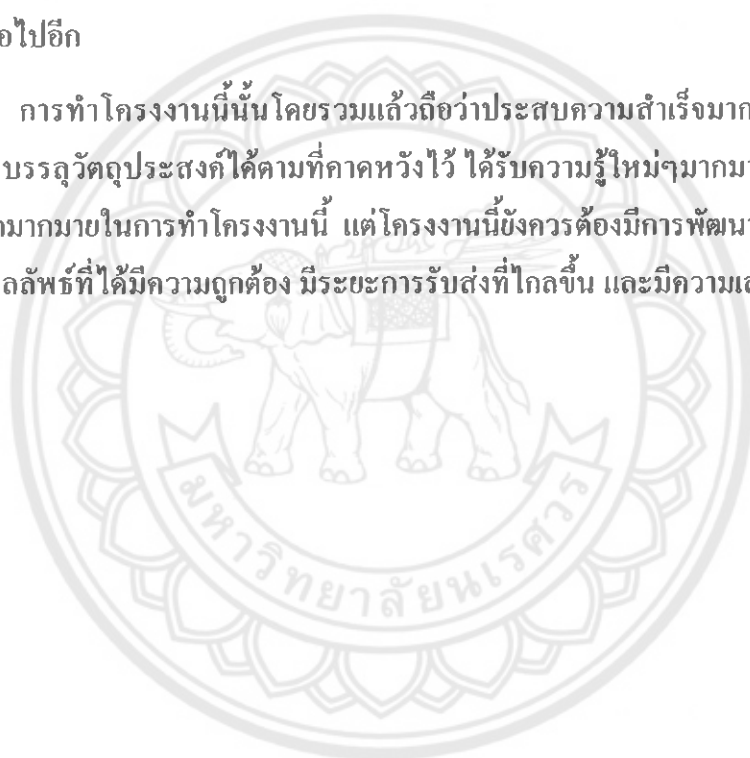
โครงการระบบจัดเก็บข้อมูลและติดตามผลแบบออนไลน์สำหรับรถยนต์ประหยัดพลังงานนี้ เป็นโครงการที่จัดทำขึ้นโดยมีวัตถุประสงค์เพื่อสร้างอุปกรณ์ที่สามารถจัดเก็บข้อมูลลงใน SD card ได้ และสามารถส่งข้อมูลผ่านระบบไร้สายมายังคอมพิวเตอร์ได้พร้อมๆกัน ซึ่งได้ทำการทดสอบกับระบบจำลองของรถประหยัดพลังงานต้นแบบ เพื่อเก็บข้อมูลที่เป็นแรงดัน ไฟฟ้าและความเร็วรถ

สำหรับการออกแบบระบบนั้นแบ่งออกเป็น 4 ส่วน คือ ส่วนแรกเป็นส่วนอินพุต ซึ่งจะรับข้อมูลที่เป็นแรงดัน ไฟฟ้า และความเร็วล้อเข้ามา ส่วนที่สองเป็นส่วนประมวลผล โดยได้ใช้ไมโครคอนโทรลเลอร์ที่มีประสิทธิภาพสูงและทันสมัยของตระกูล AVR เบอร์ ATmega32 มาทำหน้าที่ประมวลผลข้อมูล ใช้ภาษา C ในการโปรแกรมผ่านคอมไพเลอร์ AVR Studio เมื่อมีการประมวลผลเรียบร้อยแล้วจึงส่งข้อมูลไปยังส่วนต่อไป ส่วนที่สามคือส่วนเอาต์พุต ซึ่งมีทั้งระบบจัดเก็บข้อมูลลงใน SD card และระบบรับส่งข้อมูลไร้สาย ระบบ FM ที่มีชื่อว่า Futaba ทำการส่งข้อมูลจากรถไปยังคอมพิวเตอร์ และส่วนสุดท้ายคือส่วนติดต่อกับผู้ใช้ ใช้ภาษา Visual Basic ในการพัฒนา ภายในโปรแกรมมีการรับข้อมูลจากตัวรับข้อมูลระยะไกลเข้ามา และสามารถแสดงผลทั้งข้อมูลตัวเลขและข้อมูลกราฟได้

จากผลการทดสอบในบทที่ 4 ในแต่ละส่วนนั้น จะเห็นว่าความคลาดเคลื่อนในการอ่านค่าแรงดันไฟฟ้าเข้ามานั้นยังค่อนข้างสูงถึงร้อยละ 30.93 เพราะเหตุผลทั้งด้านฮาร์ดแวร์คือตัวต้านทานที่ไม่มีขยตามท้องตลาด และเหตุผลทางด้านซอฟต์แวร์คือตัวสับประสิทธิ์ที่ต้องนำไปคูณในโปรแกรมนั้น ไม่สามารถใช้เลขที่มีทศนิยมได้ เพราะในการแสดงผลใช้ตัวแปร Character ต่อมาเป็นการทดสอบการวัดความเร็วจากแบบจำลองที่ใช้ดีซีมอเตอร์ ผลที่ได้คือระบบสามารถแสดงผลออกมาได้ สามารถบ่งบอกความเร็วได้ ถัดมาคือการทดสอบการบันทึกข้อมูลลงสู่ SD card ซึ่งสามารถบันทึกลง SD card ได้จริงเป็นไฟล์ Microsoft Excel สามารถบันทึกข้อมูลจนเต็มความจุ 2 GB ได้ในระยะเวลาจนถึง 4 ปี นอกจากนี้สำหรับการทดสอบการรับส่งข้อมูลไร้สาย

ผลการทดลองบ่งชี้ว่ายิ่งขนาดข้อมูลที่รับส่งมีปริมาณมาก ระยะทางในการรับส่งข้อมูลให้มีค่าที่เสถียรนั้นย่อมน้อยตาม ในทางกลับกันหากขนาดข้อมูลมีปริมาณน้อย ระยะทางในการรับส่งข้อมูลย่อมไกลขึ้น ซึ่งระยะทางที่ระบบสามารถทำงานได้เมื่อส่งข้อมูลปริมาณมากๆนั้น ในขณะที่อยู่ที่รัศมีเพียง 10 เมตรเท่านั้น ในขณะที่เมื่อส่งข้อมูลปริมาณน้อยๆ อยู่ที่รัศมี 100 เมตร ดังนั้นจึงควรใช้ระบบรับส่งข้อมูลไร้สายนี้เฉพาะกับข้อมูลที่สำคัญๆ นอกจากนี้ผลการทดสอบโปรแกรมติดต่อกับผู้ใช้นั้น ผลลัพธ์คือสามารถทำการแสดงผลได้ดีทั้งข้อมูลตัวเลขและข้อมูลกราฟ และสุดท้ายหากเปรียบเทียบผลลัพธ์ที่ได้จากการบันทึกข้อมูลลง SD card และการส่งข้อมูลระยะไกลแล้ว ข้อมูลจาก SD card จะมีความถูกต้องและความน่าเชื่อถือมากกว่า สามารถจัดเก็บข้อมูลปริมาณมากๆได้ ส่วนระบบรับส่งข้อมูลไร้สายนั้นยังมีข้อจำกัดอยู่มาก ยังคงต้องมีการพัฒนาต่อไปอีก

การทำโครงการนี้นั้นโดยรวมแล้วถือว่าประสบความสำเร็จมากในระดับหนึ่ง เพราะสามารถบรรลุวัตถุประสงค์ได้ตามที่คาดหวังไว้ ได้รับความรู้ใหม่ๆมากมาย แม้จะมีปัญหาและอุปสรรคมากมายในการทำโครงการนี้ แต่โครงการนี้ยังควรต้องมีการพัฒนาต่ออีกมาก เพื่อให้ผลลัพธ์ที่ได้มีความถูกต้อง มีระยะการรับส่งที่ไกลขึ้น และมีความเสถียรมากยิ่งขึ้น ไปอีก



5.2 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ

5.2.1 ปัญหาด้านฮาร์ดแวร์

ตารางที่ 5.1 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ด้านฮาร์ดแวร์

ปัญหา อุปสรรค	แนวทางแก้ไข ข้อเสนอแนะ
1. อุปกรณ์อิเล็กทรอนิกส์ซื้อได้ยาก และราคาค่อนข้างแพง	1. สั่งซื้ออุปกรณ์อิเล็กทรอนิกส์ทางอินเทอร์เน็ต ซึ่งต้องแลกกับค่าขนส่งที่เพิ่มขึ้น การแก้ปัญหาที่ดีคือควรวางแผนในการซื้ออุปกรณ์หลายๆชิ้นพร้อมกันเพื่อลดค่าใช้จ่าย และควรซื้ออุปกรณ์ที่เสียบางอย่างสำรองเอาไว้ด้วย เช่น ไมโครคอนโทรลเลอร์ ไอซีต่างๆ
2. ไมโครคอนโทรลเลอร์ MCS-51 ที่ใช้ในครั้งแรก ไม่สนับสนุนการเขียนโปรแกรมให้ใช้งานกับ SD card เท่าที่ควร เพราะเป็นระบบแบบเก่า	2. เปลี่ยนไปใช้ไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATmega32 ที่สนับสนุนการเขียนโปรแกรมกับ SD card
3. อุปกรณ์การรับส่งข้อมูลระยะไกล (Futaba) มีสัญญาณรบกวนอยู่มากเพราะเป็นการสื่อสารระบบ FM ทำให้ข้อมูลมีความคลาดเคลื่อน โดยเฉพาะเสาอากาศที่ถูกสัญญาณรบกวนได้ง่าย	3. ควรเปลี่ยนไปใช้อุปกรณ์รับส่งข้อมูลระยะไกล (Futaba) ที่ย่านความถี่สูงกว่านี้ (ตัวเดิม 72 MHz) โดยอาจเปลี่ยนไปใช้ที่ย่านความถี่ 2.4 GHz ที่มีสัญญาณรบกวนน้อยกว่าและมีประสิทธิภาพดีกว่า
4. แหล่งจ่ายกระแสไฟฟ้าเพื่อเลี้ยงวงจรไม่พอเนื่องจากเสาอากาศส่งสัญญาณ FM ใช้กำลังส่งค่อนข้างมาก ซึ่งขึ้นอยู่กับความยาวของเสา	4. เพิ่มแหล่งจ่ายให้กับวงจรเป็น 2 แหล่งเพื่อจ่ายให้กับตัวส่งสัญญาณระยะไกล โดยตรงหนึ่งแหล่ง แต่ต้องแลกกับน้ำหนักและขนาดที่เพิ่มขึ้น
5. ไม่สามารถหาอุปกรณ์บางอย่างที่มีค่าในอุดมคติที่ต้องการได้	5. ใช้อุปกรณ์ที่มีค่าใกล้เคียงแทนค่าในอุดมคติและส่วนภายในโปรแกรมนั้นต้องใช้ค่าสัมประสิทธิ์คุณค่าที่อ่านมาได้

ปัญหา อุปสรรค	แนวทางแก้ไข ข้อเสนอแนะ
6. ไม่มีอุปกรณ์ที่สำคัญในการทดสอบกับระบบ เช่น Hub motor ซึ่งเป็นมอเตอร์ที่ใช้จริงในล้อรถประหยัดพลังงาน	6. ใช้การจำลองสถานการณ์และการเทียบสัดส่วนแทน เช่น ใช้คีมอเตอร์แทน Hub motor แล้วนำค่ามาเทียบสัดส่วนกัน
7. ก่อนการถอด SD card ออกจากโมดูล จำเป็นต้อง Reset วงจรก่อน ไมเช่นนั้นจะทำให้ SD card ใช้การไม่ได้	7. ศึกษาการใช้งาน SD card ให้ละเอียด และต้องไม่ลืม Reset วงจร ก่อนถอด SD card ทุกครั้ง
8. บันทึกลับ (Burn) โปรแกรมในขณะที่เชื่อมต่อกับระบบรับส่งข้อมูลระยะไกลไม่ได้	8. ขณะที่บันทึก (Burn) โปรแกรมลงไมโครคอนโทรลเลอร์ ต้องตัดการทำงานจากระบบรับส่งข้อมูลไร้สายก่อน เพราะระบบรับส่งข้อมูลไร้สายนั้นดึงไฟจากแหล่งจ่ายมาก ทำให้มีแหล่งจ่ายไม่พอสำหรับการบันทึก (Burn)

5.2.2 ปัญหาด้านซอฟต์แวร์

ตารางที่ 5.2 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ด้านซอฟต์แวร์

ปัญหา อุปสรรค	แนวทางแก้ไข ข้อเสนอแนะ
1. เนื่องจากโปรแกรมที่เขียนขึ้นมีขนาดใหญ่ โดยเฉพาะไลบรารีของ SD card ทำให้ Flash memory ของไมโครคอนโทรลเลอร์ไม่เพียงพอ จึงต้องมีการเปลี่ยนเบอร์ไมโครคอนโทรลเลอร์จาก ATmega88 เป็น ATmega32 มีผลให้เสียเวลาในการทำ ความเข้าใจกับการทำงานของไมโครคอนโทรลเลอร์ตัวใหม่มากยิ่งขึ้นไปอีก	1. ต้องวางแผนและหาข้อมูลของอุปกรณ์ให้คิก่อนซื้อ และตรวจสอบโปรแกรมที่เขียนขึ้นก่อนซื้ออุปกรณ์ว่ามีพื้นที่เท่าไร หรือหากไม่ทราบว่าโปรแกรมที่เขียนจะสิ้นสุดลงที่พื้นที่เท่าไร ควรจะซื้อไมโครคอนโทรลเลอร์ที่มีประสิทธิภาพสูง มาใช้งานตั้งแต่แรก

ปัญหา อุปสรรค	แนวทางแก้ไข ข้อเสนอแนะ
2. Flash memory ของไมโครคอนโทรลเลอร์ ไม่เพียงพอต่อการเก็บโปรแกรมของระบบ SD card,ระบบรับส่งข้อมูลทางไกลและระบบวัดความเร็ว เข้าด้วยกันทั้งหมดได้	2. โปรแกรมเฉพาะระบบ SD card และระบบรับส่งข้อมูลทางไกล ส่วนระบบวัดความเร็ว ไม่ได้โปรแกรมเข้าไปพร้อมกัน จึงควรเปลี่ยนไปใช้ไมโครคอนโทรลเลอร์ที่ดีกว่านี้
3. การคำนวณค่า Voltage ที่ต้องการนำไปแสดงผล มีความคลาดเคลื่อนจากความเป็นจริงอยู่พอสมควร เพราะไม่สามารถนำค่าที่มีจุดทศนิยมมาคำนวณเป็นค่าสัมประสิทธิ์ได้ ต้องใช้จำนวนเต็มแทนเท่านั้น	3. ต้องศึกษาการเขียนโปรแกรมภาษา C กับไมโครคอนโทรลเลอร์ตระกูล AVR ให้เข้าใจอย่างถ่องแท้ เพื่อหาวิธีที่ทำให้โปรแกรมทำงานได้อย่างถูกต้องมากขึ้น

5.2.3 ปัญหาด้านอื่นๆ

ตารางที่ 5.3 ปัญหา อุปสรรค และแนวทางแก้ไข ข้อเสนอแนะ ในด้านอื่นๆ

ปัญหา อุปสรรค	แนวทางแก้ไข ข้อเสนอแนะ
1. ค่าใช้จ่ายในการทำโครงงานค่อนข้างสูง	1. หากเป็นไปได้ทางคณะวิศวกรรมศาสตร์ ควรสนับสนุนงบประมาณการทำโครงงานให้มากขึ้น
2. เวลาว่างของคณะผู้จัดทำโครงงานไม่ตรงกัน ทำให้กระทบกับช่วงเวลาที่ทำโครงงาน	2. ควรปรับการสอบวิชาเอกให้จบก่อนเดือนมกราคม โดยที่อาจารย์อาจนัดนิสิตเรียนก่อนตั้งแต่ช่วงเดือนตุลาคม จากนั้นตั้งแต่เดือนมกราคมเป็นต้นไป เป็นระยะเวลาที่ให้ทำโครงงานอย่างเต็มที่ จะสามารถแก้ปัญหาเรื่องเวลาและเรื่องโครงงานเสร็จไม่ทันลงได้มาก
3. ข้อมูลภาษาไทยหายาก ทำให้ประสบปัญหาในการหาข้อมูลภาษาอังกฤษ	3. ฝึกทักษะภาษาอังกฤษให้อยู่ในระดับที่ดี และต้องหมั่นหาข้อมูลสนับสนุนการทำโครงงานอยู่เสมอ

5.3 แนวทางในการพัฒนาต่อไป

- 1) ในการพัฒนาต่อในอนาคตอาจเปลี่ยนไปใช้ไมโครคอนโทรลเลอร์ เบอร์ที่มีประสิทธิภาพสูงขึ้น มีหน่วยความจำมากขึ้นหรืออาจเชื่อมต่อหน่วยความจำเพิ่มขึ้น เพื่อการพัฒนาต่อยอดให้สามารถเก็บข้อมูลได้หลากหลายขึ้น หรือสามารถเพิ่มส่วนประกอบอื่นๆให้มากขึ้นอีกได้
- 2) หากมีงบประมาณมากพอควรเปลี่ยนอุปกรณ์รับส่งข้อมูลระยะไกล ไปใช้ย่านความถี่ที่สูงขึ้น เพื่อลดสัญญาณรบกวน เพิ่มเสถียรภาพของระบบ และเพิ่มระยะทางในการรับส่งข้อมูล
- 3) พัฒนาระบบให้ความคลาดเคลื่อนในการอ่านค่าแรงดันไฟฟ้าลดลงจาก 30.93 % ให้น้อยลงจนอยู่ในระดับที่พอรับได้
- 4) พัฒนาการรับส่งข้อมูลให้มีการเข้ารหัส เพื่อเพิ่มขนาดการรับส่งและความถูกต้องของข้อมูล
- 5) หากมีโอกาที่เอื้ออำนวยควรนำอุปกรณ์ที่สร้างขึ้นนี้ ไปทดสอบกับรถประหยัคพลังงานจริง เพื่อที่จะสามารถนำไปใช้ประโยชน์ได้อย่างเป็นรูปธรรม

เอกสารอ้างอิง

- [1] Wisitsak. (ไม่ระบุวันที่แต่ง). การส่งข้อมูล Analog และ Digital. สืบค้นเมื่อ 4 กรกฎาคม 2553, จาก <http://www.center072.co.cc/wisitsaks/wp.../การส่งข้อมูล-Analog-และ-Digital.docx>
- [2] ไม่ระบุผู้แต่ง. (6 พฤษภาคม 2552). Modulation/Demodulation. สืบค้นเมื่อ 6 กรกฎาคม 2553, จาก <http://www.numesai.com/Modulation/Demodulation.html>
- [3] ประจิม พลังสันติกุล. (2549). การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ AVR ด้วยภาษา C กับ WinAVR (C Compiler). กรุงเทพฯ : แอพซอฟต์เทค
- [4] Atmel Corporation. (2003). 8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash. Retrieved January 10 2011, from http://www.atmel.com/dyn/resources/prod_documents/doc2503.pdf
- [5] ไม่ระบุผู้แต่ง. (ไม่ระบุวันที่แต่ง). MAX232. สืบค้นเมื่อ 7 กรกฎาคม 2553, จาก <http://www.thaimicrotron.com/CCS-628/Reference/MAX232.htm>
- [6] อติศักดิ์ ชิมะวงศ์. (8 มกราคม 2543). พอร์ตอนุกรม. สืบค้นเมื่อ 7 กรกฎาคม 2553, จาก http://www.adisak51.com/page06_1.html
- [7] ไม่ระบุผู้แต่ง. (ไม่ระบุวันที่แต่ง). การใช้งานพอร์ตอนุกรม RS232. สืบค้นเมื่อ 7 กรกฎาคม 2553, จาก www.thaimicrotron.com/CCS-628/Reference/RS232.htm
- [8] ไม่ระบุผู้แต่ง. (ไม่ระบุวันที่แต่ง). วงจรแบ่งแรงดัน. สืบค้นเมื่อ 8 กรกฎาคม 2553, จาก <http://www.sptc.ac.th/prapruet/dc/tummmmmmmmmmmmmmmmmmm%20%20nerha/nerha8.htm>
- [9] ไม่ระบุผู้แต่ง. (ไม่ระบุวันที่แต่ง). Data logging system. สืบค้นเมื่อ 9 กรกฎาคม 2553, จาก <http://www.scribd.com/doc/Presentation-DataLogger-Know2Pro-com>
- [10] Dharmani. (January 31,2009). SD/SDHC Card Interfacing with ATmega8 /32 (FAT32 implementation). Retrieved January 10 2011, from <http://www.dharmanitech.com/2009/01/sd-card-interfacing-with-atmega8-fat32.html>
- [11] ไม่ระบุผู้แต่ง. (ไม่ระบุวันที่แต่ง). วงจรแปลงไฟ และจ่ายไฟ. สืบค้นเมื่อ 9 กรกฎาคม 2553, จาก <http://www.mind-tek.net/8051.php>



ภาคผนวก ก

รายละเอียดของไมโครคอนโทรลเลอร์ ATmega32

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 8-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 32K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 1024 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 2K Byte Internal SRAM
 - Programming Lock for Software Security
- JTAG (IEEE Std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timers/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 3 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain of 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 48-pin PDIP, 44-lead TQFP, and 44-pad MLF
- Operating Voltages
 - 2.7 - 5.5V for ATmega32L
 - 4.5 - 5.5V for ATmega32
- Speed Grades
 - 0 - 8 MHz for ATmega32L
 - 0 - 16 MHz for ATmega32
- Power Consumption at 1 MHz, 3V, 25°C for ATmega32L
 - Active: 1.1 mA
 - Idle Mode: 0.35 mA
 - Power-down Mode: < 1 µA



**8-bit AVR®
Microcontroller
with 32K Bytes
In-System
Programmable
Flash**

**ATmega32
ATmega32L**

Preliminary

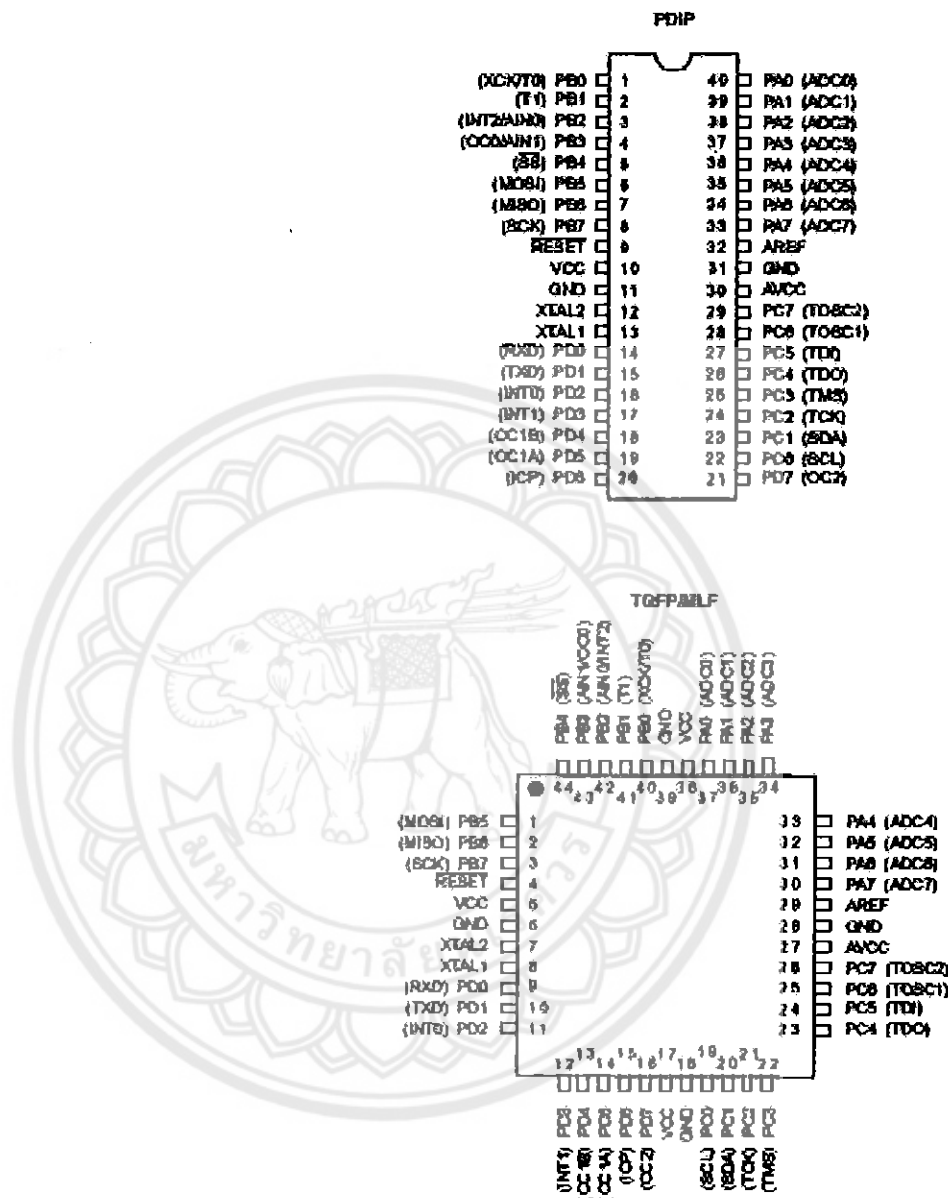
2007-AVR-1203





Pin Configurations

Figure 1. Pinouts ATmega32



Disclaimer

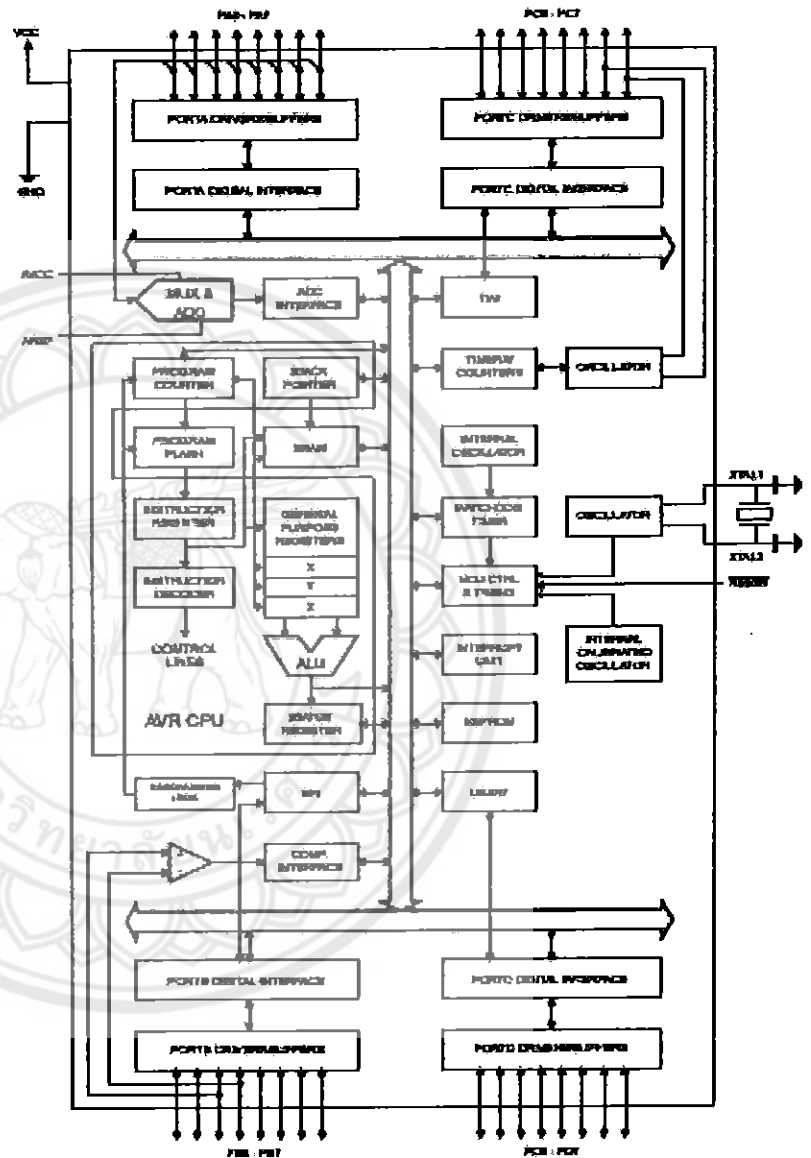
Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

Overview

The ATmega32 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



Pin Descriptions

VCC	Digital supply voltage.
GND	Ground.
Port A (PA7..PA0)	<p>Port A serves as the analog inputs to the A/D Converter.</p> <p>Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p>
Port B (PB7..PB0)	<p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port B also serves the functions of various special features of the ATmega32 as listed on page 55.</p>
Port C (PC7..PC0)	<p>Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.</p> <p>The TD0 pin is tri-stated unless TAP states that shift out data are entered.</p> <p>Port C also serves the functions of the JTAG interface and other special features of the ATmega32 as listed on page 58.</p>
Port D (PD7..PD0)	<p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega32 as listed on page 60.</p>
RESET	Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 35. Shorter pulses are not guaranteed to generate a reset.
XTAL1	Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.
XTAL2	Output from the inverting Oscillator amplifier.
AVCC	AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V _{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V _{CC} through a low-pass filter.
AREF	AREF is the analog reference pin for the A/D Converter.

Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
\$3F (\$3F)	SREG	I	T	H	S	V	N	Z	C	8
\$3E (\$3E)	SPH					SP11	SP10	SP9	SP8	10
\$3D (\$3D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	10
\$3C (\$3C)	OCR0	Timer/Counter0 Output Compare Register								80
\$3B (\$3B)	GCR	INT1	INT0	INT2				WSEL	WCE	45, 65
\$3A (\$3A)	GFR	INTF1	INTF0	INTF2						66
\$39 (\$39)	TWSK	OCIE2	TOIE2	TCIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	80, 110, 128
\$38 (\$38)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	81, 111, 128
\$37 (\$37)	SPMCR	SPMIE	RWWSB		RWWSRE	BLBSET	POWRT	PGERS	SPMEN	245
\$36 (\$36)	TWCR	TWYH1	TWEA	TWSTA	TWSTD	TWYC	TWEN		TWIE	175
\$35 (\$35)	MCUCR	SE	SM2	SM1	SMD	ISC11	ISC10	ISC01	ISC00	38, 64
\$34 (\$34)	MCUCSR	JTD	ISC2		JTRF	WDRF	BORF	EXTRF	PORF	38, 65, 226
\$33 (\$33)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	78
\$32 (\$32)	TCNT0	Timer/Counter0 (8 Bits)								80
\$31 ^(*) (\$31 ^(*))	OSCCAL	Oscillator Calibration Register								28
	ODCR	On-Chip Debug Register								222
\$30 (\$30)	SFIOR	ADTS2	ADTS1	ADTS0		ACME	PUD	PSR2	PSR10	54, 83, 129, 156, 216
\$2F (\$2F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	105
\$2E (\$2E)	TCCR1B	ICNC1	ICES1		WGM13	WGM12	CS12	CS11	CS10	108
\$2D (\$2D)	TCNT1H	Timer/Counter1 - Counter Register High Byte								109
\$2C (\$2C)	TCNT1L	Timer/Counter1 - Counter Register Low Byte								109
\$2B (\$2B)	OCR1AH	Timer/Counter1 - Output Compare Register A High Byte								109
\$2A (\$2A)	OCR1AL	Timer/Counter1 - Output Compare Register A Low Byte								109
\$29 (\$29)	OCR1BH	Timer/Counter1 - Output Compare Register B High Byte								109
\$28 (\$28)	OCR1BL	Timer/Counter1 - Output Compare Register B Low Byte								109
\$27 (\$27)	ICR1H	Timer/Counter1 - Input Capture Register High Byte								110
\$26 (\$26)	ICR1L	Timer/Counter1 - Input Capture Register Low Byte								110
\$25 (\$25)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	123
\$24 (\$24)	TCNT2	Timer/Counter2 (8 Bits)								125
\$23 (\$23)	OCR2	Timer/Counter2 Output Compare Register								125
\$22 (\$22)	ASSR					AS2	TCH2UB	OCR2UB	TCR2UB	126
\$21 (\$21)	WDTOR				WDTOE	WDE	WDF2	WDF1	WDF0	40
\$20 ^(*) (\$20 ^(*))	UBRRH	URSEL					UBRRq(1:8)			162
	UCSRC	URSEL	URSEL	UPM1	UPM0	USBS	UCS21	UCS20	UCPOL	160
\$1F (\$1F)	EEARH							EEAR8	EEAR0	17
\$1E (\$1E)	EEARL	EEPROM Address Register Low Byte								17
\$1D (\$1D)	EEDR	EEPROM Data Register								17
\$1C (\$1C)	EEDR					EERIE	EEMWE	EEWE	EERE	17
\$1B (\$1B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	62
\$1A (\$1A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	62
\$19 (\$19)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	62
\$18 (\$18)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	62
\$17 (\$17)	DDRB	DOB7	DOB6	DOB5	DOB4	DOB3	DOB2	DOB1	DOB0	62
\$16 (\$16)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	63
\$15 (\$15)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	63
\$14 (\$14)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	63
\$13 (\$13)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	63
\$12 (\$12)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	63
\$11 (\$11)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	63
\$10 (\$10)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	63
\$0F (\$0F)	SPDR	SPI Data Register								136
\$0E (\$0E)	SPSR	SPIF	WCOL						SPDIX	136
\$0D (\$0D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	134
\$0C (\$0C)	UDR	USART I/O Data Register								157
\$0B (\$0B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	UZX	MPCK	158
\$0A (\$0A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCS22	RXB8	TXB8	159
\$09 (\$09)	UBRRL	USART Baud Rate Register Low Byte								162
\$08 (\$08)	ACSR	ACD	ACBG	ACD	ACI	ACIE	ACIC	ACIS1	ACIS0	197
\$07 (\$07)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	212
\$06 (\$06)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADRE	ADPS2	ADPS1	ADPS0	214
\$05 (\$05)	ADCH	ADC Data Register High Byte								215
\$04 (\$04)	ADCL	ADC Data Register Low Byte								215
\$03 (\$03)	TWDR	Two-wire Serial Interface Data Register								177
\$02 (\$02)	TWAR	TWAS	TWAS	TWAI	TWAS	TWA2	TWA1	TWA0	TWGOE	177
\$01 (\$01)	TWSCR	TWS7	TWS6	TWS5	TWS4	TWS3		TWSP1	TWSP0	176
\$00 (\$00)	TWBR	Two-wire Serial Interface Baud Rate Register								175

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
8	2.7 - 5.5V	ATmega32L-8AC ATmega32L-8PC ATmega32L-8MC	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega32L-8AI ATmega32L-8PI ATmega32L-8MI	44A 40P6 44M1	Industrial (-40°C to 85°C)
16	4.5 - 5.5V	ATmega32-16AC ATmega32-16PC ATmega32-16MI	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega32-16AI ATmega32-16PI ATmega32-16MC	44A 40P6 44M1	Industrial (-40°C to 85°C)



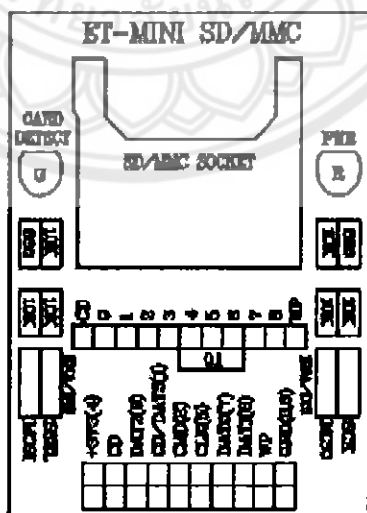


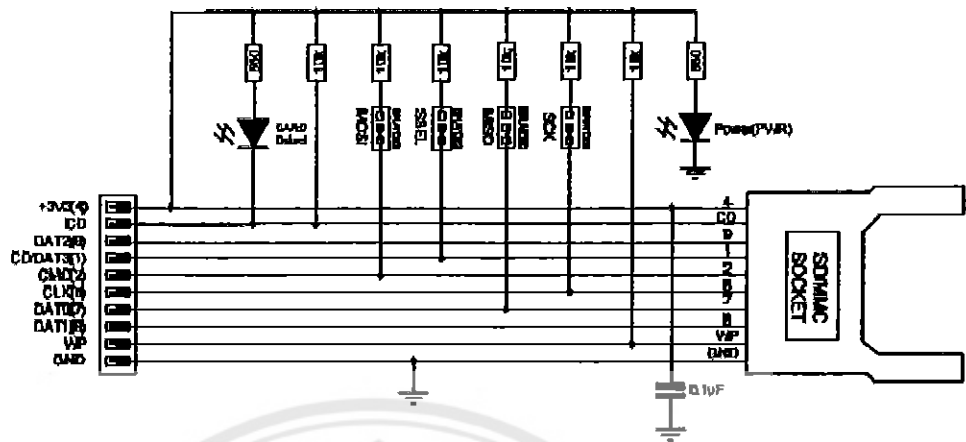
ชุดเชื่อมต่อหน่วยความจำ SD/MMC CARD (ET-MINI SD/MMC)

ET-MINI SD/MMC คือ ชุดอุปกรณ์สำหรับใช้ในการเชื่อมต่อกับอุปกรณ์หน่วยความจำ SD และ MMC CARD เช่น การเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับหน่วยความจำ Memory Card(SD/MMC) เป็นต้น ภายในชุดประกอบด้วย ช่อง (Socket) สำหรับใส่ CARD ประเภท SD และ MMC โดยจะมีการจัดขาสัญญาณออกตามที่ Connector Pin เพื่อให้สามารถนำไปต่อใช้งานได้สะดวก นอกจากนี้ยังมีวงจรต่างๆ เช่น วงจรตรวจสอบสถานะการเสียบการ์ด (CARD DETECT) และวงจรพูลอัพ (Pull-Up) สัญญาณต่างๆ อีกด้วย

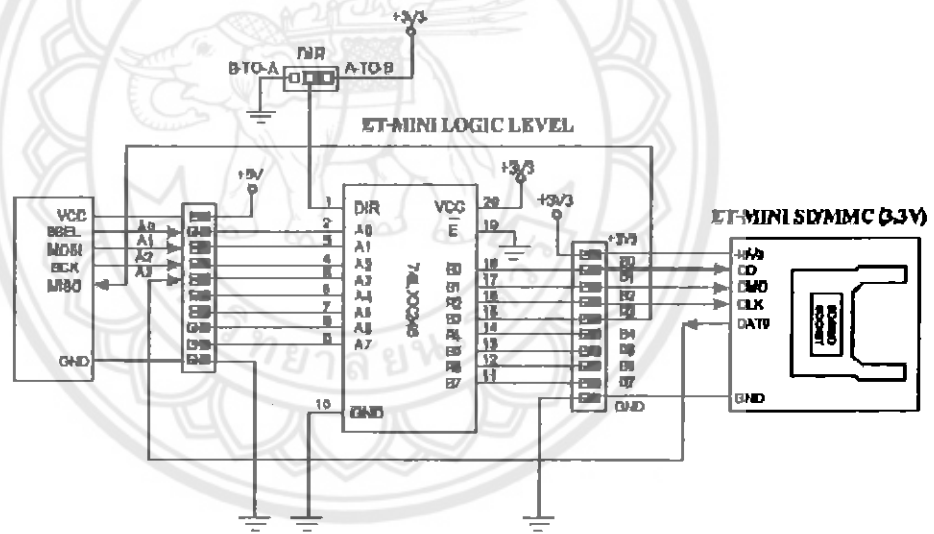
คุณสมบัติบอร์ด ET-MINI SD/MMC

- รองรับการ์ดประเภท SD และ MMC
- สามารถเลือก ใช้ (Enable) หรือ ไม่ใช้ (Disable) วงจรพูลอัพ(Pull-Up)สัญญาณต่างๆ ได้
- สามารถแสดงสถานะการเสียบการ์ด (CARD DETECT) แสดงผล โดย LED และ ให้สัญญาณเอาต์พุตออกที่ขาสัญญาณ CD โดยมีคุณสมบัติดังนี้
 CD = 1 คือ ไม่มี Card
 CD = 0 คือ มีการ์ด
- สามารถแสดงสถานะของสวิตช์ Write Protection บน SD/MMC CARD ได้ โดยจะมีสัญญาณเอาต์พุตออกที่ขาสัญญาณ WP ดังนี้
 WP = 1 คือ ตำแหน่งของสวิตช์ Write Protection อยู่ที่ตำแหน่ง OFF
 WR = 0 คือ ตำแหน่งของสวิตช์ Write Protection อยู่ที่ตำแหน่ง ON





ตัวอย่าง การเชื่อมต่อสัญญาณระหว่างไมโครคอนโทรลเลอร์ (อุปกรณ์ 5V) กับ SD/MMC CARD (อุปกรณ์ 3V)
 โดยใช้ชุด ET-MINI LOGIC LEVEL และ ET-MINI SD/MMC ในโหมด SPI



รายละเอียดขาสัญญาณของ SD CARD

การเชื่อมต่อกับ SD CARD สามารถทำได้สองแบบ ก็คือ การเชื่อมต่อแบบ SD MODE และ การเชื่อมต่อแบบ SPI MODE ดังรายละเอียดตามตารางต่อไปนี้

- รายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SD MODE

Pin No.	Name	Type ¹	Description
SD Mode			
1	CD/DAT3 ²	IO ² , PP	Card detect/Data line [Bit 3]
2	CMD	IO, PP	Command/Response
3	V _{cc}	S	Supply voltage ground
4	V _{cc}	S	Supply voltage
5	CLK	I	Clock
6	V _{cc}	S	Supply voltage ground
7	DAT0	IO, PP	Data line [Bit 0]
8	DAT1	IO, PP	Data line [Bit 1]
9	DAT2	IO, PP	Data line [Bit 2]

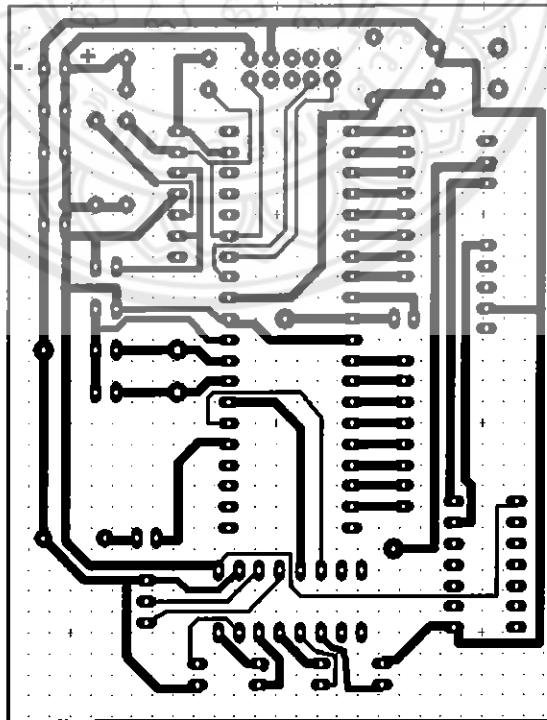
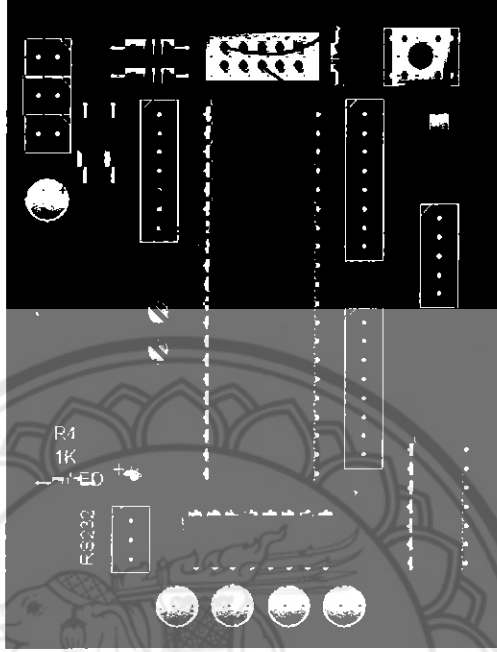
- รายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SPI MODE

SPI Mode			
1	CS	I	Chip Select (active low)
2	DataIn	I	Host-to-card Commands and Data
3	V _{cc}	S	Supply voltage ground
4	V _{cc}	S	Supply voltage
5	CLK	I	Clock
6	V _{cc}	S	Supply voltage ground
7	DataOut	O	Card-to-host Data and Status
8	RSV ⁴	—	Reserved
9	RSV ⁴	—	Reserved

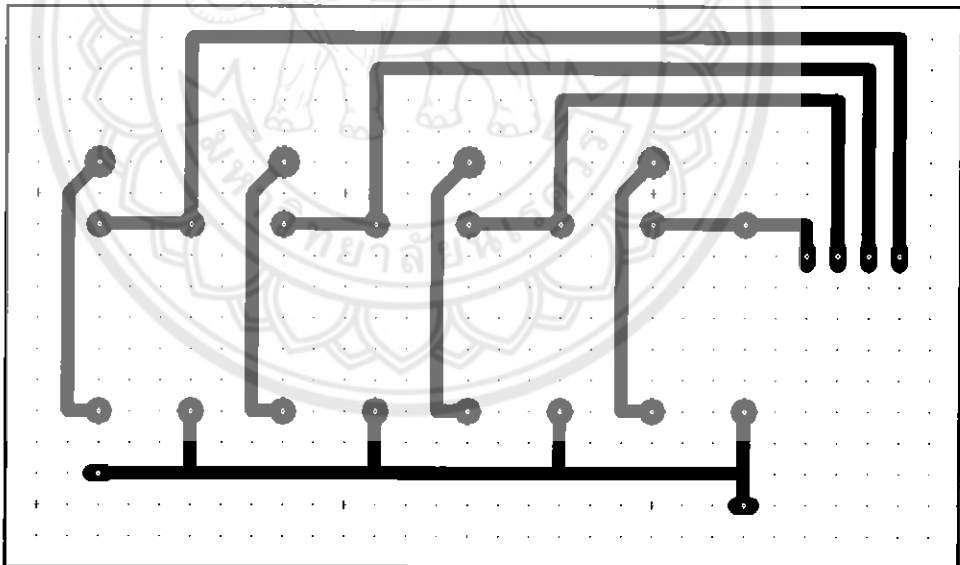
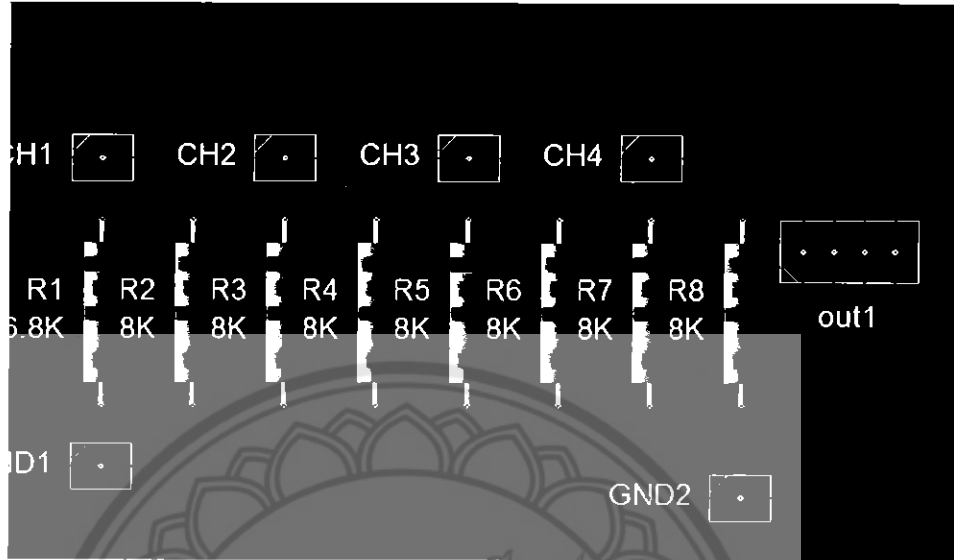
ภาคผนวก ค
ลายวงจรต่างๆ ที่ใช้ในโครงการ



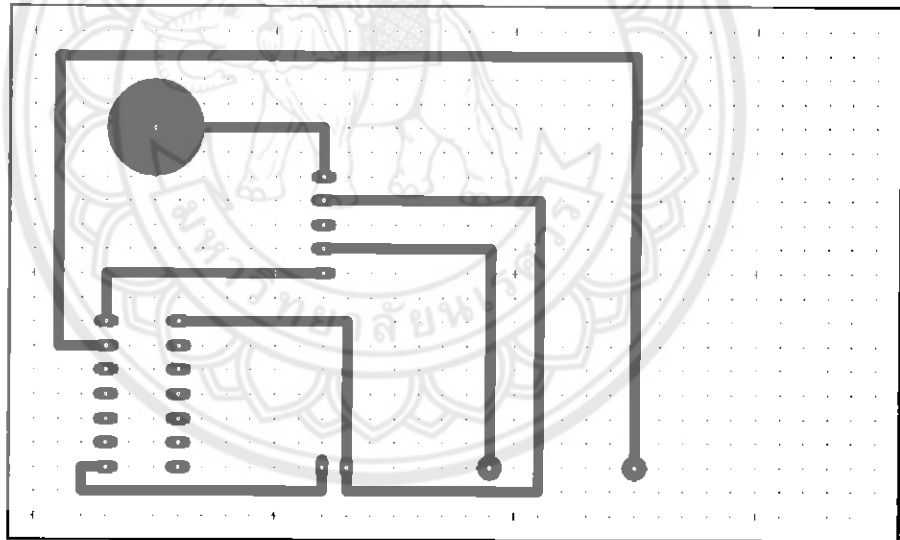
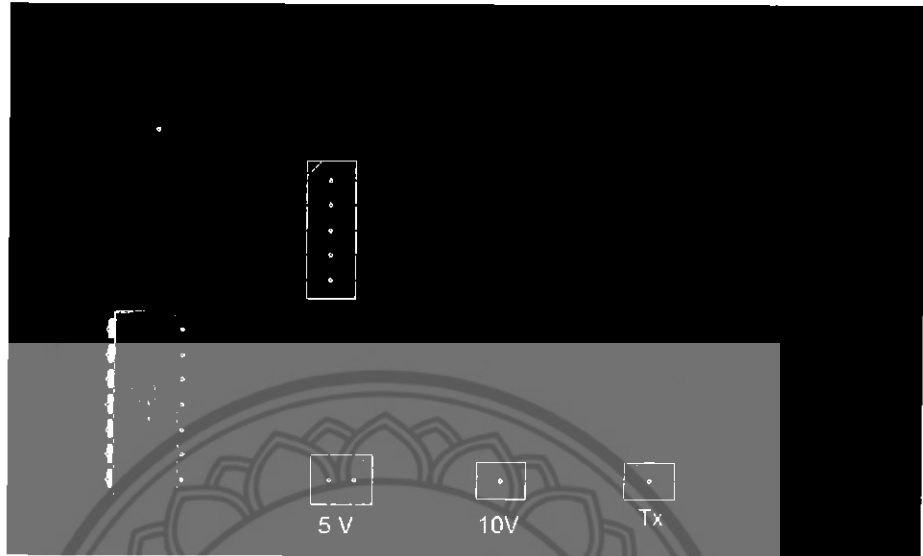
3395 ATmega32



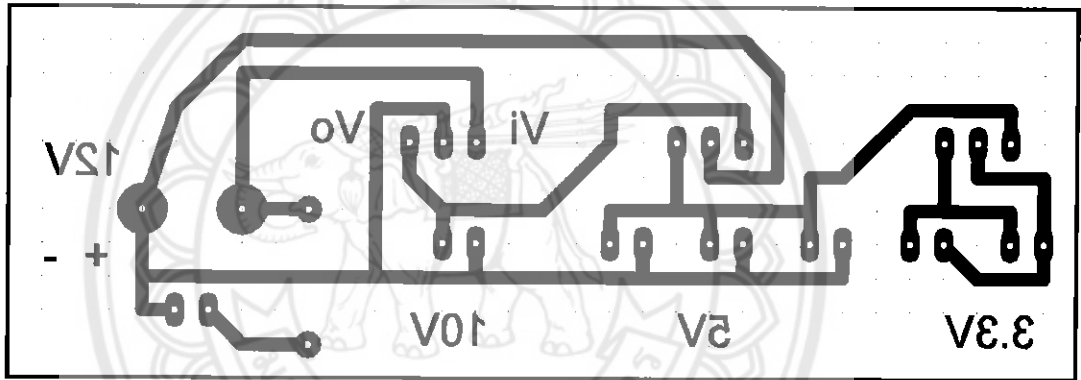
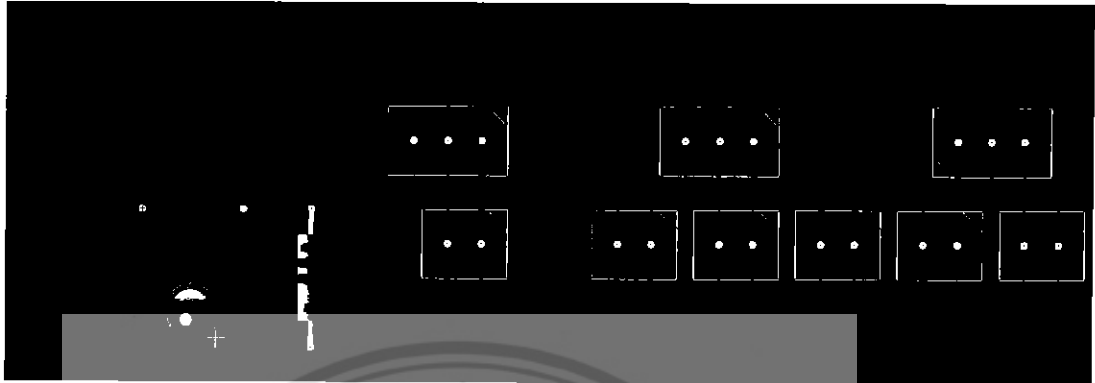
วงจรแบ่งแรงดันไฟฟ้า



วงจรส่งข้อมูล Futaba



วงจรแปลงไฟและจ่ายไฟ 10V, 5V และ 3.3 V



ประวัติผู้ดำเนินโครงการ



ชื่อ นายอภิพงศ์ ปิงยศ
 ภูมิลำเนา 130/2 หมู่ 8 ต. เมืองเก่า อ. เมือง จ. พิจิตร
 ประวัติการศึกษา
 - จบระดับมัธยมศึกษาจาก โรงเรียนพิจิตรพิทยาคม
 จ.พิจิตร
 - ปัจจุบันกำลังศึกษาระดับปริญญาตรีชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร
 Email: arm21641@hotmail.com



ชื่อ นายชวาน ประทีปศิริ
 ภูมิลำเนา 24/1 หมู่ 9 ต. เขาค้อ อ. เขาค้อ จ. เพชรบูรณ์
 ประวัติการศึกษา
 - จบระดับมัธยมศึกษาจาก โรงเรียนเพชรพิทยาคม
 จ.เพชรบูรณ์
 - ปัจจุบันกำลังศึกษาระดับปริญญาตรีชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร
 Email: lg_chuan2@hotmail.com



ชื่อ นายชาติชาย ไชยพานิช
 ภูมิลำเนา 330/2 ถ.ศรีธรรมไตรปิฎก ต. ในเมือง อ.เมือง จ. พิษณุโลก
 ประวัติการศึกษา
 - จบระดับมัธยมศึกษาจาก โรงเรียนพิษณุโลกพิทยาคม
 จ.พิษณุโลก
 - ปัจจุบันกำลังศึกษาระดับปริญญาตรีชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร
 Email: gorn_gang@hotmail.com