

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม

Anemometer and Wind vanes



T119432



โดย

นาย คุณานนท์

หันตา

นางสาว กมลชนก

ช่างสมบุญ

นาย ฤทธิพงษ์

แสงคำ

นาย กฤษณะศักดิ์

รักเนตรสาคร

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน **119432**
วัน,เดือน,ปี - 7 S.A. 2554

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ขอสงวนสิทธิ์ในประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาหรือข้อมูลของเอกสารนี้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2553

ปีการศึกษา 2553
เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม
Anemometer and Wind vanes



อาจารย์ที่ปรึกษา

อ.ดุสิต	สุขสวัสดิ์
อ.วรกัลป์	ลิ้มเจริญ
ผศ.ดร.ชาย	ชมภูอินไหว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2553

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องวัดความเร็วและเครื่องวัดทิศทางลม

ผู้จัดทำ

- 
1. นาย คุณานนท์ นันตา
 2. นางสาว กมลชนก ช่างสมบุญ
 3. นาย ธีรพงษ์ แสงคำ
 4. นาย กฤษณะศักดิ์ รักเนตรสาคร

..... อาจารย์ที่ปรึกษา
(อาจารย์ดุสิต สุขสวัสดิ์)

..... อาจารย์ที่ปรึกษา
(อาจารย์วรกัลป์ ลิ้มเจริญ)

..... อาจารย์ที่ปรึกษา
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาระดับบัณฑิตศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
(ผู้ช่วยศาสตราจารย์ ดร. ชัย ชมภูอินไหว)
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม

นาย คุณานนท์ นันตา
นาย กฤษณะศักดิ์ รักเนตรสาคร
นางสาว กมลชนก ช่างสมบุญ
นาย ณิชูพงษ์ แสงคำ
อาจารย์ดุสิต สุขสวัสดิ์ อาจารย์ที่ปรึกษา
อาจารย์วรกมลปี ลิ้มเจริญ อาจารย์ที่ปรึกษา
ผศ.ดร. ชาย ชมภูอินไหว อาจารย์ที่ปรึกษา
ปีการศึกษา 2553

บทคัดย่อ

โครงการนี้ได้นำเสนอการสร้างและพัฒนาเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม ซึ่งเป็นเครื่องมือวัดทางกายภาพที่ประยุกต์การใช้เซนเซอร์ร่วมกับไมโครคอนโทรลเลอร์เบอร์ PIC16F877A โดยมีการแสดงผลผ่านทางจอ LCD MODULE และบันทึกข้อมูลลงหน่วยความจำแบบ FLASH DRIVE USB นอกจากนี้ยังได้ส่งชุดข้อมูลค่าความเร็วลมและค่าทิศทางลมมายังโปรแกรมคอมพิวเตอร์เพื่อบันทึกข้อมูลลงตารางฐานข้อมูล โดยสามารถแสดงผลได้ 3 รูปแบบ คือ แสดงผลแบบหน้าปัดและตัวเลขดิจิทัล แสดงผลแบบฐานข้อมูล และแสดงผลแบบกราฟ จึงทำให้สะดวกต่อการดูข้อมูลในระยะไกล และข้อมูลที่ได้มีความเหมาะสมต่อการนำมาประยุกต์ใช้วิเคราะห์ศักยภาพพลังงานลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Anemometer and Wind vanes

Kunanon Nunta
Kamonchanok Changsomboon
Kritsanasak Ruknatesakorn
Nattapong Sangcome
Mr.Dusit Suksawat Adviser
Mr.Worrakan Limcharoen Adviser
Asst.Prof. Dr. Chai Chompoo-inwai Supervisor
2010

Abstract

The objective of this project is to present the design and development of anemometer and wind vane. This is a physical measuring instrument modified to share the sensor of the microcontroller (PIC16F877A). The Speed and direction is displayed on LCD MODULE and the data is recorded onto memory USB FLASH DRIVE. The measurement can be saved as database. This can present in 3 forms which displayed on meter and digital number, displayed on database and displayed on graph are equipped with Visual Studio 2008 C# to facilitate long distance access. It is also suitable for application to diagnose wind energy.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

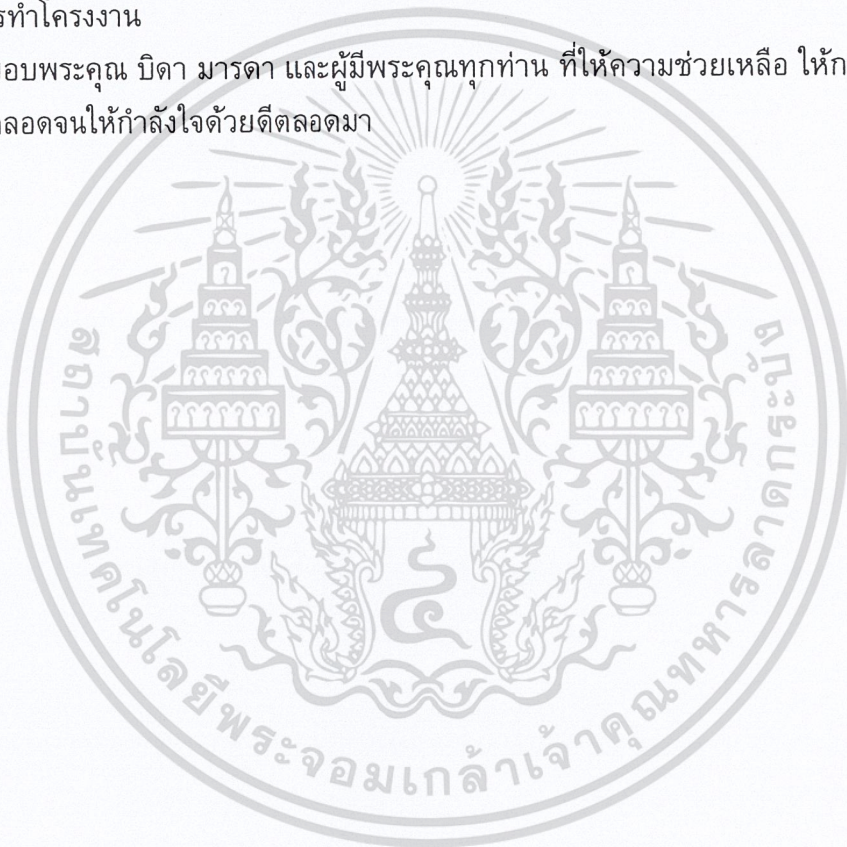
กิตติกรรมประกาศ

ขอขอบพระคุณ อาจารย์ดุสิต สุขสวัสดิ์ อาจารย์วรภัลป์ ลี้มเจริญ และ ผศ.ดร. ชาย ชมภูอินไหว อาจารย์ที่ปรึกษาโครงการ ตลอดจนอาจารย์ประจำภาควิชาทุกท่านให้วิชาความรู้ กับผู้จัดทำ ถ้าปราศจากท่านโครงการนี้ก็คงไม่สำเร็จลุล่วงไปได้

ขอขอบพระคุณ นายอำรุงฤทธิ์ นิ่มเสมอ หัวหน้างานตรวจสอบและสอบเทียบเครื่องมือ นายวัชรพล ทรัพย์วัฒน์ วิศวกรเครื่องกล 3 และเจ้าหน้าที่ของกรมอุตุนิยมวิทยาทุกท่านที่ให้ความรู้ คำแนะนำ ตลอดจนความช่วยเหลือในการทำให้ปริญญาณิพนธ์นี้สำเร็จลุล่วงได้ดี

ขอขอบคุณเพื่อนทุกคนที่ให้ความช่วยเหลือ คำแนะนำ และให้ยืมเครื่องมืออุปกรณ์ต่างๆ ในการทำโครงการ

ขอขอบพระคุณ บิดา มารดา และผู้มีพระคุณทุกท่าน ที่ให้ความช่วยเหลือ ให้การสนับสนุน ตลอดจนให้กำลังใจด้วยดีตลอดมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา **III** และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ	I
ABSTRACT	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VII
สารบัญตาราง	X
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของปัญหา.....	2
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตและข้อกำหนดของโครงการ	2
1.4 วิธีการที่ใช้ในโครงการ	2
1.5 แผนการดำเนินโครงการ	2
1.6 ประโยชน์ที่ได้รับจากการวิจัย	3
1.7 การจัดโครงสร้างของเนื้อหาภายในปฏิญานิพนธ์	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	5
2.1 ทฤษฎีเกี่ยวกับลม	5
2.1.1 ความหมายของลม.....	5
2.1.2 การเกิดลม.....	5
2.1.3 ความเร็วลม.....	5
2.1.4 การวัดความเร็วลม.....	5
2.1.5 การวัดทิศทางลม	8
2.2 เครื่องวัดความเร็วลม (Anemometer).....	9
2.2.1 เครื่องวัดความเร็วลมแบบแผ่นกระดก.....	9
2.2.2 เครื่องวัดความเร็วลมแบบลูกถ้วย	9
2.2.3 เครื่องวัดความเร็วลมแบบใช้ใบพัด.....	9
2.3 เครื่องวัดทิศทางลมหรือศรลม (Wind vane).....	10
2.4 ไมโครคอนโทรลเลอร์	11
2.4.1 ความหมายของไมโครคอนโทรลเลอร์.....	11
2.4.2 คุณสมบัติของไมโครคอนโทรลเลอร์ PIC16F877A.....	11
2.4.3 การจัดขาของไมโครคอนโทรลเลอร์ PIC16F877A	12
2.4.4 การจัดการหน่วยความจำของไมโครคอนโทรลเลอร์ PIC16F877A ..	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

2.4.5	กระบวนกรรีเซต	15
2.5	โพเทนชิโอมิเตอร์.....	15
2.5.1	ชนิดของโพเทนชิโอมิเตอร์.....	15
2.5.2	ความละเอียดของโพเทนชิโอมิเตอร์.....	16
2.5.3	การปรับสภาพสัญญาณ	16
2.6	ตัวตรวจจับที่ใช้ตรวจจับความเร็วลม	19
2.7	โครงสร้างและการทำงานของจอแสดงผล LCD.....	20
2.7.1	การเชื่อมต่อไมโครคอนโทรลเลอร์กับจอแสดงผล LCD.....	21
2.8	โครงสร้างการสร้างสัญญาณอินเทอร์รัปต์	22
2.8.1	ไทมเมอร์ในไมโครคอนโทรลเลอร์ PIC16F877A	22
2.8.2	ความรู้พื้นฐานเกี่ยวกับไทมเมอร์.....	23
2.8.3	การหาค่าเวลาในการนับแต่ละครั้ง(Time in tick).....	23
2.8.4	การหาค่าเวลาในการนับแต่ละครั้งเกี่ยวกับปริสเทลเลอร์.....	24
2.8.5	ฟังก์ชันการใช้งานไทมเมอร์ใน CCS Compiler	25
2.8.6	การทำงานในโหมดเคาน์เตอร์ของไทมเมอร์ 1	25
2.9	โครงสร้างการตรวจจับสัญญาณ.....	25
2.9.1	โมดูล Capture/Compare/PWM.....	25
2.9.2	ฟังก์ชันการใช้งานโมดูล CCP.....	26
2.9.3	โหมดตรวจจับสัญญาณอินพุต	26
2.10	โครงสร้างการรับ-ส่งข้อมูล	27
2.10.1	โมดูล USART (RS-232 Port)	27
2.10.2	โหมดอะซิงโครนัส.....	28
2.10.3	โหมดซิงโครนัส – มาสเตอร์/สเลฟ.....	28
2.10.4	ฟังก์ชันการเขียนข้อมูลพอร์ตอนุกรม	28
2.11	โครงสร้างการแปลงสัญญาณอะนาลอกเป็นดิจิตอล.....	28
2.11.1	การกำหนดช่องสัญญาณอะนาลอกเป็นดิจิตอล	29
2.11.2	การเลือกความถี่ของสัญญาณนาฬิกา	30
2.11.3	ฟังก์ชันการใช้งานโมดูล ADC ใน CCS Compiler.....	30
2.12	การเชื่อมต่อกับคอมพิวเตอร์.....	31
2.12.1	การอินเตอร์เฟสกับพอร์ตอนุกรม.....	31
2.12.2	ไอซีปรับระดับแรงดัน MAX 232.....	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปเผยแพร่โดยไม่ได้รับอนุญาต

V

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

2.13	ภาษาซีชาร์ป (C# Programming Language).....	35
2.14	Visual Studio 2008 with C#	35
2.15	Serial Port	36
2.16	Microsoft Chart Control.....	37
บทที่ 3 การออกแบบและการสร้าง		38
3.1	โครงสร้างของเครื่องวัดความเร็วลมและทิศทางลม	38
3.2	การออกแบบและการสร้าง.....	38
3.2.1	การออกแบบเครื่องวัดความเร็วลม	38
3.2.2	วงจรตรวจจับค่าความเร็วลม	41
3.2.3	การทดสอบชุดตรวจจับค่าความเร็วลม	42
3.2.4	การออกแบบเครื่องวัดทิศทางลม	42
3.2.5	วงจรตรวจวัดทิศทางลม	45
3.3	การออกแบบโปรแกรมแสดงผลการวัดค่าความเร็วลมและทิศทางลม	46
3.3.1	โปรแกรมการทำงานของไมโครคอนโทรลเลอร์.....	47
3.3.2	โปรแกรมที่ใช้คำนวณค่าความเร็วลม	49
3.3.3	โปรแกรมที่ใช้คำนวณค่าทิศทางลม.....	50
3.4	การส่งข้อมูลด้วยโมดูล USART (RS-232).....	51
3.5	การออกแบบและสร้างโปรแกรมแสดงผลบนจอคอมพิวเตอร์.....	53
3.5.1	ออกแบบโปรแกรมแสดงผลบนจอคอมพิวเตอร์	53
3.5.2	สร้างโปรแกรมแสดงผลบนจอคอมพิวเตอร์	60
บทที่ 4 การทดลองและผลการทดลอง		61
4.1	ลักษณะของพัลส์ที่ความเร็วต่างๆ	61
4.2	สอบเทียบเครื่องมือวัด.....	65
4.2.1	สอบเทียบเครื่องวัดความเร็วลม.....	65
4.2.2	สอบเทียบเครื่องวัดทิศทางลม	66
4.3	การแสดงผลข้อมูลของโปรแกรมไมโครคอนโทรลเลอร์.....	67
4.3.1	การแสดงผลข้อมูลความเร็วลมและทิศทางลมบนจอ LCD.....	68
4.3.2	การแสดงผลข้อมูลในส่วนการบันทึกข้อมูล	68
4.4	การส่งข้อมูลเข้าคอมพิวเตอร์.....	69
4.5	การประมวลผลของโปรแกรมคอมพิวเตอร์.....	70
บทที่ 5 สรุปและวิจารณ์.....		74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สารบัญภาพ

ภาพที่	หน้า
2.1	เครื่องวัดความเร็วลมแบบลูกถ้วย.....9
2.2	เครื่องวัดความเร็วลมแบบใช้ใบพัด.....10
2.3	ครลม.....10
2.4	ขาสัญญาณของไมโครคอนโทรลเลอร์ 16F877A.....12
2.5	โครงสร้างภายในของไมโครคอนโทรลเลอร์ 16F877A.....13
2.6	การจัดการพื้นที่หน่วยความจำของไมโครคอนโทรลเลอร์ 16F877A.....14
2.7	แสดงการตรวจจ็บระยะโดยใช้โพเทนชิโอมิเตอร์แบบเชิงเส้น.....15
2.8	แสดงโพเทนชิโอมิเตอร์แบบต่าง ๆ.....17
2.9	แสดงโพเทนชิโอมิเตอร์เมื่อขับโหลด.....18
2.10	แสดงหลักการตรวจจ็บความเร็วลม.....19
2.11	ขาสัญญาณของจอแสดงผล LCD ขนาด 16x2.....20
2.12	การเชื่อมต่อไมโครคอนโทรลเลอร์กับจอแสดงผล LCD แบบ 4 บิต.....21
2.13	ไดอะแกรมการทำงานของโมดูล CCP ในโหมดตรวจจ็บสัญญาณ.....27
2.14	ไดอะแกรมวงจรอินพุตแปลงสัญญาณอะนาลอกเป็นดิจิตอล.....29
2.15	การจัดค่าของคอนเนคเตอร์พอร์ตอนุกรมมาตรฐานอาร์เอสสองสามสองแบบ DB-9 ..32
2.16	การเชื่อมต่อแบบอาร์เอสสองสามสองโดยใช้สัญญาณ 3 เส้น ระหว่างอุปกรณ์ภายนอกกับคอมพิวเตอร์.....33
2.17	ตำแหน่งขาสัญญาณและวงจรภายในไอซีปรับระดับแรงดัน MAX 232.....34
2.18	คอมโพเนนต์และอีเวนต์ Serial port.....36
2.19	ตัวอย่างรูปแบบกราฟใน Microsoft Chart Control.....37
3.1	บล็อกไดอะแกรมโครงสร้างของระบบ.....38
3.2	เครื่องวัดความเร็วลม.....38
3.3	ก้านรับลม.....39
3.4	ลักษณะของการต่อลูกถ้วยเข้ากับก้านรับลม.....39
3.5	ลักษณะของแกนหมุน.....39
3.6	ส่วนประกอบ Body ของเครื่องวัดความเร็วลม.....39
3.7	ตัวจับยึดก้านรับลม.....40
3.8	ลักษณะของส่วนบังคับแกนหมุน.....40
3.9	การประกอบส่วนบังคับแกนเข้ากับท่อตรง.....40
3.10	ลักษณะของแผ่นตัดแสง.....40
3.11	การประกอบชิ้นส่วน Body ของเครื่องวัดความเร็วลม.....41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

3.12	เครื่องวัดความเร็วลมเมื่อประกอบเสร็จสมบูรณ์	41
3.13	วงจรตรวจจับค่าความเร็วลม	41
3.14	เครื่องวัดทิศทางลม	42
3.15	ลักษณะของหัวครลม	42
3.16	การเชื่อมหางรับลมเข้ากับแกนครลม	43
3.17	ลักษณะของแกนหมุน	43
3.18	ส่วนประกอบของตัว Body ของเครื่องวัดทิศทางลม	43
3.19	ลักษณะการเจาะรูที่ฝาครอบเพื่อใส่แกนของครลม	43
3.20	ลักษณะของส่วนบังคับกับแกนหมุน	44
3.21	การประกอบส่วนบังคับกับแกนเข้ากับท่อตรง	44
3.22	แผ่นยึดโพเทนชิโอมิเตอร์	44
3.23	การประกอบชิ้นส่วนของ Body ของเครื่องวัดทิศทางลม	44
3.24	เครื่องวัดทิศทางลมเมื่อประกอบเสร็จสมบูรณ์	45
3.25	โพเทนชิโอมิเตอร์	45
3.26	บล็อกไดอะแกรมการทำงานของโปรแกรมแสดงผล	46
3.27	บล็อกไดอะแกรมการทำงานของไมโครคอนโทรลเลอร์	48
3.28	ไดอะแกรมการคำนวณความเร็วลม	49
3.29	ไดอะแกรมการคำนวณทิศทางลม	50
3.30	ไดอะแกรมการรับข้อมูลด้วย USART (RS-232)	51
3.31	ไดอะแกรมการส่งข้อมูลด้วย USART (RS-232)	52
3.32	ไดอะแกรมของส่วนเชื่อมต่อข้อมูลและจัดเก็บค่าลงฐานข้อมูล	54
3.33	การใช้คอมพิวเตอร์ Comport เพื่อรับข้อมูลจากอุปกรณ์ภายนอก	55
3.34	ตารางฐานข้อมูล(.mdb) ที่เชื่อมต่อกับโปรแกรม	55
3.35	ไดอะแกรมของโปรแกรมแสดงผลแบบหน้าปัดและตัวเลขดิจิทัล	56
3.36	หน้าตาแสดงผลข้อมูลแบบหน้าปัดและตัวเลขดิจิทัล	57
3.37	ไดอะแกรมของโปรแกรมแสดงผลข้อมูลแบบฐานข้อมูล	58
3.38	หน้าตาแสดงผลข้อมูลแบบฐานข้อมูล	58
3.39	ไดอะแกรมของโปรแกรมแสดงผลข้อมูลความเร็วลมแบบกราฟ	59
3.40	หน้าตาแสดงผลข้อมูลความเร็วลมแบบกราฟ	60
4.1	รับความเร็วลมอ้างอิงจากอุโมงค์ลม	61
4.2	การวัดสัญญาณพัลส์	61

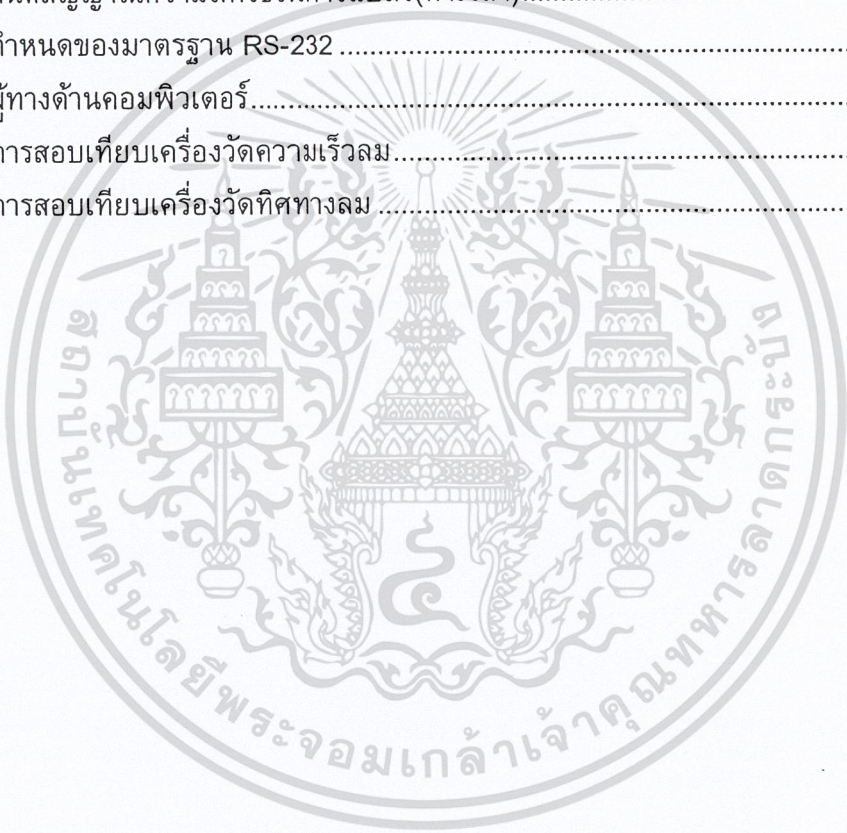
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สารบัญญภาพ(ต่อ)

4.3	ความถี่พัลส์ที่ความเร็วลม 1 เมตรต่อวินาที	62
4.4	ความถี่พัลส์ที่ความเร็วลม 2 เมตรต่อวินาที	63
4.5	ความถี่พัลส์ที่ความเร็วลม 5 เมตรต่อวินาที	63
4.6	ความถี่พัลส์ที่ความเร็วลม 10 เมตรต่อวินาที	64
4.7	ความถี่พัลส์ที่ความเร็วลม 14 เมตรต่อวินาที	64
4.8	การทดสอบความเร็วลมภายในอุโมงค์ลม	65
4.9	การสอบเทียบเครื่องวัดความเร็วลม	65
4.10	การสอบเทียบเครื่องวัดทิศทางลม	66
4.11	ตำแหน่งองศาขณะสอบเทียบ	67
4.12	การแสดงผลข้อมูลของโปรแกรมไมโครคอนโทรลเลอร์	67
4.13	LCD ขณะแสดงผลการทำงาน	68
4.14	ข้อมูลที่บันทึกผ่านโปรแกรม Notepad	68
4.15	ค่าความเร็วลมและค่าทิศทางลมที่ส่งมายังคอมพิวเตอร์	69
4.16	การเชื่อมต่อโปรแกรมคอมพิวเตอร์เข้ากับชุดประมวลผลข้อมูล	70
4.17	โปรแกรมเมื่อแสดงผลแบบหน้าปัดและตัวเลขดิจิทัล	70
4.18	ตารางฐานข้อมูล(.mdb) ที่เชื่อมต่อกับโปรแกรม	71
4.19	โปรแกรมแสดงผลแบบฐานข้อมูลตามช่วงวันและเวลาที่เลือก	71
4.20	โปรแกรมเมื่อแสดงผลความเร็วลมแบบกราฟวัน	72
4.21	โปรแกรมเมื่อแสดงผลความเร็วลมแบบกราฟเดือน	72
4.22	กราฟแสดงทิศทางลมโดยโปรแกรม SmartDraw	73
5.1	เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม	75

สารบัญญัตราง

ตารางที่	หน้า
1.1	แผนการดำเนินงาน3
2.1	ความสัมพันธ์ระหว่างหน่วยต่างๆที่ใช้ในการวัดความเร็วลม6
2.2	ขนาดของลม สัญลักษณ์ที่แสดงบนบก7
2.3	เลขรหัส ทิศนับเป็นองศาจริง และบ้อยท์ของเข็มทิศ8
2.4	โหมด CCP กับการใช้งานไทเมอร์26
2.5	โหมดแสดงความสัมพันธ์ของ CCP1 และ CCP2 กับไทเมอร์26
2.6	การกำหนดช่องสัญญาณ ADC30
2.7	กำหนดสัญญาณความถี่ที่ใช้ในการแปลง(ค่าเวลา).....30
2.8	ข้อกำหนดของมาตรฐาน RS-23232
2.9	ตัวผู้ทางด้านคอมพิวเตอร์.....33
4.1	ผลการสอบเทียบเครื่องวัดความเร็วลม66
4.2	ผลการสอบเทียบเครื่องวัดทิศทางลม67



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของปัญหา

ปัจจุบันแหล่งเชื้อเพลิงฟอสซิลในการผลิตไฟฟ้าในประเทศไทยลดลงเรื่อยๆ อีกทั้งมีราคาแพงและเกิดผลกระทบต่อสิ่งแวดล้อม แนวโน้มการใช้พลังงานลมในการผลิตไฟฟ้าจึงมีมากขึ้น เนื่องจากเป็นพลังงานสะอาดที่สามารถนำมาผลิตกระแสไฟฟ้าได้โดยตรงและใช้แล้วไม่หมดไป แต่ในการติดตั้งกังหันลมผลิตไฟฟ้านั้นจะต้องมีการสำรวจศักยภาพลมเสียก่อนว่าบริเวณนั้นมีความเร็วลมมากพอและสม่ำเสมอเพียงพอต่อการผลิตกระแสไฟฟ้าหรือไม่ สำหรับการเก็บข้อมูลลมในประเทศไทยนั้นยังมีข้อจำกัดด้านปริมาณข้อมูล เนื่องจากมีสถานีทำการตรวจวัดเพียงไม่กี่แห่ง อีกทั้งยังเก็บข้อมูลอย่างไม่ต่อเนื่อง เหตุผลอีกประการหนึ่งคือเครื่องวัดมีราคาแพงมาก เครื่องช่วยในการเก็บรวบรวมข้อมูลดังกล่าวจึงถูกจำกัดด้วยงบประมาณ

จากปัญหาดังกล่าว ทางกลุ่มจึงได้คิดริเริ่มในการพัฒนาสร้างเครื่องวัดความเร็วและทิศทางลม โดยสามารถจัดเก็บข้อมูลและแสดงผลในรูปแบบข้อมูลและกราฟได้ ทำให้เหมาะสมต่อการนำมาวิเคราะห์ศักยภาพพลังงานลม เพื่อใช้ในการตัดสินใจติดตั้งกังหันลมผลิตไฟฟ้าให้คุ้มค่าต่อเม็ดเงินลงทุน อีกทั้งข้อมูลลมที่ได้ยังสามารถนำมาวิจัยพัฒนาออกแบบใบพัดของกังหันลมและการพัฒนาระบบให้มีประสิทธิภาพเหมาะสมกับสภาพภูมิอากาศในประเทศไทยอีกด้วย

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาออกแบบและสร้างเครื่องมือวัดปริมาณทางกายภาพ
2. เพื่อประยุกต์ใช้ไมโครคอนโทรลเลอร์ในการประมวลผลสัญญาณ และเชื่อมต่อสัญญาณเข้ากับคอมพิวเตอร์ เพื่อแสดงผลผ่านทางโปรแกรมคอมพิวเตอร์
3. เพื่อศึกษาเปรียบเทียบผลที่ได้จากเครื่องมือวัดที่สร้างและค่าพลังงานจริงที่ได้จากกรมอุตุนิยมวิทยา

1.3 ขอบเขตและข้อกำหนดของโครงการ

โครงการนี้จะทำการสร้างและพัฒนาเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม โดยจะแบ่งออกเป็น 4 ส่วนคือ ส่วนการสร้างเครื่องมือวัด ส่วนการประมวลผลและเชื่อมต่อสัญญาณ ส่วนจัดเก็บข้อมูล และส่วนแสดงผลผ่านโปรแกรมคอมพิวเตอร์ ซึ่งรายละเอียดมีดังต่อไปนี้

1. การสร้างเครื่องมือวัด โดยออกแบบและสร้างเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลมให้สามารถใช้งานได้จริงและมีเสถียรภาพ โดยการใช้เทคโนโลยีรูปแบบที่แตกต่างจากเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ กรุณา
ประกอบกับวัสดุที่มีต้นทุนต่ำ
ไม่มีการแก้ไข ทุกสิ่ง อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การประมวลผลและเชื่อมต่อสัญญาณ ใช้ไมโครคอนโทรลเลอร์ PIC16F877A ขนาด 16 บิต ในการประมวลสัญญาณและคำนวณค่าความเร็วลมและทิศทางลม แสดงผลเป็น ทศนิยม 2 ตำแหน่งผ่านทางจอ LCD ขนาด 16X2 และส่งสัญญาณไปเชื่อมต่อกับ คอมพิวเตอร์โดยใช้ USART (RS-232)

3. การจัดเก็บข้อมูล บันทึกข้อมูลโดยใช้ USB FLASH DRIVE ในการสำรองไฟล์ข้อมูล และมีการบันทึกข้อมูลลงตารางฐานข้อมูลผ่านโปรแกรมคอมพิวเตอร์

4. การแสดงผลผ่านทางโปรแกรมคอมพิวเตอร์ ใช้โปรแกรม Visual Studio 2008 ภาษา C# ในการสร้างโปรแกรม โดยแสดงผลในรูปแบบหน้าต่างและเลขทศนิยม ฐานข้อมูล และกราฟ ความเร็วลม

1.4 วิธีการที่ใช้ในโครงการ

1. ออกแบบเทคโนโลยีที่ใช้ เพื่อให้เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลมทำงาน ได้ถูกต้องและมีประสิทธิภาพสูง
2. ออกแบบและสร้างเครื่องวัดความเร็วและเครื่องวัดทิศทางลม
3. เปรียบเทียบค่าความถูกต้องของเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม
4. ส่งสัญญาณมาแสดงผลผ่านโปรแกรมคอมพิวเตอร์ และจัดเก็บลงหน่วยความจำ

1.5 แผนการดำเนินโครงการ

โครงการวิจัยนี้ได้ออกแบบและสร้างเครื่องมือวัดทางกายภาพ ชุดประมวลผลและแสดงผล หน่วยความจำ และโปรแกรมคอมพิวเตอร์สำหรับแสดงผล โดยมีระยะเวลาในการ ดำเนินโครงการ 2 เดือน ซึ่งสามารถแบ่งการดำเนินงานได้ดังตารางที่ 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1.1 แผนการดำเนินงาน

การดำเนินงาน	เดือน									
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	
1. ศึกษาข้อมูลและงานวิจัยที่เกี่ยวกับเครื่องวัดความเร็วลม เครื่องวัดทิศทางลม	■	■								
2. ออกแบบเครื่องวัดความเร็วลม เครื่องวัดทิศทางลม และการแสดงผลข้อมูล		■	■							
3. สร้างเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม			■	■						
4. เขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์				■	■	■	■	■	■	■
5. สอบเทียบค่ามาตรฐานโดยกรมอุตุนิยมวิทยา								■	■	
6. ศึกษาและออกแบบโปรแกรมสำหรับการส่งข้อมูล การแสดงผล และจัดเก็บข้อมูล					■	■	■	■	■	
7. สร้างโปรแกรมสำหรับแสดงผลกับการจัดเก็บข้อมูล								■	■	
8. ทดสอบการทำงานของเครื่องวัดความเร็วลม เครื่องวัดทิศทางลม ส่วนการแสดงผลและการจัดเก็บข้อมูล								■	■	■
9. สรุปผลงานวิจัย เสนอโครงการ										■

1.6 ประโยชน์ที่ได้รับจากการวิจัย

โครงการนี้เป็นการสร้างเครื่องมือวัดความเร็วลมและเครื่องวัดทิศทางลมที่มีความแม่นยำและเสถียรภาพสูง เพื่อนำมาวัดค่าพลังงานลม ซึ่งสามารถแสดงผลผ่านโปรแกรมคอมพิวเตอร์ จึงสะดวกต่อการสังเกตค่าของพลังงานลมในระยะไกลโดยแสดงผลในรูปแบบข้อมูลและกราฟได้ ทำให้สามารถนำชุดข้อมูลที่ได้ออกไปประยุกต์ วิเคราะห์พลังงานลมเพื่อการวิจัยและพัฒนาทางด้านพลังงานต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.7 การจัดโครงสร้างของเนื้อหาภายในปฏิญานิพนธ์

บทที่ 1 เป็นการกล่าวเกี่ยวกับบทนำของโครงการที่ปฏิบัติโดยได้กล่าวถึงความเป็นมาและความสำคัญของโครงการที่เกี่ยวกับพลังงานลม รวมทั้งยังกล่าวถึงขอบเขตของโครงการและประโยชน์ที่ได้จากโครงการนี้

บทที่ 2 กล่าวถึงทฤษฎีต่างๆ ได้แก่ พลังงานลม เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม ไมโครคอนโทรลเลอร์ การเชื่อมต่อสัญญาณของพอร์ตอนุกรม ไอซีปรับระดับแรงดัน MAX232 ภาษาซีชาร์ป และโปรแกรม Visual Studio 2008

บทที่ 3 เป็นกล่าวเกี่ยวกับการออกแบบและการสร้างเครื่องมือวัด โดยอธิบายถึงโครงสร้าง การประกอบและวงจรของเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม การคำนวณพัลส์และแรงดันของไมโครคอนโทรลเลอร์เพื่อนำมาแสดงผลและบันทึกลงหน่วยความจำ การเชื่อมต่อข้อมูลผ่านพอร์ตอนุกรม การสร้างโปรแกรมสำหรับจัดเก็บข้อมูลลงฐานข้อมูลและแสดงผลผ่านโปรแกรมคอมพิวเตอร์

บทที่ 4 แสดงถึงผลการทดสอบเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม โดยทำการอ่านค่าพัลส์เพื่อคำนวณความเร็วลม และสอบเทียบเครื่องมือวัดเพื่อหาค่าความผิดพลาดของเครื่องมือ จากนั้นทดสอบการเชื่อมต่อข้อมูลของพอร์ตอนุกรมไปยังคอมพิวเตอร์ การบันทึกลงหน่วยความจำ และการแสดงผลของข้อมูล

บทที่ 5 เป็นการสรุปผลของโครงการ และข้อเสนอแนะปรับปรุงของเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม เพื่อให้มีประสิทธิภาพ เสถียรภาพ และความสามารถของเครื่องวัดที่มากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 ทฤษฎีเกี่ยวกับลม

2.1.1 ความหมายของลม

ลม คือ กระแสอากาศที่เคลื่อนที่ในแนวนอน ส่วนความกดอากาศ คือ อากาศที่เคลื่อนที่ในแนวตั้ง การเรียกชื่อลมนั้นเรียกตามทิศทางที่ลมนั้น ๆ พัดมา เช่น ลมที่พัดมาจากทิศเหนือเรียกว่า ลมเหนือ และลมที่พัดมาจากทิศใต้เรียกว่า ลมใต้ เป็นต้น ซึ่งการวัดลมจะต้องวัดใน 2 vector คือ ความเร็วลม (Speed) และทิศทางลม (Direction)

2.1.2 การเกิดลม

การเกิดลมเกิดจากอากาศที่เคลื่อนที่จากสาเหตุต่าง ๆ ดังนี้

1. ความแตกต่างของอุณหภูมิในที่สองแห่ง อากาศเมื่อได้รับความร้อนจะขยายตัว ความหนาแน่นของอากาศจะลดลง อากาศจึงลอยตัวขึ้น อากาศเย็นที่มีอุณหภูมิต่ำกว่าและมีความหนาแน่นของอากาศมากกว่าจากบริเวณใกล้เคียงจะเคลื่อนเข้ามาแทนที่ทำให้เกิดลม
2. ความแตกต่างของความกดอากาศ อากาศเมื่อได้รับความร้อนจะมีความกดอากาศต่ำ ความหนาแน่นลดลงจึงลอยตัวสูงขึ้น อากาศเย็นที่มีความหนาแน่นมากกว่าและมีความกดอากาศสูงกว่าจะเคลื่อนที่เข้ามาหาบริเวณที่มีความกดอากาศต่ำ ทำให้เกิดลม

2.1.3 ความเร็วลม

ความเร็วลม คือ การเคลื่อนที่ของอากาศที่ทำให้เกิดแรง หรือความกดอากาศที่ผ่านจุดที่กำหนดให้บนพื้นผิวโลก และแรงหรือความกดนี้เป็นสัดส่วนกับค่ากำลังสองของความเร็ว ซึ่งอธิบายในรูปของสมการได้ดังนี้

$$P = kv^2 \quad (2.1)$$

เมื่อ P = ความกดที่เกิดจากการกระทำของลม

v = ความเร็วลม

k = ค่าคงที่ของหน่วยที่ใช้

2.1.4 การวัดความเร็วลม

เนื่องจากการรายงานความเร็วลมเพื่อการพยากรณ์อากาศนั้นใช้เป็นนอต ดังนั้น เครื่องวัดความเร็วลม จึงควรแบ่งสเกลเป็นนอตด้วย สำหรับรายงานเพื่อการพยากรณ์อากาศ ต้องการค่าของความเร็วลมผิวพื้นเฉลี่ยในช่วง 10 นาที ก่อนการตรวจ หรือระหว่างที่ลมมีความเร็ว 5 นอต และต้องอ่านให้ได้ค่าใกล้เคียงนอตที่สุดถ้าลมมีความเร็วน้อยกว่า 1 นอต ให้รายงานว่า “ ลมสงบ ” ค่าของความเร็วลมนั้นจะมีหน่วยที่ใช้วัดซึ่งมีอยู่ด้วยกันหลายหน่วยดังนี้

1. นอต (หรือไมล์ทะเลต่อชั่วโมง)
2. เมตร ต่อ วินาที
3. กิโลเมตร ต่อ ชั่วโมง
4. ไมล์ ต่อ ชั่วโมง
5. ฟุต ต่อ วินาที

ตารางที่ 2.1 ความสัมพันธ์ระหว่างหน่วยต่างๆที่ใช้ในการวัดความเร็วลม

นอต (Kt.)	เมตร ต่อ วินาที (m/sec.)	ไมล์ ต่อ ชั่วโมง (m.p.h)	กิโลเมตร ต่อ ชั่วโมง(km/hr)	ฟุต ต่อ วินาที (ft/sec.)
1	0.515	1.152	1.853	1.689
1.943	1	2.237	3.6	3.281
0.868	0.447	1	1.609	1.467
0.54	0.287	0.622	1	0.911
0.592	0.305	0.682	1.097	1

การวัดความเร็วลมนั้น โดยทั่วไปจะวัดเป็นกิโลเมตรต่อชั่วโมง หรือ ไมล์ต่อชั่วโมง หรือ วัดค่าเป็น นอต สำหรับการคาดคะเนความเร็วของลมนั้น พลเรือเอกโบฟอร์ตได้เป็นผู้กำหนดการกะประมาณความเร็วลมได้ โดยเปรียบเทียบตามกำลังของลมโดยอาการของควัน บ้าง ชงบ้าง หรือ ต้นไม้ และอาคารบ้านเรือนเป็นเครื่องกำหนดโดยประมาณคาดคะเนความเร็วของลมไว้ 17 มาตรา ดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ขนาดของลม สัญลักษณ์ที่แสดงบนบก

Beaufort Number	ขนาดของลม		สัญลักษณ์ที่แสดงบนบก	หนด	กม./ชม.
				Knots	Km./hr.
0	ลมสงบ	Clam	ลมเจียบ ควันลอยขึ้นตรง	น้อยกว่า 1	น้อยกว่า 1
1	ลมเบา	Light air	ควันลอยตามลมแต่ศรลมไม่หันไปตามทิศทางลม	1 - 3	1 - 5
2	ลมอ่อน	Light breeze	รู้สึกลมพัดที่ใบหน้า ใบไม้แกว่งไกว ศรลมหันไปตามทิศทางลม	4 - 6	6 - 11
3	ลมโชย	Gentle breeze	ใบไม้และกิ่งไม้เล็กๆ กระดิกขยับ	7 - 10	12 - 19
4	ลมปานกลาง	Moderate breeze	มีฝุ่นตลบ กระจดาขปลิว กิ่งไม้เล็กขยับเขยื้อน	11 - 16	20 - 28
5	ลมแรง	Fresh breeze	ต้นไม้เล็กแกว่งไกวไปมามีระลอกน้ำ	17 - 21	29 - 38
6	ลมจัด	Strong breeze	กิ่งไม้ใหญ่ขยับเขยื้อน ได้ยินเสียงหวีดหวิวไซ้ร่มลำบาก	22 - 27	39 - 49
7	พายุเกลอ่อน	Near Gale	ต้นไม้ใหญ่ทั้งต้นแกว่งไกว เดินทวนลมไม่สะดวก	28 - 33	50 - 61
8	พายุเกล	Gale	กิ่งไม้หัก ลมต้านการเดินทาง	34 - 40	62 - 74
9	พายุเกลแรง	Strong gale	อาคารที่ไม่มั่นคงหักพัง หลังคาปลิว	41 - 47	75 - 88
10	พายุ	Storm	ต้นไม้ถอนรากล้ม เกิดความเสียหายมาก (ไม่ปรากฏบ่อย)	48 - 55	89 - 102
11	พายุใหญ่	Violent storm	เกิดความเสียหายทั่วไป (ไม่ค่อยปรากฏ)	56 - 63	103 - 117
12	พายุไต้ฝุ่น, เฮอริเคน	Typhoon Hurricane	เกิดความเสียหายทั่วไป (ไม่ค่อยปรากฏ)	มากกว่า 63	มากกว่า 117

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การวัดทิศทางลม

ทิศทางลมนั้นถือเอาลมที่พัดเข้าสถานีเป็นเกณฑ์ โดยใช้วัดเป็นองศาตามเข็มนาฬิกา โดยใช้ทิศทางเหนือจริงของสถานีเป็นหลัก คือ นับจาก 0 จนถึง 360 องศา นั่นเอง หรืออาจใช้นับเป็นป้อยท์ของเข็มทิศก็ได้ เช่น 8,16 และ 32 ป้อยท์ เป็นต้น

สำหรับรายงานทิศทางลมโดยรหัสนั้น ให้รายงานเป็นรหัส 00 – 36 ตารางที่เป็นเลขรหัส, ทิศนับเป็นองศาจริง และป้อยท์ของเข็มทิศ

ตารางที่ 2.3 เลขรหัส ทิศนับเป็นองศาจริง และป้อยท์ของเข็มทิศ

Compass Direction	Exact Equivalent in degree	Code figure	Compass direction	Exact Equivalent in degree	Code figure
Calm	-	00			
N by E	11.25	01	S by W	191.25	19
NNE	22.5	02	SSW	202.5	20
NE by N	33.75	03	SW by S	213.75	21
NE	45	05	SW	225	23
NE by E	56.25	06	SW by W	236.25	24
ENE	67.5	07	WSW	247.5	25
E by N	78.5	08	W by S	258.75	26
E	90	09	W	270	27
E by S	101.25	10	W by N	281.25	28
ESE	112.5	11	WNW	292.5	29
SE by E	123.75	12	NW by W	303.75	30
SE	135	14	NW	315	32
SE by S	146.25	15	NW by N	326.25	33
SSE	157.5	16	NNW	337.5	34
S by E	168.75	17	N by W	348.75	35
S	180	18	N	360	36
			Variable	-	99

โดยที่ N หมายถึง ทิศเหนือ (North)

E หมายถึง ทิศตะวันออก (East)

W หมายถึง ทิศตะวันตก (West)

S หมายถึง ทิศใต้ (South)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 เครื่องวัดความเร็วลม (Anemometer)

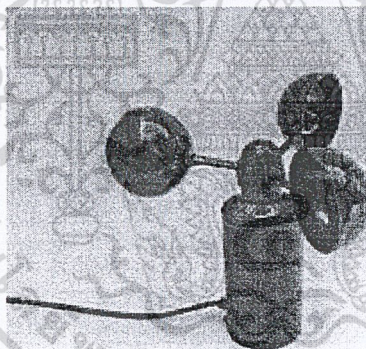
ใช้เครื่องมือที่เรียกว่า Anemometer เครื่องวัดความเร็วลมจะต้องรายงานค่าที่ตรวจได้ในระยะไกล คือจะต้องเดินสายจากเครื่องวัดมาแสดงผลยังจอมอนิเตอร์ ณ ห้องทำการที่อยู่ห่างออกไป และเครื่องมือวัดจะต้องวัดลมกระโชกได้ด้วย ดังนั้นเครื่องวัดจะต้องมีความไวสูง ซึ่งเครื่องวัดความเร็วที่ใช้โดยส่วนใหญ่มีดังนี้

2.2.1 เครื่องวัดความเร็วลมแบบแผ่นกระดาษ

เครื่องวัดความเร็วลมแบบแผ่นกระดาษเป็นแบบที่ง่ายที่สุด แต่มีความถูกต้องแม่นยำพอประมาณทำจากไม้ หรือโลหะ ทำเป็นแผ่นสี่เหลี่ยมติดอยู่บนยอดเขา สามารถหมุนไปตามทิศทางลม เมื่อมีลมมาปะทะแผ่นสี่เหลี่ยมก็จะกระดกขึ้นตามความแรงของลมที่มาปะทะ โดยจะสเกลและเข็มชี้ค่าบนสเกล

2.2.2 เครื่องวัดความเร็วลมแบบลูกถ้วย

เครื่องวัดแบบนี้ใช้การหมุนของลูกถ้วย ซึ่งเป็นลูกถ้วยครึ่งทรงกลม หรือครึ่งทรงกรวย เป็นแบบลูกถ้วย 3 ใบและ 4 ใบ โดยอาศัยลมที่มาปะทะลูกถ้วยทำให้แกนของลูกถ้วยหมุน และส่งสัญญาณจำนวนรอบมาให้เครื่องคำนวณเป็นค่าความเร็วลมอีกทีหนึ่ง

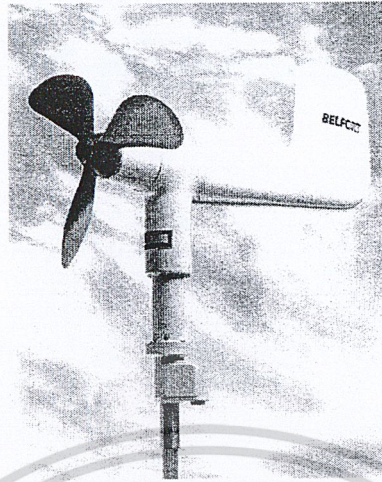


รูปที่ 2.1 เครื่องวัดความเร็วลมแบบลูกถ้วย

2.2.3 เครื่องวัดความเร็วลมแบบใช้ใบพัด (Aero vane)

เครื่องวัดแบบนี้จะมีเครื่องวัดความเร็วลม และทิศทางลมในตัวเดียวกัน มีรูปร่างลักษณะคล้ายเครื่องบิน ที่ส่วนหัวจะมีใบพัดสำหรับวัดความเร็วลม สำหรับส่วนที่เป็นลำคล้ายเครื่องบินจะทำหน้าที่เป็นคร่อมใช้สำหรับวัดทิศทางลม โดยมีหลักการทำงาน คือเมื่อมีลมมาปะทะกับเครื่องวัด ส่วนที่เป็นใบพัดก็จะหมุนแล้วส่งสัญญาณออกมาและแสดงผลเป็นค่าความเร็วลม และในขณะที่เดียวกันตัวเครื่องวัดก็จะหันไปตามทิศทางลมดังกล่าวและส่งสัญญาณออกมาเพื่อแสดงผลเป็นทิศทางต่าง ๆ

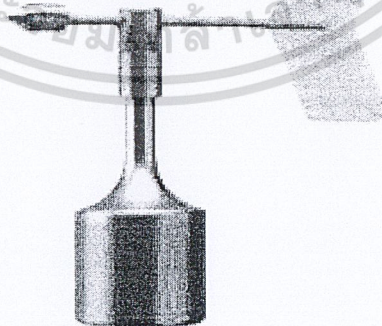
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 เครื่องวัดความเร็วลมแบบใช้ใบพัด

2.3 เครื่องวัดทิศทางลมหรือศรลม (Wind vane)

ศรลมเป็นอุปกรณ์ที่ใช้ตรวจสอบทิศทางลม โดยจะมีลักษณะเป็นลูกศรที่หางมีแผ่นรับลม เมื่อลมพัดมาปะทะหางลูกศรก็จะเกิดแรงผลักทำให้หัวลูกศรชี้ไปในทิศทางที่ลมพัดมา ศรลมนั้นมีหลายแบบแต่โดยทั่วไปแล้ว จะมีลักษณะปลายศรแหลม หรือรูปทรงกระบอก ส่วนหางศรมีรูปร่างคล้ายหางปลาบ้าง หางเครื่องบินบ้าง หางศรมีหน้าที่บังคับให้หัวศรลมหันรับลมตลอดเวลา เลี้ยวให้อยู่หนึ่งถึงบางครั้งจะเปลี่ยนทิศอย่างรวดเร็ว แต่ศรลมก็จะหมุนรับและชี้ทิศทางได้ทันทีทันที



รูปที่ 2.3 ศรลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ไมโครคอนโทรลเลอร์

2.4.1 ความหมายของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ เป็นชื่อของอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งซึ่งรวมเอาหน่วยคำนวณทางคณิตศาสตร์ ลอจิก(Arithmetic Logic Unit) และวงจรรับสัญญาณนาฬิกาไว้ด้วยกัน ซึ่งจะทำให้เราสามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี และช่วยลดจำนวนอุปกรณ์หรือขนาดของระบบ ในขณะที่มีขีดความสามารถสูงขึ้น ภายใต้งบประมาณที่เหมาะสม

ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกันคือ “ไมโคร” (Micro) ซึ่งหมายถึง ไมโครโปรเซสเซอร์ ซึ่งเป็นอุปกรณ์ประมวลผลที่มีขนาดเล็ก ภายในประกอบด้วยหน่วยประมวลผลกลางหรือซีพียู (Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก ซึ่งวงจรเชื่อมต่อหน่วยความจำและวงจรสัญญาณอีกคำหนึ่งคือ “คอนโทรลเลอร์” (Controller) หมายถึง อุปกรณ์ควบคุม ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยที่สามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

2.4.2 คุณสมบัติของไมโครคอนโทรลเลอร์ PIC16F877A

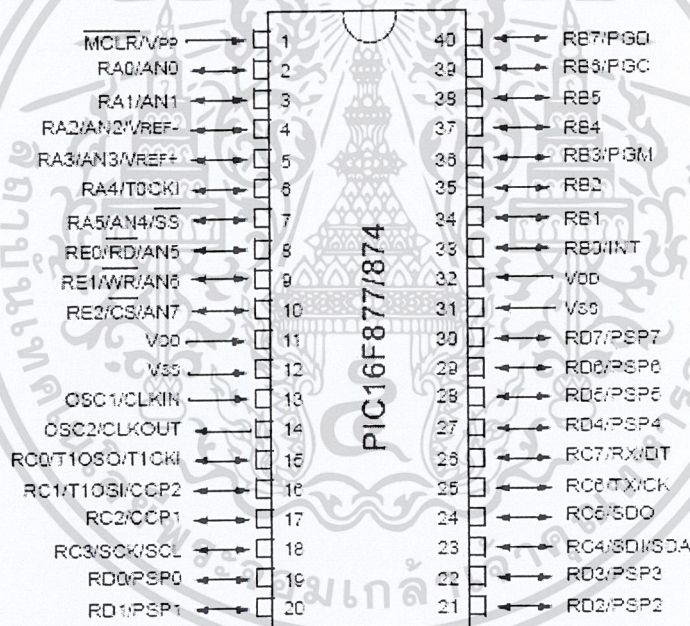
ไมโครคอนโทรลเลอร์ตระกูล PIC16FXXX ได้รับความนิยมมากที่สุดเพราะเป็นชิปรุ่นแรกที่หน่วยความจำโปรแกรมเป็นแบบแฟลช และมีหน่วยความจำข้อมูลแบบ EEPROM ไมโครคอนโทรลเลอร์ PIC16F877A เป็นชิปที่มีการใช้งานกันอย่างแพร่หลาย หาซื้อง่ายและราคาไม่แพงมากนัก ซึ่งมีคุณสมบัติหลักๆดังนี้

1. มีคำสั่งที่เป็นภาษาแอสเซมบลี 35 คำสั่ง
2. ใน 1 คำสั่งใช้เวลาทำงาน 1 ถึง 2 ไชเคิล
3. ทำงานได้สูงสุดที่สัญญาณนาฬิกาตั้งแต่ไฟตรงถึง 20 MHz
4. ทำงานแบบ Pipe-line สามารถทำงาน 2 อย่างในเวลาเดียวกันได้
5. หน่วยความจำโปรแกรมเป็นแบบ Flash มีขนาด 8k Word (1 word = 14 บิต)
6. มีหน่วยความจำข้อมูล (Data Memory RAM) ขนาด 368 ไบต์
7. มีหน่วยความจำข้อมูลแบบ EEPROM ขนาด 256 ไบต์
8. ตอบสนองการอินเทอร์รัปต์ได้ทั้งหมด 14 แหล่ง
9. มี Stack ให้ใช้ได้สูงสุด 8 ระดับ
10. มีระบบ Power On Reset , Power Up Timer , Oscillator Start-Up และ Watchdog Timer
11. มีระบบ Code Protection กันการคัดลอก
12. มีโหมดประหยัดพลังงาน (Sleep Mode)
13. สัญญาณนาฬิกามีหลายโหมดให้เลือกใช้งาน คือ อาจจะใช้ XTAL หรือ วงจร RC
14. สามารถโปรแกรมด้วยไฟ +5 VDC ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในวงการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. ใช้การโปรแกรมแบบ In – Circuit Serial Programming
16. ทำงานที่ไฟเลี้ยง 12 VDC ถึง 5.5 VDC
17. Current Sink และ Current Source อยู่ที่ 25 mA
18. มี Timer/Counter 3 ตัว คือ Timer 0 ขนาด 8 บิต, Timer 1 ขนาด 16 บิต และ Timer 2 ขนาด 8 บิต
19. มีโมดูล Capture/Compare/PWM (Pulse Width Modulation) 2 ชุด
20. มีวงจรแปลงสัญญาณอะนาลอกเป็นดิจิตอล (A/D Converter) ขนาด 10 บิต
21. มีระบบ USART สำหรับการสื่อสารแบบ RS232
22. มีระบบตรวจระดับไฟเลี้ยง (Brown-out Reset)
23. มี I/O พอร์ตทั้งหมด 5 พอร์ต แต่ละพอร์ตมีจำนวนบิตไม่เท่ากัน

2.4.3 การจัดขาของไมโครคอนโทรลเลอร์ PIC16F877A



รูปที่ 2.4 ขาสัญญาณของไมโครคอนโทรลเลอร์ PIC16F877A

ไมโครคอนโทรลเลอร์ตระกูล PIC เบอร์ 16F877A เป็นไมโครคอนโทรลเลอร์ขนาด 40 ขา มีขาสัญญาณต่างๆดังนี้

1. MCLR/Vpp: Master Clear (Reset) Input / Programming Voltage Input ทำหน้าที่เป็นขาสัญญาณรีเซ็ต (Reset) เมื่อขานี้ได้รับลอจิก 0 ไมโครคอนโทรลเลอร์จะถูกรีเซ็ตและทำหน้าที่เป็นขาสัญญาณรับแรงดัน ขณะทำการบันทึกโปรแกรมลงหน่วยความจำของไมโครคอนโทรลเลอร์
- เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. VDD: Positive Supply (+2.00 V ถึง +5.5 V) ทำหน้าที่เป็นขาไฟเลี้ยงของไมโครคอนโทรลเลอร์

3. VSS: Ground ทำหน้าที่เป็นกราวด์

4. OSC1/CLKIN: Oscillator Crystal Input / External Clock Source Input

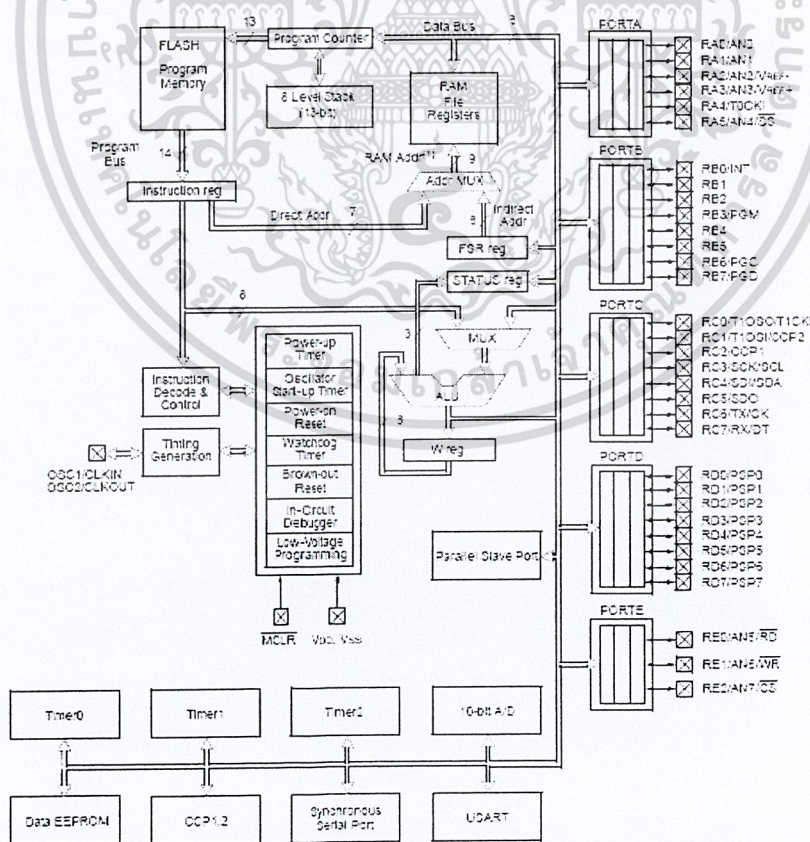
5. OSC2/CLKOUT: Oscillator Crystal Output / External Clock Source Output ทั้งสองขาทำหน้าที่เป็นขาสัญญาณสำหรับต่อคริสตัล ในกรณีที่อยู่ในโหมดการใช้สัญญาณนาฬิกาจากภายนอก (Crystal Oscillator Mode)

6. RA0-RA5: พอร์ต A มีจำนวน 6 ขา เป็นพอร์ตแบบสองทิศทาง (Bi-direction I/O Port) คือ เป็นได้ทั้งพอร์ตอนุกรมและพอร์ตเอาท์พุทใช้ในการส่งและรับข้อมูล นอกจากนี้ยังมีหน้าที่อื่นๆ

7. RB0-RB7: พอร์ต B มีจำนวน 8 ขา ขนาด 8 บิต เป็นพอร์ตแบบสองทิศทางใช้ในการส่งและรับข้อมูล นอกจากนี้บางขายังทำหน้าที่รับสัญญาณอินพุตจากการอินเทอร์รัปต์ (Interrupt) จากภายนอกด้วย

8. RC0 - RC7: พอร์ต C มีจำนวน 8 ขา ขนาด 8 บิต เป็นพอร์ตแบบสองทิศทาง ใช้ในการส่งและรับข้อมูล นอกจากนี้ยังทำหน้าที่อื่นๆ

9. RD0-RD7: พอร์ต D มีจำนวน 8 ขา ขนาด 8 บิต เป็นพอร์ตแบบสองทิศทาง ใช้ในการส่งและรับข้อมูล นอกจากนี้ยังทำหน้าที่อื่นๆ



รูปที่ 2.5 โครงสร้างภายในของไมโครคอนโทรลเลอร์ PIC16F877A

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 การจัดการหน่วยความจำของไมโครคอนโทรลเลอร์ 16F877A

ไมโครคอนโทรลเลอร์ 16F877A มีหน่วยความจำโปรแกรมแบบแฟลช ซึ่งใช้สำหรับเก็บโปรแกรมควบคุมการทำงานของ PIC มีหน่วยความจำข้อมูล (Data Memory RAM) ขนาด 368 ไบต์ หน่วยความจำของข้อมูลแบบอีอีพรอม (Data Memory EEPROM) ขนาด 256 ไบต์ และได้มีการจัดการหน่วยความจำของ PIC ออกเป็นแบงก์ต่างๆ ได้ทั้งหมด 4 แบงก์ คือ Bank 0 – Bank 3 และแต่ละแบงก์มีขนาด 128 ไบต์ ดังรูป

File Address	File Address	File Address	File Address
Indirect addr. ⁽¹⁾ 00h	Indirect addr. ⁽¹⁾ 50h	Indirect addr. ⁽²⁾ 100h	Indirect addr. ⁽¹⁾ 120h
TMR0 01h	OPTION_REG 51h	TMR0 101h	OPTION_REG 121h
PCL 02h	PCL 52h	PCL 102h	PCL 122h
STATUS 03h	STATUS 53h	STATUS 103h	STATUS 123h
FSR 04h	FSR 54h	FSR 104h	FSR 124h
PORTA 05h	TRISA 55h		
PORTE 06h	TRISB 56h	PORTE 105h	TRISB 125h
PORTC 07h	TRISC 57h		
PORT ⁽¹⁾ 08h	TRISD ⁽¹⁾ 58h		
PORT ⁽¹⁾ 09h	TRISE ⁽¹⁾ 59h		
PCLATH 0Ah	PCLATH 5Ah	PCLATH 10Ah	PCLATH 12Ah
INTCON 0Bh	INTCON 5Bh	INTCON 10Bh	INTCON 12Bh
PIR1 0Ch	PIE1 5Ch	EEDATA 10Ch	ECON1 12Ch
PIR2 0Dh	PIE2 5Dh	EEDADR 10Dh	ECON2 12Dh
TMR1L 0Eh	PCON 5Eh	EEDATH 10Eh	Reserved ⁽²⁾ 12Eh
TMR1H 0Fh		EEDARH 10Fh	Reserved ⁽²⁾ 12Fh
T1CON 10h			
TMR2 11h	SSPCON2 51h		
T2CON 12h	PR2 52h		
SSPBUF 13h	SSPAD 53h		
SSPCON 14h	SSPSTAT 54h		
CCP1L 15h			
CCP1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 58h	General Purpose Register 12 Bytes 17h	General Purpose Register 12 Bytes 127h
TXREG 19h	SFBRG 59h		
RCREG 1Ah			
CCP2L 1Bh			
CCP2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 5Eh		
ADCON0 1Fh	ADCON1 5Fh		
General Purpose Register 80 Bytes 7Fh	General Purpose Register 50 Bytes EFh	General Purpose Register 20 Bytes 17Fh	General Purpose Register 30 Bytes 1FFh
	accesses 7Ch-7Fh F0h	accesses 7Ch-7Fh 17Fh	accesses 70h-7Fh 1FDh

Legend:
 [] Unimplemented data memory locations, read as 0.
 [] Not a physical register.
 Note 1: These registers are not implemented on the PIC16F877E.
 Note 2: These registers are reserved, mainly in these registers 0-az.

รูปที่ 2.6 การจัดการพื้นที่หน่วยความจำของไมโครคอนโทรลเลอร์ PIC16F877A

แบงก์ 0 และแบงก์ 1 จะแบ่งออกเป็น 2 ส่วนเหมือนกันคือ ส่วนที่ 1 ขนาด 32 ไบต์ เป็นพื้นที่ของรีจิสเตอร์ไฟล์ที่เรียกว่า SFR (Special Function Register) ซึ่งใช้สำหรับกำหนดเงื่อนไขการทำงานและบันทึกสถานะการทำงานของไมโครคอนโทรลเลอร์ ส่วนที่ 2 ขนาด 96 ไบต์ ซึ่งเป็นพื้นที่หน่วยความจำสำหรับใช้งานทั่วไป ซึ่งใช้สำหรับเก็บผลลัพธ์และเงื่อนไขต่างๆ ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบงก์ 2 และแบงก์ 3 จะแบ่งออกเป็น 2 ส่วน แต่มีขนาดแตกต่างกันแบงก์ 0 และแบงก์ 1 คือ ส่วนที่ 1 ขนาด 16 ไบต์ เป็นพื้นที่ของรีจิสเตอร์ไฟล์ ส่วนที่ 2 ขนาด 112 ไบต์ เป็นพื้นที่หน่วยความจำสำหรับใช้งานทั่วไป

2.4.5 กระบวนการรีเซ็ต

การรีเซ็ต (Reset) ของไมโครคอนโทรลเลอร์ คือการทำให้ไมโครคอนโทรลเลอร์ เริ่มต้นทำงานใหม่ ซึ่งเมื่อไมโครคอนโทรลเลอร์ถูกรีเซ็ต จะกระโดดไปทำโปรแกรมที่ตำแหน่ง 0000H ดังนั้นตำแหน่งของโปรแกรมจะต้องเก็บไว้ที่ตำแหน่งเริ่มต้น 0000H การรีเซ็ตจะช่วยแก้ปัญหาและข้อผิดพลาดของโปรแกรม ซึ่งการรีเซ็ตของ PIC16F877A สามารถเกิดขึ้นได้ 6 กรณี คือ

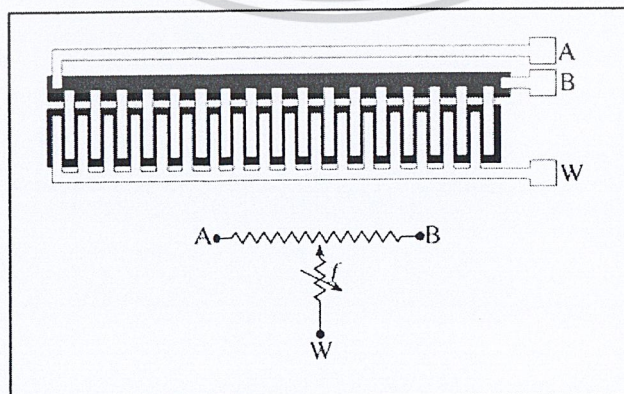
1. เพาเวอร์ออนรีเซ็ต (Power on Reset)
2. การรีเซ็ตค่าที่ขา MCLR ในโหมดปกติ
3. การรีเซ็ตที่ขา MCLR ในโหมดประหยัดพลังงาน
4. การรีเซ็ตจากวอตช์ดอกไทเมอร์ ในโหมดปกติ
5. การรีเซ็ตจากวอตช์ดอกไทเมอร์ ในโหมดประหยัดพลังงาน
6. การรีเซ็ตเนื่องจากไม่มีไฟเลี้ยงหรือไฟเลี้ยงลดต่ำลง

2.5 โปเทนชิโอมิเตอร์

ทำหน้าที่เปลี่ยนแปลงการเคลื่อนที่เชิงมุมไปเป็นค่าความต้านทาน คือ หากเราจ่ายแรงเคลื่อนทางอินพุต จะทำให้สัญญาณแรงเคลื่อนและกระแสที่ออกมาทางเอาต์พุตเปลี่ยนแปลงไปตามระยะทาง เนื่องจากการเลื่อนของ (Wiper) ที่ตัวโปเทนชิโอมิเตอร์

2.5.1 ชนิดของโปเทนชิโอมิเตอร์

(1) โปเทนชิโอมิเตอร์แบบเชิงเส้น (Linear Potentiometer) ใช้ในการวัดระยะทางที่เป็นเส้นตรง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.7 แสดงการตรวจจกระยะโดยใช้โปเทนชิโอมิเตอร์แบบเชิงเส้น
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) โปเทนชิโอมิเตอร์แบบเชิงมุม (Rotary Potentiometer) ใช้สำหรับวัดระยะในลักษณะการหมุน โปเทนชิโอมิเตอร์วัดระยะเชิงมุมบางตัวสามารถวัดระยะเชิงมุมได้ถึง 10 องศา หากมันหมุนเต็มรอบจะได้ 357 องศา แต่ในแบบหลายรอบอาจจะมีมุมในการหมุนได้ถึง 3500 องศา หรือมากกว่า ความเร็วรอบสูงสุดที่โปเทนชิโอมิเตอร์แบบไวร์วาร์ดหมุนได้ประมาณ 300 รอบ/วินาที หากมากกว่านี้จะทำให้เกิดสัญญาณรบกวนซึ่งเป็นผลมาจากการไม่สัมผัสกันอย่างต่อเนื่องของแขน ซึ่งความเร็วในการหมุนของโปเทนชิโอมิเตอร์แบบต่อเนื่องอาจจะหมุนได้ถึง 2000 รอบ/นาที โปเทนชิโอมิเตอร์แบบไวร์ที่บางมาก ๆ จะมีเส้นผ่านศูนย์กลางเพียง 0.01 มิลลิเมตรเท่านั้น โดยลวดที่ใช้อาจทำจากแพลทินัม หรือผสมนิกเกิลที่นำมาพันอยู่บนฟอรัม

2.5.2 ความละเอียดของโปเทนชิโอมิเตอร์

คือการเปลี่ยนแปลงความต้านทาน ค่าน้อยที่สุดเมื่อแขนของโปเทนชิโอมิเตอร์เคลื่อนที่จากขดลวดช่วงหนึ่งไปยังอีกช่วงหนึ่งดังนั้นความละเอียดจึงกำหนดได้จาก

$$\text{Resolution} = \frac{\text{full scale displacement}}{\text{number of turn of wire}} \quad (2.2)$$

หากพิจารณาถึงวัสดุที่ใช้ทำโปเทนชิโอมิเตอร์ที่จากฟิล์มคาร์บอน ฟิล์ม โลหะพลาสติกตัวนำ (Conductive plastic) และโลหะเซรามิก (Ceramic metal) มีข้อได้เปรียบเหนือกว่าแบบไวร์ อย่างแรก คือมีความละเอียดมากกว่า เพราะไม่มีรอบของลวดที่พันบนแกนมาเกี่ยวข้องทำให้แขน สามารถเคลื่อนที่คร่อมผ่านผิวที่เรียบของมันได้ (โดยไม่สัมผัสกับลวดจากรอบหนึ่งไปยังอีกรอบหนึ่ง) ทำให้ผลของการสึกกร่อน การสัมผัสที่ขาดช่วง และผลจากความผิดพลาดมีค่าลดลง

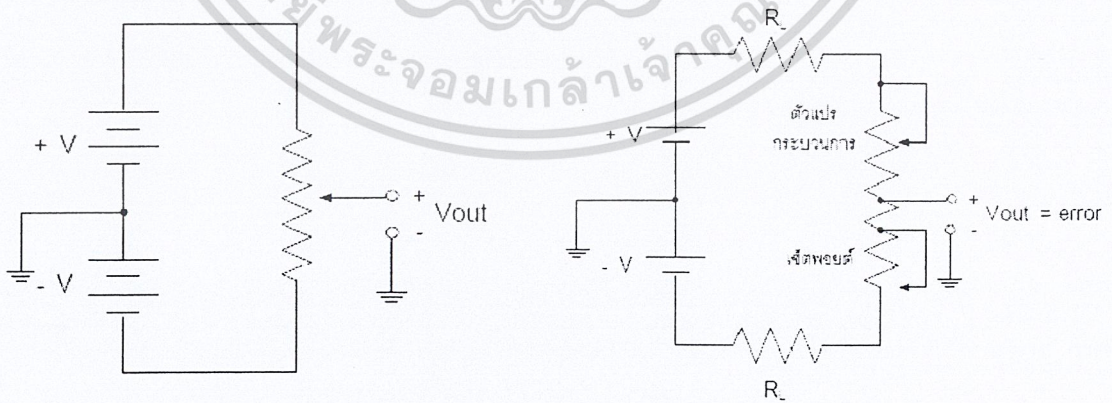
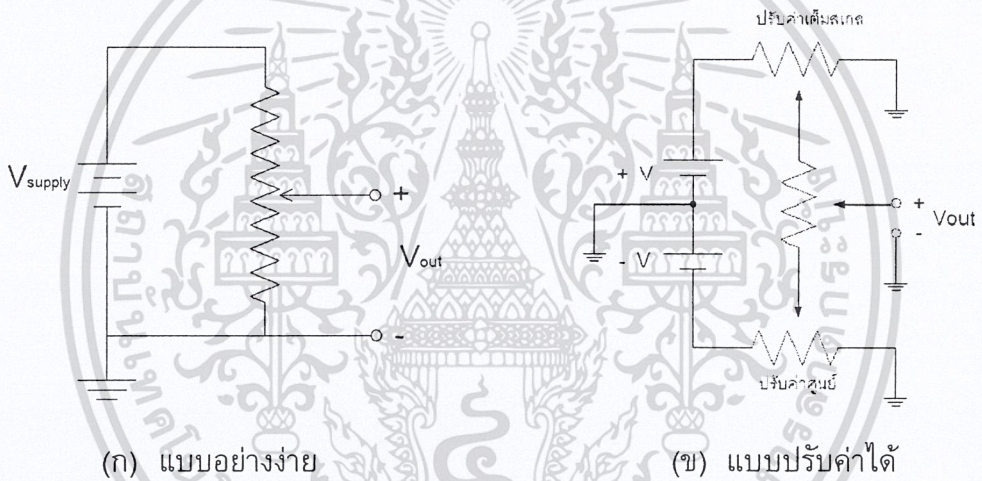
2.5.3 การปรับสภาพสัญญาณ

สัญญาณรบกวนของโปเทนชิโอมิเตอร์จะเกิดจากวัสดุที่ใช้ทำ พบว่าค่ารบกวนดังกล่าวเป็นสัดส่วนโดยตรงกับอุณหภูมิและกระแสของตัวมัน ดังนั้นเพื่อให้มีค่ารบกวนน้อยที่สุดจึงต้องรักษาแรงเคลื่อนที่จ่ายให้มีค่าต่ำสุด นอกจากนี้สัญญาณรบกวนยังเกิดจากการเคลื่อนของแขนได้อีกด้วย โปเทนชิโอมิเตอร์แบบต่อเนื่องจะสามารถหมุนได้เร็วกว่า และสร้างสัญญาณรบกวนน้อยกว่าแบบไวร์วาร์ด แต่ก็พบว่าในโปเทนชิโอมิเตอร์แบบไวร์วาร์ด จะแสดงค่าความเหนี่ยวนำและค่าการเก็บประจุออกมาด้วย ที่ความถี่ 50 เฮิร์ตซ์ ค่ารีแอกแตนซ์จะมีค่าอยู่ในย่านเมกะโอห์ม แต่ถ้ากระตุ้นโปเทนชิโอมิเตอร์ด้วยคลื่นไซน์ขนาด 10 กิโลเฮิร์ตซ์ ค่ารีแอกแตนซ์จะตกลงมาอยู่ในย่านกิโลโอห์ม ดังนั้นหากต้องมีการกระตุ้นสัญญาณ ต้องรักษาความถี่และความต้านทานพื้นฐานให้ต่ำที่สุดเท่าที่เป็นไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีหลากหลายวิธีที่จะต่อโพเทนชิโอมิเตอร์ใช้งานในระบบควบคุมวิธีง่าย ๆ คือการขับแสดงได้ในรูป (ก) จะทำให้ได้แรงเคลื่อนเอาต์พุตเป็นศูนย์ เมื่อแขนถูกขับไปยังด้านล่างสุด และจะได้แรงเคลื่อนเป็น V_{supply} เมื่อแขนอยู่ด้านบน ส่วนในรูป (ข) จะมีหลักการทำงานคือแขนไม่ขับไปยังจุดปลายสุดของโพเทนชิโอมิเตอร์ ก็สามารถออกแบบให้มีการปรับศูนย์ที่ตำแหน่งใดก็ได้ตามต้องการ การวางโพเทนชิโอมิเตอร์เพื่อขับปลายทั้งคู่ของมัน โดยกำหนดให้ศูนย์โวลต์อยู่ที่จุดศูนย์กลางจะแสดง ดังรูป (ค) และในวงจรนี้จะสามารถสร้างสัญญาณผิดพลาดที่อ้างอิงกับกราวด์ เพื่อใช้กับเครื่องควบคุมกระบวนการได้ ในรูป (ง) โพเทนชิโอมิเตอร์ที่ใช้ปรับเซตพอยต์ (Set point) จะติดตั้งอยู่บนแผงควบคุม (Control panel) ส่วนค่าตัวแปรกระบวนการ (Process variable) ซึ่งจะขับโดยตรงจากตัวแปรที่วัดได้ทำให้ได้ค่าความผิดพลาด (Error) เป็นต้น

$$\text{Error} = \text{SP} - \text{PV} \tag{2.3}$$



(ค) แบบปลายคู่ (double - ended)

(ง) แบบให้ค่าเอาต์พุตที่มีความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเซตพอยต์ (SP) และตัวแปรกระบวนการ (PV) มีค่าหรือตำแหน่งเดียวกัน ความต้านทานเดียวกัน ก็จะไม่มีความผิดพลาดเกิดขึ้น นั่นคือแรงเคลื่อนทางต้านเอาต์พุตจะมีค่าเป็นศูนย์ แต่ถ้าแขนที่ต่อกับเซตพอยต์เคลื่อนที่ขึ้นค่าความต้านทานก็จะเพิ่มขึ้น V_{out} ก็เป็นบวก ทำให้เกิดกระบวนการตอบสนองโดยการเคลื่อนที่แขนด้านบนของตัวแปรกระบวนการที่เป็นสัดส่วนเดียวกัน จะทำให้ค่า V_{out} กลับมาเป็นศูนย์โวลต์ แต่ถ้าเซตพอยต์มีค่าต่ำกว่านี้ค่าที่ได้จะตรงข้ามกัน ตัวต้านทาน R_L ที่ใส่เพิ่มเข้าไปนั้นเพื่อกำจัดการไหลของกระแส เมื่อเซตพอยต์เคลื่อนที่ลงจนหมด

ทั้งหมดที่ผ่านมายังเป็นกรณีที่ไม่มีการขับโหลด หากโพเทนชิโอมิเตอร์ที่วัดตำแหน่งต้องการมีการขับโหลด จะทำให้เกิดความไม่เป็นเชิงเส้นที่หลายๆย่านทำให้เกิดผลเสียต่อทรานสดิวเซอร์เอง เอาต์พุตที่ต้องการ (สมมติให้ไม่มีการโหลด) มีค่าเป็น

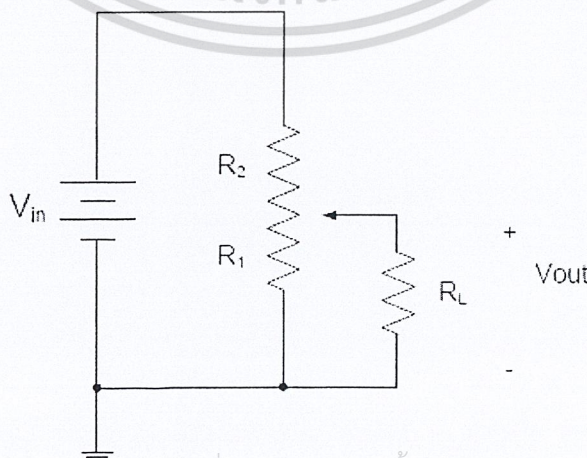
$$V_{desired} = \frac{R_1}{R_1 + R_2} V_{in} \tag{2.4}$$

อย่างไรก็ตามถ้าต่อ R_L ขนาดกับ R_1 ทำให้มีผลกลายเป็น

$$V_{actual} = \frac{[R_1 R_L / (R_1 + R_L)]}{[R_1 R_L + (R_1 + R_L)] + R_2} V_{in} \tag{2.5}$$

หรือเขียนให้ง่ายขึ้น จะได้ว่า

$$V_{actual} = \frac{R_1 R_L}{R_1 R_L + R_1 R_2 + R_2 R_L} \tag{2.6}$$



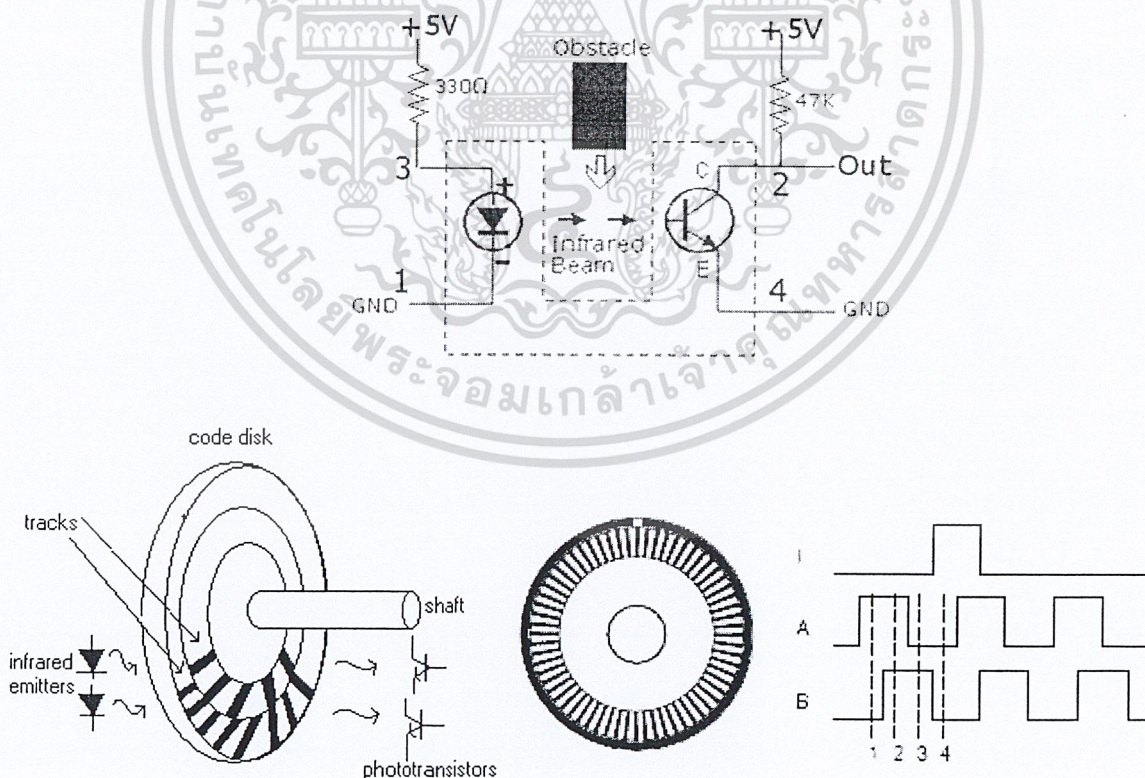
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น **รูปที่ 2.9** แสดงโพเทนชิโอมิเตอร์เมื่อขับโหลด เอกสารทุกครั้งที่มีการนำไปใช้

2.6 ตัวตรวจจับที่ใช้ตรวจจับความเร็วลม

ตัวตรวจจับความเร็วลมของเครื่องวัดความเร็วลมแบบใช้ถ้วยหมุนนั้น ที่ใช้กันอยู่มีอยู่ด้วยกัน 2 แบบคือ

1. ใช้อาการหมุนของเครื่องกำเนิดไฟฟ้าเล็กๆ (Electrical generator) ต่อกระแสไฟฟ้าจากเครื่องกำเนิดไฟฟ้านี้ไปเข้าหน้าปัด ซึ่งแบ่งสเกลไว้เป็นค่าของความเร็วลมอาศัยหลักการหมุน

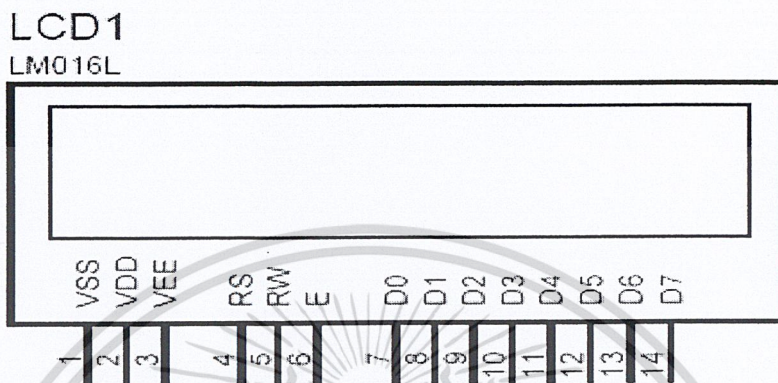
2. ใช้หลักการหมุนของแผ่นตรวจจับ เมื่อแผ่นตรวจจับหมุนจะได้สัญญาณออกมาเป็นพัลส์ซึ่งเป็นสัญญาณรูปสี่เหลี่ยม เมื่อแผ่นตรวจจับหมุนครบหนึ่งรอบจะได้สัญญาณที่เป็นรูปพัลส์ออกมาได้ถึง 20 พัลส์ ซึ่งการทำงานของตัวตรวจจับความเร็วลมแบบนี้ เมื่อหมุนแผ่นตรวจจับตัวรับสัญญาณก็สามารถรับสัญญาณจากตัวส่งได้โดยผ่านร่องแผ่นตรวจจับในด้านสว่างก็ได้สัญญาณเอาต์พุตออกมาสภาวะหนึ่ง และเมื่อหมุนแผ่นตรวจจับถึงด้านมืด สัญญาณจากตัวส่งจะถูกกันไว้ตัวรับก็ไม่สามารถรับสัญญาณนั้นได้ สัญญาณที่ออกมาทางด้านเอาต์พุตก็จะออกมาอีกสภาวะหนึ่งในขณะที่หมุนแผ่นตรวจจับอย่างต่อเนื่อง สัญญาณที่ได้จะสลับกันอย่างต่อเนื่องด้วยซึ่งสัญญาณรูปพัลส์ที่เกิดขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษายกเว้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.10 แสดงหลักการตรวจจับความเร็วลม
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 โครงสร้างและการทำงานของจอแสดงผล LCD

การใช้งานจอแสดงผล LCD ร่วมกับ PIC จะต้องเขียนโปรแกรมเพื่อควบคุมการแสดงผลที่หน้าจอ โดยการส่งข้อมูลออกทางบัสข้อมูล (Data Bus) ของจอแสดงผล LCD โดยการส่งข้อมูลมี 2 แบบ คือ แบบ 8 บิต ส่งข้อมูลออกขา D0 – D7 และแบบ 4 บิต ส่งข้อมูลออกขา D4 – D7 นอกจากนี้ยังมีขาสัญญาณต่างๆ อีกหลายขา ดังรูป



รูปที่ 2.11 ขาสัญญาณของจอแสดงผล LCD ขนาด 16x2

ขา 1 VSS ต่อลงกราวด์
 ขา 2 VDD เป็นขาไฟเลี้ยงให้กับจอแสดงผล LCD ขนาด +5 VDC
 ขา 3 VEE เป็นขาที่ใช้สำหรับปรับค่าความสว่างของหน้าจอแสดงผล LCD
 ขา 4 RS เป็นขาแสดงสถานะของข้อมูลที่ส่งออกทางขา data ว่าเป็นคำสั่งหรือข้อมูล โดยถ้าขานี้เป็นลอจิก 0 ข้อมูลที่ส่งเข้ามาคือคำสั่ง ถ้าขานี้เป็นลอจิก 1 ข้อมูลที่ส่งเข้ามาคือข้อมูลสำหรับแสดงผล

ขา 5 RW เป็นขาแสดงสถานะของการอ่านหรือเขียนข้อมูลของจอแสดงผล LCD โดยถ้าขานี้เป็นลอจิก 0 จะเป็นการเขียนข้อมูลลงในจอแสดงผล LCD ถ้าขานี้เป็นลอจิก 1 จะเป็นการอ่านข้อมูลจากจอแสดงผล LCD

ขา 6 E เป็นขากำหนดการทำงานของจอแสดงผล LCD

ขา 7 – 14 (D0-D7) เป็นขาสำหรับเขียนหรืออ่านข้อมูลระหว่างจอแสดงผล LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

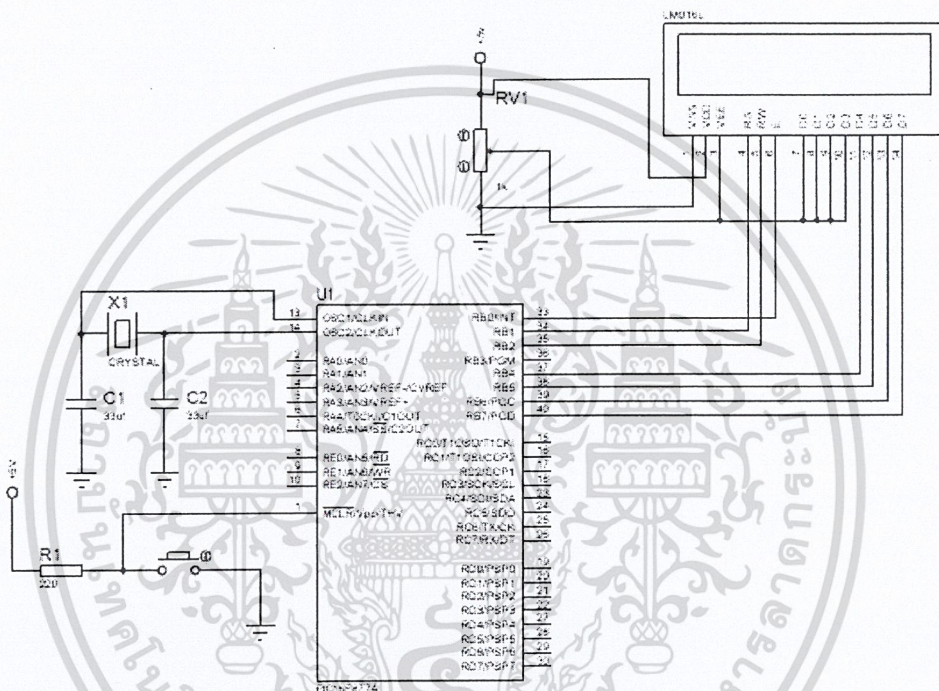
จอแสดงผล LCD ที่มีจำหน่ายกันทั่วไป มีมากมายหลายแบบและหลายขนาด เช่น ขนาด 16 ตัวอักษร จนถึง 40 ตัวอักษร หรือจำนวนบรรทัดมีตั้งแต่ 1 บรรทัดจนถึง 4 บรรทัด ซึ่งจำนวนขาที่ต่อใช้งานจะมีจำนวนเท่ากัน แต่จำนวนตัวอักษรและจำนวนบรรทัดที่แตกต่างกันสามารถแสดงผลได้ โดยการเขียนโปรแกรมจัดตำแหน่งหน่วยความจำของจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 การเชื่อมต่อไมโครคอนโทรลเลอร์กับจอแสดงผล LCD

การเชื่อมต่อจอแสดงผล LCD แบบ 4 บิต

การเชื่อมต่อไมโครคอนโทรลเลอร์กับจอแสดงผล LCD ขนาด 4 บิต ขาที่ใช้สำหรับส่งข้อมูลระหว่างจอแสดงผล LCD กับไมโครคอนโทรลเลอร์จะใช้ 4 ขา หรือ ขา D4-D7 สำหรับขา 1 (VSS) ต่อเข้ากับกราวด์ ขา 2 (VDD) ต่อเข้ากับไฟ +5VDC ขา 3 (VEE) ต่อเข้ากับตัวต้านทานปรับค่าได้ เพื่อทำการปรับค่าความสว่างของหน้าจอ ขา 4 (RS) ต่อเข้ากับขา RB1 ขา 5 (RW) ต่อเข้ากับขา RB2 และ ขา 6 (E) ต่อเข้ากับขา RB0 ดังรูป



รูปที่ 2.12 การเชื่อมต่อไมโครคอนโทรลเลอร์กับจอแสดงผล LCD แบบ 4 บิต

ในการเลือกใช้การเชื่อมต่อจอแสดงผล LCD แบบ 4 บิต เพราะจะมีฟังก์ชันต่างๆ อยู่ใน CCS C คอมไพเลอร์ รวมถึง Source Code จะอยู่ใน folder Drivers (C:\program file\PIICC\Drivers) ซึ่งสามารถเรียกใช้ได้เลย การเรียกใช้ไฟล์ lcd.c สามารถเรียกใช้ได้โดยการประกาศ `#include <lcd.c>` ที่ไฟล์ส่วนหัวของโปรแกรม เมื่อทำการประกาศที่ส่วนหัวแล้วก็จะสามารถเรียกใช้ได้ทุกฟังก์ชันใน lcd.c ซึ่งทำให้สะดวกและเขียนโปรแกรมง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 โครงสร้างการสร้างสัญญาณอินเทอร์รัปต์

ไทเมอร์/เคาน์เตอร์และวอตช์ด็อกไทเมอร์

ไทเมอร์ (Timer) ของไมโครคอนโทรลเลอร์ PIC สามารถกำหนดโหมดการทำงานได้สองรูปแบบ คือ

การทำงานในโหมดไทเมอร์

หรือตัวจับเวลาเป็นการนับสัญญาณนาฬิกาภายในตัวไมโครคอนโทรลเลอร์ โดยสัญญาณที่ใช้ นับเกิดจากความถี่ของออสซิลเลเตอร์ ที่ถูกหารด้วย 4 ($F_{osc}/4$) หรือการนับ 1 ครั้งจะใช้เวลา เท่ากับ $4/F_{osc}$ (จาก $T=1/f$)

การทำงานในโหมดเคาน์เตอร์

จะเป็นการนับสัญญาณนาฬิกาจากภายนอก ที่ผ่านเข้ามาทางขาพอร์ตที่ได้กำหนดไว้ เช่น ใน ไมโครคอนโทรลเลอร์ PIC 16F877A ขาพอร์ตของไทเมอร์ 0 เมื่อใช้นับสัญญาณนาฬิกาจาก ภายนอกจะถูกกำหนดไว้ที่ขา RA4/T0CK1 และไทเมอร์ 1 ที่ขา RC0/T1CK1 เป็นต้น

2.8.1 ไทเมอร์ในไมโครคอนโทรลเลอร์ PIC16F877A

ไมโครคอนโทรลเลอร์ PIC16F877A มีไทเมอร์ให้ใช้งาน 3 ตัว คือ ไทเมอร์ 0 (Timer 0), ไทเมอร์ 1 (Timer 1) และไทเมอร์ 2 (Timer 2) โดยมีรายละเอียดดังนี้

- ไทเมอร์ 0 ขนาด 8 บิต เป็นได้ทั้งไทเมอร์และเคาน์เตอร์

อินพุตรับสัญญาณนาฬิกาได้ทั้งจากภายใน (Instruction Clock) โดยสัญญาณนาฬิกาได้มาจาก $F_{osc}/4$ และสัญญาณนาฬิกาจากภายนอก (External Clock) ผ่านขา RA4/T0CK1

- ช่วงการนับ 0-255 นับค่าได้สูงสุด 256

- มีปริสเกลเลอร์ (การหารสัญญาณนาฬิกา) 1,2,4,5,6,8,16,32,64,128,256

- สร้างสัญญาณอินเทอร์รัปต์ได้เมื่อเกิดโอเวอร์โฟลว์ เนื่องจากการนับ

- ไทเมอร์ 1 ขนาด 16 บิต เป็นได้ทั้งไทเมอร์และเคาน์เตอร์

อินพุตรับสัญญาณนาฬิกาได้ทั้งจากภายใน (Instruction Clock) โดยสัญญาณนาฬิกา ได้มาจาก $F_{osc}/4$ และภายนอก (external Clock) ผ่านขา RC0/T1OSO/T1CK1

ช่วงการนับ 0-65535 นับค่าได้สูงสุด 65536 มีปริสเกลเลอร์ (การหารสัญญาณนาฬิกา) 1,2,4,8 สร้างสัญญาณอินเทอร์รัปต์ได้เมื่อเกิดโอเวอร์โฟลว์เนื่องจากการนับไทเมอร์ 2 ขนาด 8 บิต เป็น

ไทเมอร์เท่านั้น อินพุตรับสัญญาณนาฬิกาได้จากภายใน (Instruction Clock) โดยสัญญาณ นาฬิกาได้มาจาก $F_{osc}/4$ เท่านั้น กำหนดขอบเขตการนับได้ โดยกำหนดค่าในรีจิสเตอร์ คาบเวลา (PR2) ช่วงการนับ 0-1 ถึง 0-255 มีปริสเกลเลอร์ (การหารสัญญาณนาฬิกา) 1,4,16

สร้างสัญญาณอินเทอร์รัปต์ได้เมื่อเกิดโอเวอร์โฟลว์ได้ 1-16 ช่วง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 ความรู้พื้นฐานเกี่ยวกับไทเมอร์

เนื่องจากการทำงานของไทเมอร์เมื่อเป็นการนับสัญญาณพิกากภายในหรือทำงานในโหมดไทเมอร์ ค่าเวลาในการนับแต่ละครั้งจะถูกหารด้วย 4 หรือเท่ากับ $4/F_{osc}$ (F_{osc} คือ ความถี่ของออสซิลเลเตอร์ที่ใช้งาน) พื้นฐานที่สำคัญที่ต้องทำความเข้าใจก่อนคือเรื่องของเวลาในการนับแต่ละครั้ง เพื่อที่จะได้สามารถนำไทเมอร์ไปประยุกต์ใช้งานได้ต่อไป

2.8.3 การหาค่าเวลาในการนับแต่ละครั้ง (Time in Tick)

เมื่อใช้ความถี่ออสซิลเลเตอร์เท่ากับ 4 MHz เราสามารถหาค่าเวลาในการนับแต่ละครั้ง หรือการเพิ่มค่าแต่ละครั้งในรีจิสเตอร์ TMRx (x คือหมายเลขกำหนดของไทเมอร์ที่ใช้งาน) ได้ดังนี้จากสูตร

$$T = 1/f$$

เพราะฉะนั้นเวลา

$$T = 1/ [F_{osc}/4]$$

$$= 4/F_{osc}$$

$$= 4/ (4 \text{ MHz})$$

$$= 1 \text{ ไมโครวินาที}$$

หมายความว่าในการนับของไทเมอร์แต่ละครั้งใช้เวลาเท่ากับ 1 ไมโครวินาที ที่ความถี่ออสซิลเลเตอร์ 4 MHz เมื่อใช้ความถี่ออสซิลเลเตอร์เท่ากับ 20 MHz เราสามารถหาค่าเวลาในการนับแต่ละครั้งได้ดังนี้

$$T = 1/ [F_{osc}/4]$$

$$= 4/F_{osc}$$

$$= 4/ (20 \text{ MHz})$$

$$= 0.2 \text{ ไมโครวินาที}$$

หมายความว่าเมื่อใช้ความถี่ออสซิลเลเตอร์ 20 MHz ในการหนึ่งครั้งไทเมอร์ จะใช้เวลาเท่ากับ 0.2 ไมโครวินาที หรือเมื่อเวลาผ่านไป 0.2 ไมโครวินาที ค่าในรีจิสเตอร์ TMRx จะเพิ่มขึ้น 1 ค่า เช่น ไทเมอร์ 0 ของ PIC16F877A มีขนาด 8 บิต หรือนับค่าได้สูงสุดเท่ากับ 255 (เกิดจาก 2^8) เมื่อนับค่าถึง 255 แล้วจะกลับไปเมื่อเริ่มต้นนับค่า 0 ใหม่อีกครั้งเรียกเหตุการณ์นี้ว่าการเกิดโอเวอร์โฟลว์ เมื่อรวมการนับจาก 255 และวนกลับไปนับที่ 0 จะนับได้เท่ากับ 256 ครั้ง แล้วเมื่อเริ่มนับค่า 0 ใหม่วนการนับแบบนี้ไปตลอดการทำงานของไทเมอร์ และถ้ามีการกำหนดการอินเตอร์รัปต์เนื่องจากไทเมอร์เมื่อเกิดโอเวอร์โฟลว์ก็จะเกิดอินเตอร์รัปต์ตามมา จึงสามารถหาค่าเวลาในการนับทั้งหมดเมื่อเกิดโอเวอร์โฟลว์ได้เท่ากับ (เวลาที่ใช้ในการนับทั้งหมดจนเกิดโอเวอร์โฟลว์)

$$T_{\text{overflow}} = 256 \times 0.2 \text{ ไมโครวินาที}$$

$$= 51.2 \text{ ไมโครวินาที}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพราะฉะนั้นหากต้องการใช้ไทเมอร์นับเวลา 1 วินาที จะต้องเกิดโอเวอร์โฟลว์กี่ครั้ง หาได้ดังนี้

เวลา 51.2 ไมโครวินาที เกิดโอเวอร์โฟลว์ 1 ครั้ง

เวลา 1 วินาที เกิดโอเวอร์โฟลว์ $(1 \times 1) / 51.2 \text{ us}$

0.01953125 us

19531.25 ครั้ง

จะเห็นว่าเวลาในการนับเวลา 1 วินาที จะเกิดโอเวอร์โฟลว์ทั้งสิ้น 19531.25 ครั้ง

2.8.4 การหาค่าเวลาในการนับแต่ละครั้งเกี่ยวกับปริสเกลเลอร์

เนื่องจากไทเมอร์ของไมโครคอนโทรลเลอร์ PIC มีคุณสมบัติการหารความถี่ด้วยค่าปริสเกลเลอร์เพื่อลดทอนความถี่ลง เราจึงสามารถนำค่าปริสเกลเลอร์มาใช้ลดทอนความถี่ทำให้การนับแต่ละครั้งใช้เวลามากขึ้น แต่ค่าความละเอียดของเวลาจะลดลง เช่น ที่ความถี่ออสซิลเลเตอร์ 20 MHz การนับ 1 ครั้งใช้เวลาเท่ากับ 0.2 ไมโครวินาที เกิดโอเวอร์โฟลว์ใช้เวลาเท่ากับ 51.2 ไมโครวินาที เมื่อใช้ปริสเกลเลอร์ (ที่ไทเมอร์ 0 กำหนดค่าปริสเกลเลอร์ได้ 1,2, 4,5,6,8,16,32,64,128,256) จะเกิดการเปลี่ยนแปลงของเวลาดังนี้

การนับ 1 ครั้งใช้เวลาเท่ากับ

$$\begin{aligned} T &= 1 / [F_{osc} / 4 \times 256] \\ &= (4 \times 256) / (20 \text{ MHz}) \\ &= 51.2 \text{ us} \end{aligned}$$

$$\begin{aligned} T_{\text{overflow}} &= 51.2 \times 256 \\ &= 13107.2 \text{ ไมโครวินาที} \\ &= 13.1072 \end{aligned}$$

นั่นหมายความว่า การเกิดโอเวอร์โฟลว์หนึ่งครั้งใช้เวลาเท่ากับ 13.1072 มิลลิวินาที หากต้องการเวลา 1 วินาที ทำให้เขียนโปรแกรมหาเวลา 1 วินาทีได้ด้วยการนับการเกิดโอเวอร์โฟลว์ของไทเมอร์ 0 ให้ได้ 76 ครั้ง

จะเห็นได้ว่าการใช้งานปริสเกลเลอร์เข้ามาช่วยหารความถี่จะทำให้เราสามารถนับเวลาได้นานขึ้นแต่ความละเอียดของเวลาในการนับแต่ละครั้งจะลดลง ตัวแปรที่นำมาใช้ในการเก็บค่าการนับก็จะใช้ขนาดลดลงได้ ทำให้ประหยัดหน่วยความจำข้อมูล (RAM Memory) ได้มากขึ้น การเลือกใช้งานตัวหารความถี่จึงต้องเลือกให้เหมาะสมกับการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.5 ฟังก์ชันการใช้งานไทเมอร์ใน CCS Compiler

ไทเมอร์ 0

setup_timer_0 (mode), get_timer0(), set_timer0()

ไทเมอร์ 1

setup_timer_1 (mode), get_timer1(), set_timer1()

ไทเมอร์ 2

setup_timer_2 (mode, period, postscale), get_timer2(), set_timer2()

mode กำหนดโหมดการทำงาน

period เป็นคาบเวลาที่กำหนดในรีจิสเตอร์ PR2 ควบคุมการรีเซตไทเมอร์ 2

postscale เป็นค่าที่กำหนดจำนวนการรีเซตของไทเมอร์ 2 ก่อนที่จะเกิดอินเตอร์รัปต์

2.8.6 การทำงานในโหมดเคาน์เตอร์ของไทเมอร์ 1

การนับสัญญาณนาฬิกาภายนอกของไทเมอร์ 1 จะแบ่งออกได้เป็น 2 แบบคือ

1. เคาน์เตอร์โหมดแบบซิงโครนัส (สัญญาณนาฬิกาภายนอกสัมพันธ์กับสัญญาณนาฬิกา-ภายใน) เมื่อเข้าสู่โหมดสลีป การทำงานของไทเมอร์จะหยุดลงด้วย
2. เคาน์เตอร์โหมดแบบอะซิงโครนัส (สัญญาณนาฬิกาภายนอกไม่สัมพันธ์กับสัญญาณนาฬิกา-ภายใน) เป็นตัวนับอิสระ ไทเมอร์ 1 จะทำงานต่อเนื่องแม้ว่าจะเข้าสู่โหมดสลีปหรือโหมดประหยัดพลังงาน หากมีการเซตอินเตอร์รัปต์ไว้จะทำให้ไมโครคอนโทรลเลอร์ wake-up ออกจากโหมดสลีป

2.9 โครงสร้างการตรวจจับสัญญาณ

2.9.1 โมดูล Capture/Compare/PWM

ไมโครคอนโทรลเลอร์ PIC16F877 จะมีโมดูล Capture/Compare/PWM หรือ CCP ให้ใช้งานได้ 2 ตัว คือ CCP1 (ขา RC2) และ CCP2 (ขา RC1) โดยโมดูล CCP แบ่งโหมดการทำงาน 3 โหมด คือ ตรวจจับสัญญาณ (Input Capture) เปรียบเทียบข้อมูล (Output Compare) และสร้างสัญญาณพัลส์ PWM (Output PWM) มีรีจิสเตอร์ที่เกี่ยวข้องได้แก่

โมดูล CCP1 Capture/Compare/PWM Register1 (CCPR1) ประกอบด้วยรีจิสเตอร์ขนาด 8 บิต 2 ตัว คือ CCPR1L (Low byte) และ CCPR1H (high byte)

โมดูล CCP/ Capture/Compare/PWM Register2 (CCPR2) ประกอบด้วยรีจิสเตอร์ขนาด 8 บิต 2 ตัว คือ CCPR2L (Low byte) และ CCPR2H (high byte)

รีจิสเตอร์ไทเมอร์ 1 และ 2 เนื่องจากโมดูล CCP ต้องอาศัยฐานเวลาในการเปลี่ยนแปลงค่ารายละเอียดแสดงตามตารางที่ 2.9 และ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 โหมด CCP กับการใช้งานไทเมอร์

โหมดการทำงานของโมดูล CCP	ไทเมอร์ที่ใช้งาน
โหมดตรวจจับสัญญาณ (Capture mode)	ไทเมอร์ 1
โหมดเปรียบเทียบสัญญาณ (Compare mode)	ไทเมอร์ 1
โหมดสร้างสัญญาณทางความกว้างของพัลส์ (PWM mode)	ไทเมอร์ 2

ตารางที่ 2.5 แสดงความสัมพันธ์ของ CCP1 และ CCP2 กับไทเมอร์

โหมด CCP1	โหมด CCP2	ผลที่เกิดขึ้นกับโมดูล CCP1 และ CCP2
Capture	Capture	ใช้ฐานเวลาไทเมอร์ 1 ร่วมกัน (TMR1 time-base)
Capture	Compare	ถ้ามีการกำหนดคุณสมบัติกระตุ้นสัญญาณพิเศษ (special event trigger) ในโหมดเปรียบเทียบข้อมูล ไทเมอร์ 1 จะถูกเคลียร์
Compare	Compare	ถ้ามีการกำหนดคุณสมบัติกระตุ้นสัญญาณพิเศษ ในโหมดเปรียบเทียบข้อมูลตัวใดตัวหนึ่งหรือทั้งสอง ไทเมอร์ 1 จะถูกเคลียร์
PWM	PWM	สัญญาณทั้งสองจะมีความถี่และความเร็วเดียวกัน
PWM	Capture	ไม่มีผลกระทบต่อกัน
PWM	Compare	ไม่มีผลกระทบต่อกัน

2.9.2 ฟังก์ชันการใช้งานโมดูล CCP

ติดตั้งการใช้งานโมดูล CCP

setup_ccp1 (mode), setup_ccp2 (mode)

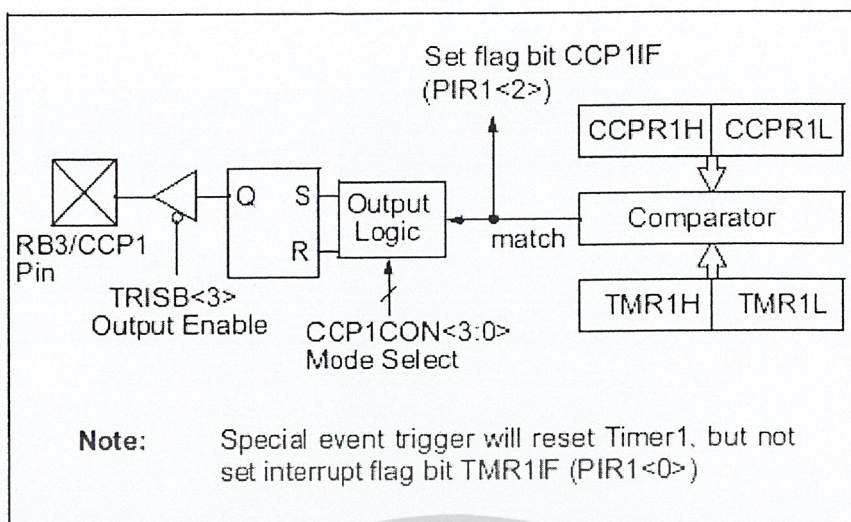
คอนโทรลไจเร็กตีฟสำหรับโมดูล CCP

#int_ccp1, #int_ccp2

2.9.3 โหมดตรวจจับสัญญาณอินพุต

โหมดตรวจจับสัญญาณอินพุต ใช้ในการจับสัญญาณจากภายนอกที่ขา CCPx เมื่อตรงตามเงื่อนไข การตรวจจับสัญญาณที่กำหนดขึ้นจะนำค่าในรีจิสเตอร์ TMR1H และ TMR1L กำหนดให้กับรีจิสเตอร์ CCPR1H และ CCPR1L หากมีการเซตบิตอินเตอร์รัปต์จะเกิดสัญญาณอินเตอร์รัปต์ขึ้น รายละเอียดการทำงานแสดงดังรูปที่ โหมดนี้ถูกนำมาประยุกต์ใช้งานในเรื่องการจับสัญญาณความถี่ หรือการคำนวณหาความกว้างของสัญญาณพัลส์ เป็นต้น

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ไตอะแกรมการทำงานของโมดูล CCP ในโหมดตรวจจับสัญญาณ

โหมดตรวจจับสัญญาณสามารถกำหนดเงื่อนไขการตรวจจับสัญญาณได้ 4 รูปแบบคือ
 โหมดตรวจจับสัญญาณ ทุกขอบขาลงของสัญญาณ
 โหมดตรวจจับสัญญาณ ทุกขอบขาขึ้นของสัญญาณ
 โหมดตรวจจับสัญญาณ ทุกขอบขาขึ้นที่ 4 ของสัญญาณ
 โหมดตรวจจับสัญญาณ ทุกขอบขาขึ้นที่ 16 ของสัญญาณ

2.10 โครงสร้างการรับ-ส่งข้อมูล

2.10.1 โมดูล USART (RS-232 Port)

โมดูล USART (Universal Synchronous Asynchronous Receiver Transmitter) ใน PIC 16F877A สามารถกำหนดการทำงานได้ 3 โหมดคือ

1. โหมดอะซิงโครนัสแบบฟูลดูเพล็กซ์ (Asynchronous (full duplex))
2. โหมดซิงโครนัส – มาสเตอร์ (synchronous – Master (half duplex))
3. โหมดซิงโครนัส – สเลฟ (synchronous – Slave (half duplex))

โหมดจัดขาพอร์ตสำหรับการสื่อสารข้อมูลอนุกรมที่ขา RC6/TX/CK และ RC7/RX/DT ดังนี้ ขาพอร์ต RC6/TX/CK เป็นขาพอร์ตสำหรับส่งข้อมูลในโหมดอะซิงโครนัส และขาสัญญาณนาฬิกาในโหมดซิงโครนัส RC7/RX/DT เป็นขาพอร์ตสำหรับรับข้อมูลในโหมดอะซิงโครนัส และขาข้อมูลในโหมดซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.2 โหมดอะซิงโครนัส

โหมดอะซิงโครนัสแบบฟูลดูเพล็กซ์ เป็นการรับข้อมูล 2 ทิศทางได้ในเวลาเดียวกัน นิยมนำมาใช้ในการเชื่อมต่อกับไมโครคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม (RS-232) นอกจากนี้ยังนำมาใช้เชื่อมต่อกับระบบ Embedded systems ที่ใช้สายในการเชื่อมต่อเพียง 2 เส้นในการรับและส่งข้อมูลที่เรียกว่า RX และ TX

2.10.3 โหมดซิงโครนัส – มาสเตอร์/สเลฟ

โหมดซิงโครนัส – มาสเตอร์/สเลฟ เป็นการรับส่งข้อมูลแบบทางเดียว ไม่สามารถรับและส่งได้พร้อมกัน นิยมนำมาใช้ในการติดต่อกับอุปกรณ์ไอซี เช่น ไอซี A/D, D/A และไอซี หน่วยความจำอีอีพรอมอนุกรม เป็นต้น

2.10.4 ฟังก์ชันการอ่านเขียนข้อมูลพอร์ตอนุกรม

คอนโทรลไตรีกตีฟสำหรับโมดูล USART

#use delay() ติดตั้งการใช้งานสัญญาณนาฬิกาหลัก

#use rs232() ติดตั้งการใช้งานพอร์ตอนุกรม

ฟังก์ชันใช้งานโมดูล USART

enable_interrupt() เปิดการใช้งานอินเตอร์รัปต์เนื่องจากพอร์ตอนุกรม

getc() ฟังก์ชันอ่านข้อมูลจากพอร์ตอนุกรมทีละอักขระ

putc() ฟังก์ชันเขียนข้อมูลจากพอร์ตอนุกรมทีละอักขระ

printf() ฟังก์ชันเขียนข้อมูลไปที่พอร์ตอนุกรมแบบสตริง

2.11 โครงสร้างการแปลงสัญญาณอะนาลอกเป็นดิจิตอล

โมดูลแปลงสัญญาณอะนาลอกเป็นดิจิตอล

โมดูลแปลงสัญญาณอะนาลอกเป็นดิจิตอล (Analog-to-Digital) หรือ ADC สำหรับ PIC16F877A นั้น โมดูล ADC จะมีขนาด 10 บิต 8 ช่อง ที่ขาพอร์ต RA0/AN0, RA1/AN1, RA2/AN2, RA3/AN3, RA5/AN4, RE0/AN5, RE1/AN6, RE2/AN7 โดยการแปลงสัญญาณอะนาลอกเป็นดิจิตอลและเป็นแบบประมาณค่า (Successive Approximate) โดยมี วงจรสุ่มและเก็บสัญญาณ (sample and hold) แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 การกำหนดช่องสัญญาณ ADC

PCFG3: PCFG0	AN7 ⁽¹⁾ RE2	AN6 ⁽¹⁾ RE1	AN5 ⁽¹⁾ RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN/ Refs ⁽²⁾
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input D = Digital I/O

2.11.2 การเลือกความถี่ของสัญญาณนาฬิกา

สามารถกำหนดค่าเวลาในการแปลงสัญญาณได้โดยกำหนดความถี่ ซึ่งจะต้องคำนึงถึงสัญญาณนาฬิกาด้วย เพื่อให้การแปลงค่าสัญญาณถูกต้องที่สุด โดยที่ค่าเวลาที่น้อยที่สุดและยังคงความถูกต้องของการแปลงสัญญาณจะอยู่ที่ 1.6 us

ตารางที่ 2.7 กำหนดสัญญาณความถี่ที่ใช้ในการแปลง (ค่าเวลา)

AD Clock Source (TAD)		Maximum Device Frequency
Operation	ADCS1:ADCS0	Max.
2Tosc	00	1.25 MHz
8Tosc	01	5 MHz
32Tosc	10	20 MHz
RC ^(1, 2, 3)	11	(Note 1)

Note 1: The RC source has a typical TAD time of 4 us, but can vary between 2-6 us.

Note 2: When the device frequencies are greater than 1 MHz, the RC A/D conversion clock source is only recommended for SLEEP operation.

Note 3: For extended voltage devices (LC), please refer to the Electrical Characteristics (Sections 15.1 and 15.2)

2.11.3 ฟังก์ชันใช้งานโมดูล ADC ใน CCS Compiler

setup_adc_ports()	กำหนดพอร์ตอะนาลอกที่ใช้งาน
setup_adc()	กำหนดสัญญาณนาฬิกา
set_adc_chanel()	กำหนดช่องที่จะอ่านสัญญาณอะนาลอก

เอกสารนี้เป็น real_adc() วนไว้สำหรับการใช้งานอ่านสัญญาณอะนาลอก เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 การเชื่อมต่อกับเครื่องคอมพิวเตอร์

หากเราต้องการที่จะเคลื่อนย้ายข้อมูลระหว่างคอมพิวเตอร์กับอุปกรณ์อื่นๆ หรือคอมพิวเตอร์ด้วยกันมี 2 ทางเลือกนั่นคือการรับส่งข้อมูลแบบอนุกรมและแบบขนาน

2.12.1 การอินเตอร์เฟซกับพอร์ตอนุกรม

เป็นวิธีการอินเตอร์เฟซของพีซีวิธีหนึ่งเพื่อทำการเชื่อมต่อกับอุปกรณ์อื่นๆ ซึ่งวิธีนี้จะเป็นการติดต่อแบบอนุกรมผ่านทางพอร์ตอนุกรม ซึ่งการเชื่อมต่อแบบนี้จะมีความเร็วในการเชื่อมต่อช้ากว่าพอร์ตขนาน แต่อุปกรณ์ที่ใช้พอร์ตขนานจะมีราคาแพงกว่า และจะต้องใช้สายสัญญาณมากกว่า การส่งข้อมูลทางพอร์ตอนุกรมนั้นในเครื่องพีซีจะใช้มาตรฐาน RS-232 ซึ่งเป็นมาตรฐานการส่งข้อมูลระยะทางไม่เกิน 15 เมตร ส่วนชิปที่ใช้ในการควบคุมการทำงานนั้นยังคงมีการพัฒนาขึ้นมาใหม่เรื่อยๆ เราจึงต้องทำความเข้าใจมาตรฐานที่ใช้อยู่ตลอดเวลา เพื่อนำไปประยุกต์และใช้งานได้ตามความต้องการได้เสมอ

2.12.1.1 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัส คือการรับและการส่งข้อมูลผ่านสายโดยไม่จำเป็นต้องมีสัญญาณนาฬิกามาร่วมด้วยเหมือนกับการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้การกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนั้นว่า อัตราการถ่ายถอดข้อมูลหรือบอดเรต(Baud rate) มีหน่วยเป็นบิตต่อวินาที (Bit per second: bps)

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับส่งข้อมูลแบบอะซิงโครนัสเรียกว่า Universal Asynchronous Receiver/Transmitter หรือ UART อัตราความเร็วในการรับส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสคือ ค่าบอดเรต ซึ่งก็คือ ค่าจำนวนบิตต่อวินาทีที่ใช้ในการรับและส่งข้อมูลบอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรมอาร์เอสสองสามสอง

2.12.1.2 มาตรฐานพอร์ตอนุกรมแบบอาร์เอสสองสามสอง (RS-232)

เพื่อที่จะให้อุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้ มาตรฐานหลายชนิดได้รับการออกแบบขึ้น มาตรฐานที่ใช้กันกว้างขวางที่สุดคือ RS-232 ถูกประกาศในปี 1969 โดย Electronic Industries Association (EIA) ซึ่งเป็นการกำหนดมาตรฐานต่างๆ เช่น ลักษณะการเชื่อมต่อ สัญญาณทางไฟฟ้าที่ใช้เพื่อป้องกันไม่ให้อุปกรณ์ส่งข้อมูลภายในสายเส้นเดียวกัน

คุณสมบัติทางไฟฟ้าของพอร์ตอนุกรมตามมาตรฐาน RS-232 พอสรุปได้ดังนี้

- Logic '0' หรือ สเปซ (Space) มีค่า +3 โวลต์ ถึง +25 โวลต์
- Logic '1' หรือ มาร์ค (Mark) มีค่า -3 โวลต์ ถึง -25 โวลต์
- ช่วง +3 โวลต์ถึง -3 โวลต์ เป็นช่วงบอกค่าไม่ได้
- ความต่างศักย์เทียบกับกราวด์ไม่เกิน 25 โวลต์และกระแสไม่เกิน 500 มิลลิแอมป์

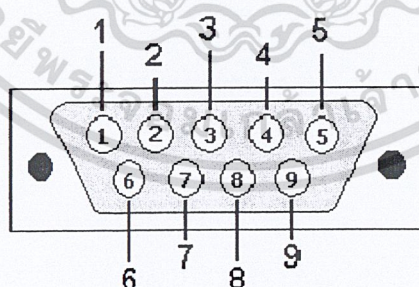
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูอาจารย์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.8 ข้อกำหนดของมาตรฐาน RS-232

Specifications	RS-232
Mode of Operation	Single-Ended
Total Number of Drivers and Receivers on One Line	1 Driver and 1 Receiver
Maximum Cable Length	50 FT.
Maximum Data Rate	20 KBPS
Maximum Driver Output Voltage	+/-25 V
Driver Output Signal Level (Loaded Min.)	+/-5 V to +/-15 V
Driver Output Signal Level (Unloaded Max)	+/-25 V
Driver Load Impedance (Ω)	3k to 7k
Max. Driver Current in High Z State (Power On)	N/A
Max. Driver Current in High Z State (Power Off)	+/-6 mA @ +/-2 V
Slew Rate (Max.)	30 V/ μ S
Receiver Input Voltage Range	+/-15 V
Receiver Input Sensitivity	+/-3 V
Receiver Input Resistance (Ohms)	3k to 7k

2.12.1.3 ลักษณะของหัวต่อตามมาตรฐาน RS-232 (DB-9)

พอร์ตอนุกรมมีหัวต่อ 2 แบบคือ แบบ DB-25 และ DB-9 (ซึ่งในที่นี่จะอธิบายเฉพาะ ส่วนของ DB-9) ซึ่งทั้ง 2 แบบจะเป็นชนิดตัวผู้ทางด้านของคอมพิวเตอร์ ดังนั้นอุปกรณ์ที่จะนำมาต่อกับคอมพิวเตอร์จึงต้องใช้หัวต่อชนิดตัวเมีย



รูปที่ 2.15 การจัดขาของคอนเนคเตอร์พอร์ตอนุกรมมาตรฐานอาร์เอสสองสามสองแบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

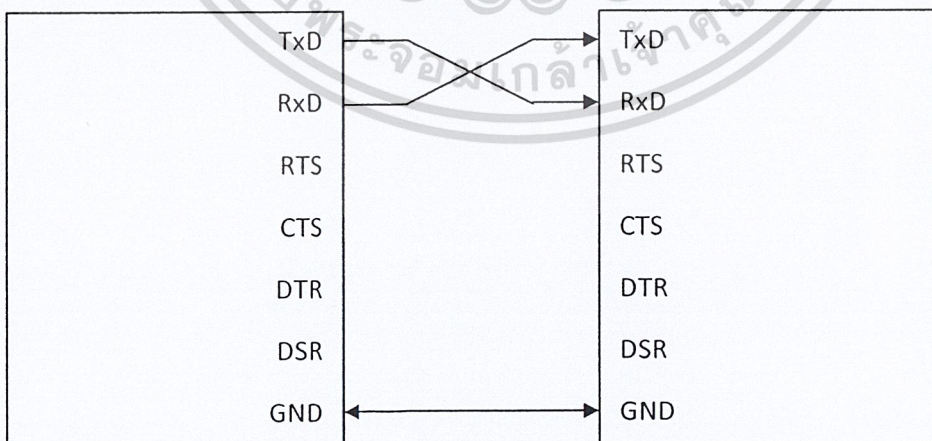
ตารางที่ 2.9 ตัวผู้ทางด้านคอมพิวเตอร์

Pin	Signal	Direction	Description
1	CD	←	Carrier Detect
2	RxD	←	Receive Data
3	TxD	→	Transmit Data
4	DRT	→	Data Terminal Ready
5	GND	-	System Ground
6	DSR	←	Data Set Ready
7	RTS	→	Require to Send
8	CTS	←	Clear to Send
9	RI	←	Ring Indicator

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกดังในรูปที่ 2.16 ลูกศรแสดงถึงทิศทางของข้อมูล เป็นการเชื่อมต่อในลักษณะที่ใช้สายสัญญาณเพียง 3 สาย โดยสายเส้นหนึ่งสำหรับในการส่งข้อมูล สายอีกเส้นหนึ่งใช้สำหรับในการรับข้อมูล และสายเส้นสุดท้ายจะใช้สำหรับไว้เป็นกราวด์

รายละเอียดของพอร์ตอนุกรมอาร์เอสสองสามสองที่ใช้งาน มีดังนี้

- Receive Data : RD ขานี้ใช้สำหรับเพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บในรีจิสเตอร์บัฟเฟอร์
- Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยที่จะนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลออกไป
- Signal Ground ขากราวด์ของระบบ

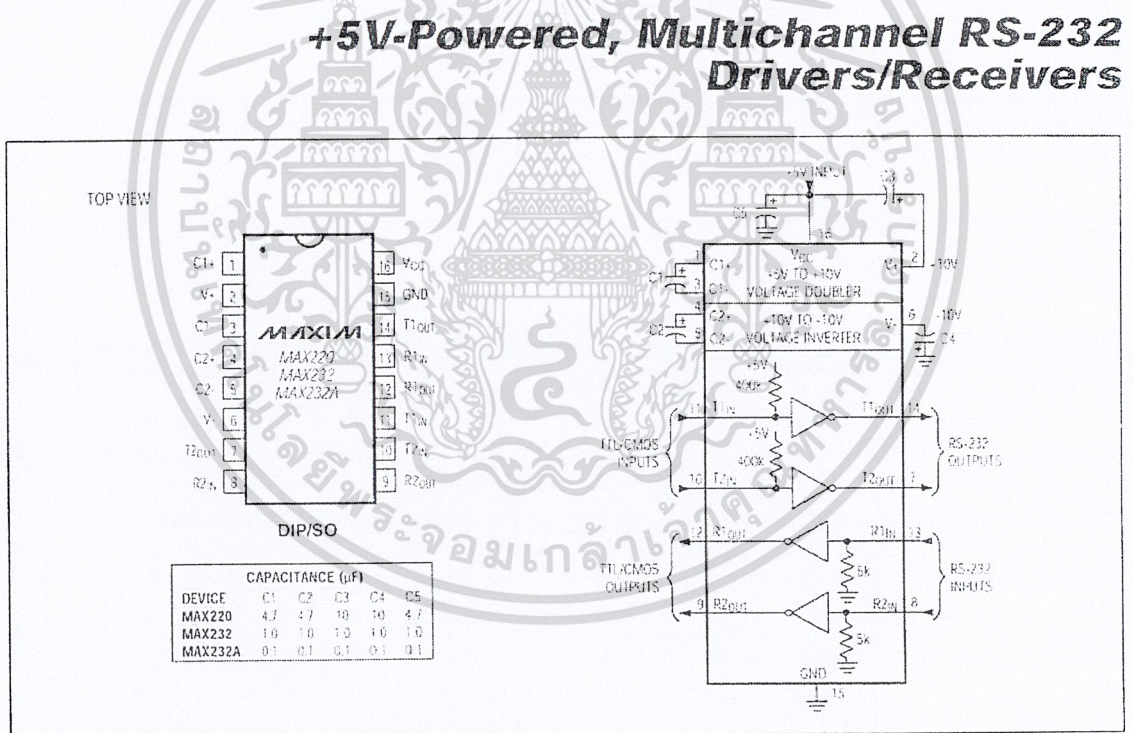


รูปที่ 2.16 การเชื่อมต่อแบบอาร์เอสสองสามสองโดยใช้สายสัญญาณ 3 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับระหว่างอุปกรณ์ภายนอกกับคอมพิวเตอร์รุ่นให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.2 ไอซีปรับระดับแรงดัน MAX 232

ไอซีปรับระดับแรงดัน MAX 232 เป็นไอซี 16 ขาที่ใช้ทำหน้าที่ในการรับส่งข้อมูลแบบอนุกรม (serial) ระหว่างอุปกรณ์ TTL/CMOS กับพอร์ท RS-232 โดยไอซีปรับระดับแรงดัน MAX 232 มีช่องทางในการรับข้อมูลแบบอนุกรม (serial) 2 ช่องทางและมีช่องทางในการส่งข้อมูลแบบอนุกรม (serial) 2 ช่องทาง โครงสร้างภายในของไอซีปรับระดับแรงดัน MAX 232 ยังมีวงจรทวีแรงดัน (voltage doubler) และวงจรอินเวอร์สแรงดัน (voltage inverter) วงจรทวีแรงดันเป็นวงจรที่ทำหน้าที่ยกระดับแรงดันจาก 5 โวลต์เป็น 10 โวลต์ และวงจรอินเวอร์สแรงดันเป็นวงจรที่ทำหน้าที่กลับสัญญาณจาก +10 โวลต์เป็น -10 โวลต์ วงจรทั้งสองทำหน้าที่หลักก็คือขยายสัญญาณก่อนที่จะส่งออกไปยังพอร์ท RS-232 รูปตำแหน่งขาสัญญาณและวงจรเทียบการทำงานของ MAX 232 ดังแสดงในรูปที่



รูปที่ 2.17 ตำแหน่งขาสัญญาณและวงจรภายในไอซีปรับระดับแรงดัน MAX 232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของไอซีปรับระดับแรงดัน MAX 232

1. เมื่อส่งข้อมูลออก โดยไอซีปรับระดับแรงดัน MAX 232 จะรับข้อมูลจากอุปกรณ์ TTL/CMOS เข้ามาที่ขา TX IN จากนั้นไอซีปรับระดับแรงดัน MAX 232 จะทำการยกกระดัดแรงดันของข้อมูลให้เป็น +10 โวลต์หรือ -10 โวลต์ และส่งออกมาที่ขา TX OUT ผ่านพอร์ท RS-232 เพื่อส่งข้อมูลต่อไปยังอุปกรณ์ภายนอก

2. เมื่อรับข้อมูลเข้า โดยไอซีปรับระดับแรงดัน MAX 232 จะรับข้อมูลที่ส่งมาจากอุปกรณ์ภายนอกผ่านพอร์ท RS-232 มายังขา RX IN ของไอซีปรับระดับแรงดัน MAX 232 จากนั้นไอซีปรับระดับแรงดัน MAX 232 จะทำการแปลงระดับแรงดันของข้อมูลให้เป็น 0 โวลต์หรือ 5 โวลต์ และส่งออกที่ขา RX OUT เพื่อทำการส่งข้อมูลต่อไปที่อุปกรณ์ TTL/CMOS

2.13 ภาษาซีชาร์ป (C# Programming Language)

ภาษาซีชาร์ป (C# Programming Language) เป็นภาษาโปรแกรมเชิงวัตถุ ทำงานบนดอตเน็ตเฟรมเวิร์ก(.NET Framework) พัฒนาโดยบริษัทไมโครซอฟท์ มีรากฐานมาจากภาษาซี ภาษาซีพลัสพลัสและภาษาอื่นๆ (โดยเฉพาะภาษาเดลไฟและจาวา) ปัจจุบันภาษาซีชาร์ปเป็นภาษามาตรฐานรองรับโดย ECMA และ ISO ภาษาซีชาร์ปถูกพัฒนาขึ้นโดยเป็นส่วนหนึ่งในการพัฒนาโครงสร้างพื้นฐานของดอตเน็ตเฟรมเวิร์กเป็นการนำข้อดีของภาษาต่างๆ มาปรับปรุงเพื่อให้มีความเป็นโอโอพี (OOP) อย่างถึงที่สุด ขณะเดียวกันก็ลดความซับซ้อนในโครงสร้างของภาษาลง (เรียบง่ายกว่าภาษาซีพลัสพลัส และมีเครื่องหมายน้อยลง (เมื่อเทียบกับ Java)

ภาษาซีชาร์ปเป็นภาษาที่ทำให้โปรแกรมรันบนซอฟต์แวร์พื้นฐานบนคอมพิวเตอร์ได้เวลาเขียนโปรแกรมจะง่าย เพราะสามารถใช้รันบนอุปกรณ์ใดก็ได้ที่มีซอฟต์แวร์พื้นฐานภาษาซีชาร์ปนี้อยู่ภายใต้โครงสร้างเทคโนโลยีดอตเน็ต

2.14 Visual Studio 2008 with C#

เป็นโปรแกรมชุด แนวคิดคล้ายๆกับ Microsoft Office เพียงแต่เป็นชุดด้านการพัฒนาแอปพลิเคชัน ซึ่งได้รวบรวมเอาเครื่องมือพัฒนาแอปพลิเคชันหลายๆตัวเข้าไว้ด้วยกัน ซึ่งช่วยในการเขียนโปรแกรมในภาษา C# ซึ่งภาษา C# เป็นภาษาโปรแกรมยุคใหม่ที่สร้างขึ้นมาสำหรับการพัฒนาซอฟต์แวร์ภายใต้เทคโนโลยี Microsoft.NET ซึ่งเป็นเทคโนโลยีที่ได้รับความนิยมสูงที่สุดในปัจจุบัน โปรแกรม Visual C# 2008 นั้น มีข้อดีหลายประการคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ภาษา C# ที่เขียนง่าย เพราะมีความกับภาษา C, C++ และ Java ทำให้เข้าใจง่ายและเหมาะสมต่อการศึกษของผู้สนใจ

2. มีการพัฒนาของภาษา เพราะ C# เป็นภาษาใหม่ แต่ถูกพัฒนามาจากภาษา C++ และ Java จึงได้แก้ไขข้อบกพร่องของภาษาเหล่านั้น ทำให้เป็นภาษาที่สวยงามและมีประสิทธิภาพ

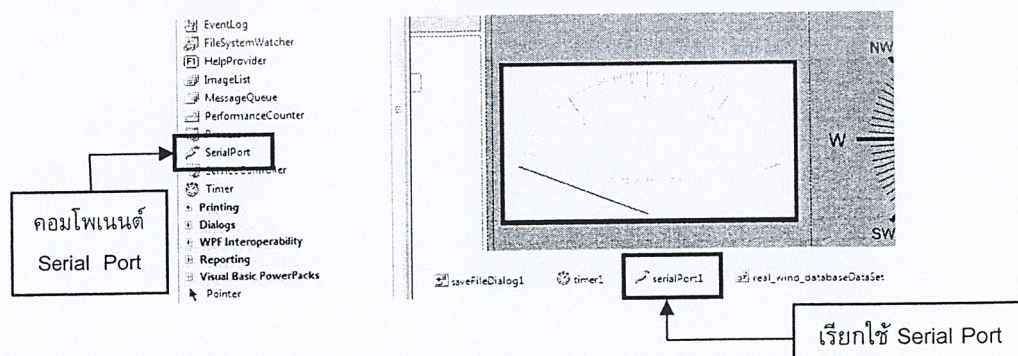
3. ภาษา C# มีความนิยมมากขึ้นเรื่อยๆ และเกิดจากบริษัทไมโครซอฟท์ทำให้มั่นใจได้ว่า Visual C# 2008 จะมีการพัฒนา ปรับปรุงและคงอยู่ต่อไปอีกเรื่อยๆ

นอกจากนี้แล้วแนวทางการพัฒนาแอปพลิเคชันในปัจจุบันไม่ว่าจะด้วยภาษา หรือเครื่องมือใดๆ ต่างก็ใช้สิ่งที่เรียกว่าองค์ประกอบทางซอฟต์แวร์ (Software Component) แทบทั้งสิ้น Visual Studio นั้นได้มีการพัฒนาองค์ประกอบทางซอฟต์แวร์เป็นอย่างดีโดยบริษัทไมโครซอฟท์รวมทั้งบริษัทซอฟต์แวร์อื่นๆอีกมากมาย อีกทั้งผู้ใช้งานก็สามารถสร้าง Control ขึ้นมาใช้เองได้ด้วย จึงทำให้พร้อมนำมาประกอบกันเป็นแอปพลิเคชันได้อย่างรวดเร็วและง่ายดาย

2.15 Serial Port

โปรแกรม Visual Studio 2008 จะมีคอมโพเนนต์ที่ใช้สำหรับอินเตอร์เฟสกับอุปกรณ์ภายนอก เช่น ทำการส่งค่าหรือรับค่าระหว่างอุปกรณ์ภายนอกกับคอมพิวเตอร์ โดยโปรแกรม Visual Studio 2008 ตัวคอนโทรลที่ชื่อว่า Serial Port มาให้แล้ว ทำให้สะดวกต่อนักพัฒนาโปรแกรม เพราะเพียงแค่กำหนดรูปแบบการรับ-ส่งข้อมูลและกำหนดพอร์ตที่ต่าง ๆ ของคอมโพเนนต์นี้เท่านั้น ไม่จำเป็นต้องเขียนซอร์สโค้ดการติดต่อระหว่างอุปกรณ์ให้ยุ่งยาก จึงช่วยให้นักพัฒนาสามารถพัฒนาโปรแกรมได้เร็วมากและเป็นมาตรฐานเดียวกันของโปรแกรมเมอร์ทุกคน

ในปัจจุบันการติดต่อผ่าน Serial port ระหว่าง Computer กับ อุปกรณ์ภายนอกนี้เป็นที่นิยมมาก เพราะง่ายและสะดวกในการสร้าง และนำไปใช้ได้ทั้ง Microcontroller, PLC หรือนำไปประยุกต์ใช้ในอุตสาหกรรมก็ได้

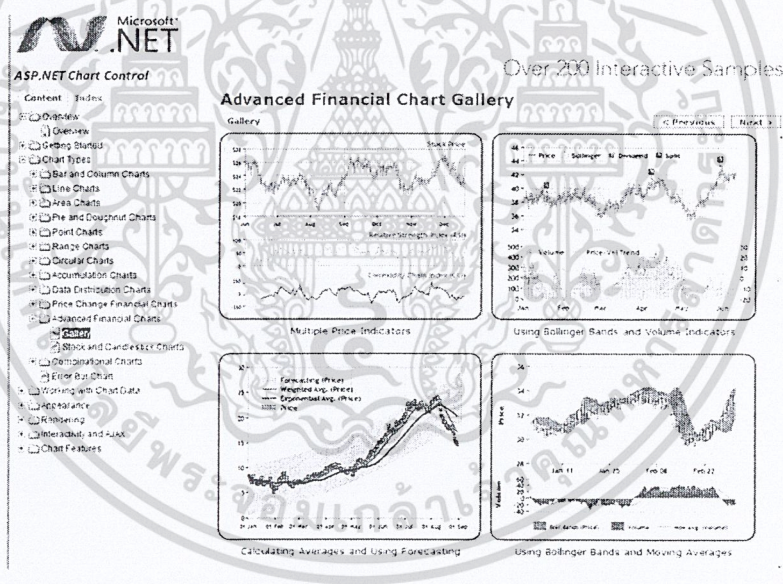


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.18 คอนโพเนนต์และอีเวนต์ Serial port
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16 Microsoft Chart Control

Microsoft Chart Control คือคอมโพเนนต์ที่บริษัทไมโครซอฟท์ได้ผลิตขึ้นมา เพื่อให้ นักพัฒนาใช้สร้างรายงานที่มีแผนภูมิ และกราฟได้อย่างรวดเร็ว โดยปราศจากการใส่ซอร์สโค้ด ใดๆ ซึ่งอาศัยพื้นฐานบน Dudas Chart Control version 5.5 สำหรับใช้งานบน .NET Framework 3.5 และ Visual Studio 2008 ซึ่งจะถูกใส่มาพร้อมใน .NET Framework 4.0 สามารถดาวน์โหลดคอมโพเนนต์จากเว็บไซต์ของไมโครซอฟท์ได้โดยตรง โดยไม่เสียค่าใช้จ่าย ใดๆ

Microsoft Chart Control นี้สามารถพัฒนาได้ทั้งบนเว็บแอปพลิเคชัน (ASP.NET) หรือโปรแกรมบนวินโดวส์ (Window Form) และสร้างรายงานแผนภูมิ หรือกราฟกับข้อมูลที่มีความซับซ้อนเช่น การวิเคราะห์เกี่ยวกับสถิติ การเงิน ได้ง่ายขึ้นและมีรูปแบบที่สวยงาม โดยสามารถสร้างได้ทั้งในรูปแบบ 2 มิติ และ 3 มิติ สามารถประกาศแผนภูมิ แบบสแตติก (statically) หรือแบบไดนามิก (Dynamically) ก็ได้ นอกจากนี้ยังสนับสนุนการทำงานกับ AJAX สำหรับเว็บแอปพลิเคชัน ทำให้นักพัฒนาสามารถสร้างรายงานแผนภูมิกราฟแบบ Real Time ได้อีกด้วย



รูปที่ 2.19 ตัวอย่างรูปแบบกราฟใน Microsoft Chart Control

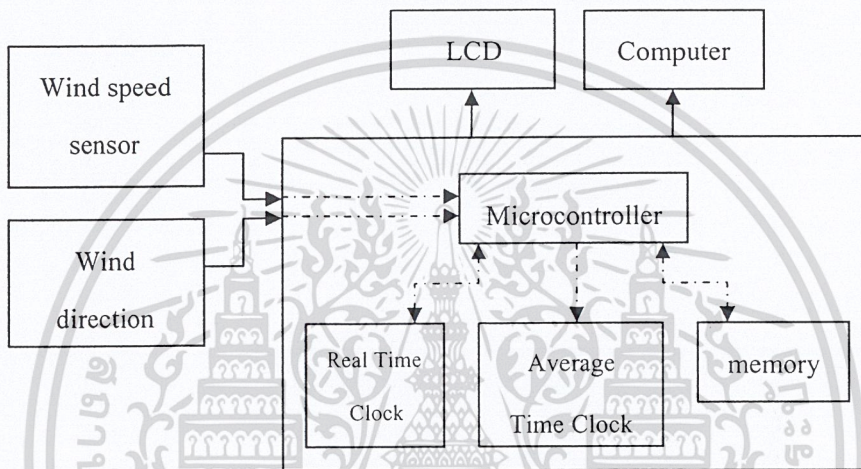
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

3.1 โครงสร้างของเครื่องวัดความเร็วลมและทิศทางลม

เครื่องวัดความเร็วลมและทิศทางลมนี้มีการตรวจวัดอยู่ 2 อย่าง คือ ความเร็วและทิศทางลม โดยข้อมูลต่างๆจะถูกแปลงค่าที่ได้จากเครื่องมือวัดส่งออกมาแสดงผลเป็นค่าดิจิทัลยังหน้าจอ LCD มีการเก็บข้อมูลลง USB FLASH DRIVE และใช้โปรแกรมประยุกต์ที่ประมวลผลบนคอมพิวเตอร์แสดงค่าต่างๆที่ตรวจวัดได้ โดยโครงสร้างทั้งหมดของระบบแสดงดังรูป

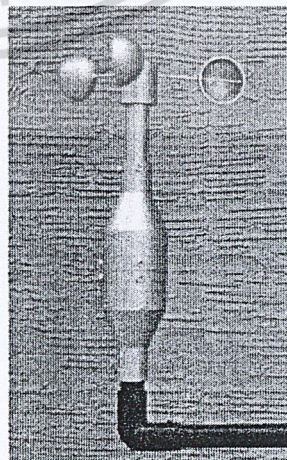
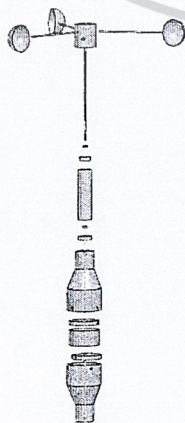


รูปที่ 3.1 บล็อกไดอะแกรมโครงสร้างของระบบ

3.2 การออกแบบและการสร้าง

3.2.1 การออกแบบเครื่องวัดความเร็วลม

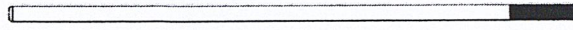
ในการสร้างเครื่องวัดความเร็วลมนั้น ในโครงงานนี้ได้เครื่องตรวจวัดความเร็วลมแบบลูกถ้วยทรงกลมสามแกน โดยโครงสร้างของเครื่องวัดความเร็วลมจะมีลักษณะดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

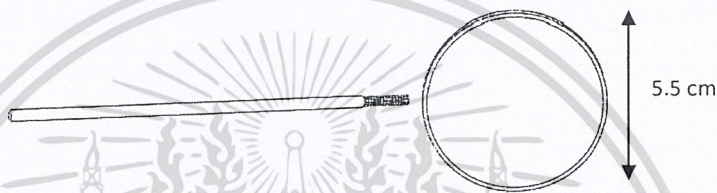
รูปที่ 3.2 เครื่องวัดความเร็วลม

1. นำแท่งสแตนเลสขนาดเส้นผ่านศูนย์กลาง 3.10 มิลลิเมตร จำนวน 3 แท่ง มาตัดให้มีความยาว 11 เซนติเมตร จากนั้นนำปลายด้านหนึ่งของแท่งสแตนเลสทั้ง 3 แท่งมาทำเกลียว



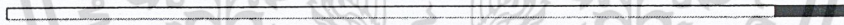
รูปที่ 3.3 ก้านรับลม

2. ในการทำลูกถ้วยเราจัดหาพลาสติกเครื่องทรงกลมขนาดเส้นผ่านศูนย์กลาง ทำการเจาะรูที่บริเวณขอบของลูกถ้วยให้ทะลุตามแนวเส้นผ่านศูนย์กลางเพื่อจะนำแกนที่ได้จากในข้อหนึ่งด้านที่ทำเกลียวมาประกอบกับลูกถ้วยดังรูป



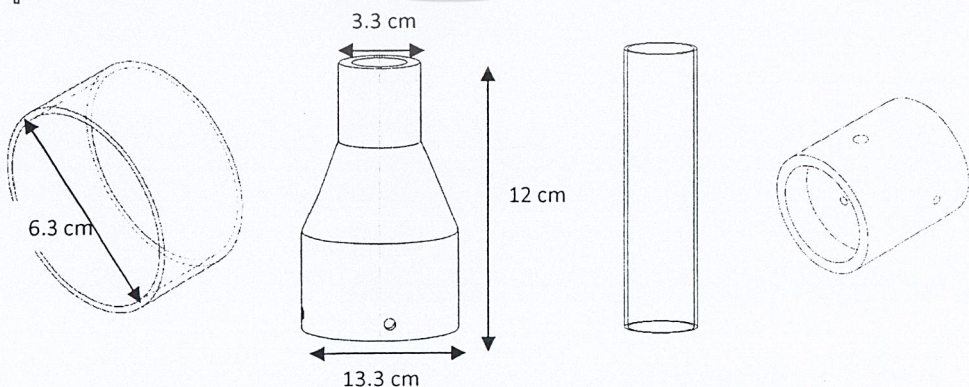
รูปที่ 3.4 ลักษณะของการต่อลูกถ้วยเข้ากับก้านรับลม

3. นำแท่งสแตนเลสขนาดเส้นผ่านศูนย์กลาง มาตัดให้มีความยาว 16.5 เซนติเมตร เพื่อที่จะนำมาเป็นแกนหมุน (shaft) นำปลายด้านหนึ่งมาทำเกลียวเพื่อใช้ในการยึดแผ่นตัดแสง



รูปที่ 3.5 ลักษณะของแกนหมุน

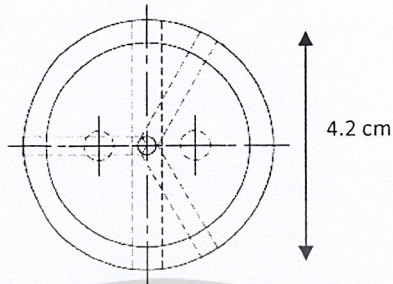
4. นำท่อพีวีซีซึ่งประกอบไปด้วย ท่อเชื่อม, ท่อลดตรง, ท่อตรงและฝาครอบ ขนาดต่างๆ ดังแสดงในรูปมาทำเป็นส่วนประกอบของชิ้น (Body)



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การใช้นโยบายที่การศึกษานานาชาติไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 ส่วนประกอบ Body ของเครื่องวัดความเร็วลม

5. นำฝาครอบมาหล่อด้วยเรซินให้เหลือพื้นที่ว่างจากขอบประมาณ 1.5 เซนติเมตร จากนั้นทำการเจาะรู 3 รูทำมุมกัน 120 องศา เพื่อใส่แกนของลูกถ้วยรับลมทั้ง 3 ก้าน และเจาะรูที่จุดกึ่งกลางของเรซินสำหรับใส่แกนหมุน (Shaft) หลังจากนั้นจึงทำการเจาะรูพร้อมทำเกลียวสำหรับใส่ยึดยึดแกนทั้งหมดให้ติดอยู่กับฝาครอบ



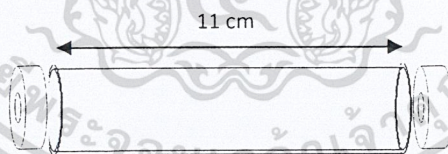
รูปที่ 3.7 ตัวจับยึดก้านรับลม

6. นำแผ่นอะคริลิกหนา 8 มิลลิเมตร มาตัดเป็นวงกลมจำนวน 2 แผ่นขนาดดังรูป โดยทำการเจาะรูตรงกลางสำหรับใส่ตลับลูกปืน



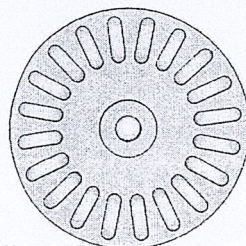
รูปที่ 3.8 ลักษณะของส่วนบังคับแกนหมุน

7. นำแผ่นอะคริลิกที่ได้จากข้อ 6 มาตอกอัดลงในท่อพีวีซีทั้งสองด้านหลังจากนั้นนำแกนหมุนสอดผ่านตลับลูกปืนจากด้านหนึ่งไปทะลุปลายอีกด้านหนึ่ง



รูปที่ 3.9 การประกอบส่วนบังคับแกนเข้ากับท่อตรง

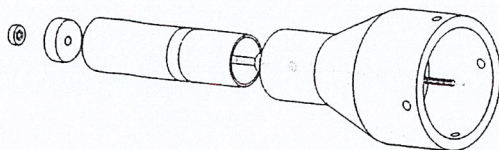
8. การทำแผ่นตัดแสงเราได้จัดหาแผ่นตัดแสงที่มีเส้นผ่านศูนย์กลาง ซึ่งจะมีช่องว่างสลับกันโดยการหมุนของแผ่นตัดแสงหนึ่งรอบจะได้จำนวนพัลส์เท่ากับ 20 พัลส์ ซึ่งมีลักษณะดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.10 ลักษณะของแผ่นตัดแสง

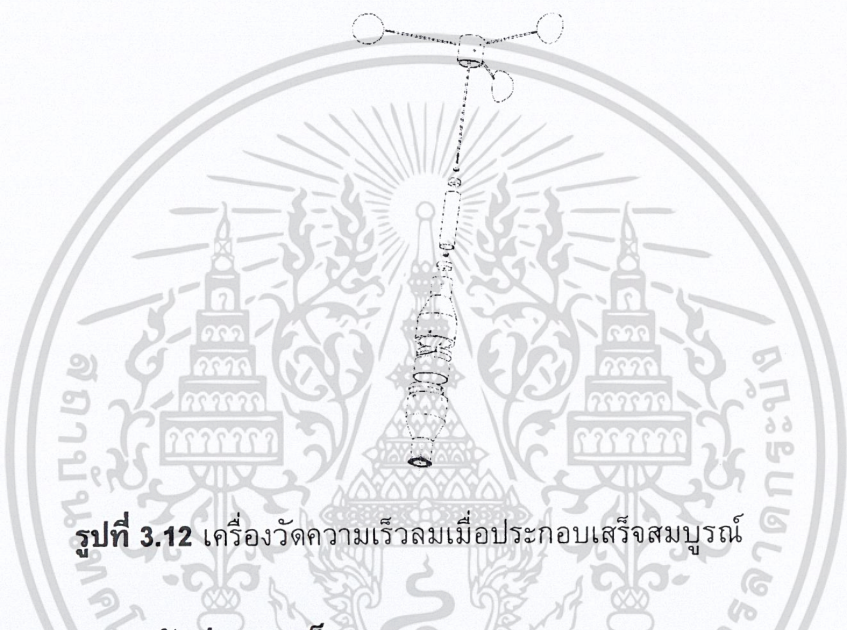
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. นำแกนหมุนที่ได้จากขั้นตอนในข้อ 7 มาสวมเข้ากับท่อพีวีซีลดตรงดังรูป



รูปที่ 3.11 การประกอบชิ้นส่วน Body ของเครื่องวัดความเร็วลม

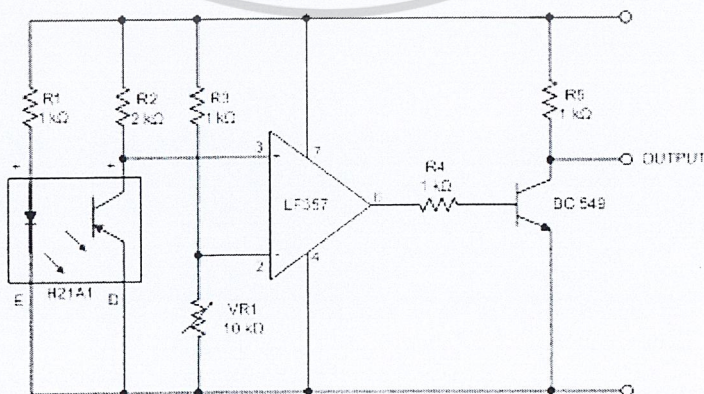
10. นำฝาครอบที่ประกอบด้วยแกนลูกถ้วยรับลมเรียบร้อยแล้วมาสวมเข้ากับปลายของแกนหมุน (Shaft) และทำการขันน็อตยึดแกนหมุน (Shaft) เข้ากับฝาครอบ



รูปที่ 3.12 เครื่องวัดความเร็วลมเมื่อประกอบเสร็จสมบูรณ์

3.2.2 วงจรตรวจจับค่าความเร็วลม

โครงสร้างของวงจรที่ใช้สำหรับตรวจจับความเร็วลมจะใช้วงจรจากกรมอุตสาหกรรมศึกษา โดยแสดงดังภาพ โดยวงจรนี้จะใช้หลักการของการตรวจจับแสงที่ผ่านเข้าโฟโตทรานซิสเตอร์เพื่อกำเนิดเป็นแรงดันไฟฟ้าแล้วทำการขยายสัญญาณโดยใช้อปแอมป์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.13 วงจรตรวจจับค่าความเร็วลม
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 การทดสอบชุดตรวจจับค่าความเร็วลม

ในการทดสอบชุดตรวจจับค่าความเร็วลมเริ่มต้นจะทดสอบการหมุนของก้านลมว่าสามารถหมุนได้ดีหรือไม่ หลังจากนั้นจะทดสอบวงจรตรวจจับแสงเมื่อก้านรับลมหมุนโดยจะมีขั้นตอนดังต่อไปนี้

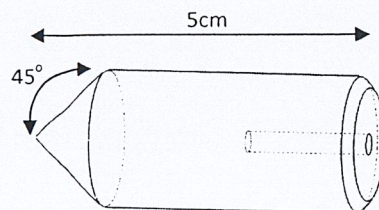
1. ป้อนไฟฟ้ากระแสตรงขนาด 5 โวลต์ให้กับวงจรตรวจจับค่าความเร็วลม
2. ใช้ออสซิลโลสโคปวัดสัญญาณอินพุตที่ขา 3 ของไอซี LF 357
3. หมุนก้านรับลมไปหนึ่งรอบ แล้วสังเกตที่ออสซิลโลสโคปจะมีสัญญาณพัลส์เกิดขึ้น
4. ใช้ออสซิลโลสโคปวัดสัญญาณเอาต์พุตที่ขาคอนแลคเตอร์ของทรานซิสเตอร์ BC549
5. ทำการปรับแต่งสัญญาณเอาต์พุตที่ตัวต้านทานปรับค่าได้เปรียบเทียบกับสัญญาณอินพุตโดยให้มีลักษณะใกล้เคียงกันมากที่สุด

3.2.4 การออกแบบเครื่องวัดทิศทางลม



รูปที่ 3.14 เครื่องวัดทิศทางลม

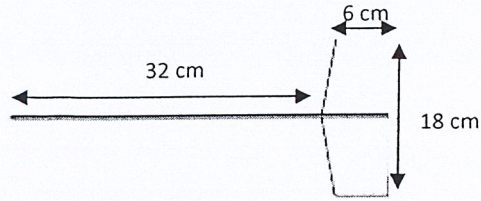
1. นำแท่งสแตนเลสขนาดเส้นผ่านศูนย์กลาง ตัดให้มีความยาว 5 เซนติเมตร เพื่อใช้สำหรับเป็นหัวคร ทำการกลึงปลายด้านหนึ่งให้ลู่ลมและนำปลายอีกด้านหนึ่งมาเจาะรูที่กึ่งกลางของแท่งสำหรับใส่แกนของครลม หลังจากนั้นเจาะรูพร้อมสต๊าฟเกลียวสำหรับยึดแกนกับหัวครลมดังรูป



รูปที่ 3.15 ลักษณะของหัวครลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะกิจของทางนี้ ไปลงญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. นำแผ่นสแตนเลสหนา 1 มิลลิเมตร มาตัดให้มีขนาดดังรูปเพื่อใช้สำหรับทำหางรับลม หลังจากนั้นทำการเชื่อมหางรับลมเข้ากับแกนของศรลมดังรูป



รูปที่ 3.16 การเชื่อมหางรับลมเข้ากับแกนศรลม

3. นำแท่งสแตนเลสขนาดเส้นผ่านศูนย์กลาง ตัดให้มีความยาว 22 เซนติเมตร เพื่อใช้สำหรับเป็นแกนหมุน (Shaft)



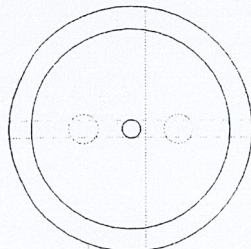
รูปที่ 3.17 ลักษณะของแกนหมุน

4. นำท่อพีวีซี ซึ่งประกอบไปด้วย ท่อลดตรง, ท่อตรงและฝาครอบ ขนาดต่างๆ ดังแสดงในรูปมาทำเป็นส่วนประกอบของโครง (Body)



รูปที่ 3.18 ส่วนประกอบ Body ของเครื่องวัดทิศทางลม

5. นำฝาครอบมาหล่อด้วยเรซินให้เหลือพื้นที่ว่างจากขอบประมาณ 1.5 เซนติเมตร และเจาะรูตามแนวเส้นผ่านศูนย์กลางของฝาครอบให้ทะลุมาอีกด้านหนึ่งสำหรับใส่แกนของศรลม หลังจากนั้นทำการเจาะรูและสตาฟเกลียวเพื่อยึดแกนศรลมเข้ากับตัวฝาครอบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.19 ลักษณะการเจาะรูที่ฝาครอบเพื่อใส่แกนของศรลม
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น ยกเว้นหากมีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. นำแผ่นอะคริลิกหนา 8 มิลลิเมตร มาตัดเป็นวงกลมจำนวน 2 แผ่นขนาดดังรูป 3.20 โดยทำการเจาะรูตรงกลางสำหรับใส่ตั้บลูกปืน



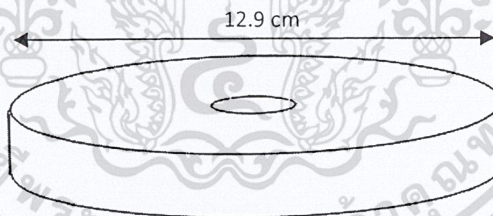
รูปที่ 3.20 ลักษณะของส่วนบังคับแกนหมุน

7. นำแผ่นอะคริลิกที่ได้จากข้อ 6 มาดอกลงที่ปลายของท่อพีวีซีทั้งสองด้านหลังจากนั้นนำแกนหมุนสอดผ่านตั้บลูกปืนจากด้านหนึ่งไปทะลุปลายอีกด้านหนึ่ง ซึ่งจะได้แกนหมุนดังรูป



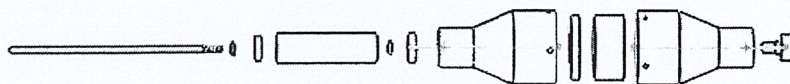
รูปที่ 3.21 การประกอบส่วนบังคับแกนเข้ากับท่อตรง

8. นำแผ่นอะคริลิกหนา 8 มิลลิเมตรตัดเป็นวงกลมเส้นผ่านศูนย์กลาง จากนั้นนำมาเจาะรูตรงกลางสำหรับใส่ตัวโพเทนชิโอมิเตอร์



รูปที่ 3.22 แผ่นยึดโพเทนชิโอมิเตอร์

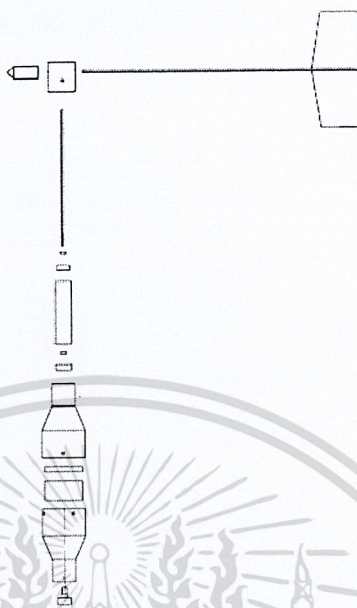
9. นำแผ่นอะคริลิกที่ได้จากข้อที่ 8 มาใส่ในท่อพีวีซีสตรงและนำโพเทนชิโอมิเตอร์มาต่อผ่านรูของแผ่นอะคริลิกในส่วนท้าย



รูปที่ 3.23 การประกอบชิ้นส่วน Body ของเครื่องวัดทิศทางลม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปเผยแพร่โดยไม่ได้รับอนุญาตจะถือว่าผิดกฎหมาย
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

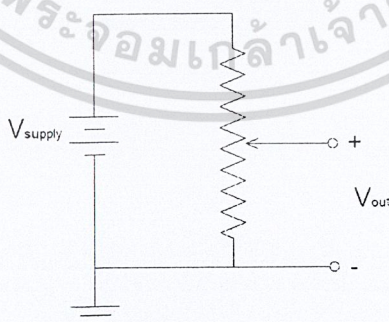
10. นำฝาครอบที่ประกอบกับแกนของโครงมเรียงร้อยแล้วมาสวมเข้ากับปลายของแกนหมุน (Shaft) และทำการขันน็อตยึดแกนหมุน (Shaft) เข้ากับฝาครอบ



รูปที่ 3.24 เครื่องวัดทิศทางลมเมื่อประกอบเสร็จสมบูรณ์

3.2.5 วงจรตรวจวัดทิศทางลม

เซนเซอร์สำหรับตรวจวัดทิศทางลม จะใช้โพเทนชิโอมิเตอร์ที่มีลักษณะการเปลี่ยนแปลงแบบเชิงมุมซึ่งมีคุณสมบัติ คือแรงเคลื่อนไฟฟ้าเอาต์พุตจะเปลี่ยนแปลงไปตามมุมที่เปลี่ยนแปลงไป โดยที่การเปลี่ยนแปลงมุมนั้นจะส่งผลให้ความต้านทานภายในโพเทนชิโอมิเตอร์เปลี่ยนแปลงด้วยเช่นกัน โดยที่การเปลี่ยนแปลงนั้นจะมีลักษณะเป็นเชิงเส้น



รูปที่ 3.25 โพเทนชิโอมิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาการแบ่งแรงดันของวงจร จะได้ว่า

$$V_o = \frac{\theta}{\theta_{max}} V \quad (3.1)$$

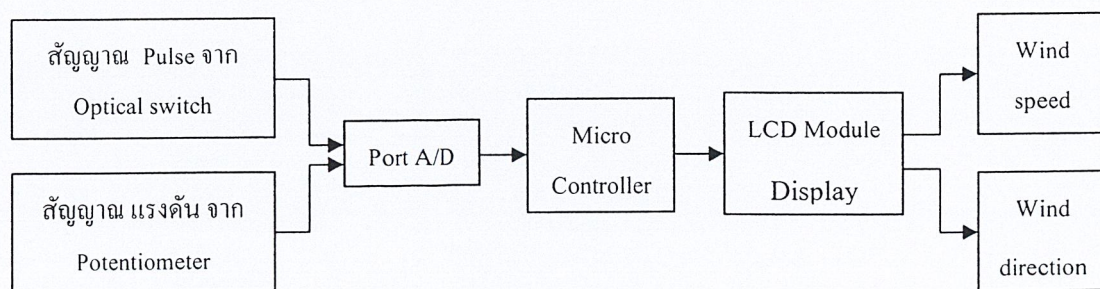
เมื่อ	V	คือ แรงเคลื่อนไฟฟ้าที่ป้อนให้กับ Potentiometer
	V_o	คือ แรงเคลื่อนไฟฟ้าเอาต์พุต
	θ	คือ มุมที่เปลี่ยนแปลง
	θ_{max}	คือ มุมสูงสุดของ Potentiometer

สำหรับโพเทนชิโอมิเตอร์ที่ใช้กับเซนเซอร์ตัวนี้จะมีค่าความต้านทานสูงสุด ณ มุมสูงสุด ซึ่งมีค่าเท่ากับ 10 กิโลโอห์ม โดยที่ป้อนแรงเคลื่อนไฟฟ้า 5 โวลต์ ให้กับโพเทนชิโอมิเตอร์ ซึ่งจะทำให้ได้แรงเคลื่อนเอาต์พุตมีค่าอยู่ในช่วง 0 ถึง 5 โวลต์ ดังนั้นจึงไม่ต้องใช้วงจรปรับสภาพสัญญาณดังนั้นก็ทิศทางหรือมุมที่ได้จะมีค่าดังต่อไปนี้

$$\theta = \frac{V_o}{V} \theta_{max} \quad (3.2)$$

3.3 การออกแบบโปรแกรมแสดงผลการวัดค่าความเร็วลมและทิศทางลม

การเขียนโปรแกรมในการวัดค่าความเร็วลมและทิศทางลม ค่าความเร็วลมและทิศทางลมจะแสดงผลข้อมูลที่ได้ผ่านทางจอ LCD MODULE ซึ่งแสดงทั้งค่าความเร็วลมและทิศทางลม โดยการคำนวณและประมวลผลด้วย Microcontroller เบอร์ 16F877A ทำหน้าที่หลักในการคำนวณค่าโดยสัญญาณ pulse จากเครื่องวัดความเร็วลม และ สัญญาณแรงดัน จากเครื่องวัดทิศทางลม โดยแสดงผลผ่านจอ LCD และผ่านทางสายสัญญาณ RS-232 โดยในส่วนโปรแกรมหลักในการวัดความเร็วลมจะใช้สัญญาณจาก Opto Sensor ในการสร้างพัลส์ และทิศทางลมจะใช้สัญญาณแรงดัน 0-5 V ซึ่งจะสามารถอธิบายได้ตามบล็อกไดอะแกรมต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น **รูปที่ 3.26** บล็อกไดอะแกรมการทำงานของโปรแกรมแสดงผล

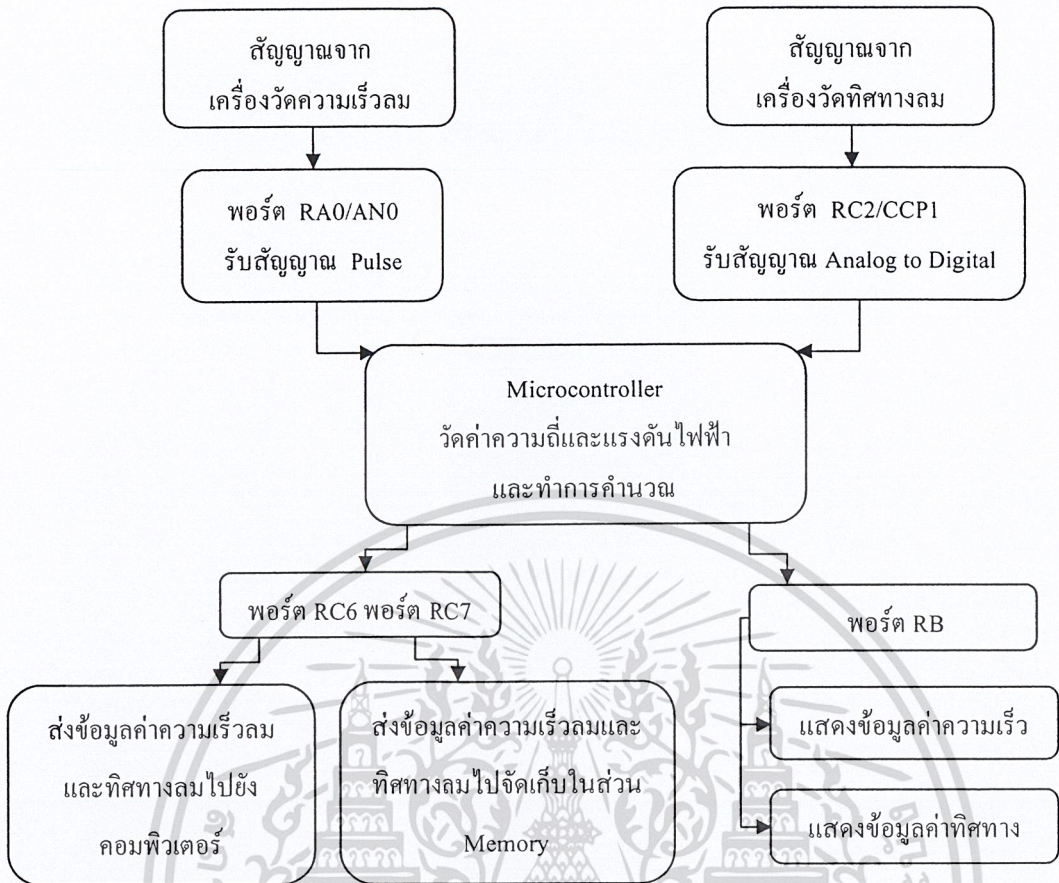
เมื่อทำการออกแบบการทำงานของวงจรวัดความเร็วและทิศทางลมแล้ว ในส่วนต่อไปจะเป็นการแสดงถึงรายละเอียดของวงจร ซึ่งประกอบไปด้วยวงจรวัดความเร็วลมและวงจรวัดทิศทางลม โดยมีหลักการการทำงานที่แตกต่างกันออกไป มีรายละเอียดดังต่อไปนี้

ในส่วนของการควบคุมการทำงานของอุปกรณ์ทุกตัวจะใช้ Microcontroller เบอร์ 16F877A เป็นส่วนควบคุมทั้งในการรับสัญญาณ การคำนวณ การติดต่อสื่อสารผ่าน RS-232 และการแสดงผลผ่านจอ ซึ่งมีการสื่อสารตามฟังก์ชันที่ต้องการเนื่องจากมีการทำงานในโหมด Analog to Digital โหมด timer 1 โหมด CCP1 หรือโหมด Capture สัญญาณอินเตอร์รัปต์ ซึ่งมีความจำเป็นในการคำนวณหาค่าความเร็วลมและทิศทางลม โดยการเขียนโปรแกรมการทำงานด้วยภาษา CCS Compiler บนโปรแกรม PICC Compiler และทำการจำลองโปรแกรมการใช้งานด้วยโปรแกรม Proteus เพื่อวิเคราะห์การทำงาน และทำการสร้างวงจบบน PCB ผ่านโปรแกรม Protel 99SE ในการทำแผ่น PCB

3.3.1 โปรแกรมการทำงานของไมโครคอนโทรลเลอร์

การทำงานของโปรแกรมทั้งหมดจะเริ่มจากการรับข้อมูลผ่านทางพอร์ต RA0/AN0 และ พอร์ต RC2/CCP1 เพื่อทำการรับสัญญาณจากเครื่องมือวัดความเร็วลมและทิศทางลมส่งต่อไปยังส่วนประมวลผลโดย Microcontroller เพื่อทำการหาค่าที่ได้จากเครื่องมือวัดความเร็วลมและทิศทางลม แสดงผลบนจอ LCD ขนาด 16x2 โดยแสดงค่าความเร็วลมในแถวบน และค่าทิศทางลมในแถวล่าง หลังจากนั้นทำการบันทึกข้อมูลลงใน USB FLASH DRIVE โดยผ่านการเชื่อมต่อกับ ET-USB FLASH DRIVE โดยการส่งข้อมูลผ่าน USART(RS-232) ในการเขียนข้อมูลลงใน FLASH DRIVE โดยสามารถสรุปการทำงานของวงจรได้ตามบล็อกไดอะแกรมต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 บล็อกไดอะแกรมการทำงานของไมโครคอนโทรลเลอร์

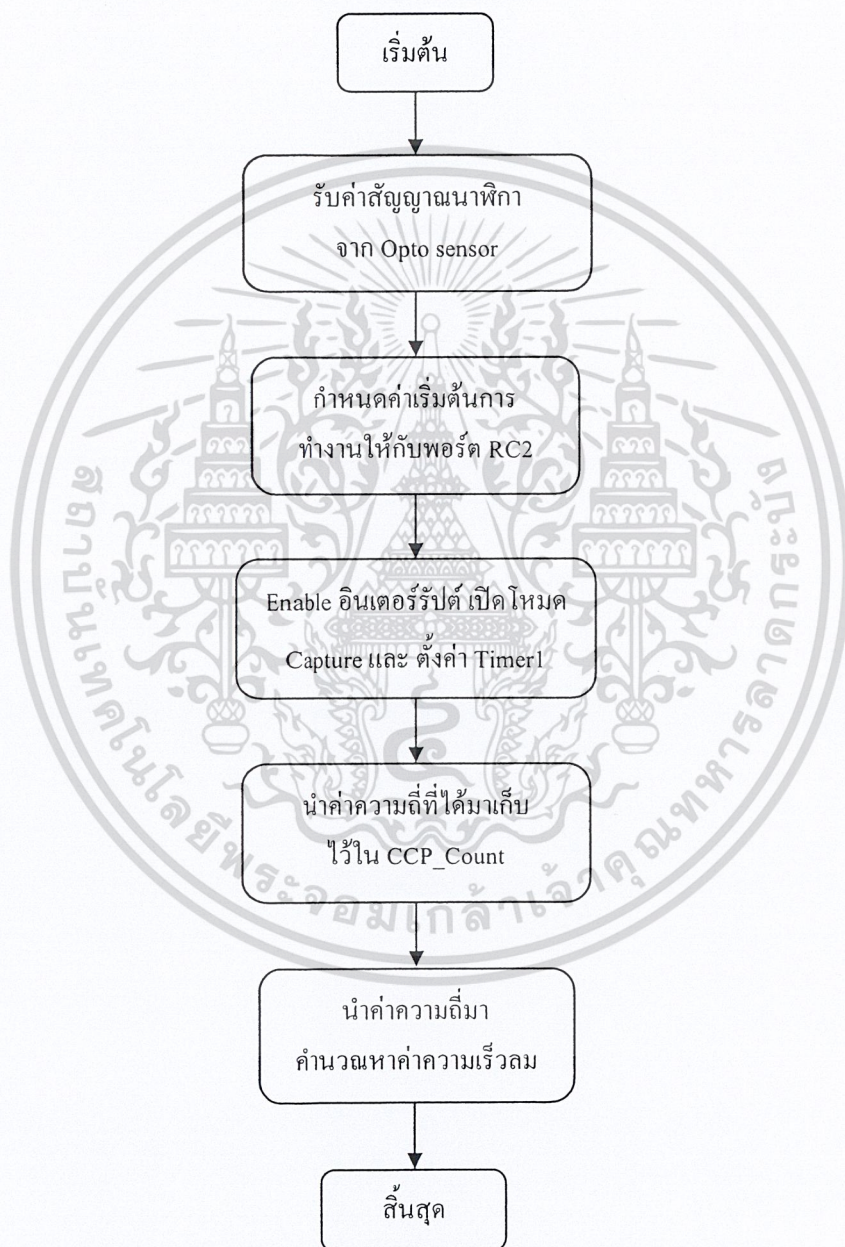
ตารางที่ 3.1 การกำหนดตำแหน่งพอร์ตให้กับอุปกรณ์

อุปกรณ์	ขาสัญญาณ	พอร์ต
สัญญาณจาก opto sensor	Pulse	RC2/CCP1
สัญญาณแรงดัน 0-5 V	Analog	RA0/AN0
LCD Module	RS(Register Select)	RB1/RD1
	R/W(Read/Write)	RB2/RD2
	E(Enable)	RB0/RD0
LCD Module	DATA	RB4/RD4
	DATA	RB5/RD5
	DATA	RB6/RD6
	DATA	RB7/RD7
ชุดส่งถ่ายข้อมูล RS-232	RX	RC7
	TX	RC6

เอกสารนี้จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 โปรแกรมที่ใช้คำนวณค่าความเร็วลม

จะรับสัญญาณนาฬิกาขนาด 0-5 โวลต์ มาโดยผ่านทางพอร์ต RC2/CCP1 ซึ่งในโหมดนี้จะใช้ timer1 ในการอินเทอร์รัปต์สัญญาณและ Capture สัญญาณหลังจากนั้นทำการเขียนโปรแกรมโดยใช้โหมด timer 1 ในการจับสัญญาณความถี่ของ pulse ในช่วงเวลา 1 วินาทีโดยทำการปรับเทียบค่าความเร็วลม ณ ตำแหน่งลมต่างๆ เพื่อทำการคำนวณค่าซึ่งมีหลักการทำงานดังแผนภาพต่อไปนี้



รูปที่ 3.28 ไตอะแกรมการคำนวณความเร็วลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 โปรแกรมที่ใช้คำนวณค่าทิศทางลม

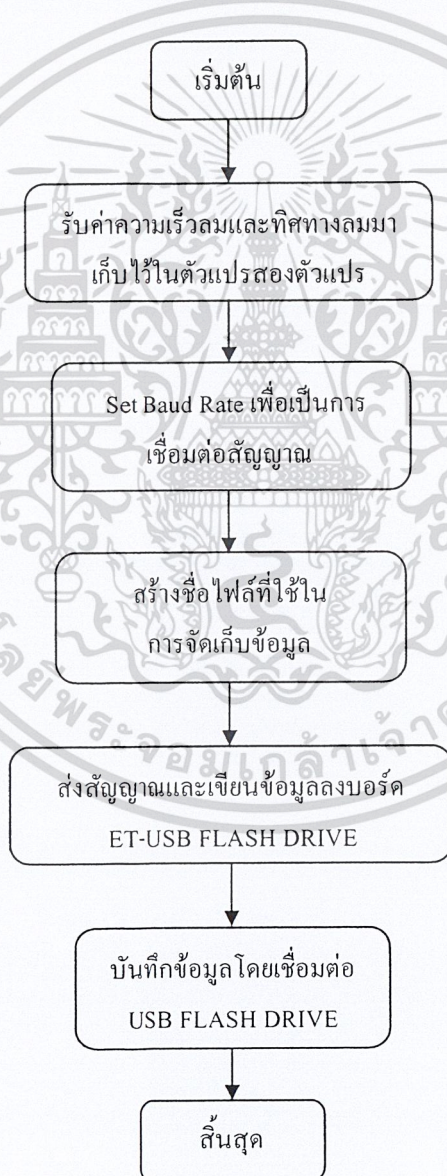
จะรับแรงดันขนาด 0-5 V มาโดยผ่านทางพอร์ต RA0/AN0 ซึ่งในไมโครนี้จะใช้การทำงานในไมโคร A/D มาทำการรับสัญญาณโดยการจ่ายแรงดันผ่าน Potentiometer 360 ซึ่งสามารถหมุนได้รอบจึงเหมาะแก่การนำมาทำการสร้างเครื่องมือในการวัดทิศทางลม โดยจะให้สัญญาณ 0-5 V หลังจากนั้นทำการเขียนโปรแกรมโดยใช้ไมโคร AN0 ในการแปลงสัญญาณ Analog เป็นสัญญาณ Digital โดยที่แรงดัน 0-5 V มีค่าเท่ากับ 256 bit โดยทำการปรับเทียบค่าทิศทางลม ณ ตำแหน่งแรงดันต่างๆ เพื่อทำการคำนวณค่าซึ่งมีหลักการทำงานดังแผนภาพต่อไปนี้



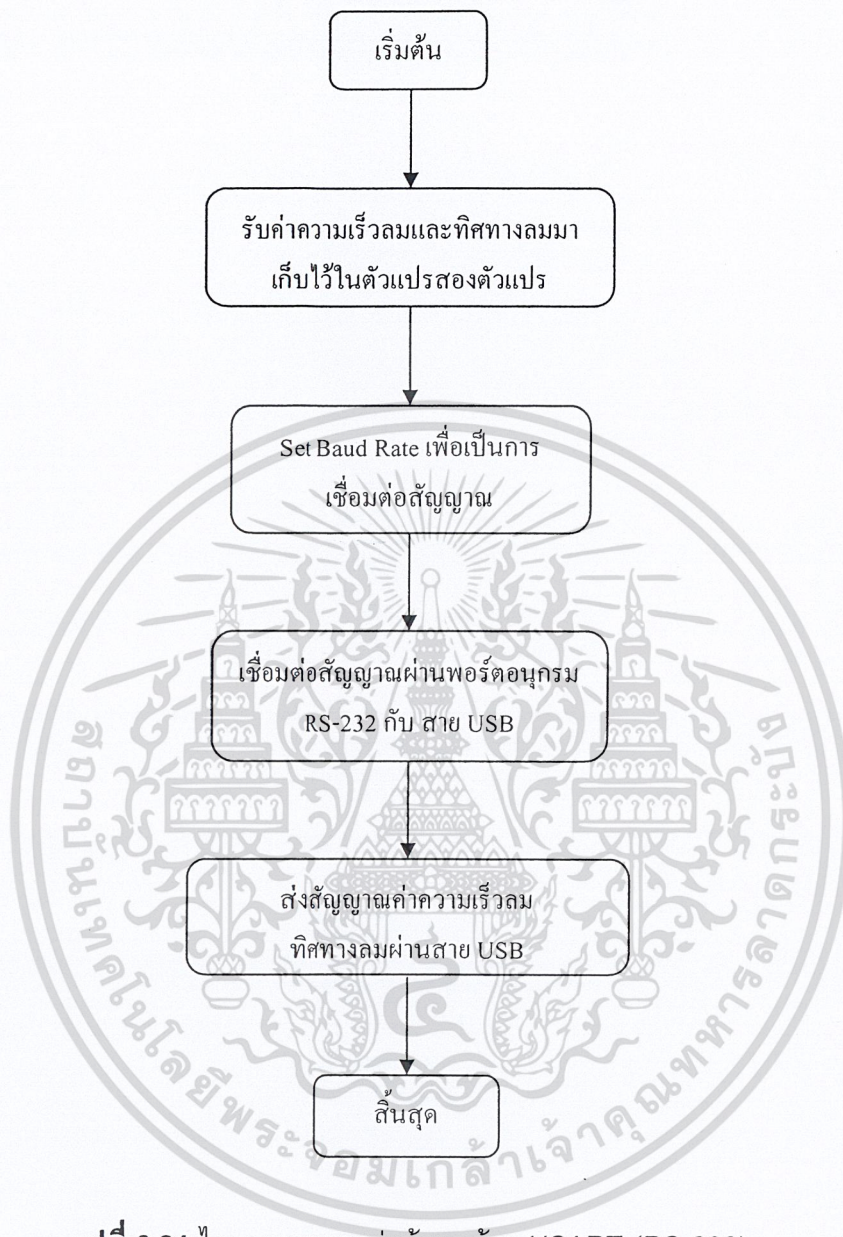
เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.29 ไดอะแกรมการคำนวณทิศทางลมให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การส่งข้อมูลด้วยโมดูล USART (RS-232)

การส่งข้อมูลผ่าน USART(RS-232) เพื่อทำการส่งสัญญาณโดยทำการส่งข้อมูล ความเร็วลมและทิศทางลม โดยหลักการทำงานจะทำงานร่วมกับบอร์ด ET-USB FLASH DRIVE และสาย USB ซึ่งทำหน้าที่เพื่อทำการบันทึกข้อมูลลง USB FLASH DRIVE และส่งข้อมูลไปยังคอมพิวเตอร์โดยการส่งข้อมูลจะทำการส่งสัญญาณผ่านทางพอร์ต RX และ TX ของไมโครคอนโทรลเลอร์ โดยการทำงานของ USART ซึ่งใช้งานในโหมดอะซิงโครนัสแบบ Full duplex ซึ่งสามารถรับส่งข้อมูลได้ 2 ทิศทางในเวลาเดียวกัน โดยสามารถเขียนคำสั่งในการ Read/Write Open file กับบอร์ด ET-USB FLASH DRIVE ได้ และสามารถนำสายสัญญาณไปเชื่อมต่อกับสาย USB ประมวลผลผ่านคอมพิวเตอร์ โดยใช้บอร์ดต่อร่วมกับคอมพิวเตอร์ได้ ซึ่งการทำงานรับส่งข้อมูลมีหลักการทำงานดังแผนภาพดังต่อไปนี้



เอกสารนี้เป็นเอกสาร **รูปที่ 3.30** ไดอะแกรมการบันทึกข้อมูลด้วย USART (RS-232) ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.31 ไตอะแกรมการส่งข้อมูลด้วย USART (RS-232)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบและสร้างโปรแกรมแสดงผลบนจอคอมพิวเตอร์

3.5.1 ออกแบบโปรแกรมแสดงผลบนจอคอมพิวเตอร์

ในการสร้างโปรแกรมแสดงผลบนจอคอมพิวเตอร์ ขั้นตอนแรกจะต้องกำหนดการทำงานของโปรแกรมก่อนว่า โปรแกรมที่จะสร้างนั้นสามารถทำงานอะไรได้บ้าง แล้วจึงมาออกแบบหน้าต่างของแอปพลิเคชันให้มีรูปแบบที่เข้าใจง่าย ไม่ทำงานซ้ำซ้อน ฟังก์ชันการทำงานครบถ้วนตามที่กำหนด ซึ่งอาจจะร่างแบบคร่าวๆ ลงในกระดาษเสียก่อน แล้วจึงลงมือสร้างโปรแกรมจริง เพื่อให้ง่ายต่อการเห็นภาพหน้าต่างแอปพลิเคชันและเครื่องมือที่ต้องใช้ในโปรแกรม

หลังจากที่ได้ออกแบบหน้าต่างของโปรแกรมแล้วต่อไปจะกำหนดการทำงานของขั้นตอนการทำงานของโปรแกรม โดยสามารถแบ่งการทำงานของโปรแกรมออกเป็นโปรแกรมต่างๆ ได้ 4 ส่วนหลักๆ ได้แก่

1. ส่วนเชื่อมต่อข้อมูลระหว่างอุปกรณ์และคอมพิวเตอร์
1. ส่วนแสดงข้อมูลแบบหน้าปัดและตัวเลขดิจิทัล
2. ส่วนแสดงข้อมูลแบบฐานข้อมูล
3. ส่วนแสดงข้อมูลความเร็วลมแบบกราฟ

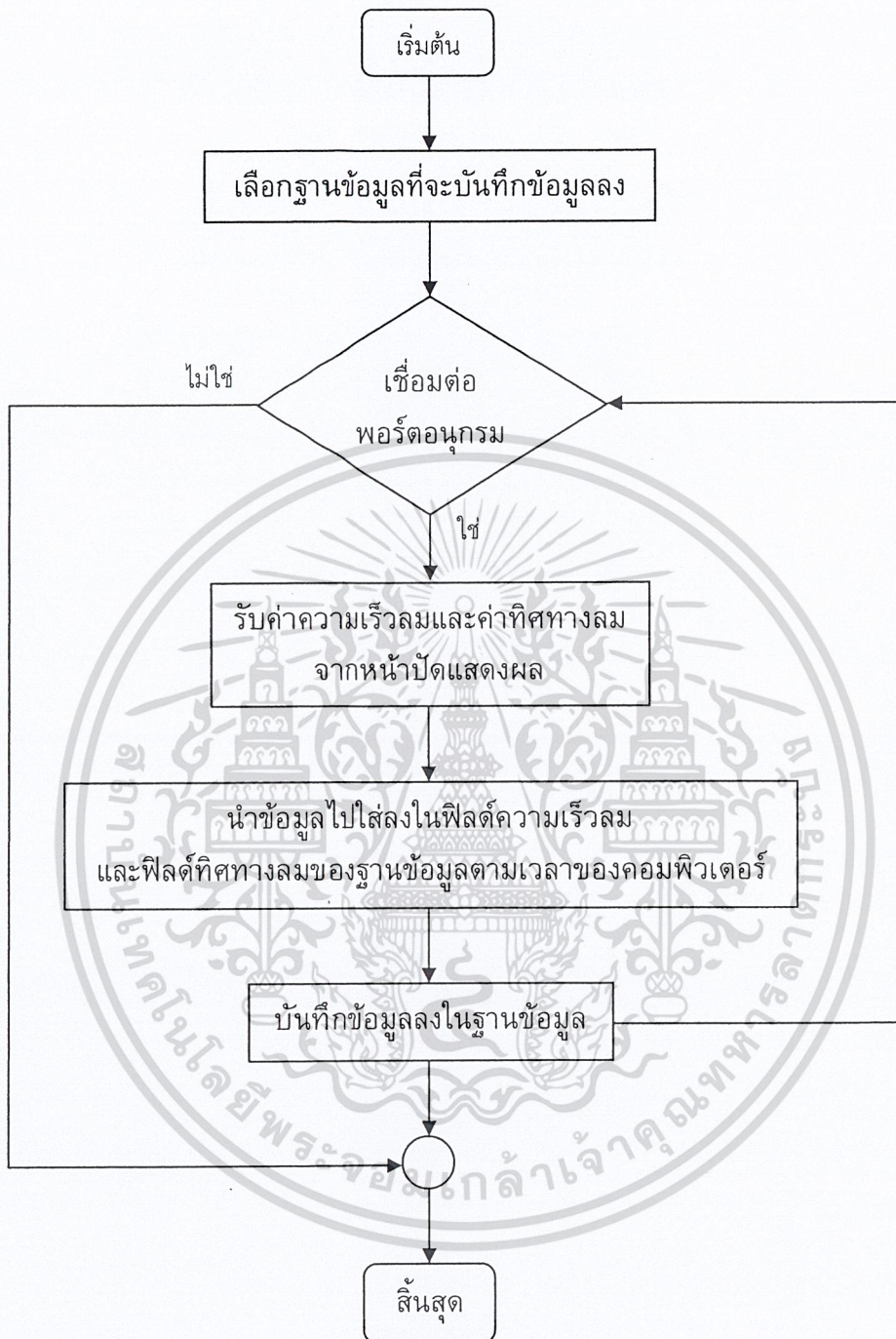
การอธิบายการทำงานของในแต่ละส่วนโปรแกรม จะใช้ไดอะแกรมแทนลำดับขั้นตอนในแต่ละลำดับขั้นแนวคิด

3.5.1.1 ส่วนเชื่อมต่อข้อมูลระหว่างอุปกรณ์และคอมพิวเตอร์

โปรแกรมจะทำการรับข้อมูลค่าความเร็วลมและค่าทิศทางลมจากไมโครคอนโทรลเลอร์ มาเก็บเป็นไฟล์ฐานข้อมูล Access database File (.mdb) โดยในฐานข้อมูลจะเก็บค่าความเร็วลม ค่าทิศทางลมตามค่าวันเดือนปีและเวลาของข้อมูลนั้นๆ

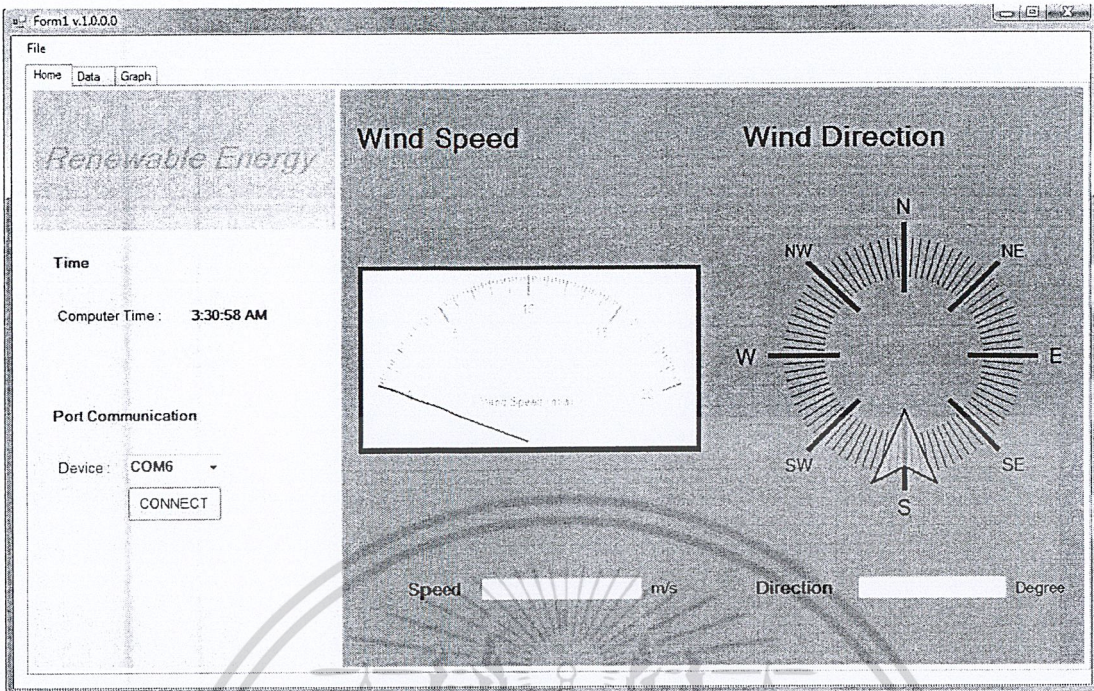
ออกแบบให้โปรแกรมรับข้อมูลจากไมโครคอนโทรลเลอร์ด้วยการสื่อสารข้อมูลแบบอนุกรม โดยเชื่อมต่ออุปกรณ์กับคอมพิวเตอร์ผ่านทางคอนโพรต Comport ของโปรแกรม จากนั้นส่งค่าความเร็วลมและค่าทิศทางลมไปเก็บยังไฟล์ฐานข้อมูล Access database File (.mdb) ร่วมกับเวลา Timer ของคอมพิวเตอร์ที่เป็นตัวรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

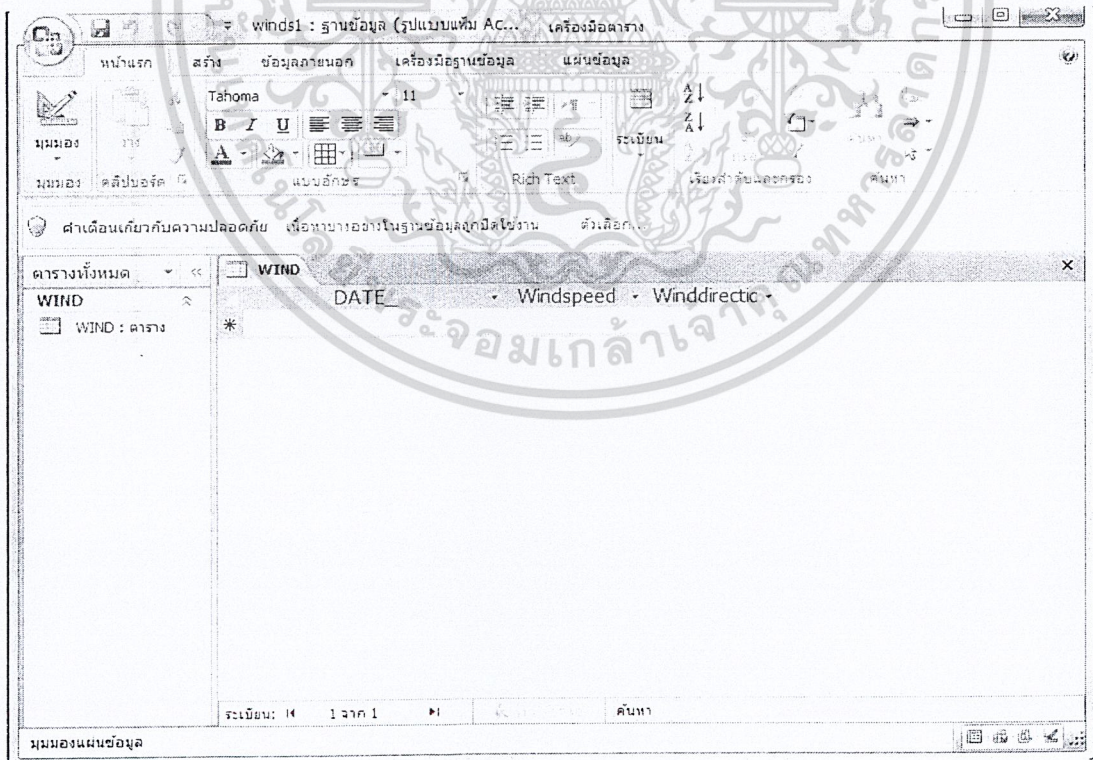


รูปที่ 3.32 ไตอะแกรมของส่วนเชื่อมต่อข้อมูลและจัดเก็บค่าลงฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



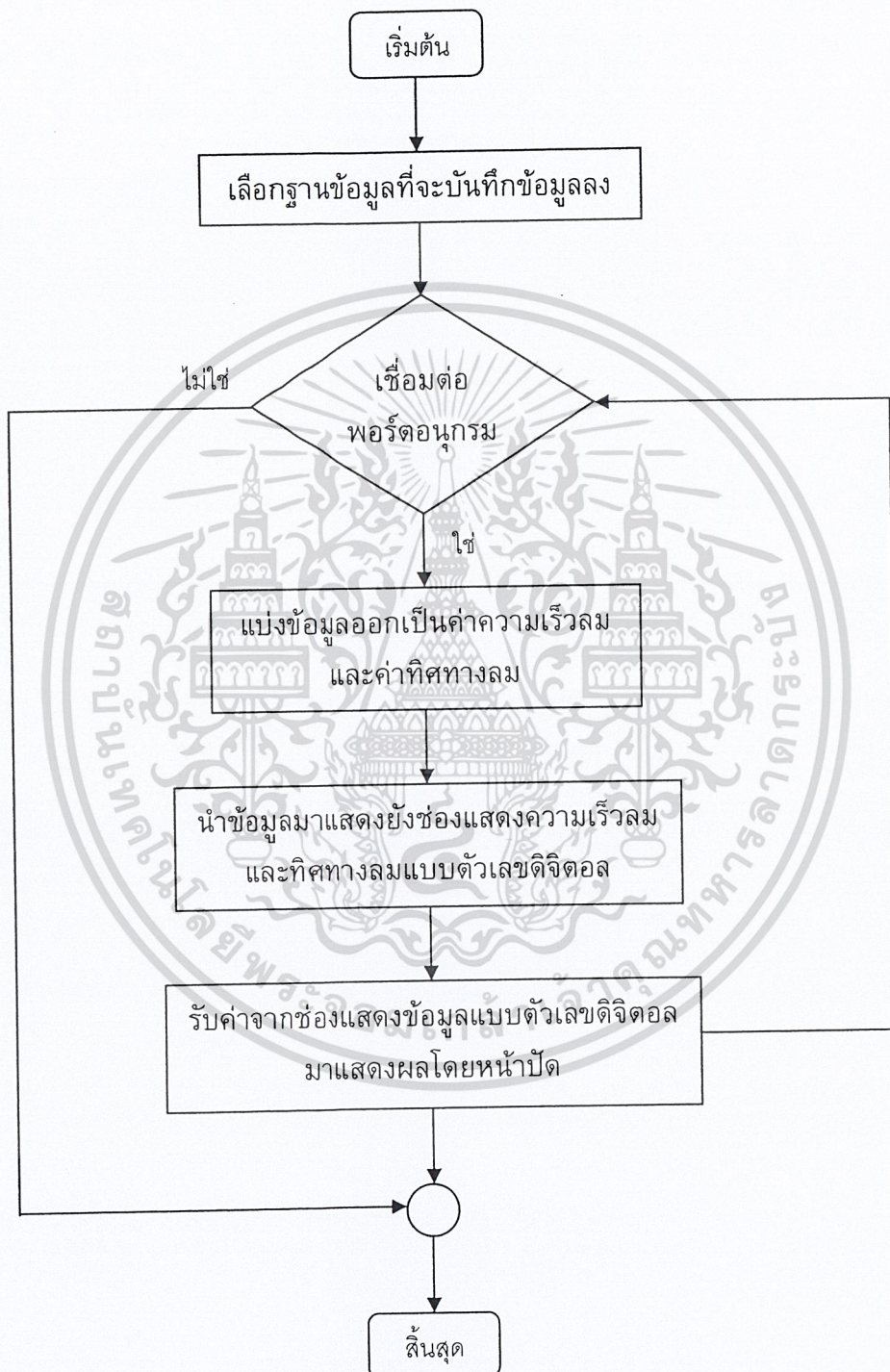
รูปที่ 3.33 การใช้คอมโพเนนต์ Comport เพื่อรับข้อมูลจากอุปกรณ์ภายนอก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.34 ตารางฐานข้อมูล(.mdb) ที่เชื่อมต่อกับโปรแกรม
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

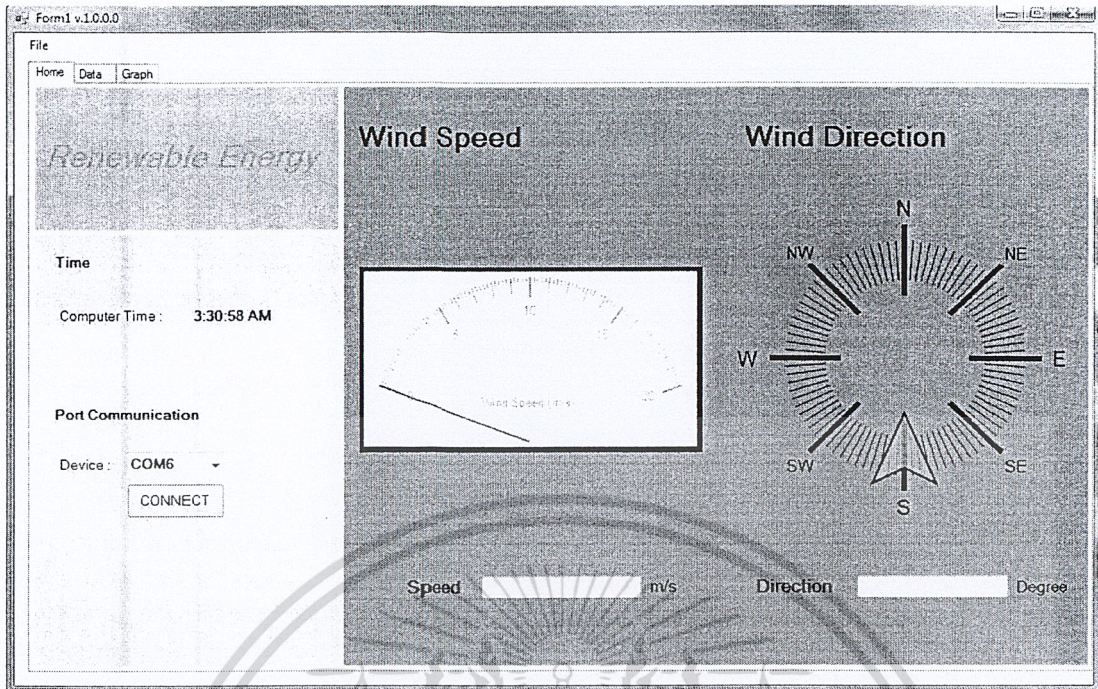
3.5.1.2 ส่วนแสดงข้อมูลแบบหน้าปัดและตัวเลขดิจิทัล

ได้ออกแบบให้มีการแสดงผลแบบหน้าปัดและตัวเลขดิจิทัลเพื่อให้สะดวกต่อการดูข้อมูลที่เวลาจริงในระยะไกล อีกทั้งยังง่ายและเร็วต่อการสังเกตค่าที่เวลาจริงมากกว่ารูปแบบอื่นๆ



รูปที่ 3.35 ไตอะแกรมของโปรแกรมแสดงข้อมูลแบบหน้าปัดและตัวเลขดิจิทัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



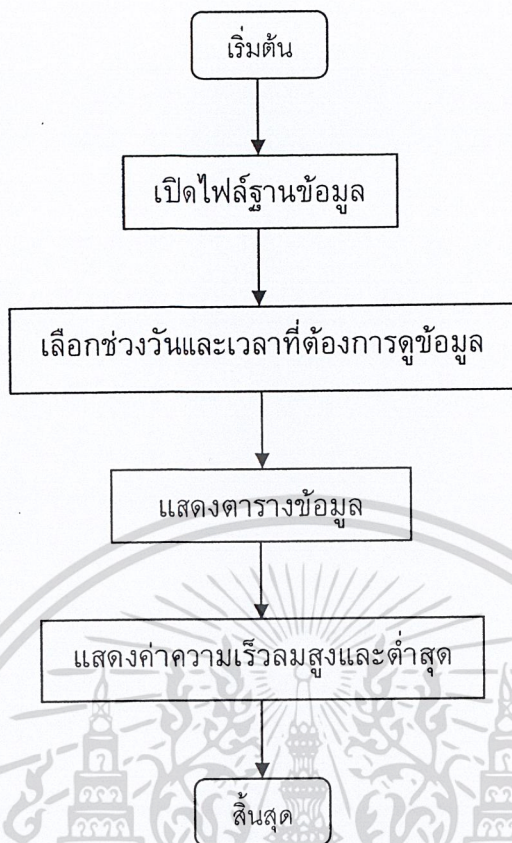
รูปที่ 3.36 หน้าต่างแสดงข้อมูลแบบหน้าปัดและตัวเลขดิจิทัล

3.5.1.3 ส่วนแสดงข้อมูลแบบฐานข้อมูล

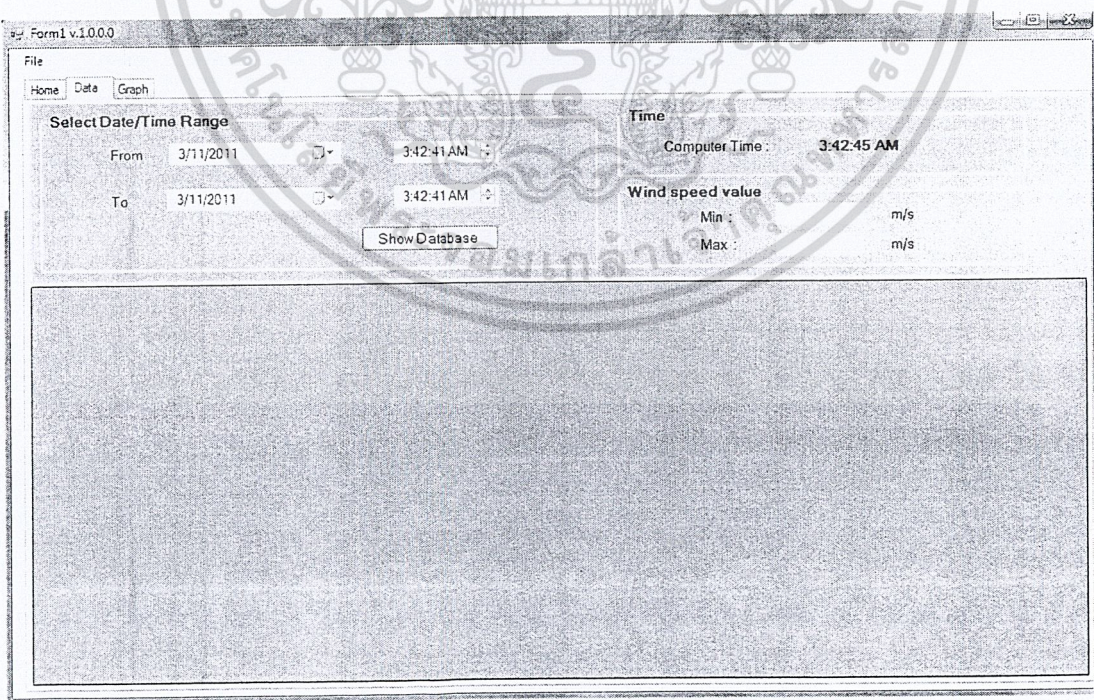
ได้ออกแบบให้มีการแสดงผลแบบฐานข้อมูลเพื่อใช้สำหรับวิเคราะห์ สังเกตข้อมูลที่มีจำนวนมาก ๆ อย่างละเอียด ซึ่งเราสามารถนำข้อมูลจากฐานข้อมูลนี้ไปประยุกต์ใช้ในการหาค่าพลังงานลม หรือนำไปประยุกต์ คำนวณร่วมกับโปรแกรมอื่น ๆ ได้ต่อไป โดยได้แสดงผลค่าความเร็วลมและทิศทางลมร่วมกับวันเวลาของข้อมูล ณ ขณะนั้น

ได้อะแกรรมของโปรแกรมแสดงข้อมูลแบบฐานข้อมูล จะแสดงดังรูปที่ 3.37 หลักการคือ เริ่มต้นจะต้องเปิดไฟล์ฐานข้อมูล Access database File (.mdb) โดยเป็นไฟล์ข้อมูลเดียวกับที่ได้เชื่อมต่อเพื่อรับข้อมูลจากเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลมมาเก็บไว้ แล้วจึงเลือกช่วงวันเดือนปีและเวลาที่ต้องการดูข้อมูลก่อน จากนั้นจะนำค่าความเร็วลมและทิศทางลม มาใส่ในตารางข้อมูลตามวัน เดือน และปี สุดท้ายก็จะแสดงตารางข้อมูลยังฟิลด์ของฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.37 ไดอะแกรมของโปรแกรมแสดงข้อมูลแบบฐานข้อมูล

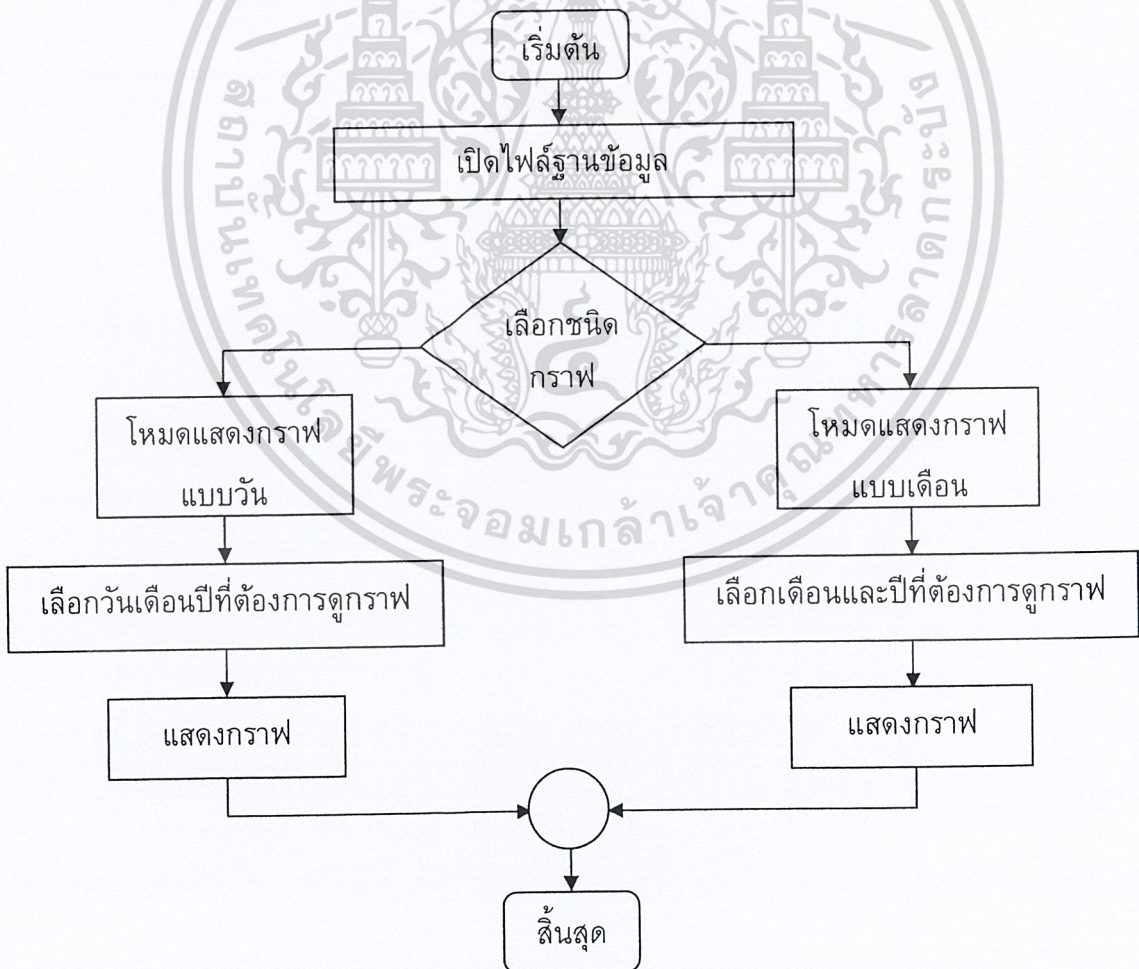


เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.38 หน้าต่างแสดงข้อมูลแบบฐานข้อมูลให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

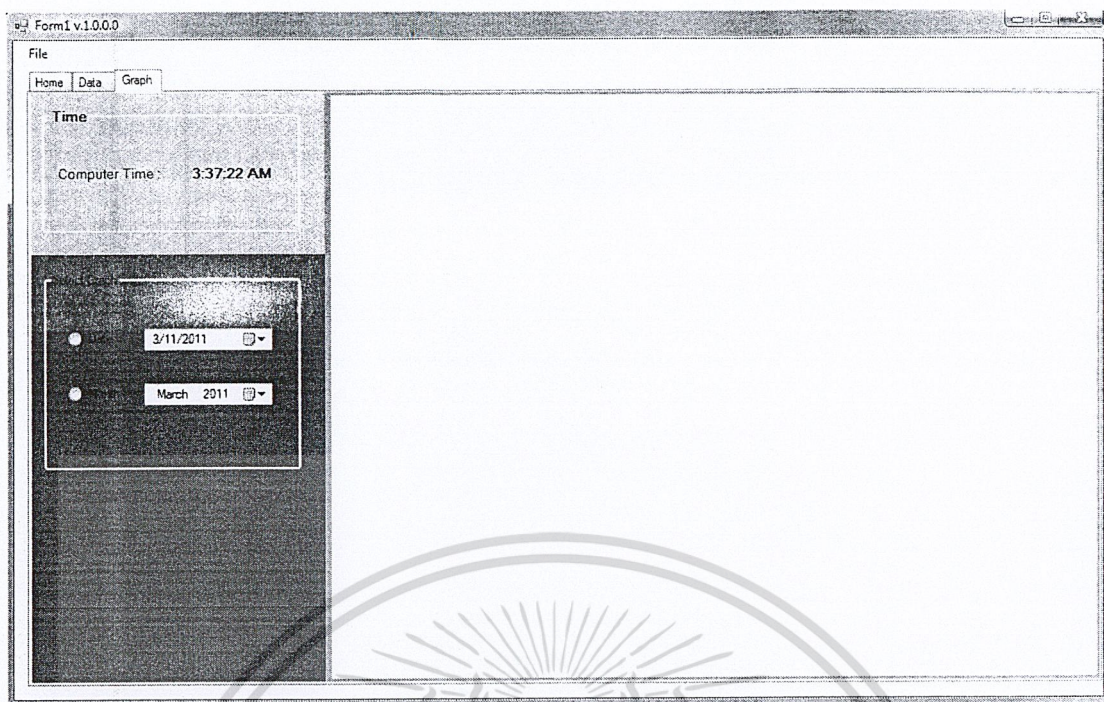
3.5.1.4 ส่วนแสดงข้อมูลความเร็วลมแบบกราฟ

ได้ออกแบบให้มีการแสดงผลข้อมูลความเร็วลมแบบกราฟ เพื่อให้สะดวกต่อการสังเกต และเปรียบเทียบข้อมูลที่มีจำนวนมาก โดยเมื่อดูกราฟแบบวันก็จะทำให้เห็นว่าความเร็วลมที่ เวลาต่างๆในวันที่เลือกนั้นเป็นอย่างไร มีการเปลี่ยนอย่างไร และเมื่อดูกราฟแบบเดือนก็จะทำให้เห็นว่าความเร็วลมในแต่ละวันของเดือนนั้นเป็นอย่างไร มีการเปลี่ยนแปลงอย่างไร โดยการที่ไม่ได้ออกแบบให้แสดงผลค่าทิศทางลมแบบกราฟนั้น เนื่องจากข้อมูลมีจำนวนมาก ทำให้แสดงผลของข้อมูลได้ไม่ดี นำมาเปรียบเทียบและสรุปผลไม่ได้

ได้อะแกรมของโปรแกรมแสดงข้อมูลความเร็วลมแบบกราฟ มีหลักการคือ เริ่มต้น จะต้องเปิดไฟล์ฐานข้อมูล Access database File(.mdb) โดยเป็นไฟล์ข้อมูลเดียวกับที่ได้ เชื่อมต่อเพื่อรับข้อมูลจากเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลมมาเก็บไว้ แล้วจึงเลือก โหมดการแสดงผลของกราฟ ซึ่งจะมีให้เลือก 2 โหมดคือ โหมดแสดงกราฟแบบวันและโหมด แสดงกราฟแบบเดือน ซึ่งถ้าเลือกวันก็ต้องเลือกวันที่ต้องการจะดูข้อมูลก่อน สุดท้ายก็จะนำ ข้อมูลความเร็วลมของวันนั้นมาพล็อตกราฟและแสดงรูปกราฟออกมา โดยโหมดการแสดงผล กราฟแบบเดือนก็จะมีลักษณะเช่นเดียวกัน



เอกสารนี้เป็นเอกสารที่ **รูปที่ 3.39** ได้อะแกรมของโปรแกรมแสดงข้อมูลความเร็วลมแบบกราฟ ระเบียบด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.40 หน้าต่างแสดงข้อมูลความเร็วลมแบบกราฟ

3.5.2 สร้างโปรแกรมแสดงผลบนจอคอมพิวเตอร์

หลังจากที่ได้ออกแบบหน้าจอโปรแกรม และการทำงานของโปรแกรมต่อไปจะเป็นขั้นตอนในการสร้างโปรแกรม โดยจะมีลำดับขั้นดังนี้

1. สร้างหน้าจอของโปรแกรมตามที่ยกแบบไว้ โดยนำเอาคอนโทรลต่างๆ ที่ต้องการในทูลบ็อกซ์มาวางไว้บนฟอร์ม
2. กำหนดค่าพรีอเพอร์ตี้ของคอนโทรลต่างๆตามที่ได้เลือกไว้
3. เขียนโปรแกรม หรือซอร์สโค้ดให้กับออปเจ็คแต่ละตัวตามที่ได้กำหนดไว้โดยรายละเอียดของซอร์สโค้ดของโปรแกรมทั้งหมดจะอยู่ในภาคผนวก
4. ทดสอบและรันโปรแกรมว่าทำงานถูกต้องตามเงื่อนไขหรือไม่
5. เมื่อโปรแกรมสามารถทำงานได้ตามเงื่อนไขก็ทำการบันทึกโปรแกรม และฟอร์มที่สร้างไว้
6. การคอมไพล์ เพื่อแปลงโปรแกรมโดยรวบรวมจากส่วนประกอบต่างๆ ออกมาเป็นไฟล์ที่สามารถทำงานได้เลย ซึ่งเรียกว่า เอ็กซีคิวทีฟไฟล์

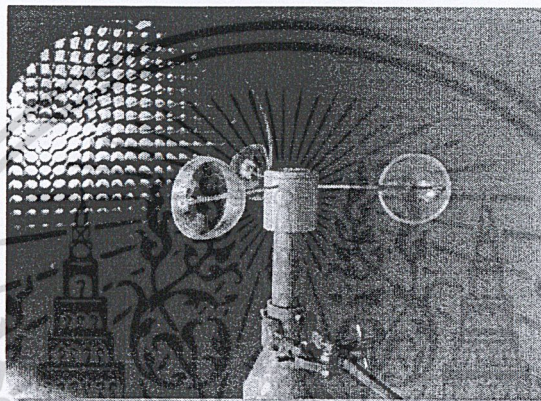
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

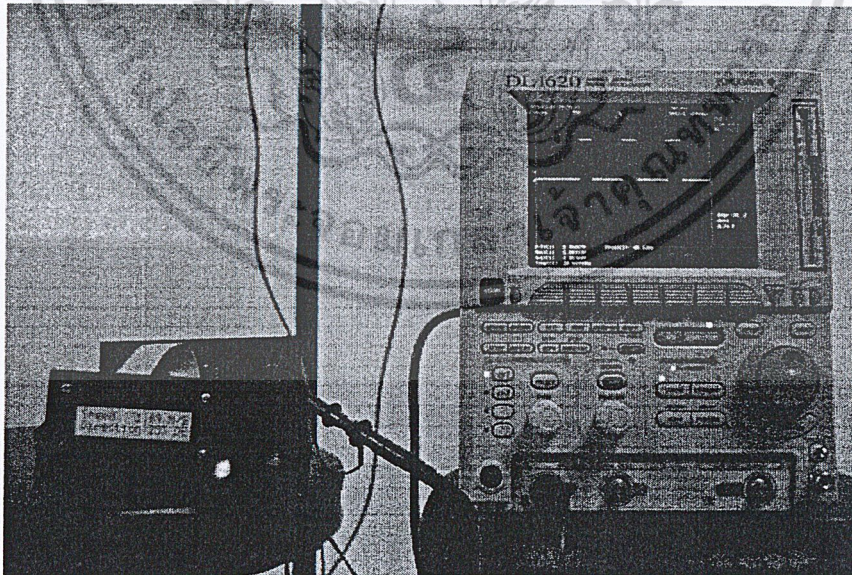
การทดลองและผลการทดลอง

4.1 ลักษณะของพัลส์ที่ความเร็วต่างๆ

การคำนวณหาค่าความเร็วลมสามารถคำนวณได้จากค่าความถี่ของพัลส์ จำนวนรอบที่เคลื่อนที่ได้ ซึ่งจะมีความแตกต่างกันที่ความเร็วต่างกัน โดยเปรียบเทียบความเร็วลมอ้างอิงจากอุโมงค์ลมของกรมอุตุนิยมวิทยา ซึ่งสามารถปล่อยความเร็วลมที่มีค่าสม่ำเสมอ ทำให้ค่าที่คำนวณได้มีความใกล้เคียงความเป็นจริง



รูปที่ 4.1 รับความเร็วอ้างอิงจากอุโมงค์ลม



รูปที่ 4.2 การวัดสัญญาณพัลส์

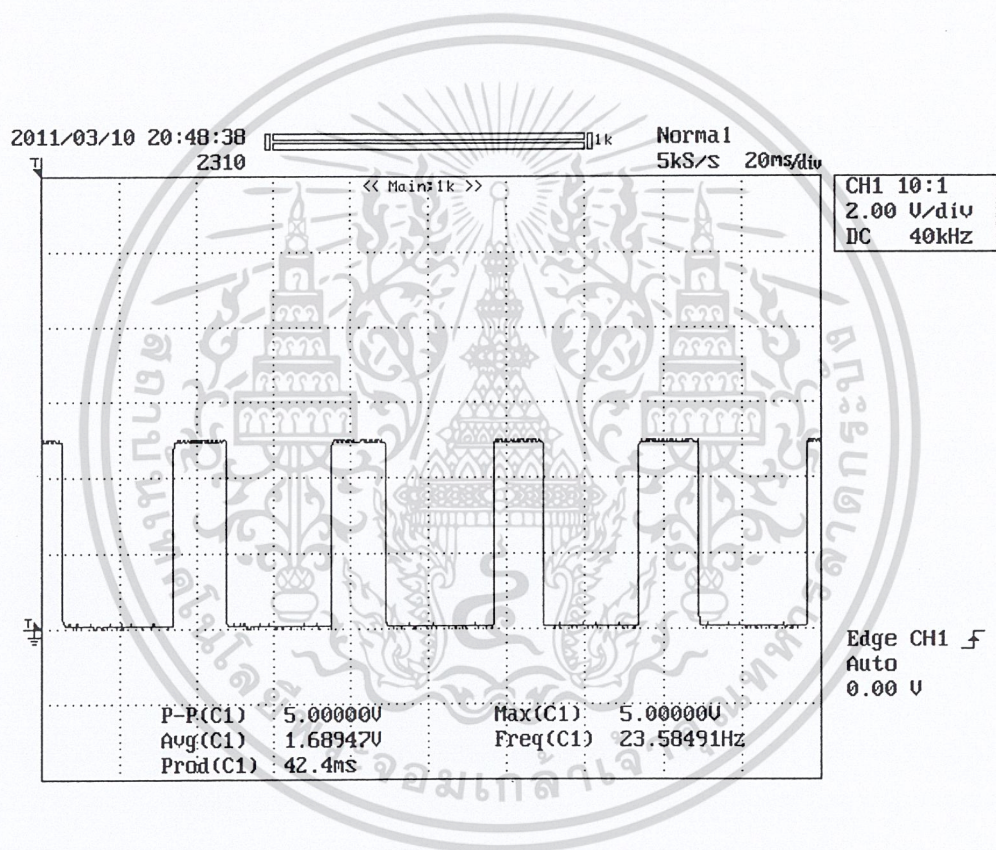
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณหาความเร็วต่าง ๆ ตามสมการต่อไปนี้

$$v = \frac{2\pi rN}{t} \quad (4.1)$$

จากสมการได้คำนวณหาค่าความเร็วเริ่มต้นซึ่งได้แสดงที่ความเร็วต่าง ๆ ดังนี้

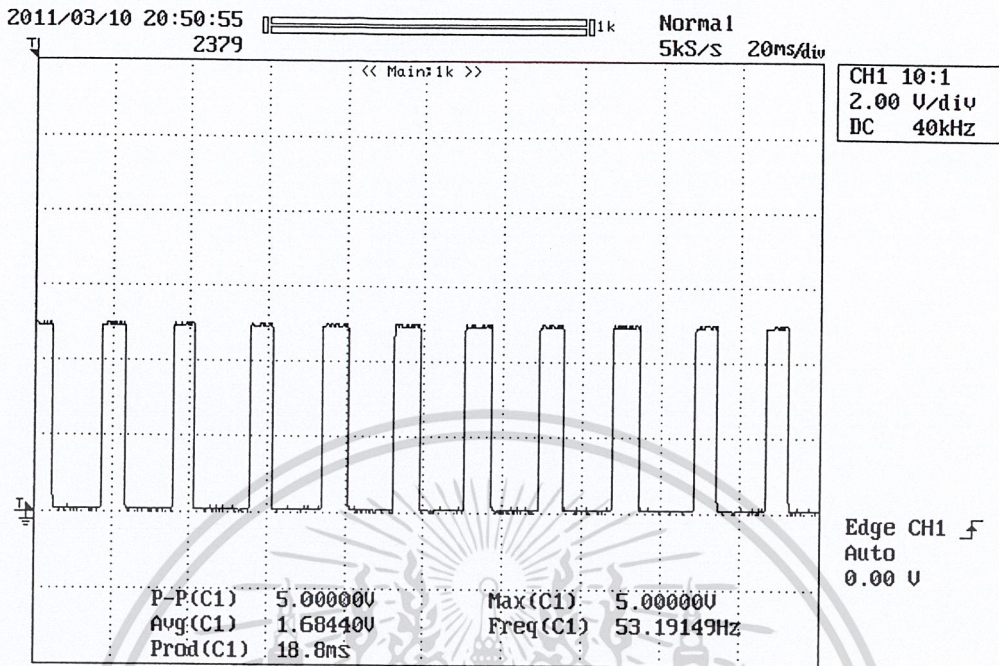
ที่ความถี่ 23.58 Hz จากการคำนวณ $v = \frac{2 \times 3.14 \times 11 \times 23.58 \times 1}{100 \times 17} = 0.95 \text{ M/S}$



รูปที่ 4.3 ความถี่พัลส์ที่ความเร็วลม 1 เมตรต่อวินาที

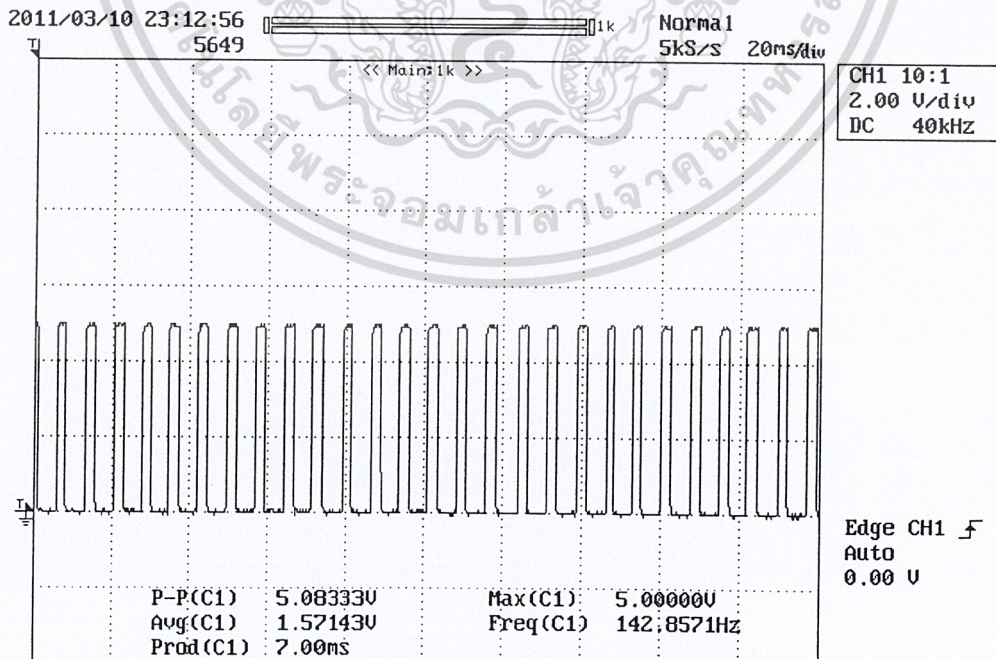
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ความถี่ 53.19 Hz จากการคำนวณ $v = \frac{2 \times 3.14 \times 11 \times 53.19 \times 1}{100 \times 18} = 2.04 \text{ M/S}$



รูปที่ 4.4 ความถี่พัลส์ที่ความเร็วลม 2 เมตรต่อวินาที

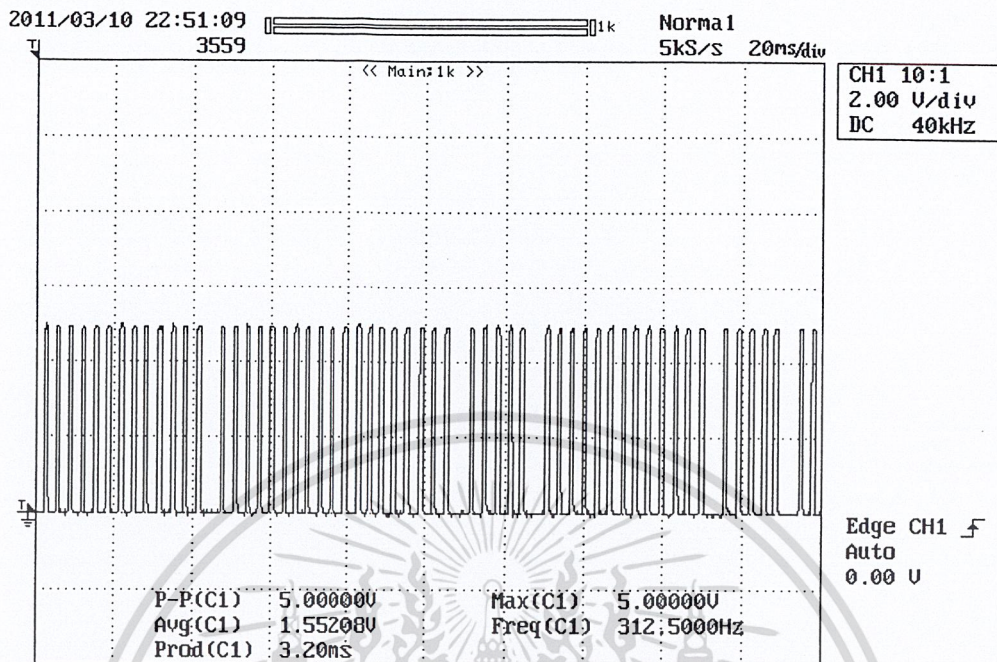
ที่ความถี่ 142.85 Hz จากการคำนวณ $v = \frac{2 \times 3.14 \times 11 \times 142.85 \times 1}{100 \times 20} = 4.93 \text{ M/S}$



รูปที่ 4.5 ความถี่พัลส์ที่ความเร็วลม 5 เมตรต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในอาคารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ความถี่ 312.5 Hz จากการคำนวณ $v = \frac{2 \times 3.14 \times 11 \times 312.5 \times 1}{100 \times 21} = 10.27 \text{ M/S}$



รูปที่ 4.6 ความถี่พัลส์ที่ความเร็วลม 10 เมตรต่อวินาที

ที่ความถี่ 454.54 Hz จากการคำนวณ $v = \frac{2 \times 3.14 \times 11 \times 454.54 \times 1}{100 \times 22.5} = 13.95 \text{ M/S}$



รูปที่ 4.7 แสดงความถี่พัลส์ที่ความเร็วลม 14 เมตรต่อวินาที

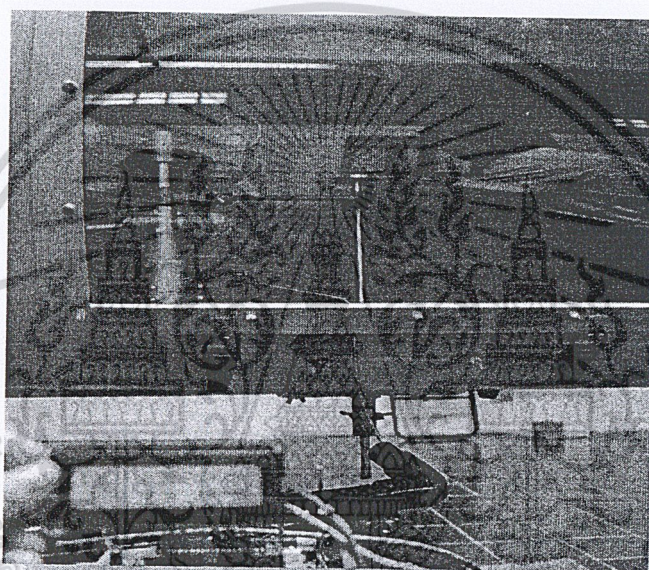
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 สอบเทียบเครื่องมือวัด

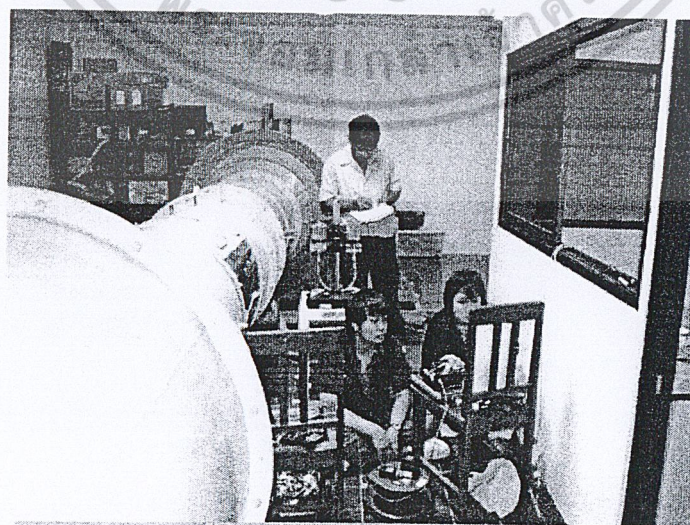
เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลมจะต้องผ่านการสอบเทียบค่ามาตรฐานของเครื่องมือวัดจากกรมอุตุนิยมวิทยาก่อนการนำไปใช้จริง

4.2.1 สอบเทียบเครื่องวัดความเร็วลม

ทำการสอบเทียบเครื่องวัดความเร็วลมโดยนำเครื่องวัดความเร็วลมที่สร้างไว้เข้าไปในอุโมงค์ลมและนำสายสัญญาณออกมาภายนอกเพื่อทำการวัดความเร็วลมที่ได้ กรมอุตุนิยมวิทยาจะทำการปล่อยลมจากเครื่อง Digital 645 ผ่านอุโมงค์ลม ตั้งแต่ 1 m/s จนถึง 15 m/s จากนั้นทำการอ่านผลความเร็วลมที่ได้และบันทึกค่าเพื่อนำมาเปรียบเทียบกับความถูกต้อง



รูปที่ 4.8 การทดสอบความเร็วลมภายในอุโมงค์ลม



รูปที่ 4.9 การสอบเทียบเครื่องวัดความเร็วลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

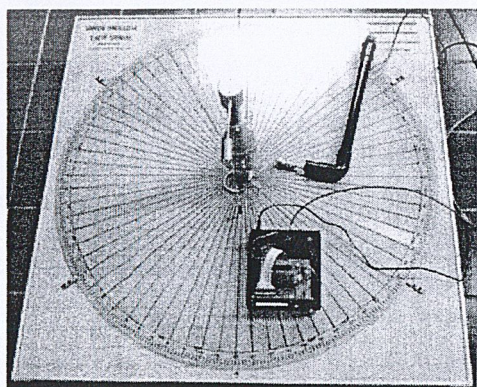
ผลการสอบเทียบความเร็วลม

ตารางที่ 4.1 ผลการสอบเทียบเครื่องวัดความเร็วลม

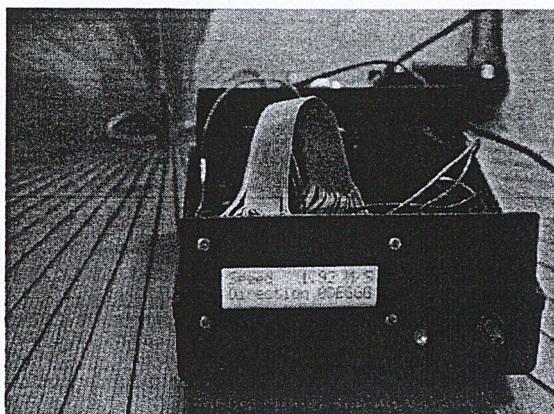
Ultrasonic Anemometer (m/sec)	Velocity of Tested Anemometer (m/sec)	Correction (m/sec)
1.01	0.95	0.06
2.01	2.00	0.01
3.00	3.00	0.00
4.00	3.96	0.04
5.01	4.93	0.08
6.01	6.13	-0.12
7.00	6.96	0.04
8.00	8.01	-0.01
9.01	8.91	0.10
10.01	10.00	0.01
11.02	10.98	0.04
12.03	12.02	0.01
13.02	12.90	0.12
14.01	14.00	0.01
15.01	15.03	-0.02

4.2.2 สอบเทียบเครื่องวัดทิศทางลม

ทำการสอบเทียบเครื่องวัดทิศทางลมโดยนำเครื่องวัดทิศทางลมมาวางไว้บนชุดวัดทิศทางลม โดยต่อนางเครื่องให้หน้า 0 องศาของเครื่องวัดทิศทางลมวางลงให้ตรงกับ 0 องศาของชุดวัดทิศทางลม จากนั้นทำการหมุนนครของเครื่องวัดทิศทางลมแล้วทำการอ่านผลทิศทางลมที่ได้และบันทึกค่าเพื่อนำมาเปรียบเทียบความถูกต้อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.10 การสอบเทียบเครื่องวัดทิศทางลม
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุผลแบบลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ตำแหน่งองศาขณะทดสอบเทียบ

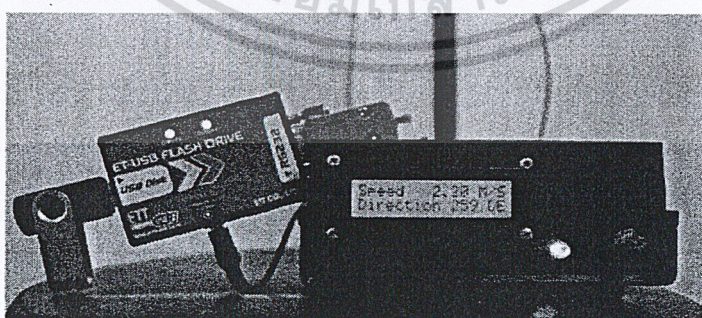
ผลการสอบเทียบทิศทางลม

ตารางที่ 4.2 ผลการสอบเทียบเครื่องวัดทิศทางลม

Wind Direction (DEG)	Tested Wind Direction (DEG)
0	0
90	90
180	180
270	270

4.3 การแสดงผลข้อมูลของโปรแกรมไมโครคอนโทรลเลอร์

ในรูปที่ 4.12 แสดงผลข้อมูลบนจอ LCD MODULE ขนาด 16 x 2 และการเชื่อมต่อกับบอร์ด ET-USB FLASH DRIVE ในส่วนการบันทึกข้อมูล

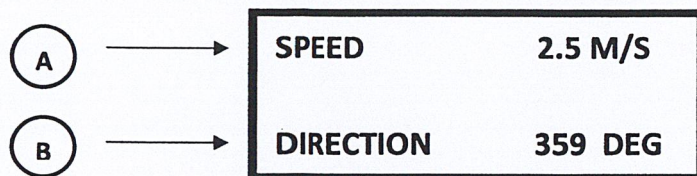


รูปที่ 4.12 การแสดงผลข้อมูลของโปรแกรมไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1 การแสดงผลข้อมูลความเร็วลมและทิศทางลมบนจอ LCD

การแสดงผลบนจอ LCD โดยการแสดงค่าบนจอ LCD MODULE โดยในแถวแรกจะเป็นการแสดงค่าความเร็วลม และในแถวที่สองจะแสดงค่าทิศทางลม



รูปที่ 4.13 LCD ขณะแสดงผลการทำงาน

ส่วน A เป็นส่วนที่แสดงค่าความเร็วลมในหน่วยเมตรต่อวินาที(M/S)

ส่วน B เป็นส่วนที่แสดงค่าทิศทางลมในหน่วยองศา(DEG)

จากการแสดงผลข้อมูลค่าความเร็วลมและทิศทางลม พบว่าค่าที่แสดงบนจอ LCD ไม่มีความผิดพลาดเกิดขึ้นค่าที่ได้มีความผิดพลาดไม่เกิน 10 % และแสดงผลทุกๆ 4 วินาทีในการแสดงผล

4.3.2 การแสดงผลข้อมูลในส่วนการบันทึกข้อมูล

การบันทึกข้อมูลในส่วนนี้จะใช้บันทึกข้อมูลในไฟล์ .txt และแสดงผลการบันทึกโดยโปรแกรม notepad โดยการเขียนข้อมูลบนบอร์ด ET-USB FLASH DRIVE โดยเชื่อมต่อกับไมโครคอนโทรลเลอร์ด้วยพอร์ตอนุกรม (RS-232) ซึ่งได้ผลการทดลองดังนี้

```

WIND2 - Notepad
File Edit Format View Help
1.03 289.00
0.98 307.00
1.03 286.00
1.11 289.00
0.90 16.00C
0.70 16.00C
0.65 256.00
0.70 321.00
0.80 286.00
0.70 270.00
0.65 259.00
1.20 275.00
1.41 255.00
1.03 217.00
0.75 232.00
0.55 237.00
0.45 253.00
0.65 265.00
0.80 335.00
0.55 214.00
0.45 245.00
0.25 325.00
0.75 314.00
1.07 301.00
1.11 348.00
1.15 323.00
1.49 310.00
1.11 292.00
0.80 316.00
0.65 334.00
0.75 348.00
0.80 293.00
0.50 218.00
0.45 272.00
  
```

รูปที่ 4.14 ข้อมูลที่บันทึกผ่านโปรแกรม Notepad

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

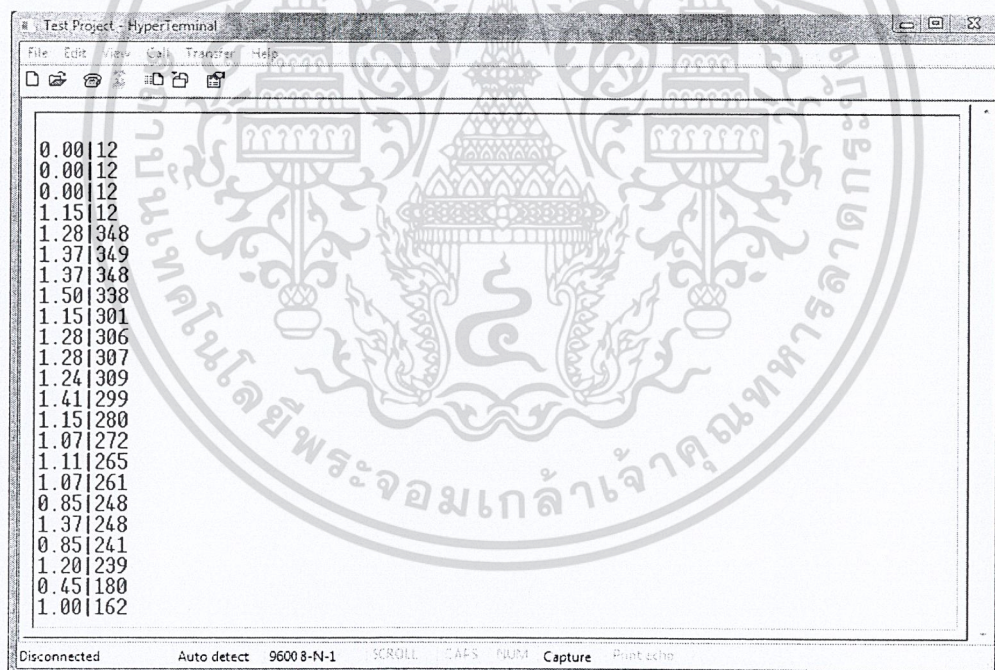
จากการจัดเก็บข้อมูลค่าความเร็วลมและทิศทางลมซึ่งบันทึกค่าความเร็วลมและทิศทางลมทุกๆ 4 วินาที มีการจัดเก็บค่าได้อย่างต่อเนื่องแต่พบว่ายังมีปัญหาที่เกิดจากการเขียนข้อมูลผิดพลาดลงบน USB FLASH DRIVE แต่ข้อมูลมีความถูกต้องและมีความสัมพันธ์กับหน้าจอแสดงผลไม่มีค่าความผิดพลาดเกิดขึ้น

4.4 การส่งข้อมูลเข้าคอมพิวเตอร์ (Protocol)

ในรูปที่ 4.15 แสดงผลการเชื่อมต่อข้อมูลระหว่างเครื่องมือวัดกับคอมพิวเตอร์ โดยแสดงค่าความเร็วลมและทิศทางลมที่ไม่โครคอนโทรลเลอร์ส่งให้กับคอมพิวเตอร์ ซึ่งได้ข้อมูลมาจากเครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม

โดยข้อมูลที่ส่งเข้ามามีจำนวน 9 ไบต์

- ไบต์ที่ 1 ถึง ไบต์ที่ 5 เป็นข้อมูลค่าความเร็วลมเป็นจำนวนเต็ม 2 หลัก คั่นด้วย “.” และตามด้วยทศนิยมอีก 2 ตำแหน่ง
- ไบต์ที่ 6 เป็นเครื่องหมาย “|” สำหรับคั่นระหว่างค่าความเร็วลมและค่าทิศทางลม
- ไบต์ที่ 7 ถึง ไบต์ที่ 9 เป็นข้อมูลค่าทิศทางลมเป็นจำนวนเต็ม 3 หลัก

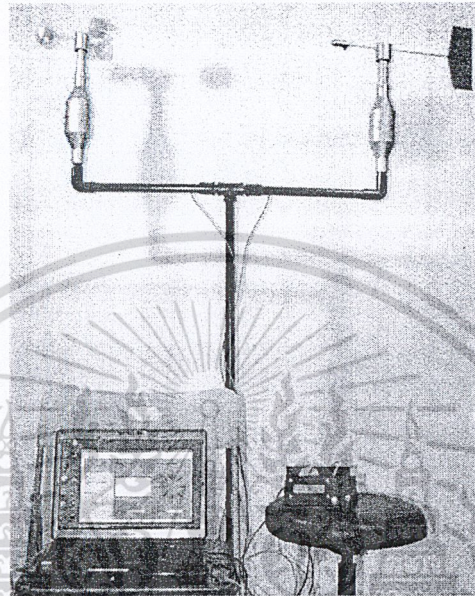


รูปที่ 4.15 ค่าความเร็วลมและค่าทิศทางลมที่ส่งมายังคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

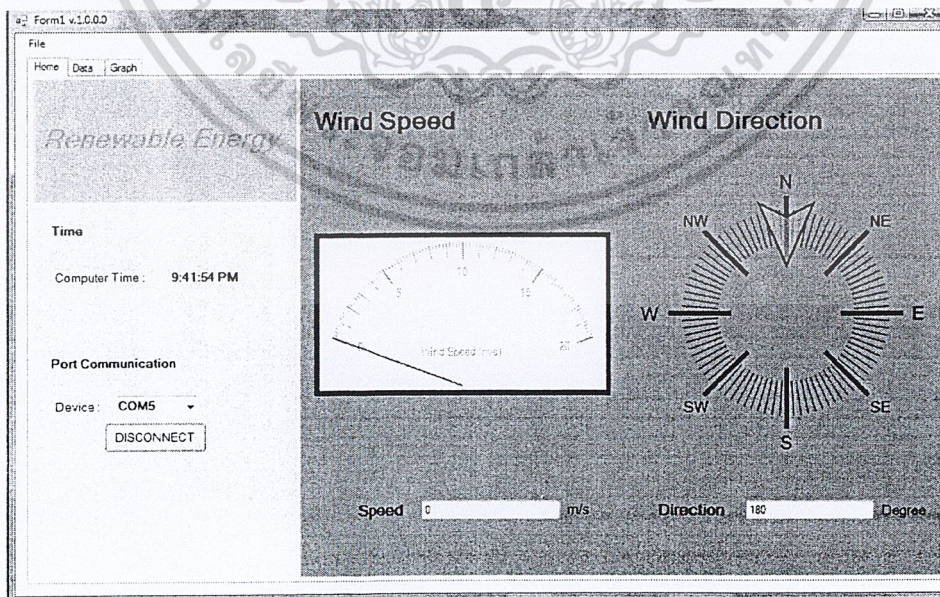
4.5 การประมวลผลของโปรแกรมคอมพิวเตอร์

เชื่อมต่อโปรแกรมแสดงผลผ่านคอมพิวเตอร์กับชุดประมวลผลข้อมูลลม ทำการต่อสายสัญญาณเข้ากับเครื่องมือวัด แล้วทำการให้ลมแก่เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม สังเกตและเปรียบเทียบความเร็วลมและทิศทางลมที่แสดงผลผ่านหน้าจอ LCD กับคอมพิวเตอร์



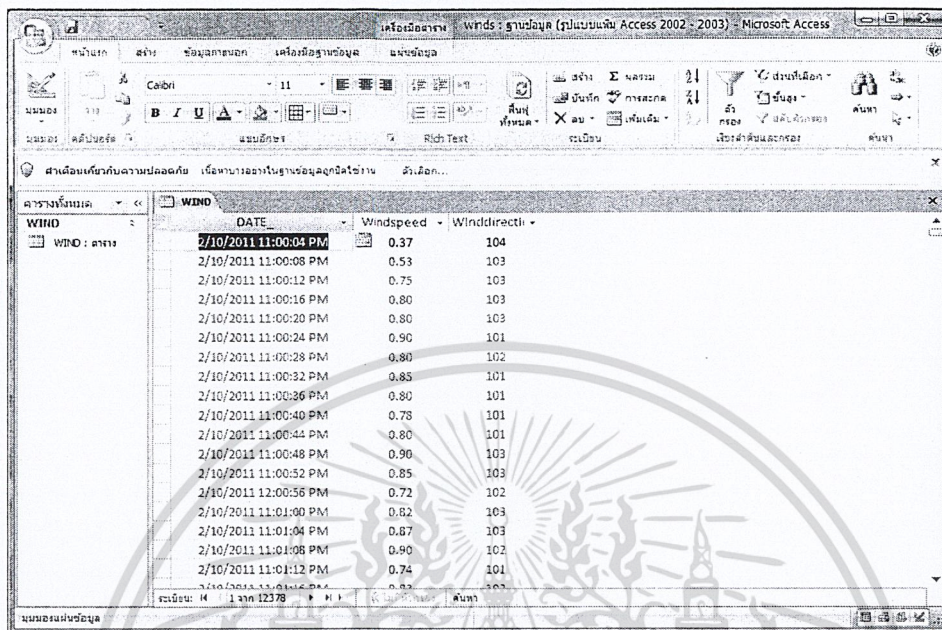
รูปที่ 4.16 การเชื่อมต่อโปรแกรมคอมพิวเตอร์เข้ากับชุดประมวลผลข้อมูลลม

เมื่อทำการเชื่อมต่อข้อมูลแล้ว จะสามารถดูค่าความเร็วลมและทิศทางลมจากเข็มชี้ของหน้าปัด พร้อมทั้งแสดงผลเป็นตัวเลขดิจิทัล



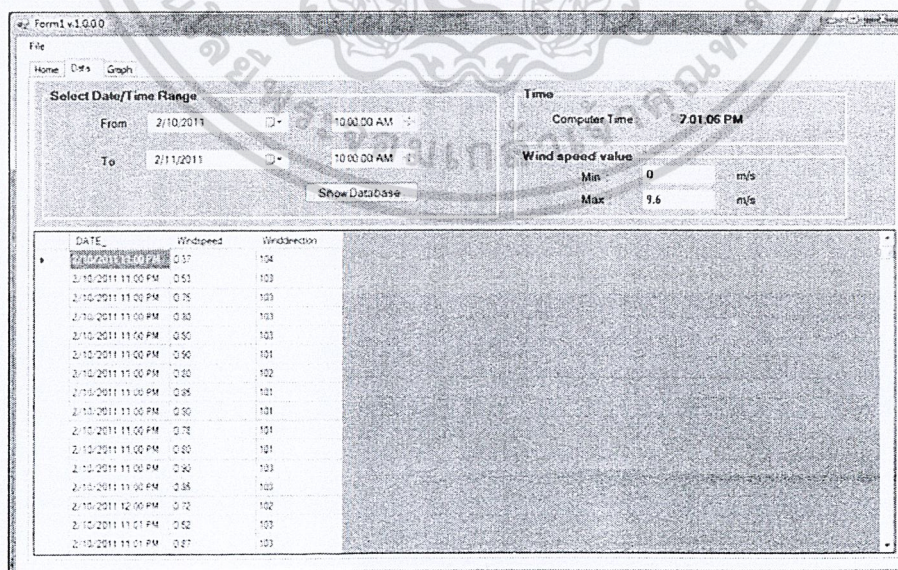
เอกสารนี้เป็นเอกสารรูปที่ 4.17 โปรแกรมเมื่อแสดงผลแบบหน้าปัดและตัวเลขดิจิทัลใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการเชื่อมต่อโปรแกรมกับชุดประมวลข้อมูล โปรแกรมจะแบ่งค่าความเร็วลม และค่าทิศทางลมออกจากกัน และจะทำการบันทึกลงไฟล์ความเร็วมและทิศทางลมของฐานข้อมูล(.mdb) ร่วมกับวันเดือนปีและเวลาของข้อมูลที่เข้ามาโดยอัตโนมัติ



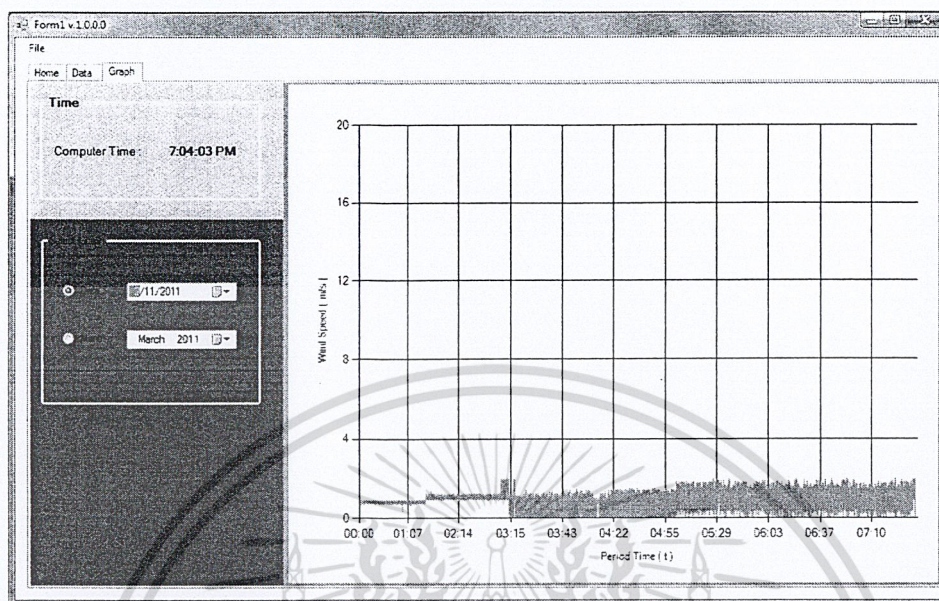
รูปที่ 4.18 ตารางฐานข้อมูล(.mdb) ที่เชื่อมต่อกับโปรแกรม

ทำการเลือกไฟล์ฐานข้อมูล และช่วงวันเวลาที่ต้องการดูข้อมูล โปรแกรมจะแสดงผลค่าความเร็วลมและค่าทิศทางลมร่วมกับวันเดือนปีและเวลาของข้อมูลนั้น พร้อมทั้งแสดงค่าสูงสุดและต่ำสุดของความเร็วลมในช่วงวันเวลาที่เลือกด้วย



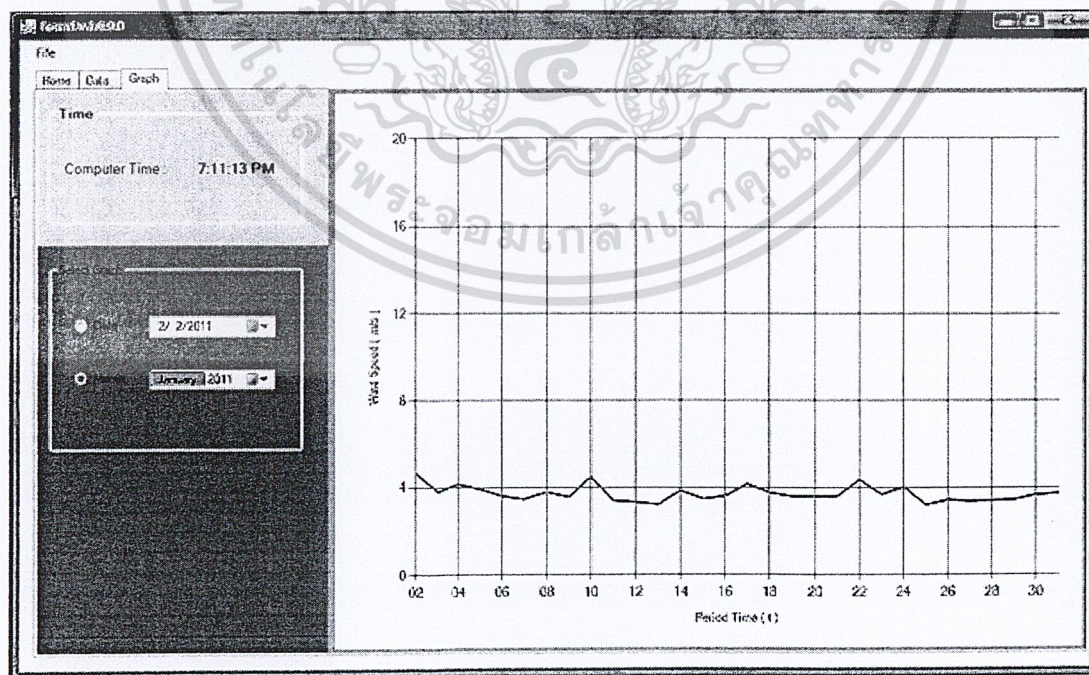
เอกสารนี้เป็นเอกสารรูปที่ 4.19 โปรแกรมแสดงผลแบบฐานข้อมูลตามช่วงวันและเวลาที่เลือก ประโยชน์ด้านการค้าไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการเลือกไฟล์ฐานข้อมูล ชนิดของกราฟแสดงความเร็วลมแบบวัน และวันที่ที่ต้องการ โปรแกรมจะแสดงผลความเร็วลมร่วมกับเวลาดังรูป



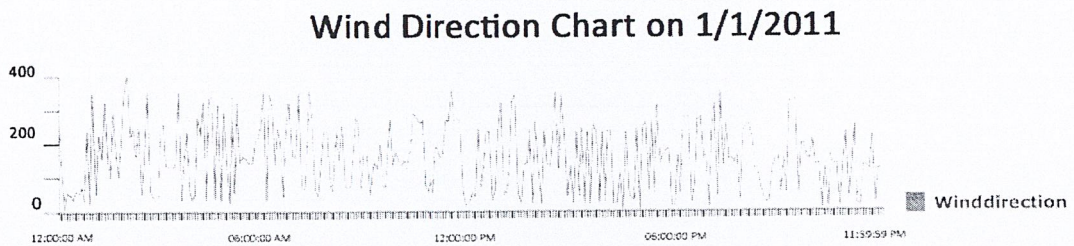
รูปที่ 4.20 โปรแกรมเมื่อแสดงผลความเร็วลมแบบกราฟวัน

ทำการเลือกไฟล์ฐานข้อมูล ชนิดของกราฟแสดงความเร็วลมแบบเดือน และเดือนที่ต้องการ โปรแกรมจะแสดงผลความเร็วลมร่วมกับเวลาดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น รูปที่ 4.21 โปรแกรมเมื่อแสดงผลความเร็วลมแบบกราฟเดือน

โดยโปรแกรมสามารถตัดลอกข้อมูลไปประยุกต์ใช้ร่วมกับโปรแกรมอื่นๆได้ เช่น ใช้กับโปรแกรม SmartDraw เพื่อพลอตกราฟ หรือโปรแกรม Microsoft Excel เพื่อนำความเร็วลมไปคำนวณหาพลังงานลมแล้วนำไปหาพลังงานไฟฟ้าต่อไป



รูปที่ 4.22 กราฟแสดงทิศทางลมโดยโปรแกรม SmartDraw



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์

เครื่องมือวัดความเร็วลมสามารถตรวจจับค่าความเร็วโดยใช้การคำนวณจากพัลส์ที่ได้จากการตรวจจับของเซนเซอร์แสง การจะได้ค่าความเร็วลมที่แท้จริงนั้นจะต้องนำเครื่องมือไปวัดและเปรียบเทียบค่ากับลมจากอุโมงค์ลมที่กรมอุตุนิยมวิทยา ซึ่งจะต้องมีความเป็นเชิงเส้นในทุกย่านความเร็ว เมื่อพบว่าเครื่องมือวัดไม่เป็นเชิงเส้นแล้ว จะต้องคำนวณปรับเพื่อให้ความเร็วลมในทุกย่านการทำงานของเครื่องมือวัดมีความเป็นเชิงเส้นและถูกต้อง

เครื่องมือวัดทิศทางความเร็วลมสามารถตรวจจับทิศทางได้โดยโพเทนชิโอมิเตอร์ซึ่งสามารถตรวจจับระยะทางของการเคลื่อนที่เชิงมุม ซึ่งเครื่องมือวัดทิศทางจะต้องวัดได้ทุกทิศทางหรือ 360 องศา ดังนั้นจึงควรใช้โพเทนชิโอมิเตอร์ที่หมุนได้รอบ 360 องศา

สำหรับการคำนวณค่าความเร็วลมและทิศทางลมนั้นจะใช้ไมโครคอนโทรลเลอร์ในการประมวลผล โดยใช้โปรแกรม PICC Compiler ภาษา CCS ในการเขียนโปรแกรม เพราะมีความนิยม และมีฟังก์ชันหลากหลายทำให้ง่ายและสะดวกต่อการเขียน เนื่องจากการปรับแต่งเซนเซอร์ให้ถูกต้องนั้นทำได้ยาก จึงทำการปรับแต่งความถูกต้องและแม่นยำของการวัดด้วยการคำนวณผ่านไมโครคอนโทรลเลอร์ นอกจากนี้ไมโครคอนโทรลเลอร์ยังสามารถส่งค่าไปบันทึกยังหน่วยความจำ เป็นการช่วยสำรองข้อมูลได้อีกทางหนึ่ง

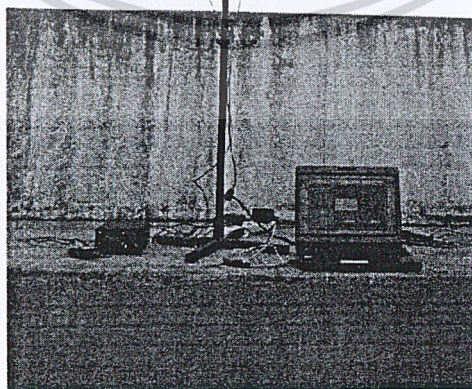
ในส่วนของแสดงผลผ่านโปรแกรมคอมพิวเตอร์นั้น สร้างโปรแกรมโดยใช้โปรแกรม Visual Studio 2008 ด้วยภาษา C# โปรแกรมแสดงผลสามารถจัดเก็บข้อมูลความเร็วลมและทิศทางลมที่รับมาจากเครื่องมือวัดได้โดยอัตโนมัติ และแสดงผลได้ 3 รูปแบบ คือแสดงผลแบบหน้าปัดและตัวเลขดิจิทัล แสดงผลแบบตารางฐานข้อมูล แสดงผลแบบกราฟความเร็วลม

ในปัจจุบันพลังงานทดแทนมีบทบาทมากขึ้นเรื่อยๆ การสร้างเครื่องมือวัดได้เองจึงช่วยสนับสนุนการทำวิจัยด้านพลังงานลมให้มีความแพร่หลายและมีการเก็บข้อมูลอย่างต่อเนื่องมากยิ่งขึ้น โดยผลที่ได้จากเครื่องมือวัดความเร็วและเครื่องมือวัดทิศทางลมจะทำให้เห็นศักยภาพพลังงาน ณ จุดที่ตั้งเครื่อง เพื่อนำมาคำนวณหาความคุ้มค่าในการติดตั้งเครื่องกำเนิดไฟฟ้าพลังงานลมต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสนอแนะ

1. เครื่องวัดมีข้อจำกัดในเรื่องของอายุการใช้งานเนื่องจากวัสดุที่ใช้ทำมาจากพลาสติกสำเร็จ หากต้องการความแข็งแรงควรใช้วัสดุที่เป็นโลหะแต่จะมีปัญหาทางด้านการขึ้นรูปชิ้นส่วน
2. Opto sensor อาจมีข้อจำกัดในการอ่านค่า Pulse ณ จุดที่มีความเร็วลมสูงมากๆ
3. การประมวลผลด้วย PIC 16F877A เพียงตัวเดียวจะทำให้เกิดข้อจำกัดในเรื่องของ Timeclock ทำให้เครื่องวัดทิศทางลมมีการตอบสนองช้า เพราะต้องรอการประมวลผลที่เวลาเท่ากับเครื่องวัดความเร็วลม
4. หากต้องการความละเอียดของค่าความเร็วลมควรใช้แผ่นตัดแสงที่ให้จำนวน Pulse ต่อรอบสูงกว่า
5. หากต้องการทำชุดเครื่องมือวัดหลายๆชุดโดยใช้ไมโครคอนโทรลเลอร์ในการประมวลผลเพียงตัวเดียวหรือใช้ชุดแสดงผลเพียงชุดเดียวควรใช้ PIC ที่มีจำนวนบิตดีที่สูงขึ้น เช่น 18F
6. โปรแกรม Visual Studio นั้น ควรใช้โปรแกรมรุ่นล่าสุด เนื่องจากมี Control ที่มากกว่ารุ่นเก่าๆ ทั้งให้ใช้งานได้มากกว่า ง่ายกว่าและไม่ต้องมีการเขียน Source code ให้ยุ่งยาก เช่น การแสดงผลแบบกราฟที่ Visual Studio 2010 มี Chart Control มาให้ใช้ได้เลย ทำให้สามารถแสดงผลกราฟได้ทุกรูปแบบโดยที่ไม่ต้องคำนวณและเขียนโปรแกรมเพิ่มเติม



เอกสารนี้เป็นเอกสารที่สงวน **รูปที่ 5.1** เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม มาไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

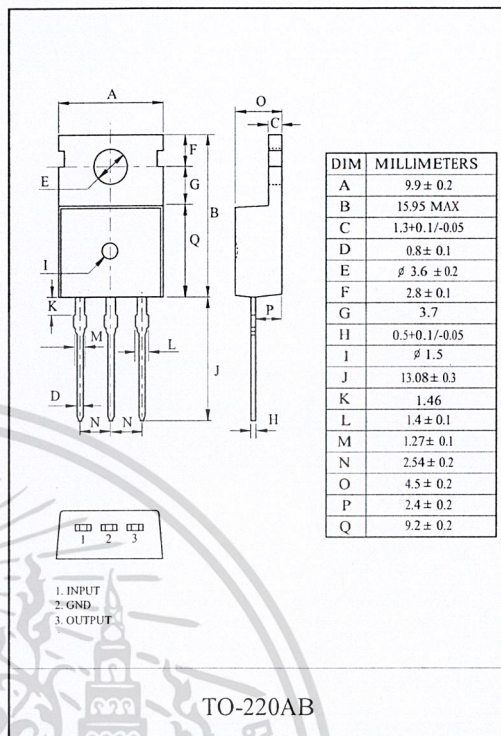
THREE TERMINAL POSITIVE VOLTAGE REGULATORS 5V, 6V, 7V, 8V, 9V, 10V, 12V, 15V, 18V, 20V, 24V.

FEATURES

- Internal Thermal Overload Protection.
- Internal Short Circuit Current Limiting.
- Output Current up to 1.5A.
- Satisfies IEC-65 Specification. (International Electronical Commission).
- Package is TO-220AB

LINE-UP

ITEM	OUTPUT VOLTAGE (Typ.)	UNIT
KIA7805AP	5	V
KIA7806AP	6	
KIA7807AP	7	
KIA7808AP	8	
KIA7809AP	9	
KIA7810AP	10	
KIA7812AP	12	
KIA7815AP	15	
KIA7818AP	18	
KIA7820AP	20	
KIA7824AP	24	



MAXIMUM RATINGS (Ta=25°C)

CHARACTERISTIC	SYMBOL	RATING	UNIT
Input Voltage	V _{IN}	35	V
		40	
Power Dissipation-1 (No Heatsink)	AP	P _{D2}	W
Power Dissipation-2 (Infinite Heatsink)	AP	P _{D2}	
Operating Junction Temperature	T _j	-40 ~ 150	°C
Storage Temperature	T _{stg}	-55 ~ 150	°C
Maximum Junction Temperature	T _{j(max)}	150	°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

KIA7805AP~KIA7824AP

KIA7805AP

ELECTRICAL CHARACTERISTICS ($V_{IN}=10V$, $I_{OUT}=500mA$, $0^{\circ}C \leq T_j \leq 125^{\circ}C$)

CHARACTERISTIC	SYMBOL	TEST CIRCUIT	TEST CONDITION	MIN.	TYP.	MAX.	UNIT	
Output Voltage	V_{OUT}	Fig. 1	$T_j=25^{\circ}C$, $I_{OUT}=100mA$	4.8	5.0	5.2	V	
Input Regulation	Reg line	Fig. 1	$T_j=25^{\circ}C$	$7.0V \leq V_{IN} \leq 25V$	-	3	100	mV
				$8.0V \leq V_{IN} \leq 12V$	-	1	50	
Load Regulation	Reg load	Fig. 1	$T_j=25^{\circ}C$	$5mA \leq I_{OUT} \leq 1.5A$	-	15	100	mV
				$250mA \leq I_{OUT} \leq 750mA$	-	5	50	
Output Voltage	V_{OUT}	Fig. 1	$7.0V \leq V_{IN} \leq 20V$	4.75	-	5.25	V	
Quiescent Current	I_B	Fig. 1	$T_j=25^{\circ}C$, $I_{OUT}=5mA$	-	4.2	8.0	mA	
Quiescent Current Change	ΔI_B	Fig. 1	$7.0V \leq V_{IN} \leq 25V$	-	-	1.3	mA	
Output Noise Voltage	V_{NO}	Fig. 2	$T_a=25^{\circ}C$, $10Hz \leq f \leq 100kHz$	-	50	-	μV_{rms}	
Ripple Rejection Ratio	RR	Fig. 3	$f=120Hz$, $8.0V \leq V_{IN} \leq 18V$,	62	78	-	dB	
Dropout Voltage	V_D	Fig. 1	$I_{OUT}=1.0A$, $T_j=25^{\circ}C$	-	2.0	-	V	
Short Circuit Current Limit	I_{SC}	Fig. 1	$T_j=25^{\circ}C$	-	1.6	-	A	
Average Temperature Coefficient of Output Voltage	TC_{VO}	Fig. 1	$I_{OUT}=5mA$, $0^{\circ}C \leq T_j \leq 125^{\circ}C$	-	-0.6	-	mV/ $^{\circ}C$	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



深圳秋田视佳实业有限公司
 地址：深圳市华侨城东部工业区文昌街东北 C-7 栋
 电话:(086)0755-26919178 传真: (086)0755 -26911092
 网址: [Http://www.av-display.com.cn](http://www.av-display.com.cn)

SHENZHEN AV-DISPLAY CO.,LTD
 Address: North East C-7 Building, Wenchang Street
 Eastern District, OCT, ShenZhen, China
 TEL: (086)0755-26919178 FAX: (086)0755-26911092
[Http://www.av-display.com.cn](http://www.av-display.com.cn)

SPECIFICATION FOR LCM MODULE

MODULE NO.: ABC016002A07-GHY
DOC.REVISION: 00

Customer Approval:

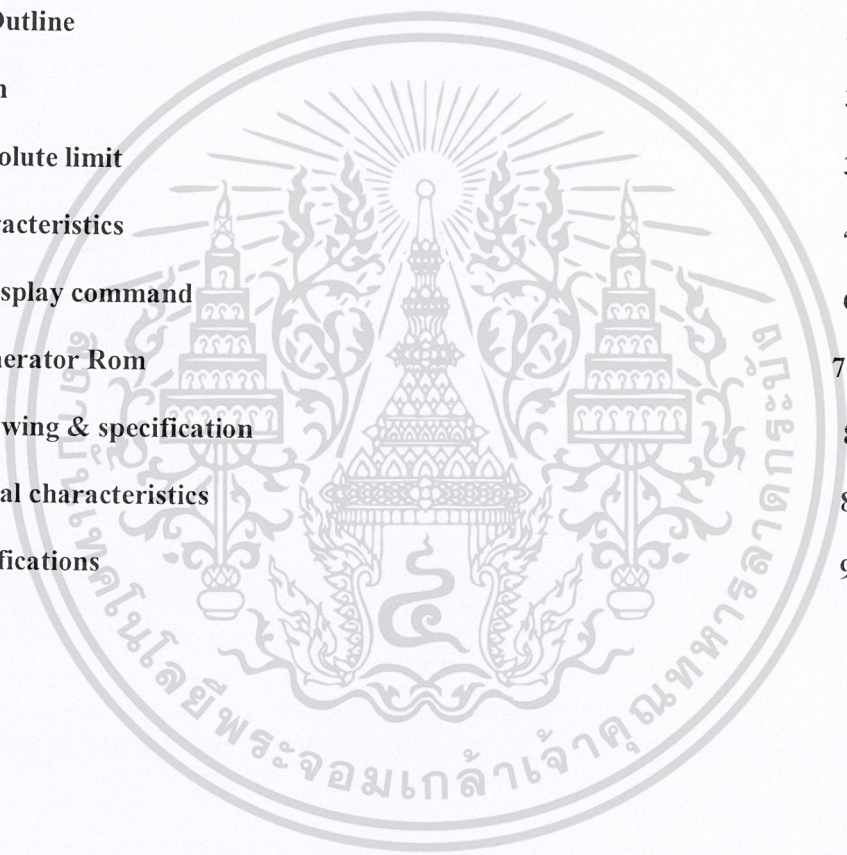
--

	SIGNATURE	DATE
PREPARED BY (RD ENGINEER)		Apr-11-2006
PREPARED BY (QA ENGINEER)		
CHECKED BY		
APPROVED BY		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CONTENTS

1. Functions & Features	1
2. Mechanical specifications	1
3. Block diagram	1
4. Dimensional Outline	2
5. Pin description	3
6. Maximum absolute limit	3
7. Electrical characteristics	4~5
8. Control and display command	6
9. Character Generator Rom	7
10. Backlight drawing & specification	8
11. Electro-Optical characteristics	8
12. Quality Specifications	9~17



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. FUNCTIONS & FEATURES

1.1. Format	: 16x2 characters
1.2. LCD mode	: STN / Positive Transflective Mode / Grey
1.3. Viewing direction	: 6 o'clock
1.4. Driving scheme	: 1/16 Duty , 1/5 Bias
1.5. Power supply voltage (V _{DD})	: 5.0V
1.6. LCD driving voltage(V _{op})	: 4.2V
1.7. Operation temp	: -20~70°C
1.8. Storage temp	: -30~80°C
1.9. Backlight color	: Y-G

2. MECHANICAL SPECIFICATIONS

2.1. Module size	: 80.0mm(L)*36.0mm(W)*14.5(H) max mm
2.2. Viewing area	: 62.2mm(L)*17.9mm(W)
2.3. Character pitch	: 3.55mm(L)*5.95mm(W)
2.4. Character size	: 2.95mm(L)*5.55mm(W)
2.5. Dot pitch	: 0.55mm(L)*0.65mm(W)
2.6. Dot size	: 0.60mm(L)*0.70mm(W)
2.7. Weight	: Approx.

3. BLOCK DIAGRAM

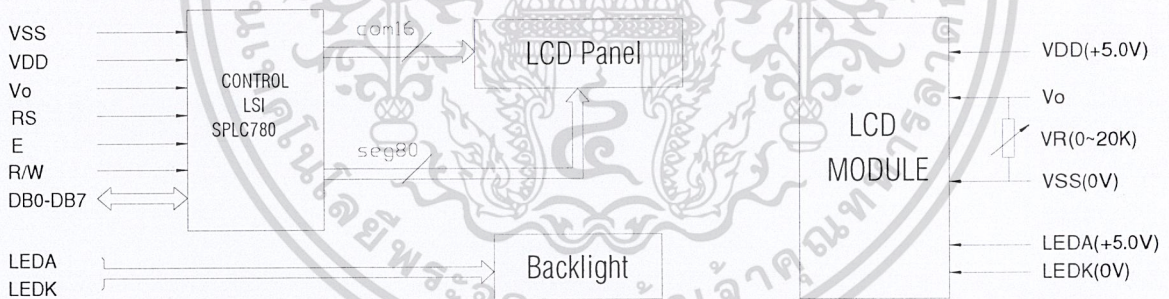


Figure1. Block diagram

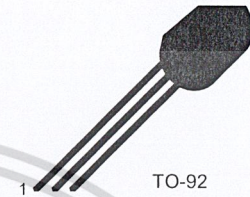
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 1 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



BC546/547/548/549/550

Switching and Applications

- High Voltage: BC546, $V_{CE0}=65V$
- Low Noise: BC549, BC550
- Complement to BC556 ... BC560



TO-92
1. Collector 2. Base 3. Emitter

NPN Epitaxial Silicon Transistor

Absolute Maximum Ratings $T_a=25^{\circ}C$ unless otherwise noted

Symbol	Parameter	Value	Units
V_{CB0}	Collector-Base Voltage : BC546	80	V
	: BC547/550	50	V
	: BC548/549	30	V
V_{CE0}	Collector-Emitter Voltage : BC546	65	V
	: BC547/550	45	V
	: BC548/549	30	V
V_{EB0}	Emitter-Base Voltage : BC546/547	6	V
	: BC548/549/550	5	V
I_C	Collector Current (DC)	100	mA
P_C	Collector Power Dissipation	500	mW
T_J	Junction Temperature	150	$^{\circ}C$
T_{STG}	Storage Temperature	-65 ~ 150	$^{\circ}C$

Electrical Characteristics $T_a=25^{\circ}C$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units	
I_{CBO}	Collector Cut-off Current	$V_{CB}=30V, I_E=0$			15	nA	
h_{FE}	DC Current Gain	$V_{CE}=5V, I_C=2mA$	110		800		
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage	$I_C=10mA, I_B=0.5mA$		90	250	mV	
		$I_C=100mA, I_B=5mA$		200	600	mV	
$V_{BE(sat)}$	Base-Emitter Saturation Voltage	$I_C=10mA, I_B=0.5mA$		700		mV	
		$I_C=100mA, I_B=5mA$		900		mV	
$V_{BE(on)}$	Base-Emitter On Voltage	$V_{CE}=5V, I_C=2mA$	580	660	700	mV	
		$V_{CE}=5V, I_C=10mA$			720	mV	
f_T	Current Gain Bandwidth Product	$V_{CE}=5V, I_C=10mA, f=100MHz$		300		MHz	
C_{ob}	Output Capacitance	$V_{CB}=10V, I_E=0, f=1MHz$		3.5	6	pF	
C_{ib}	Input Capacitance	$V_{EB}=0.5V, I_C=0, f=1MHz$		9		pF	
NF	Noise Figure : BC546/547/548	$V_{CE}=5V, I_C=200\mu A$ $f=1KHz, R_G=2K\Omega$		2	10	dB	
			: BC549/550		1.2	4	dB
			: BC549		1.4	4	dB
			: BC550		1.4	3	dB

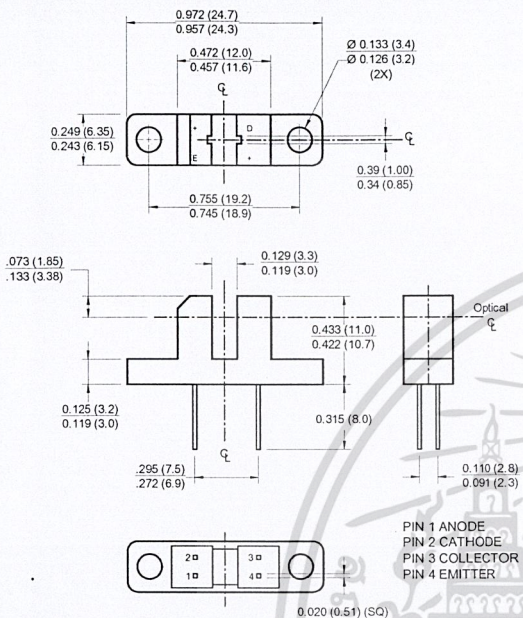
h_{FE} Classification

Classification	A	B	C
h_{FE}	110 ~ 220	200 ~ 450	420 ~ 800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

H21A1 / H21A2 / H21A3 PHOTOTRANSISTOR OPTICAL INTERRUPTER SWITCH

PACKAGE DIMENSIONS

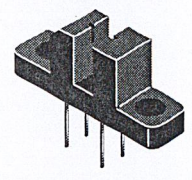


NOTES:

- Dimensions for all drawings are in inches (mm).
- Tolerance of $\pm .010$ (.25) on all non-nominal dimensions unless otherwise specified.

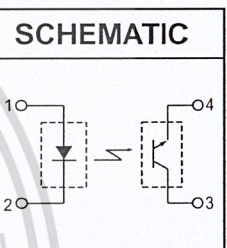
DESCRIPTION

The H21A1, H21A2 and H21A3 consist of a gallium arsenide infrared emitting diode coupled with a silicon phototransistor in a plastic housing. The packaging system is designed to optimize the mechanical resolution, coupling efficiency, ambient light rejection, cost and reliability. The gap in the housing provides a means of interrupting the signal with an opaque material, switching the output from an "ON" to an "OFF" state.



FEATURES

- Opaque housing
- Low cost
- .035" apertures
- High $I_{C(ON)}$



- Derate power dissipation linearly 1.33 mW/°C above 25°C.
- RMA flux is recommended.
- Methanol or isopropyl alcohols are recommended as cleaning agents.
- Soldering iron tip 1/16" (1.6mm) minimum from housing.

ABSOLUTE MAXIMUM RATINGS (T_A = 25°C unless otherwise specified)

Parameter	Symbol	Rating	Unit
Operating Temperature	T _{OPR}	-55 to +100	°C
Storage Temperature	T _{STG}	-55 to +100	°C
Soldering Temperature (Iron) ^(2,3 and 4)	T _{SOL-I}	240 for 5 sec	°C
Soldering Temperature (Flow) ^(2 and 3)	T _{SOL-F}	260 for 10 sec	°C
INPUT (EMITTER)			
Continuous Forward Current	I _F	50	mA
Reverse Voltage	V _R	6	V
Power Dissipation ⁽¹⁾	P _D	100	mW
OUTPUT (SENSOR)			
Collector to Emitter Voltage	V _{CEO}	30	V
Emitter to Collector Voltage	V _{ECO}	4.5	V
Collector Current	I _C	20	mA
Power Dissipation (T _C = 25°C) ⁽¹⁾	P _D	150	mW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

H21A1 / H21A2 / H21A3

PHOTOTRANSISTOR

OPTICAL INTERRUPTER SWITCH

ELECTRICAL / OPTICAL CHARACTERISTICS (T_A = 25°C)(All measurements made under pulse condition)

PARAMETER	TEST CONDITIONS	SYMBOL	DEVICES	MIN	TYP	MAX	UNITS
INPUT (EMITTER)							
Forward Voltage	I _F = 60 mA	V _F	All	—	—	1.7	V
Reverse Breakdown Voltage	I _R = 10 μA	V _R	All	6.0	—	—	V
Reverse Leakage Current	V _R = 3 V	I _R	All	—	—	1.0	μA
OUTPUT (SENSOR)							
Emitter to Collector Breakdown	I _F = 100 μA, E _e = 0	BV _{ECO}	All	6.0	—	—	V
Collector to Emitter Breakdown	I _C = 1 mA, E _e = 0	BV _{CEO}	All	30	—	—	V
Collector to Emitter Leakage	V _{CE} = 25 V, E _e = 0	I _{CEO}	All	—	—	100	nA
COUPLED							
On-State Collector Current	I _F = 5 mA, V _{CE} = 5 V	I _{C(ON)}	H21A1	0.15	—	—	mA
			H21A2	0.30	—	—	
			H21A3	0.60	—	—	
	I _F = 20 mA, V _{CE} = 5 V		H21A1	1.0	—	—	
			H21A2	2.0	—	—	
			H21A3	4.0	—	—	
	I _F = 30 mA, V _{CE} = 5 V		H21A1	1.9	—	—	
			H21A2	3.0	—	—	
			H21A3	5.5	—	—	
Saturation Voltage	I _F = 20 mA, I _C = 1.8 mA	V _{CE(SAT)}	H21A2/3	—	—	0.40	V
	I _F = 30 mA, I _C = 1.8 mA		H21A1	—	—	0.40	V
Turn-On Time	I _F = 30 mA, V _{CC} = 5 V, R _L = 2.5 KΩ	t _{on}	All	—	8	—	μs
Turn-Off Time	I _F = 30 mA, V _{CC} = 5 V, R _L = 2.5 KΩ	t _{off}	All	—	50	—	μs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

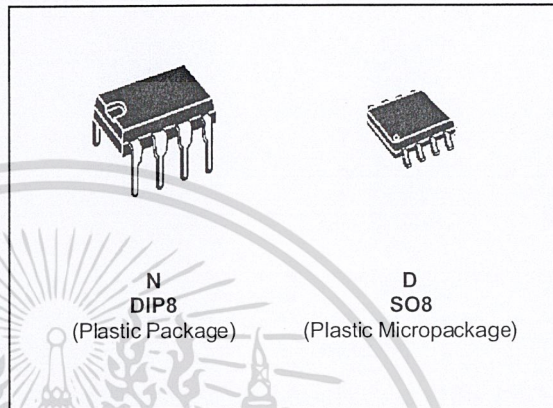
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



LF155-LF255-LF355
LF156-LF256-LF356
LF157-LF257-LF357

WIDE BANDWIDTH SINGLE J-FET OPERATIONAL AMPLIFIERS

- HIGH INPUT IMPEDANCE J-FET INPUT STAGE
- HIGH SPEED J-FET OP-AMPS : up to 20MHz, 50V/μs
- OFFSET VOLTAGE ADJUSTMENT DOES NOT DEGRADE DRIFT OR COMMON-MODE REJECTION AS IN MOST OF MONOLITHIC AMPLIFIERS
- INTERNAL COMPENSATION AND LARGE DIFFERENTIAL INPUT VOLTAGE CAPABILITY (UP TO V_{CC}^+)



TYPICAL APPLICATIONS

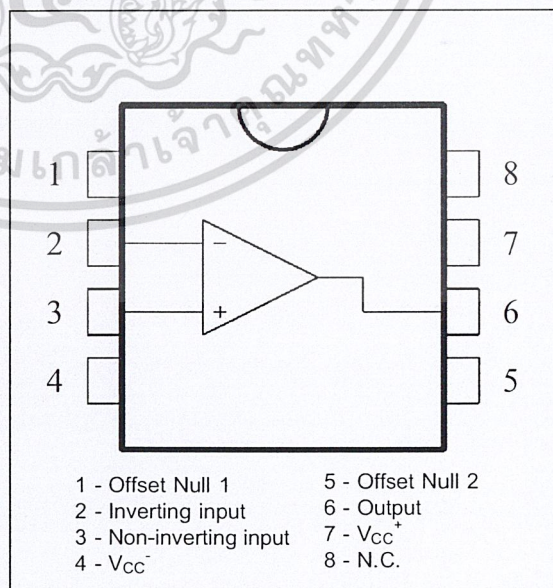
- PRECISION HIGH SPEED INTEGRATORS
- FAST D/A AND CONVERTERS
- HIGH IMPEDANCE BUFFERS
- WIDEBAND, LOW NOISE, LOW DRIFT AMPLIFIERS
- LOGARITHMIC AMPLIFIERS
- PHOTOCCELL AMPLIFIERS
- SAMPLE AND HOLD CIRCUITS

ORDER CODES

Part Number	Temperature Range	Package	
		N	D
LF355, LF356, LF357	0°C, +70°C	•	•
LF255, LF256, LF257	-40°C, +105°C	•	•
LF155, LF156, LF157	-55°C, +125°C	•	•

Example : LF355N

PIN CONNECTIONS (top view)



DESCRIPTION

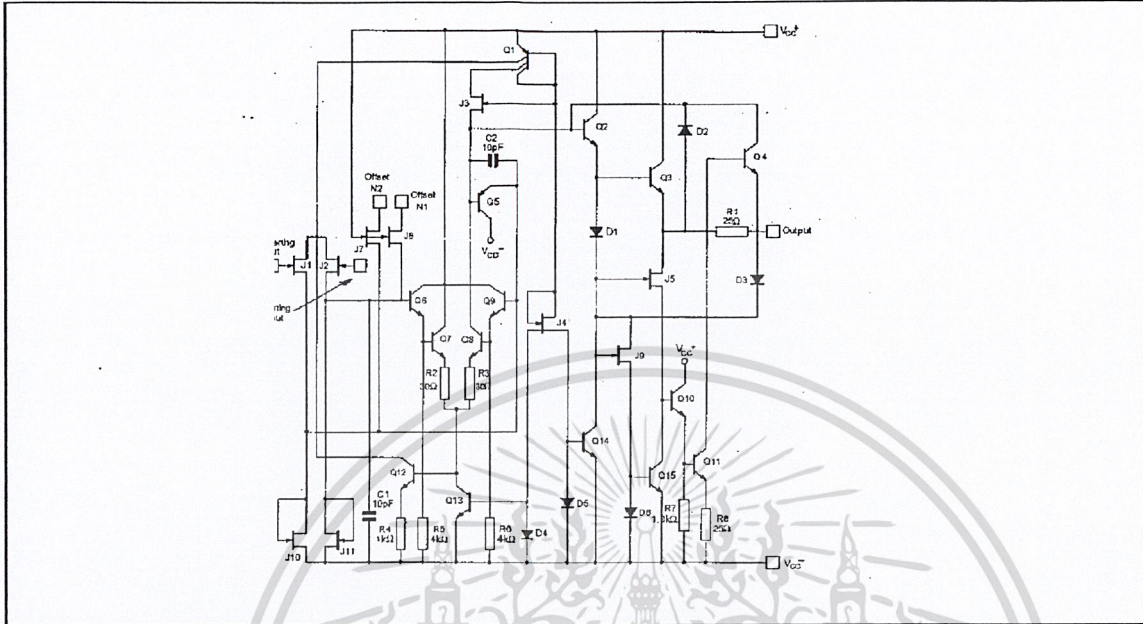
These circuits are monolithic J-FET input operational amplifiers incorporating well matched, high voltage J-FET on the same chip with standard bipolar transistors.

This amplifiers feature low input bias and offset currents, low input offset voltage and input offset voltage drift, coupled with offset adjust which does not degrade drift or common-mode rejection.

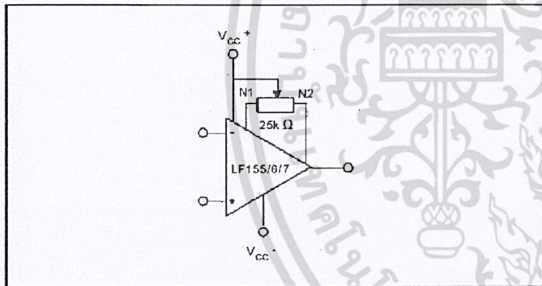
The devices are also designed for high slew rate, wide bandwidth, extremely fast settling time, low voltage and current noise and a low 1/f noise level.

LF155 - LF156 - LF157

SCHEMATIC DIAGRAM



V_{io} ADJUSTMENT



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit	
V _{cc}	Supply Voltage	±22	V	
V _i	Input Voltage - (note 1)	±20	V	
V _{id}	Differential Input Voltage	±40	V	
P _{tot}	Power Dissipation	570	mW	
	Output Short-circuit Duration	Infinite		
T _{oper}	Operating Free Air Temperature Range	LF155-LF156-LF157 LF255-LF256-LF257 LF355-LF356-LF357	-55 to +125 -40 to +105 0 to 70	°C
T _{stg}	Storage Temperature Range		-65 to 150	°C

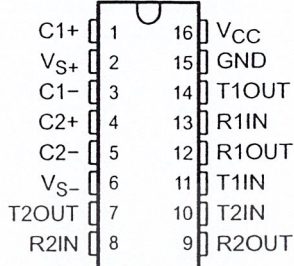


MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

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- μ F 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- μ F 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

TA	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
 ให้บริการเรียนเพื่อการศึกษาเท่านั้น ไม่นับญาติหน้าไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัด INSTRUMENTS อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L - FEBRUARY 1989 - REVISED MARCH 2004

Function Tables

EACH DRIVER

INPUT T _{IN}	OUTPUT T _{OUT}
L	H
H	L

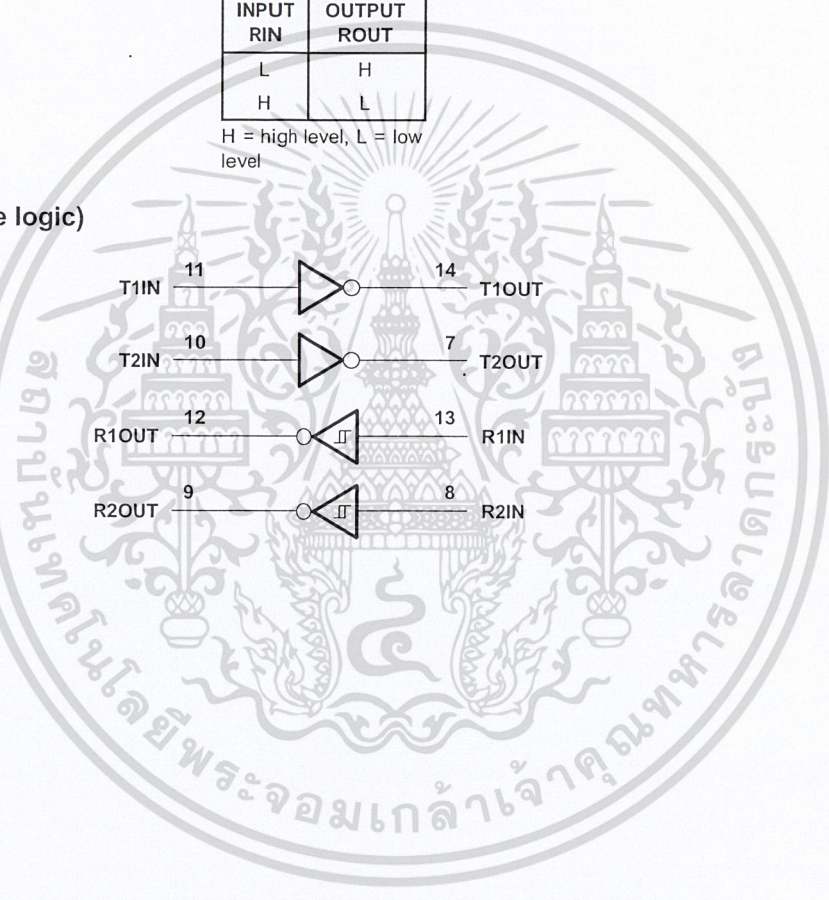
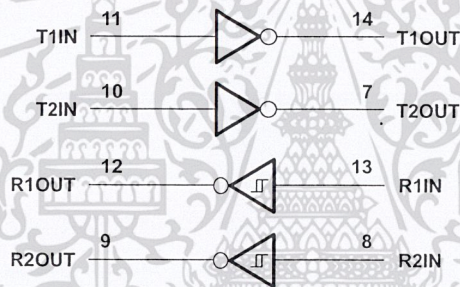
H = high level, L = low level

EACH RECEIVER

INPUT R _{IN}	OUTPUT R _{OUT}
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)



MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L – FEBRUARY 1989 – REVISED MARCH 2004

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1)	–0.3 V to 6 V
Positive output supply voltage range, V_{S+}	$V_{CC} - 0.3$ V to 15 V
Negative output supply voltage range, V_{S-}	–0.3 V to –15 V
Input voltage range, V_i : Driver	–0.3 V to $V_{CC} + 0.3$ V
Receiver	± 30 V
Output voltage range, V_o : T1OUT, T2OUT	$V_{S-} - 0.3$ V to $V_{S+} + 0.3$ V
R1OUT, R2OUT	–0.3 V to $V_{CC} + 0.3$ V
Short-circuit duration: T1OUT, T2OUT	Unlimited
Package thermal impedance, θ_{JA} (see Notes 2 and 3): D package	73°C/W
DW package	57°C/W
N package	67°C/W
NS package	64°C/W
Operating virtual junction temperature, T_J	150°C
Storage temperature range, T_{stg}	–65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltages are with respect to network GND.

2. Maximum power dissipation is a function of $T_J(\text{max})$, θ_{JA} , and T_A . The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(\text{max}) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability.

3. The package thermal impedance is calculated in accordance with JEDEC 51-7.

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN, T2IN)	2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)			0.8	V
R1IN, R2IN	Receiver input voltage			± 30	V
T_A	Operating free-air temperature	MAX232	0	70	°C
		MAX232I	–40	85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 4 and Figure 4)

PARAMETER	TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
I_{CC} Supply current	$V_{CC} = 5.5$ V, All outputs open, $T_A = 25^\circ\text{C}$		8	10	mA

‡ All typical values are at $V_{CC} = 5$ V and $T_A = 25^\circ\text{C}$.

NOTE 4: Test conditions are C1–C4 = 1 μF at $V_{CC} = 5 \pm 0.5$ V.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการบันทึกข้อมูลค่าความเร็วลมและทิศทางลม

```
#include <16F877A.h>           // header file for PIC16F877A

/***** Setting configuration fuses */

#fuses HS,NOWDT,NOPROTECT,NOLVP // Configuration word

#device ADC=8

#use delay(clock=20000000)     // Change OSC here (default 20MHz)

/***** Library function */

#use fast_io(B)

#define use_portb_lcd

#include <lcd.c>                // Driver for common LCD modules (1)

#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7) // standard output

#byte TRIS_C = 0x87           // TRISC File Address

#bit RC2 = TRIS_C.2          // TRISC.2 bit address

int i;

float A;

unsigned int16 ADC1,V;

unsigned int16 CCP_Count;

unsigned int tick = 0;

BOOLEAN sec = 0;

//-----:CCP1 Interrupt

#INT_CCP1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void CCP1_isr() {

    CCP_Count++;

}

#INT_TIMER1

void TIMER1_isr()

{

    if (tick++ >=20) {

        sec = 1;

        tick = 0;

    }

}

/***** Main Functions */

void main()

{

    SET_TRIS_B(0x00); // set port B is output

    setup_adc_ports(ALL_ANALOG);

    setup_adc(ADC_CLOCK_INTERNAL);

    // unsigned int i=255;

    RC2 = 1; // Set CCP1/RC2 input

    CCP_Count=0;

    i=0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

enable_interrupts(INT_CCP1); // Enable interrupt CCP1

enable_interrupts(INT_TIMER1); // Enable interrupt TIMER1

enable_interrupts(GLOBAL); // All interrupts ON

setup_ccp1(CCP_CAPTURE_RE); // Configure CCP1 to capture Every rising edge

setup_timer_1(T1_INTERNAL|T1_DIV_BY_4); // Setup timer1

set_timer1(0); // Start timer

lcd_init(); // LCD init (2)

lcd_putc("\fAnemo & Windvane\n"); // use function lcd_putc() (3)

delay_ms(1000);

/** use function printf() + lcd_putc() */
printf(lcd_putc, "\fSpeed");
printf(lcd_putc, "\nDirection");

while (1) {

    if(CCP_Count<=15&&CCP_Count>=0)

    {

        lcd_gotoxy(9,1);

        A = (CCP_Count)/14;

        printf(lcd_putc, "%.2f M/S",A);

        delay_ms(1000);

        CCP_Count = 0;

        sec = 0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

}

if(CCP_Count<=26&&CCP_Count>15)
{
lcd_gotoxy(9,1);
A = (CCP_Count)/15;
printf(lcd_putc,"%f M/S",A);
delay_ms(1000);
CCP_Count = 0;
sec = 0;

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

if(CCP_Count<=39&&CCP_Count>26)

{

    lcd_gotoxy(9,1);

    A = (CCP_Count)/20;

    printf(lcd_putc,"%0.2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=89&&CCP_Count>39)

{

    lcd_gotoxy(8,1);

    A = (CCP_Count)/21;

    printf(lcd_putc,"%0.2f M/S",A);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_ms(1000);

CCP_Count = 0;

sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);
}
if(CCP_Count<=140&&CCP_Count>89)
{
    lcd_gotoxy(9,1);

    A = (CCP_Count)/23;

    printf(lcd_putc,"%0.2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

        set_adc_channel(0);

        ADC1 = Read_adc();

        V =ADC1*1.411;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=219&&CCP_Count>140)

{

    lcd_gotoxy(9,1);

    A = (CCP_Count)/24;

    printf(lcd_putc,"%0.2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=249&&CCP_Count>219)

{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcd_gotoxy(9,1);

A = (CCP_Count)/25;

printf(lcd_putc,"%0.2f M/S",A);

delay_ms(1000);

CCP_Count = 0;

sec = 0;

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

}

if(CCP_Count>249)

{

lcd_gotoxy(8,1);

A = (CCP_Count)/25;

printf(lcd_putc,"%0.2f M/S",A);

delay_ms(1000);

CCP_Count = 0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sec = 0;

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

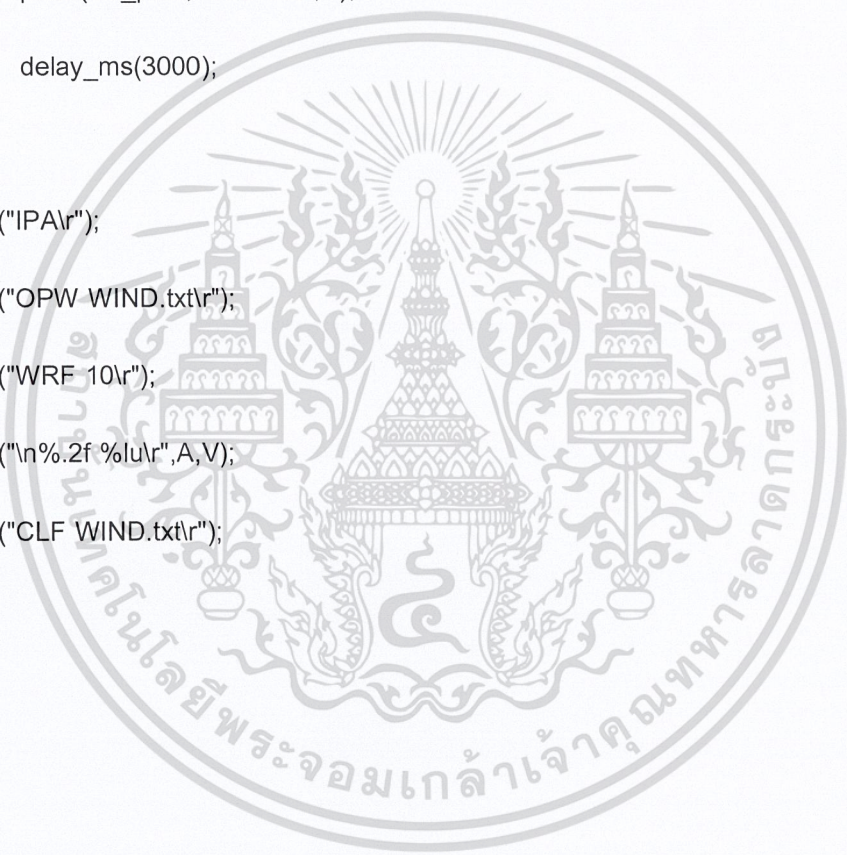
printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

}

printf("IPA\r");
printf("OPW WIND.txt\r");
printf("WRF 10\r");
printf("\n%.2f %lu\r",A,V);
printf("CLF WIND.txt\r");
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการส่งค่าความเร็วลมและทิศทางลมผ่านพอร์ตอนุกรม RS-232

```
#include <16F877A.h>           // header file for PIC16F877A

/***** Setting configuration fuses */

#fuses HS,NOWDT,NOPROTECT,NOLVP // Configuration word

#device ADC=8

#use delay(clock=20000000)     // Change OSC here (default 20MHz)

/***** Library function */

#use fast_io(B)

#define use_portb_lcd

#include <lcd.c>                // Driver for common LCD modules (1)

#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7) // standard output

#byte TRIS_C = 0x87           // TRISC File Address

#bit RC2 = TRIS_C.2          // TRISC.2 bit address

int i;

float A;

unsigned int16 ADC1,V;

unsigned int16 CCP_Count;

unsigned int tick = 0;

BOOLEAN sec = 0;

//-----:CCP1 Interrupt

#INT_CCP1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void CCP1_isr() {

    CCP_Count++;

}

#INT_TIMER1

void TIMER1_isr()

{

    if (tick++ >=20) {

        sec = 1;

        tick = 0;

    }

}

/***** Main Functions */

void main()

{

    SET_TRIS_B(0x00); // set port B is output

    setup_adc_ports(ALL_ANALOG);

    setup_adc(ADC_CLOCK_INTERNAL);

    // unsigned int i=255;

    RC2 = 1; // Set CCP1/RC2 input

    CCP_Count=0;

    i=0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

enable_interrupts(INT_CCP1); // Enable interrupt CCP1

enable_interrupts(INT_TIMER1); // Enable interrupt TIMER1

enable_interrupts(GLOBAL); // All interrupts ON

setup_ccp1(CCP_CAPTURE_RE); // Configure CCP1 to capture Every rising edge

setup_timer_1(T1_INTERNAL|T1_DIV_BY_4); // Setup timer1

set_timer1(0); // Start timer

lcd_init(); // LCD init (2)

lcd_putc("\fAnemo & Windvane\n"); // use function lcd_putc() (3)

delay_ms(1000);

/** use function printf() + lcd_putc() */

printf(lcd_putc,"\fSpeed");

printf(lcd_putc,"\nDirection");

while (1) {

    if(CCP_Count<=15&&CCP_Count>=0)

    {

        lcd_gotoxy(9,1);

        A = (CCP_Count)/14;

        printf(lcd_putc,"%f M/S",A);

        delay_ms(1000);

        CCP_Count = 0;

        sec = 0;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

}

if(CCP_Count<=26&&CCP_Count>15)
{
lcd_gotoxy(9,1);
A = (CCP_Count)/15;
printf(lcd_putc,"%f M/S",A);
delay_ms(1000);
CCP_Count = 0;
sec = 0;
set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

if(CCP_Count<=39&&CCP_Count>26)

{

    lcd_gotoxy(9,1);

    A = (CCP_Count)/20;

    printf(lcd_putc,"%0.2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=89&&CCP_Count>39)

{

    lcd_gotoxy(8,1);

    A = (CCP_Count)/21;

    printf(lcd_putc,"%0.2f M/S",A);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_ms(1000);

CCP_Count = 0;

sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);
}
if(CCP_Count<=140&&CCP_Count>89)
{
    lcd_gotoxy(9,1);

    A = (CCP_Count)/23;

    printf(lcd_putc,"%2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V =ADC1*1.411;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=219&&CCP_Count>140)

{

    lcd_gotoxy(9,1);

    A = (CCP_Count)/24;

    printf(lcd_putc,"%0.2f M/S",A);

    delay_ms(1000);

    CCP_Count = 0;

    sec = 0;

    set_adc_channel(0);

    ADC1 = Read_adc();

    V=ADC1*1.411;

    lcd_gotoxy(11,2);

    printf(lcd_putc,"%luDEG",V);

    delay_ms(3000);

}

if(CCP_Count<=249&&CCP_Count>219)

{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcd_gotoxy(9,1);

A = (CCP_Count)/25;

printf(lcd_putc,"%0.2f M/S",A);

delay_ms(1000);

CCP_Count = 0;

sec = 0;

set_adc_channel(0);

ADC1 = Read_adc();

V=ADC1*1.411;

lcd_gotoxy(11,2);

printf(lcd_putc,"%luDEG",V);

delay_ms(3000)
}

if(CCP_Count>249)
{

lcd_gotoxy(8,1);

A = (CCP_Count)/25;

printf(lcd_putc,"%0.2f M/S",A);

delay_ms(1000);

CCP_Count = 0;

sec = 0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
set_adc_channel(0);  
  
ADC1 = Read_adc();  
  
V=ADC1*1.411;  
  
lcd_gotoxy(11,2);  
  
printf(lcd_putc,"%luDEG",V);  
  
delay_ms(3000);  
  
}  
printf("\n%.2f|%lu\r",A,V);  
}  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Data.OleDb;
using System.Windows.Forms.DataVisualization.Charting;

namespace Project
{
    public partial class Form1 : Form
    {
        string strConn = "Provider = Microsoft.Jet.OLEDB.4.0;";
        string pathfile;
        OleDbConnection connectObj = new OleDbConnection();
        OleDbDataAdapter da;
        System.Data.DataSet ds = new DataSet();
        OleDbCommand cm = new OleDbCommand();
        OleDbConnection cn = new OleDbConnection();

        public Form1()
        {
            InitializeComponent();
            timer1.Start();
            FillPort();
            this.Text += " v." + Application.ProductVersion;
            chart1.Series["Series1"].ChartType = SeriesChartType.Line;
            chart1.ChartAreas["ChartArea1"].AxisY.Minimum = 0;
            chart1.ChartAreas["ChartArea1"].AxisY.Maximum = 20;
            chart1.ChartAreas["ChartArea1"].AxisX.IsMarginVisible = false;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

circleControl1.PrimaryMarkerAngle = Convert.ToSingle(270);
}

public void FillPort()
{
    // ** Fill port com add to combo box.
    string[] portName = System.IO.Ports.SerialPort.GetPortNames();
    for (int i = 0; i < portName.Length; i++)
    {
        Port.Items.Add(portName[i]);
    }
    Port.SelectedIndex = 0;
}

private void timer1_Tick_1(object sender, EventArgs e)
{
    label17.Text = DateTime.Now.ToLongTimeString();
    label18.Text = DateTime.Now.ToLongTimeString();
    label19.Text = DateTime.Now.ToLongTimeString();
}

private void oToolStripMenuItem_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "MDB Files|*.mdb"; //ให้ openFileDialog1 ใช้ระบบการ
แสดงผลไฟล์ได้แค่ .mdb
    openFileDialog1.DefaultExt = "*.mdb"; //ให้ openFileDialog1 มีค่านามสกุลตั้งเดิมใน
การแสดงผลแค่ .mdb
    openFileDialog1.FileName = ""; //ลบชื่อไฟล์ในช่อง FileName ออก
    openFileDialog1.Multiselect = false; //ไม่สามารถเลือกได้หลายไฟล์
    if (openFileDialog1.ShowDialog() == DialogResult.OK) //ถ้า openFileDialog1 แสดง
ได้อะลือกขึ้นมาแล้วเลือกไฟล์ โดยมีค่าเท่ากับ ผลลัพธ์ของ Dialog คือ OK
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        pathfile = openFileDialog1.FileName;
        strConn = "Provider = Microsoft.Jet.OLEDB.4.0;";
        strConn = strConn + "Data Source = " + pathfile;
    }
}

private void button2_Click(object sender, EventArgs e)
{
    try
    {
        connectObj.ConnectionString = strConn;
        connectObj.Open();
        string startdate = dateTimePicker1.Value.ToShortDateString();
        string enddate = dateTimePicker2.Value.ToShortDateString();
        string starttime = dateTimePicker3.Value.ToShortTimeString();
        string endtime = dateTimePicker4.Value.ToShortTimeString();

        string strSQL = "select DATE_, Windspeed, Winddirection from WIND where
DATE_ between #" + startdate + " " + starttime + "# "
        + "and #" + enddate + " " + endtime + "#";
        da = new OleDbDataAdapter(strSQL, strConn);
        if (ds.Tables.Contains("WIND"))
        {
            ds.Tables.Remove("WIND");
            dataGridView1.DataSource = null;
        }
        da.Fill(ds, "WIND");
        dataGridView1.DataSource = ds.Tables[0];
        strSQL = "select MIN(WINDSPEED),MAX(WINDSPEED) from WIND where
DATE_ between #" + startdate + " " + starttime + "# "
        + "and #" + enddate + " " + endtime + "#";
        cm.CommandText = strSQL;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cn.ConnectionString = strConn;
cn.Open();
cm.Connection = cn;
OleDbDataReader rd = cm.ExecuteReader();
if (rd.Read() != false)
{
    MaxBox.Text = rd[0].ToString();
    MinBox.Text = rd[1].ToString();
}
rd.Dispose();
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "ERROR_OPENFILE",
MessageBoxButtons.OK, MessageBoxIcon.Error);
}
finally
{
    connectObj.Close();
    cn.Close();
}
}

private void mnuClose_Click(object sender, EventArgs e)
{
    Application.Exit();
}

private void But_comport_Click(object sender, EventArgs e)
{
    if (But_comport.Text == "CONNECT")
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        But_comport.Text = "DISCONNECT";
        serialPort1.PortName = Port.SelectedItem.ToString();
        serialPort1.Open();
    }
    else
    {
        But_comport.Text = "CONNECT";
        serialPort1.Close();
    }
}

private void AddLog(String sDirection, String sSpeed)
{
    try
    {
        connectObj.ConnectionString = strConn;
        connectObj.Open();

        int dir = int.Parse(sDirection);
        double speed = double.Parse(sSpeed);
        string strSQL = "INSERT INTO wind (DATE_, Windspeed, Winddirection) " +
            "VALUES (#"+DateTime.Now.ToShortDateString()+
            "+DateTime.Now.ToLongTimeString()+"#, "+
            """+speed+""", """+dir+"")";

        cm.CommandText = strSQL;
        cn.ConnectionString = strConn;
        cn.Open();
        cm.Connection = cn;
        cm.ExecuteNonQuery();
    }
    catch (Exception ex)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        MessageBox.Show(ex.Message, "ERROR_OPENFILE",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    finally
    {
        connectObj.Close();
        cn.Close();
    }
}

string[] str_split = { "0", "0" };
byte[] buffer = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };

private void dateTimePicker5_ValueChanged(object sender, EventArgs e)
{
    radioButton1.Checked = true; // ทำเครื่องหมายถูกที่ Day
    try
    {
        connectObj.ConnectionString = strConn;
        connectObj.Open();
        string startdate = dateTimePicker5.Value.ToShortDateString();
        string enddate = dateTimePicker5.Value.ToShortDateString();
        string starttime = "00:00:00"; // dateTimePicker3.Value.ToShortTimeString();
        string endtime = "23:59:59"; // dateTimePicker4.Value.ToShortTimeString();

        string strSQL = "select Format(DATE_, \"Short Time\") AS T,Windspeed from
        WIND where DATE_ between #" + startdate + " " + starttime + "# "
            + "and #" + enddate + " " + endtime + "#";
        Clipboard.SetText(strSQL);

        cm.CommandText = strSQL;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cn.ConnectionString = strConn;
cn.Open();
cm.Connection = cn;
OleDbDataReader rd = cm.ExecuteReader();
if (rd.Read() != false)
{
    chart1.Series["Series1"].Points.DataBindXY(rd, "T", rd, "Windspeed");
}
rd.Dispose();
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "ERROR_OPENFILE",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
}
finally
{
    connectObj.Close();
    cn.Close();
}
}

private void dateTimePicker6_ValueChanged(object sender, EventArgs e)
{
    radioButton2.Checked = true; // ทำเครื่องหมายถูกที่ Month
    try
    {
        connectObj.ConnectionString = strConn;
        connectObj.Open();
        DateTime std = new DateTime(dateTimePicker6.Value.Year,
        dateTimePicker6.Value.Month, 1);
        string startdate = std.ToShortDateString();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int last_day = DateTime.DaysInMonth(std.Year,std.Month);
DateTime std2 = new DateTime(dateTimePicker6.Value.Year,
dateTimePicker6.Value.Month, last_day);

string enddate = std2.ToShortDateString();

string starttime = "00:00:00";// dateTimePicker3.Value.ToShortTimeString();
string endtime = "23:59:59"; // dateTimePicker4.Value.ToShortTimeString();

string strSQL = "select Format(DATE_, '\dd\') As D , AVG(Windspeed) AS
AvgWindspeed "+
                "from WIND " +
                "where DATE_ between #"+startdate+" "+starttime+"# and
#"+enddate+" "+endtime+"# "+
                "group by Format(DATE_, '\dd\') "+
                "ORDER BY Format(DATE_, '\dd\')";

cm.CommandText = strSQL;
cn.ConnectionString = strConn;
cn.Open();
cm.Connection = cn;
OleDbDataReader rd = cm.ExecuteReader();
chart1.Series["Series1"].Points.Clear();
if (rd.Read() != false)
{
    chart1.Series["Series1"].Points.DataBindXY(rd, "D", rd, "AvgWindspeed");
}
rd.Dispose();
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "ERROR_OPENFILE",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

finally
{
    connectObj.Close();
    cn.Close();
}
}

private void serialPort1_DataReceived(object sender,
System.IO.Ports.SerialDataReceivedEventArgs e)
{
    try
    {
        text_speed.Text = "";
        serialPort1.Read(buffer, 0, 10);

        string temp = System.Text.Encoding.ASCII.GetString(buffer).Trim();
        //MessageBox.Show(temp);
        if (!temp.Contains("|")) return;
        str_split = temp.Split('|');
        // แสดงผลบนจอภาพ
        circleControl1.PrimaryMarkerAngle = Convert.ToSingle(180 -
(Convert.ToSingle(str_split[1]) - 90) % 360);
        analogMeter1.Value = float.Parse(str_split[0]);
        text_speed.Text = analogMeter1.Value.ToString();
        text_Direct.Text = str_split[1]; //(circleControl1.PrimaryMarkerAngle -
90).ToString();//

        //ใส่ Database
        AddLog(text_Direct.Text, analogMeter1.Value.ToString());
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
catch (System.Exception ex)
{
    MessageBox.Show(ex.ToString(), "ERROR_LIVEDATA",
    MessageBoxButtons.OK);
}
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ใบรับรอง

หนังสือฉบับนี้ขอรับรองว่า โครงการเรื่อง เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม (Anemometer and Wind vanes) ที่นักศึกษาสาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังได้มาทำการ ศึกษาข้อมูล สร้าง สอบเทียบและปรับแต่งค่าของเครื่องวัดความเร็วและทิศทางลม มีนักศึกษาในการจัดทำโครงการนี้จำนวน 4 คน ซึ่งมีรายชื่อดังต่อไปนี้

1. นายคุณานนท์ นันตา รหัสนักศึกษา 49010094
2. นางสาวกมลชนก ช่างสมบัติ รหัสนักศึกษา 50010015
3. นายกฤษณะศักดิ์ รักเนตรสาคร รหัสนักศึกษา 50010047
4. นายฉัฐพงษ์ แสงคำ รหัสนักศึกษา 50010475

โดยมีเจ้าหน้าที่ของกรมอุตุนิยมวิทยา สำนักกองเครื่องมือ ส่วนมาตรฐานเครื่องมือ เป็นที่ปรึกษา อธิบาย ให้คำแนะนำ ควบคุม ดูแล

ปรากฏว่า เครื่องมือดังกล่าวได้ถูกสร้างขึ้นตามทฤษฎีและหลักวิชาการ พร้อมทำการสอบเทียบผลการตรวจวัดซึ่งค่าที่ได้ถูกต้องอยู่ในเกณฑ์ที่ยอมรับได้ จึงสามารถนำไปใช้งานได้จริง

ลงชื่อ.....
(นายอัครฤทธิ นิมเสมอ)

ตำแหน่ง ผู้อำนวยการส่วนมาตรฐานเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น 4 กุมภาพันธ์ 2554 ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THAI METEOROLOGICAL DEPARTMENT



4353 Sukhumvit, Bangna, Bangkok 10260 Tel. 0-2396-0156, 0-2399-0469

Calibration Certificate

Issued by : Calibration & Test Section : Meteorological Instruments Division

Date of Issue 4 February, 2011

Order No MD-031/11

Page : 1 of 2

Object : เครื่องวัดความเร็วลมและเครื่องวัดทิศทางลม

Manufacturer : -

Type : -

Serial No. : -

Customer : สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ถนนฉลองกรุง แขวงลาดกระบัง เขตลาดกระบัง 10520.

Calibration Condition : Temperature 24.5 °C Barometric Pressure 1009.6 hPa

NATIONAL STANDARD WIND TUNNEL

: HOOK GAGE NO 1425 Pitot Tube Theodor Friedrichs Type 0800.0000 serial 9023

: Thermal Anemometer 642 S/N 91563 : Wind Aloft Plotting Board

N.I.S.T. Test Reference Number 731/241460

: Ultrasonic Anemometer Model DA-650-3TV (sensor TR-90AH)

Serial Number 110730029 (sensor 120629586)

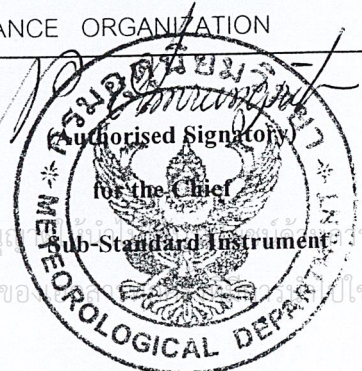
JAPAN QUALITY ASSURANCE ORGANIZATION

Checked by *Watchapol*

Mr. Watchapol Subwat
Mechanical Engineer

Signed : *Amrungrit*

Mr. Amrungrit Nimsamer



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นำไปใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของ



THAI METEOROLOGICAL DEPARTMENT

4353 Sukhumvit, Bangna, Bangkok 10260 Tel. 0-2396-0156,0-2399-0469

The Result of Calibration

Order No MD-031/11

4 February, 2011

Page : 2 of 2

FAN SPEED rpm	Ultrasonic Anemometer m/sec	HOOK GAGE NO. 1425			TESTED ANEMOMETER			
		Pressure inches	Vacumm inches	Pressure hPa	Pressure hPa	Correction hPa	Velocity m/sec	Correction m/sec
45	1.01	-	-	-	-	-	0.95	0.06
82	2.01	-	-	-	-	-	2.00	0.01
116	3.00	-	-	-	-	-	3.00	0.00
151	4.00	-	-	-	-	-	3.96	0.04
185	5.01	-	-	-	-	-	4.93	0.08
220	6.01	-	-	-	-	-	6.13	-0.12
254	7.00	-	-	-	-	-	6.96	0.04
287	8.00	-	-	-	-	-	8.01	-0.01
322	9.01	-	-	-	-	-	8.91	0.10
357	10.01	-	-	-	-	-	10.00	0.01
392	11.02	-	-	-	-	-	10.98	0.04
426	12.03	-	-	-	-	-	12.02	0.01
459	13.02	-	-	-	-	-	12.90	0.12
493	14.01	-	-	-	-	-	14.00	0.01
527	15.01	-	-	-	-	-	15.03	-0.02

Wind Aloft Plotting Board.	
US. DEPARTMENT OF COMMERCE WEATHER BUREAU	
WIND DIRETION	TESTED WIND DIRECTION
0	0
90	90
180	180
360	

Checked by :

Watchapol

Mr. Watchapol Subwat

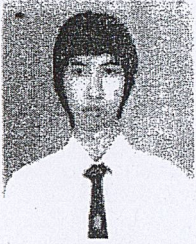
Mechanical Engineer



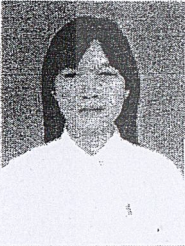
Meteorological Instruments Division

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรนำออกเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องสงวนลิขสิทธิ์ไว้ซึ่งการนำไปใช้

ประวัติผู้เขียน



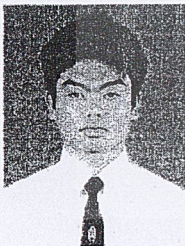
นาย คุณานนท์ นันตา เกิดวันที่ 11 มีนาคม 2530
ที่อยู่ 79 / 5 ม.19 ถ.บางนา-ตราด ต.บางปะกง อ.บางปะกง จ.ฉะเชิงเทรา
โทรศัพท์ 087-685-8275 Email : Suwannapum_new@hotmail.com



นางสาว กมลชนก ช่างสมบุญ เกิดวันที่ 19 มิถุนายน 2532
ที่อยู่ 193/5 ถ.อินทรีรี ต.แม่สอด อ.แม่สอด จ.ตาก 63110
โทรศัพท์ 083-206-1474 Email : Capacitior@hotmail.com



นาย กฤษณะศักดิ์ รักเนตรสาคร เกิดวันที่ 24 กุมภาพันธ์ 2532
ที่อยู่ 20/3 ม.3 ต.หนองอ้อ อ.บ้านโป่ง จ.ราชบุรี 70110
โทรศัพท์ 081-629-5734 Email : puggo_4@hotmail.com



นาย ณัฐพงษ์ แสงคำ เกิดวันที่ 21 พฤษภาคม 2532
ที่อยู่ 8/1 ม.6 ต.คลองคะเชนทร์ อ.เมือง จ.พิจิตร 66000
โทรศัพท์ 086-776-4775 Email : auyok_20@hotmail.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้